Oracle® WebCenter

Administrator's Guide for Application Adapters

11*g* Release 1 (11.1.1)

E17953-03

February 2013

Documentation for administrators that describes how to configure Imaging and Managed Attachments solutions using the Oracle WebCenter application adapters for Oracle E-Business Suite, Oracle PeopleSoft, and other business applications.

ORACLE

Oracle WebCenter Administrator's Guide for Application Adapters, 11*g* Release 1 (11.1.1)

E17953-03

Primary Author: Sarah Howland

Contributor: Oracle WebCenter development, product management, and quality assurance teams

Contents

Preface xvii

Audience xvii
Documentation Accessibility xvii
Related Documents xvii
Conventions xvii

What's New In This Guide xix

New and Changed Features in 11g Release 1 (11.1.1.7) xix
New and Changed Features in 11g Release 1 (11.1.1.6) xx

1 Introduction to Application Adapter Solutions

1.1 About Application Extension Framework (AXF) 1-1
1.2 About the BPM Imaging Solution 1-2
1.2.1 Business User View of the BPM Imaging Solution 1-2
1.2.1.1 Imaging Scenario 1: Entering Sales Quotes 1-3
1.2.2 About the Business Tier Agility Administrator Role 1-3
1.2.3 About AXF Commands Used by the BPM Imaging Solution 1-3
1.2.4 About BPM Imaging User Interface Components 1-4
1.2.4.1 About the Solution Workspace 1-4
1.2.4.2 About the Solution Application 1-5
1.3 About the BPEL Imaging Solution 1-6
1.3.1 Business User View of the BPEL Imaging Solution 1-6
1.3.1.1 Imaging Scenario 1: Processing Invoices 1-7
1.3.1.2 Imaging Scenario 2: Capturing Supporting Employee Documents 1-7
1.3.1.3 Imaging Scenario 3: Viewing Supporting Employee Documents 1-7
1.3.2 About AXF Commands Used by the BPEL Imaging Solution 1-8
1.3.3 About BPEL Imaging User Interface Components 1-8
1.3.3.1 About the Task List 1-9
1.3.3.2 About the Task Viewer 1-9
1.3.3.3 About the Enumeration Picker 1-10
1.3.3.4 About the Identity Picker 1-10
1.3.3.5 About Comments 1-11
1.4 About the Managed Attachments Solution 1-12
1.4.1 Business User View of the Managed Attachments Solution 1-13
1.4.1.1 Attachments Scenario 1: Capturing Identity Documents 1-13

1.4.1.2 Attachments Scenario 2: Viewing and Editing Identity Documents 1-14
1.4.2 About the AXF Command and Service Used By the Attachments Solution 1-14
1.5 About Adapter Components 1-15
1.5.1 Business Application Plug-In 1-15
1.5.2 AXF Infrastructure 1-16
1.5.2.1 Solution Mediator 1-16
1.5.2.2 AXF Request /Response 1-16
1.5.2.3 Command Mediator 1-16
1.5.2.4 AXF Commands 1-16
1.5.2.5 Conversation 1-17
1.5.3 AXF User Interface Components 1-17
1.5.4 Content Server Components (Managed Attachments) 1-17
1.5.5 AXF Solution Accelerators 1-17
1.6 About Adapter Security and Authentication 1-17
1.7 Adapter System Requirements 1-18
1.7.1 Oracle E-Business Suite 1-19
1.7.2 Oracle PeopleSoft Enterprise 1-19
1.7.3 SOA 11g (Imaging Solutions Only) 1-19
1.7.4 Oracle BPM Suite (BPM Imaging Solution Only) 1-20
1.7.5 Additional Managed Attachments Requirements 1-20
1.8 Adapter Configuration Roadmap 1-20

2 Configuring the Oracle E-Business Suite Plug-In

2.1 Configuring the Oracle E-Business Suite Database 2-1
2.1.1 Creating the AXF Oracle E-Business Suite Configuration Schema User (AXF) 2-2
2.1.2 PL/SQL Procedures Provided 2-2
2.1.3 Running the PL/SQL Scripts 2-3
2.2 Compiling Oracle E-Business Suite Forms 2-5
2.2.1 Compiling AXF_CUSTOM.pll Using the Command Line 2-8
2.3 Configuring Oracle Application Framework (OAF) Pages Via Personalization 2-8
2.3.1 Compiling Java Files, Copying Class Files, and Importing XML Files 2-9
2.3.2 Personalizing OAF Pages to Add Managed Attachments 2-12
2.3.3 Configuring Other Services on OAF Pages (Optional) 2-14
2.4 Securing Adapter Communications for Oracle E-Business Suite 2-15
2.4.1 Securing Communications Through SOAP Security 2-15
2.4.2 Securing Communications Through SSL (Oracle E-Business Suite) 2-16
2.4.2.1 Configuring the Oracle Wallet 2-16
2.4.2.2 Configuring the Integration for SSL 2-16
2.5 Configuring Oracle E-Business Suite Logging 2-17
2.5.1 Configuring Logging for Forms Applications 2-17
2.5.2 Configuring Logging for OAF Applications 2-18
2.6 Setting User Locales 2-18
2.7 Configuring Solutions for Oracle E-Business Suite 2-18
2.8 Uninstalling AXF From Oracle E-Business Suite 2-18
2.8.1 Uninstalling From Oracle E-Business Suite Forms 2-19
2.8.2 Uninstalling From Oracle E-Business Suite OAF 2-19

3 Configuring the Oracle PeopleSoft Plug-In

3.1 Importing the Oracle PeopleSoft Project ... 3-1
3.2 Configuring and Securing Adapter Communications for Oracle PeopleSoft ... 3-2
3.2.1 Securing Communications Through SSL (Oracle PeopleSoft) ... 3-3
3.2.2 Configuring Integration Broker to Communicate With AXF ... 3-3
3.2.2.1 Configuring Communication With an 11g AXF Server ... 3-3
3.2.2.2 Configuring Communication With a 10g AXF Server ... 3-4
3.2.2.3 Encrypting the Gateway Password ... 3-5
3.2.2.4 Setting Up the Service Operation Routings ... 3-5
3.2.2.5 Validating Domain Status ... 3-5
3.2.3 Configuring AXF Access For Oracle PeopleSoft Users ... 3-6
3.2.3.1 Verifying the AXF_ROLE for Oracle PeopleSoft Users ... 3-6
3.2.3.2 Assigning Users the AXF_ROLE ... 3-6
3.3 Placing AXF Components on Oracle PeopleSoft Pages ... 3-7
3.4 Configuring Oracle PeopleSoft Logging ... 3-9
3.5 Configuring Solutions for Oracle PeopleSoft ... 3-9
3.6 Uninstalling AXF From Oracle PeopleSoft ... 3-10

4 Configuring AXF Settings

4.1 Configuring Logging for AXF for BPM ... 4-1
4.1.1 Advanced Configuration ... 4-2
4.1.2 Configuring DMS Logging ... 4-4
4.1.2.1 Clearing DMS Logging ... 4-4
4.2 Configuring Logging for AXF for BPEL ... 4-4
4.2.1 Using Enterprise Manager ... 4-4
4.2.2 Using Application Server ... 4-5

5 Configuring the BPM Imaging Solution

5.1 Understanding Solution Application Administration ... 5-1
5.1.1 About the Dynamic Page Template and Page Components ... 5-2
5.1.1.1 About the Action Control ... 5-3
5.1.1.2 About the Update Status Icon ... 5-3
5.1.1.3 About Dynamic Tabs ... 5-3
5.1.1.4 About Page Components ... 5-3
5.1.2 About Business Rules and Dictionaries ... 5-4
5.1.2.1 About Editing Dictionaries ... 5-5
5.1.2.2 About Rulesets and Bucketsets ... 5-5
5.1.2.3 About Inputs ... 5-6
5.1.3 About Functional Blocks ... 5-6
5.1.4 About Control Pages ... 5-7
5.1.4.1 About the Data Control Page ... 5-8
5.1.4.2 About the Identity Browser Control Page ... 5-8
5.1.4.3 About the Comments Control Page ... 5-9
5.1.5 About Prompts and Validations ... 5-9
5.1.6 About Packages ... 5-10
5.1.7 About Contexts ... 5-11

5.1.8 About Data Fields, Tables, and Lookups 5-12
5.1.8.1 About Data Fields 5-12
5.1.8.2 About Tables 5-13
5.1.8.3 About Lookups 5-14
5.1.9 About Disabling or Discarding Business Rule Changes 5-15
5.2 Getting Started Configuring the Solution Application 5-15
5.2.1 Running or Applying a BPM Imaging Solution 5-15
5.2.2 Starting Up Solution Administration 5-15
5.2.3 Testing the Solution as a User Through the Command Driver Page 5-16
5.2.3.1 Using the Command Executor 5-18
5.2.4 Injecting Tasks for Use in the HelloBPM Solution 5-18
5.2.5 Managing Business Rules 5-19
5.2.6 Managing Contexts 5-23
5.2.6.1 Creating a Context 5-23
5.2.6.2 Updating a Context 5-24
5.2.6.3 Deleting a Context 5-24
5.2.7 Resetting a Dictionary's Custom Rules to Product Settings 5-24
5.3 Understanding the HelloBPM Solution 5-25
5.3.1 What is the HelloBPM Solution? 5-25
5.3.2 Overview of the HelloBPM Sales Quote System 5-26
5.3.3 HelloBPM User Roles and Workflow 5-26
5.3.4 How the HelloBPM Solution Uses Contexts 5-27
5.3.5 Business Rules for the Sales Quote Entry Context 5-28
5.3.5.1 Action Rulesets for the SalesQuoteEntry Context 5-29
5.3.5.2 Data Rulesets for the SalesQuoteEntry Context 5-29
5.3.5.3 DynamicTab Rulesets for the SalesQuoteEntry Context 5-30
5.3.5.4 Data Rulesets for the SalesQuoteEntry_AccountManagerReason Context 5-31
5.3.5.5 Table Rulesets for the SalesQuoteEntry_ProductItem Context 5-31
5.3.5.6 Table Rulesets for the SalesQuoteEntry_LicenseTerm Context 5-31
5.3.6 Business Rules for the Base Context 5-32
5.3.6.1 FunctionalBlock Rulesets for the Base Context 5-32
5.3.6.2 LookupAndValidator Rulesets for the Base Context 5-34
5.4 HelloBPM Solution Use Cases 5-35
5.4.1 Use Case: Add and Configure an Action Link 5-36
5.4.2 Use Case: Navigate to a Data Control Page 5-37
5.4.3 Use Case: Add a New Context 5-38
5.4.4 Use Case: Add a Data Field in a New Section 5-40
5.4.5 Use Case: Set a Data Field's Background Color 5-42
5.4.6 Use Case: Change the Display Order of Data Items or Sections 5-42
5.4.7 Use Case: Set a Section to Initially Display Closed 5-43
5.4.8 Use Case: Hide all Table Row Buttons 5-43
5.4.9 Use Case: View a Static Lookup's Configuration 5-44
5.4.10 Use Case: View a Dynamic Lookup's Configuration 5-44
5.5 Configuring BPM Views 5-45
5.6 Using Oracle Fusion Middleware Control to Manage AXF for BPM 5-45
5.6.1 Redeploying AXF for BPM Libraries and Applications 5-45
5.6.2 Temporarily Turning Off Business Rule Modifications 5-46

5.6.3 Moving from a Test to a Production AXF for BPM Environment 5-46

6 Configuring the BPEL Imaging Solution

6.1 Configuring the BPEL Connection 6-1
6.1.1 Creating a CSF Credential Alias 6-1
6.1.2 Creating a Connection in Imaging Connections 6-2
6.1.3 Referencing the Connection in the AXF_SOLUTION_ATTRIBUTES Table 6-2
6.2 Configuring the AXF Tables or Applying a Solution Accelerator 6-3
6.3 Testing Functionality Using the AXF Driver Page 6-3
6.4 Configuring Imaging Solution Options 6-3
6.4.1 Configuring Automatic Imaging Viewer Login 6-3
6.4.2 Allowing Oracle WebLogic Server/Business Application Users to Perform Tasks .. 6-4
6.4.3 Configuring Clustering for AXF in Imaging Managed Servers 6-4
6.4.4 Configuring Autotask Locking 6-4
6.4.5 Updating the Task Payload Using XPATH 6-5
6.4.5.1 Example PL/SQL Procedure For Updating the Task Payload 6-5
6.4.6 Targeting a Single Conversation With Multiple Window Instances Displayed 6-6
6.4.7 Adding a Validation 6-6
6.4.8 Deploying Custom Commands 6-7
6.4.9 Configuring Chained Commands and User Interface Components 6-7
6.4.10 Configuring or Disabling a Dynamic Data Table 6-8
6.4.10.1 Formatting XML Data For a Dynamic Data Table 6-8
6.4.10.2 Hiding the Bottom Panel in the Task Viewer 6-9
6.4.11 Reenabling Paperclip Attachments (Oracle E-Business Suite and Imaging Only) 6-9
6.4.12 Resizing Oracle PeopleSoft Popup Solution Windows 6-10

7 Configuring the Managed Attachments Solution

7.1 System Requirements for the Managed Attachments Solution 7-1
7.2 Running Configuration Scripts to Enable Managed Attachments 7-2
7.2.1 Configuring AXF Elements for Managed Attachments 7-3
7.2.2 Enabling Managed Attachments for Oracle E-Business Suite Forms Applications .. 7-4
7.2.3 Enabling Managed Attachments for Oracle E-Business Suite OAF Applications 7-5
7.2.3.1 Populating the OAF AXF Database Tables With Sample Key Values 7-5
7.2.3.2 Identifying Key Values for Your OAF Pages 7-6
7.2.3.2.1 With Both Attachment Information and Primary Keys Present 7-7
7.2.3.2.2 With Attachment Information Present, But Not Primary Keys 7-7
7.2.3.2.3 With Neither Attachment Information Nor Primary Keys Present 7-8
7.2.4 Enabling Managed Attachments for Oracle PeopleSoft 7-11
7.3 Configuring the Content Server Components 7-12
7.3.1 Enabling Content Server Components 7-12
7.3.1.1 Backing Up Customizations Before Upgrading 7-13
7.3.2 Verifying Required Content Server Components and Configuration 7-13
7.3.3 Testing the Managed Attachments Solution 7-14
7.4 Configuring the Enterprise Application Adapter Framework 7-14
7.4.1 Deploying the AppAdapterUniversal Component on Content Server 7-14
7.4.1.1 Optionally Renaming Component Files 7-15

7.4.1.2 Optionally Customizing the Framework Layout, Images, and Profile 7-15
7.4.2 Configuring the SOAP Service Call and Managed Attachments URL Display 7-15
7.4.2.1 Securing Communications Through SSL (Other Business Applications) 7-18
7.5 Configuring Managed Attachments Settings 7-18
7.5.1 Working With Global Configuration Variables 7-21
7.5.2 Alternatively Editing Settings in the Preferences.hda File 7-24
7.6 Customizing the Managed Attachments Solution 7-24
7.6.1 Customizing Settings Based on Users' Assigned Roles 7-25
7.6.2 Customizing Settings Based on Users' Business Context 7-26
7.6.3 Customizing Settings at Business Entity or Application Levels 7-27
7.6.4 Customizing the Document Type Used For New Attachments 7-28
7.6.5 Customizing the Content Server Profile Used 7-29
7.6.5.1 Changing the Profile Globally 7-29
7.6.5.2 Specifying a Profile For a Specific Business Entity or Application 7-29
7.6.6 Hiding or Displaying Managed Attachments Page Buttons 7-30
7.6.7 Customizing Scanning 7-30
7.6.7.1 Enabling or Disabling Scanning 7-31
7.6.7.2 Configuring an Alternate Scan Profile For a Business Entity 7-31
7.6.7.3 Modifying the Document Classification Used 7-32
7.6.8 Customizing Managed Attachments Page Images 7-32
7.6.9 Customizing the Style Sheet (CSS) Used 7-32
7.6.10 Configuring Attachment Relationship Attributes 7-32
7.6.11 Configuring Revision-Specific Mode 7-34
7.6.12 Changing the Result Count 7-35
7.6.13 Changing the Result Template 7-35
7.6.14 Setting Default Columns 7-36
7.6.15 Enabling the Paperclip with Attachments (Oracle E-Business Suite Forms Only) 7-37
7.6.16 Passing Extra Metadata Values From a Business Application to Content Server 7-37
7.6.16.1 Passing Values Via Enterprise Application Adapter Framework 7-38
7.6.17 Resizing Oracle PeopleSoft Popup Solution Windows 7-39
7.6.18 Configuring an Entity's Private Attachments Security Group 7-39
7.6.19 Restoring Business Application User Pass Through for Managed Attachments 7-40
7.6.20 Renaming the Managed Attachments Button For OAF Entities 7-40
7.7 Configuring Authentication and Security 7-41
7.7.1 Configuring a CSF Credential for Managed Attachments 7-41
7.7.1.1 Creating a Credential Using Enterprise Manager 7-41
7.7.1.2 Creating a Credential Using WLST 7-42
7.7.2 Configuring User Authentication 7-42
7.7.3 Configuring Document Security 7-42
7.7.4 Securing Communications Between AXF and Content Server 7-43
7.7.4.1 Using the Content Server System Properties Application 7-43
7.7.4.2 By Editing the Content Server Config.cfg File 7-43
7.7.5 Managing Business Application and Content Server Users 7-44
7.8 Configuring Oracle AutoVue Document Viewing 7-44
7.8.1 About Configuring Oracle AutoVue 7-44
7.8.2 Requirements For Oracle VueLink for Content Server 7-45
7.8.3 Enabling AutoVue For the Adapter 7-45

7.9 Configuring Oracle Distributed Document Capture 7-46
7.9.1 About Document Scanning Using Oracle Distributed Document Capture 7-46
7.9.2 Oracle Distributed Document Capture and Related Components Requirements .. 7-47
7.9.3 Configuring Oracle Distributed Document Capture For the Adapter 7-47
7.9.4 Configuring Content Server for Distributed Document Capture Via Adapter 7-54
7.9.5 Testing the Oracle Distributed Document Capture Via Adapter Configuration 7-55
7.10 Testing the Managed Attachments Solution 7-56
7.11 Configuring Content Server Logging 7-57
7.12 Uninstalling Adapter Components on Content Server 7-57

A BPM Imaging Solution Reference

A.1 AXF for BPM Business Rule Reference A-1
A.1.1 Page Component Parameters A-1
A.1.1.1 Parameters for the Image Page Component A-2
A.1.1.2 Parameters for the Image Data Page Component A-2
A.1.1.3 Parameters for the Data Page Component A-3
A.1.1.4 Parameters for the Table Data Page Component A-3
A.1.1.5 Parameters for the Master Detail Page Component A-4
A.1.1.6 Parameters for the Identity Browser Page Component A-4
A.1.1.6.1 Configuring the Identity Browser in Explicit or Relative Mode A-5
A.1.1.7 Parameters for the Comment Page Component A-5
A.1.2 Control Page Parameters A-5
A.1.2.1 Data Control Page Parameters A-5
A.1.2.2 Identity Browser Control Page Parameters A-6
A.1.2.3 Comment Control Page Parameters A-6
A.1.3 Action Dictionary A-7
A.1.3.1 Action A-7
A.1.3.2 Section A-7
A.1.3.3 ActionEnableModeException A-7
A.1.3.4 SectionEnableModeException A-8
A.1.3.5 ActionParameter A-8
A.1.3.6 SectionParameter A-8
A.1.4 Data Dictionary A-8
A.1.4.1 Data A-8
A.1.4.2 Section A-9
A.1.4.3 DataParameter A-9
A.1.4.4 SectionParameter A-9
A.1.4.5 DataComponentAttribute A-10
A.1.4.6 DataValidator A-10
A.1.5 DynamicTab Dictionary A-10
A.1.5.1 Tab A-11
A.1.5.2 TabParameter A-11
A.1.6 FunctionalBlock Dictionary A-11
A.1.6.1 NamedFunctionalBlock A-11
A.1.6.2 NamedFunctionalBlockSequence A-12
A.1.6.3 NamedFunctionalBlockParameter A-12
A.1.6.4 NamedFunctionalBlockPrompt A-12

A.1.6.5 NamedFunctionalBlockValidator A-13
A.1.7 LookupAndValidator Dictionary A-13
A.1.7.1 StaticLookup A-14
A.1.7.2 DynamicLookup A-14
A.1.7.3 DynamicLookupBindVariable A-14
A.1.7.4 LookupParameter A-15
A.1.7.5 DynamicLookupRelatedBinding A-15
A.1.7.6 Validator A-15
A.1.7.7 ValidatorParameter A-16
A.1.8 TableData Dictionary A-16
A.1.8.1 Table A-16
A.1.8.2 Column A-17
A.1.8.3 TableParameter A-17
A.1.8.4 ColumnParameter A-18
A.1.8.5 ColumnComponentAttribute A-18
A.1.8.6 ColumnValidator A-18
A.1.9 Business Rule Entity Relationships A-19
A.2 AXF Tables for the BPM Imaging Solution A-22
A.2.1 AXF_SOLUTIONS Table A-23
A.2.1.1 Column Description A-23
A.2.1.2 Example Implementation A-23
A.2.2 AXF_SOLUTION_ATTRIBUTES Table A-23
A.2.2.1 Column Description A-24
A.2.2.2 Example Implementation A-24
A.2.3 AXF_COMMANDS Table A-24
A.2.3.1 Column Description A-24
A.2.3.2 Example Implementation A-24
A.2.4 AXF_SOLUTION_PARAMETERS Table A-25
A.2.4.1 Column Description A-25
A.2.4.2 Example Implementation A-26
A.2.5 AXF_XPATH_ATTRIBUTES Table A-26
A.2.5.1 Column Description A-26
A.2.5.2 Example Implementation A-26
A.2.6 AXF_XPATH_NAMESPACES Table A-27
A.2.6.1 Column Description A-27
A.2.6.2 Example Implementation A-27
A.3 AXF Commands for the BPM Imaging Solution A-27
A.3.1 Open Solution Workspace Command A-28
A.3.1.1 Open Solution Workspace Command Parameters A-28
A.3.1.2 Example Implementation A-28
A.3.2 Update Task Command A-28
A.3.2.1 Update Task Command Parameters A-28
A.3.2.2 System Attributes A-29
A.3.2.3 Example Implementation A-29
A.3.3 Document Search Command A-29
A.3.3.1 Document Search Command Parameters A-30
A.3.3.2 Example Implementation A-30

A.4 Oracle E-Business Suite Forms Tables for the BPM Imaging Solution A-30
A.4.1 About the Oracle E-Business Suite AXF Tables For the BPM Imaging Solution..... A-30
A.4.2 AXF_CONFIGS Table (Oracle E-Business Suite) A-31
A.4.2.1 Column Description A-31
A.4.3 AXF_COMMANDS Table (Oracle E-Business Suite) A-32
A.4.3.1 Column Description A-32
A.4.4 AXF_COMMAND_PARAMETERS Table (Oracle E-Business Suite) A-33
A.4.4.1 Column Description A-33
A.4.5 AXF_PROPERTIES Table (Oracle E-Business Suite) A-33
A.4.5.1 Column Description A-34
A.5 Oracle PeopleSoft Tables for the BPM Imaging Solution..... A-34
A.5.1 About the Oracle PeopleSoft AXF Tables For the BPM Imaging Solution..... A-34
A.5.2 PS_AXF_CONFIG Table A-35
A.5.2.1 Column Description A-35
A.5.3 PS_AXF_COMMANDS Table..... A-36
A.5.3.1 Column Description A-36
A.5.4 PS_AXF_COMMAND_PARAMS Table..... A-36
A.5.4.1 Column Description A-37
A.5.5 PS_AXF_PROPERTIES Table..... A-37
A.5.5.1 Column Description A-37

B BPEL Imaging Solution Reference

B.1 AXF Tables For the BPEL Imaging Solution B-1
B.1.1 AXF_SOLUTIONS Table B-2
B.1.1.1 Column Description B-3
B.1.1.2 Example Implementation B-3
B.1.2 AXF_SOLUTION_ATTRIBUTES Table..... B-3
B.1.2.1 Column Description B-4
B.1.2.2 Example Implementation B-4
B.1.3 AXF_COMMANDS Table B-4
B.1.3.1 Column Description B-5
B.1.3.2 Example Implementation B-5
B.1.4 AXF_SOLUTION_PARAMETERS Table B-6
B.1.4.1 Column Description B-6
B.1.4.2 Example Implementation B-7
B.1.5 AXF_ACTION_MENU Table..... B-8
B.1.5.1 Column Description B-8
B.1.5.2 Example Implementation B-8
B.1.6 AXF_ACTIONS Table B-9
B.1.6.1 Column Description B-9
B.1.6.2 Example Implementation B-9
B.1.7 AXF_XPATH_ATTRIBUTES Table..... B-10
B.1.7.1 Column Description B-10
B.1.7.2 Example Implementation B-11
B.1.8 AXF_XPATH_NAMESPACES Table..... B-11
B.1.8.1 Column Description B-11
B.1.8.2 Example Implementation B-12

B.2 AXF User Components for the BPEL Imaging Solution B-12
B.2.1 Solution Editor B-12
B.2.2 Task List Web Tool B-12
B.2.2.1 Task List Parameters B-13
B.2.2.2 Example Implementation B-14
B.2.3 Task Viewer Web Tool B-14
B.2.3.1 AXF_METADATA_BLOCKS Table B-16
B.2.3.1.1 Column Description B-16
B.2.3.1.2 Example Implementation B-17
B.2.3.2 AXF_METADATA_ATTRIBUTES Table B-17
B.2.3.2.1 Column Description B-17
B.2.3.2.2 Example Implementation B-18
B.2.3.3 Comments B-18
B.2.4 Enumeration Picker Web Tool B-19
B.2.4.1 Enumeration Picker Parameters B-20
B.2.4.2 AXF_ENUM_TYPES Table B-21
B.2.4.3 AXF_ENUM_ITEMS Table B-21
B.2.5 Identity Picker Web Tool B-22
B.2.5.1 Identity Picker Parameters B-23
B.2.5.2 Example Implementation B-24
B.2.6 Account Distribution Coding Form (Oracle E-Business Suite Only) B-24
B.2.6.1 AXF_LOOKUPS Script for Coding Form Lookup Elements B-25
B.2.6.2 Example Account Distribution Coding Form Solution B-26
B.2.6.3 Example Supporting Commands for the Account Distribution Coding Form B-26
B.2.6.4 Example OpenTask and AutoOpenTask Commands B-27
B.3 AXF Commands for the BPEL Imaging Solution B-27
B.3.1 Open Task Command B-28
B.3.1.1 Open Task Command Parameters B-28
B.3.1.2 Example Implementation B-28
B.3.2 Autotask Command B-29
B.3.2.1 Autotask Command Parameters B-29
B.3.2.2 Example Implementation B-29
B.3.3 Release Task Command B-29
B.3.3.1 Release Task Command Parameters B-29
B.3.3.2 Example Implementation B-30
B.3.4 Complete Task Command B-30
B.3.4.1 Complete Task Command Parameters B-30
B.3.4.2 Example Implementation B-30
B.3.5 Redirect Command B-31
B.3.5.1 Redirect Command Parameters B-31
B.3.5.2 Example Implementation B-31
B.3.6 Update Task Command (BPEL) B-31
B.3.6.1 Update Task Command Parameters B-32
B.3.6.2 System Attributes B-32
B.3.6.3 Example Implementation B-33
B.3.7 Update Task From Procedure Command B-33
B.3.7.1 Update Task From Procedure Command Parameters B-33

B.3.7.2 Example Implementation B-33
B.3.7.3 Example PL/SQL Procedure For Updating the Task Payload B-35
B.3.8 Terminate Conversation Command B-36
B.3.9 Validate Task Command B-36
B.3.9.1 Validate Task Command Parameters B-36
B.3.9.2 Example Implementation B-36
B.4 Oracle E-Business Suite Forms Tables For the BPEL Imaging Solution B-37
B.4.1 About the Oracle E-Business Suite AXF Tables For the BPEL Imaging Solution B-37
B.4.2 AXF_CONFIGS Table (Oracle E-Business Suite) B-37
B.4.2.1 Column Description B-38
B.4.2.2 Example Implementation B-38
B.4.3 AXF_COMMANDS Table (Oracle E-Business Suite) B-39
B.4.3.1 Column Description B-39
B.4.3.2 Example Implementation B-39
B.4.4 AXF_COMMAND_PARAMETERS Table (Oracle E-Business Suite) B-40
B.4.4.1 Column Description B-40
B.4.4.2 Example Implementation B-40
B.4.5 AXF_PROPERTIES Table (Oracle E-Business Suite) B-41
B.4.5.1 Column Description B-41
B.4.5.2 Example Implementation B-42
B.5 Oracle PeopleSoft Tables For the BPEL Imaging Solution B-42
B.5.1 About the Oracle PeopleSoft AXF Tables For the BPEL Imaging Solution B-42
B.5.2 PS_AXF_CONFIG Table B-43
B.5.2.1 Column Description B-43
B.5.2.2 Example Implementation B-44
B.5.3 PS_AXF_COMMANDS Table B-44
B.5.3.1 Column Description B-44
B.5.3.2 Example Implementation B-45
B.5.4 PS_AXF_COMMAND_PARAMS Table B-45
B.5.4.1 Column Description B-45
B.5.4.2 Example Implementation B-46
B.5.5 PS_AXF_PROPERTIES Table B-46
B.5.5.1 Column Description B-46
B.5.5.2 Example Implementation B-46

C Managed Attachments Solution Reference

C.1 AXF Tables For the Managed Attachments Solution C-1
C.1.1 AXF_SOLUTIONS Table C-2
C.1.1.1 Column Description C-2
C.1.1.2 Example Implementation C-2
C.1.2 AXF_SOLUTION_ATTRIBUTES Table C-3
C.1.2.1 Column Description C-3
C.1.2.2 Example Implementation C-3
C.1.3 AXF_COMMANDS Table C-3
C.1.3.1 Column Description C-3
C.1.3.2 Example Implementation C-4
C.1.4 AXF_SOLUTION_PARAMETERS Table C-4

C.1.4.1 Column Description ... C-4
C.1.4.2 Example Implementations ... C-5
C.2 Oracle E-Business Suite Forms Tables For the Managed Attachments Solution ... C-7
C.2.1 AXF_CONFIGS Table (Oracle E-Business Suite Forms) ... C-8
C.2.1.1 Column Description ... C-9
C.2.1.2 Example Implementation ... C-9
C.2.2 AXF_PROPERTIES Table (Oracle E-Business Suite Forms) ... C-9
C.2.2.1 Column Description ... C-10
C.2.2.2 Example Implementation ... C-10
C.2.3 AXF_FND_MAP Table (Oracle E-Business Suite Forms) ... C-10
C.2.3.1 Column Description ... C-11
C.2.3.2 Example Implementation ... C-11
C.2.4 AXF_COMMANDS Table (Oracle E-Business Suite Forms) ... C-11
C.2.4.1 Column Description ... C-11
C.2.4.2 Example Implementation ... C-12
C.2.5 AXF_COMMAND_PARAMETERS Table (Oracle E-Business Suite Forms) ... C-12
C.2.5.1 Column Description ... C-12
C.2.5.2 Example Implementation ... C-12
C.3 Oracle E-Business Suite OAF Tables For the Managed Attachments Solution ... C-13
C.3.1 OAF_AXF_CONFIGS Table (Oracle E-Business Suite OAF) ... C-13
C.3.1.1 Column Description ... C-14
C.3.1.2 Example Implementation ... C-14
C.3.2 OAF_AXF_PROPERTIES Table (Oracle E-Business Suite OAF) ... C-14
C.3.2.1 Column Description ... C-15
C.3.2.2 Example Implementation ... C-15
C.3.3 OAF_AXF_COMMANDS Table (Oracle E-Business Suite OAF) ... C-15
C.3.3.1 Column Description ... C-16
C.3.3.2 Example Implementation ... C-16
C.3.4 OAF_AXF_CMD_PARAMS Table (Oracle E-Business Suite OAF) ... C-16
C.3.4.1 Column Description ... C-17
C.3.4.2 Example Implementation ... C-17
C.4 Oracle PeopleSoft Tables For the Managed Attachments Solution ... C-18
C.4.1 PS_AXF_CONFIG Table ... C-19
C.4.1.1 Column Description ... C-19
C.4.1.2 Example Implementation ... C-20
C.4.2 PS_AXF_COMMANDS Table ... C-20
C.4.2.1 Column Description ... C-20
C.4.2.2 Example Implementation ... C-21
C.4.3 PS_AXF_CMD_PARAMS Table ... C-21
C.4.3.1 Column Description ... C-22
C.4.3.2 Example Implementation ... C-22
C.4.4 PS_AXF_PROPERTIES Table ... C-23
C.4.4.1 Column Description ... C-24
C.4.4.2 Example Implementation ... C-24
C.5 Content Server Tables For the Managed Attachments Solution ... C-24
C.5.1 AFGrants Table ... C-24
C.5.1.1 Column Description ... C-24

C.5.2 AFObjects Table C-25
C.5.2.1 Column Description C-25
C.5.3 AFRelationshipAttributes Table C-25
C.5.3.1 Column Description C-25
C.5.4 AFKeys Table C-26
C.5.4.1 Column Description C-26

Preface

This document describes Oracle business application solution configurations for Oracle WebCenter systems.

Audience

This document is intended for administrators configuring integration solutions between Oracle business applications and Oracle WebCenter systems.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Related Documents

For end-user adapter information, see the *Oracle WebCenter User's Guide for Application Adapters*.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the page, or text that you enter.

What's New In This Guide

This section describes new Oracle WebCenter Application Adapter features. It contains the following topics:

- "New and Changed Features in 11g Release 1 (11.1.1.7)"
- "New and Changed Features in 11g Release 1 (11.1.1.6)"

New and Changed Features in 11g Release 1 (11.1.1.7)

This section covers the following topics:

- "New AXF for BPM Infrastructure Adds Configuration Ease (11.1.1.7)"
- "Managed Attachments Solution Features Added (11.1.1.7)"

New AXF for BPM Infrastructure Adds Configuration Ease (11.1.1.7)

This release introduces a new AXF infrastructure based on Business Process Management (Oracle BPM) rather than Business Process Execution Language (BPEL). AXF for BPM provides the framework for highly configurable human task solutions.

Instead of a solution developer modifying tables behind-the-scenes, a business agility tier administrator can now customize the user's business task experience through business rules, and see changes immediately reflected without redeploying.

This release includes both AXF for BPM and AXF for BPEL infrastructures. Note that you can use either AXF infrastructure with the Managed Attachments solution.

AXF for BPM information is included throughout this guide, with most new information contained in the following areas:

- Section 1.2, "About the BPM Imaging Solution"
- Chapter 5, "Configuring the BPM Imaging Solution"
- Appendix A, "BPM Imaging Solution Reference"

Managed Attachments Solution Features Added (11.1.1.7)

The following table describes new Managed Attachments solution features included with the application adapters.

Managed Attachments Solution Features	Description	Refer To
Role specific configuration	Role specific configuration entry has been added to the Managed Attachments Administration page which allows you to control the configurable features and button visibility, based on roles assigned to the user.	Section 7.6.1
Business context specific configuration	Business context specific configuration has been added to the Managed Attachments Administration page which allows you to control the configurable features and button visibility by passing the businessObjectContext from the business application.	Section 7.6.2
Configuration settings order of precedence	The order of precedence of the configuration settings, from highest to lowest is: role specific, business context specific, application entity specific, application instance specific, and global.	Section 7.6.1
Support for new search engine	A new search engine called DATABASE.METADATA.AFLIST.STANDARD has been added, providing flexibility in listing documents. Unlike the DATABASE.METADATA.AFLIST engine, which displays released and unreleased documents, DATABASE.METADATA.AFLIST.STANDARD displays released documents only.	Section 7.5.1

New and Changed Features in 11g Release 1 (11.1.1.6)

This section covers the following topics:

- "BPEL Imaging Solution Features Added (11.1.1.6)"
- "Managed Attachments Solution Features Added (11.1.1.6)"

BPEL Imaging Solution Features Added (11.1.1.6)

The following table describes new BPEL Imaging solution features included with the application adapters.

BPEL Imaging Solution Features	Description	Refer To
Administration interface (Solution Editor) for making web tool changes	After solution configuration, use the Solution Editor in Imaging to quickly make changes to the user interface components without directly modifying the underlying tables. For example, change BPEL views available in the Task List, task actions listed in the Task Viewer, or enumeration values available on an Enumeration Picker page.	Section B.2.1
Account Distribution Coding Form	This AXF web interface allows for entering and selecting detailed document information, including line-item level values. The invoice processing solution accelerator uses the coding form to provide an account distribution entry form for qualified users to enter codes for non-PO invoices.	Section B.2.6
Using Imaging with OAF applications	Optionally you can provide access to the Imaging solution instead of the Managed Attachments solution on OAF (Oracle Application Framework) pages. For example, include a button that allows users to access the supplier maintenance functionality available in the Imaging solution.	Section 2.3.3
Searching for tasks in the Task List	If viewing many tasks in a Task List, users can search for a specific task by a key identifier. For example, users might search for an invoice task by its invoice number.	"Searching for Tasks" in *Oracle WebCenter User's Guide for Application Adapters*
Sorting tasks in the Task List	End-users can sort the Task List in ascending or descending order by clicking a column heading.	"Sorting Tasks" in *Oracle WebCenter User's Guide for Application Adapters*

Managed Attachments Solution Features Added (11.1.1.6)

The following table describes new Managed Attachments solution features included with the application adapters.

> **Note:** For information about additional certifications, such as for Oracle E-Business Suite or Oracle PeopleSoft plug-ins, see supported configuration and certification information at:
>
> http://www.oracle.com/technetwork/middleware/webcenter/content/documentation/documentation-155348.html

Managed Attachments Solution Features	Description	Refer To
Support for Oracle E-Business Suite OAF applications	Integrate Managed Attachments functionality into OAF (Oracle Application Framework) pages by adding a Managed Attachments button via page personalization and configuration.	Section 2.3
Support for configuring Managed Attachments on other business applications	Using the Enterprise Application Adapter Framework component, integrate Managed Attachments functionality into a business application other than Oracle E-Business Suite or Oracle PeopleSoft.	Section 7.4
Administration interface for customizing Managed Attachments functionality	As an alternative to editing the preferences.hda file, use this interface to configure global variables and add attachment relationships and entity-specific configurations.	Section 7.5.1
Customization variables added	▪ **AppAdapterCheckinPrivate:** Enables you to make private document check-in the default when users check in documents, instead of shared. ▪ **AppAdapterExtraCheckinMetadata** and AppAdapterExtraScanMetadata: See item that follows in this table regarding passing extra metadata. ▪ **AppAdapterHelpIconVisible:** Enables you to display or hide the user Help icon on the Managed Attachments page. ▪ **AppAdapterViewUnreleased:** Enables you to display or hide documents in the attachments list before they are released. In addition, **ResultCount** (number of attachments to display per page) and **SearchEngineName** (search engine to use) are now variables you can set and modify on the Configure Managed Attachments page.	Section 7.5.1
Pass extra metadata for use in Content Server	Enables you to pass metadata on the AXF grant call to be prepopulated during checkin.	Section 7.6.16
Simplified URL displayed to end-users	A ticket mechanism now hides detail such as business object IDs from end-users on the Managed Attachments URL.	
CSF enhancements in the Managed Attachments security model	You must now store administrator credentials in a credential store framework (CSF) using a configurable key name. The Managed Attachments solution compares the user name sent by Oracle E-Business Suite or Oracle PeopleSoft in the Web Services call with the user name stored in the CSF. If the administrator user names do not match, an exception occurs that prevents access to the Managed Attachments page.	Section 7.7.1
Resize Oracle PeopleSoft Popup Solution Windows	You can change or remove the default system-wide size for all Oracle PeopleSoft popup windows that display as part of solutions.	Section 6.4.12 and Section 7.6.17

Managed Attachments Solution Features	Description	Refer To
Private attachments security group configurable for specific business entity	On a per entity level, you can override the default security group for private attachments, and assign documents to an alternate private attachments security group.	Section 7.6.18
Users can search within the attachments list	If viewing many attachments on the Managed Attachments page, users can perform a search within the attachments list (for example, search for a document by its name).	"Searching Within the Attachments List" in *Oracle WebCenter User's Guide for Application Adapters*.

1

Introduction to Application Adapter Solutions

This chapter describes the application adapters for Oracle WebCenter. The adapters include the **Imaging Solution**, which provides imaging, capture, and workflow capabilities using Oracle WebCenter Content: Imaging, and the **Managed Attachments Solution**, which enables business users to attach, scan, and retrieve document attachments stored in an Oracle WebCenter Content Server (Content Server) repository.

This chapter covers the following topics:

- Section 1.1, "About Application Extension Framework (AXF)"
- Section 1.2, "About the BPM Imaging Solution"
- Section 1.3, "About the BPEL Imaging Solution"
- Section 1.4, "About the Managed Attachments Solution"
- Section 1.5, "About Adapter Components"
- Section 1.6, "About Adapter Security and Authentication"
- Section 1.7, "Adapter System Requirements"
- Section 1.8, "Adapter Configuration Roadmap"

1.1 About Application Extension Framework (AXF)

Oracle WebCenter Content business application integration solutions use the **Application Extension Framework (AXF)** infrastructure, a command-driven, web services integration, to integrate a business application such as Oracle E-Business Suite or Oracle PeopleSoft with a content management application such as Imaging or Content. AXF's open Java-based architecture enables integrators to configure and modify multiple Oracle WebCenter Content business application integrations solutions separate from the business applications themselves.

AXF is provided in two versions:

- **AXF for BPM**

  The newer AXF for BPM infrastructure takes advantage of the application development and customization capabilities provided by Oracle technologies such as BPM Suite (Oracle Business Process Management Suite), ADF (Oracle Application Development Framework), MDS (Metadata Repository), and Business Rules, to create a set of reusable and configurable business components for use in customizing WebCenter Content business application integration solutions.

- **AXF for BPEL**

  The older AXF for BPEL infrastructure relies on AXF database tables (WebCenter Imaging tables) as the basis for configuring AXF solutions, commands, and web tools, implemented and customized by a solution developer or solution accelerator.

  Note: Both AXF for BPM and AXF for BPEL are included in Imaging installation. The BPEL version is automatically included with Imaging and the BPM version is available for selection.

An **AXF solution** is a micro-application whose components are created using either AXF infrastructure. Solutions are installed on top of a base configuration of core AXF and business application files standard to solutions that use AXF functionality.

Both **Imaging solutions** (the BPM imaging solution and the BPEL imaging solution) provide user interface components that form a task list that lists assigned human workflow tasks and a task detail area where users acquire, complete, and release human workflow tasks.

You can use either AXF structure with the **Managed Attachments solution**. (If using both solutions, it is recommended that you use the same AXF version.)

1.2 About the BPM Imaging Solution

This section covers the following topics:

- Section 1.2.1, "Business User View of the BPM Imaging Solution"
- Section 1.2.2, "About the Business Tier Agility Administrator Role"
- Section 1.2.3, "About AXF Commands Used by the BPM Imaging Solution"
- Section 1.2.4, "About BPM Imaging User Interface Components"

1.2.1 Business User View of the BPM Imaging Solution

From a business user's perspective, the integration is virtually seamless. End-users use the BPM Imaging solution to:

- Launch the BPM imaging solution from their business application (or via an email notifying them of an assigned task), and select and perform workflow tasks. For example, users processing tasks might select a custom button, link, or menu command integrated into their business application, initiating the following processes:
  - A SOAP request is generated and sent to AXF, passing the selected command along with additional parameters such as an AXF solution (SalesQuoteEntry), an AXF command (StartSalesQuoteEntry), and a user name.
  - AXF returns an OPEN_BROWSER command with the URL to launch, the AXF Solution Workspace.
  - The business application displays the BPM Views, Task List, and Task Details in the AXF Solution Workspace page in a new browser window.
- The user selects a task from the task list, and its details display in the solution application area. The user reviews and keys entries in the business application and/or the task detail area while viewing images and related values.

- The business user performs actions related to the workflow task, such as routing, canceling, updating, and completing tasks.

1.2.1.1 Imaging Scenario 1: Entering Sales Quotes

The HelloBPM solution installed with AXF provides an example imaging reference solution based on sales quote entry. It provides a non-production example for use in confirming installation and system operation and as the basis for portions of the documentation. It works as follows:

1. A workflow process automatically generates user tasks.

   A sales quote is uploaded, data values are assigned, and a task for processing the quote is generated. Typically, tasks are pooled into views from which groups of users select. A user may have access to tasks in multiple views.

2. From the business application, the user launches the imaging solution, by clicking a button or link or selecting a command called **StartSalesQuoteEntry**.

3. The user selects a task from those listed for a selected assignee. Once a task is selected (acquired), it is no longer available to other users.

4. In the solution application, users view the task's image, key entries based on the image, and perform related commands.

   Additional action commands are provided in top portion of the solution application. Users might route the task to another user or user group for approval, add comments for others to view, or route the task for account management.

5. Users complete the task and begin another, if desired.

> **Note:** For details about imaging solution user tasks, see the *Oracle WebCenter User's Guide for Application Adapters*.

1.2.2 About the Business Tier Agility Administrator Role

This administrator configures the solution to match particular business needs by making changes to business rules in Solution Administration. For example, this administrator can add or change task actions, configure navigation to control pages, configure page components, change data and table display and editing, add validations and prompts, and add or change tabs. After saving, changes are immediately visible in the user interface (Section 5.2.3).

Business tier agility administrators must have access to WebCenter Content Solution Administration, which provides access to the Business Rule Editor. (These administrators must be assigned to the axfadmin group in Oracle WebLogic Server.) They must also have an understanding of Oracle Business Rules, the BPM processes, and their payload. For more information, see Chapter 5.

1.2.3 About AXF Commands Used by the BPM Imaging Solution

The reusable AXF commands allow you to implement the functionality described below. For information about these commands, including their parameters and example implementations, see Section A.3.

BPM AXF Command	Description
Open Solution Workspace	Returns an instruction to the requester to open a browser with a URL to the AXF solution workspace. See Section A.3.1.

BPM AXF Command	Description
Update Task	Updates the BPM task referenced in the AXF conversation associated with the AXF request. Updates BPM task payload data given the arguments passed in the request and signals the AXF solution application. See Section A.3.2.
Document Search	Retrieves documents stored in the document repository system for display in the viewer. A URL is configured in AXF configuration which specifies the search URL with optional parameters that can be passed from the business application and appended to the search URL string. See Section A.3.3.

1.2.4 About BPM Imaging User Interface Components

This section covers the following topics:

- Section 1.2.4.1, "About the Solution Workspace"
- Section 1.2.4.2, "About the Solution Application"

1.2.4.1 About the Solution Workspace

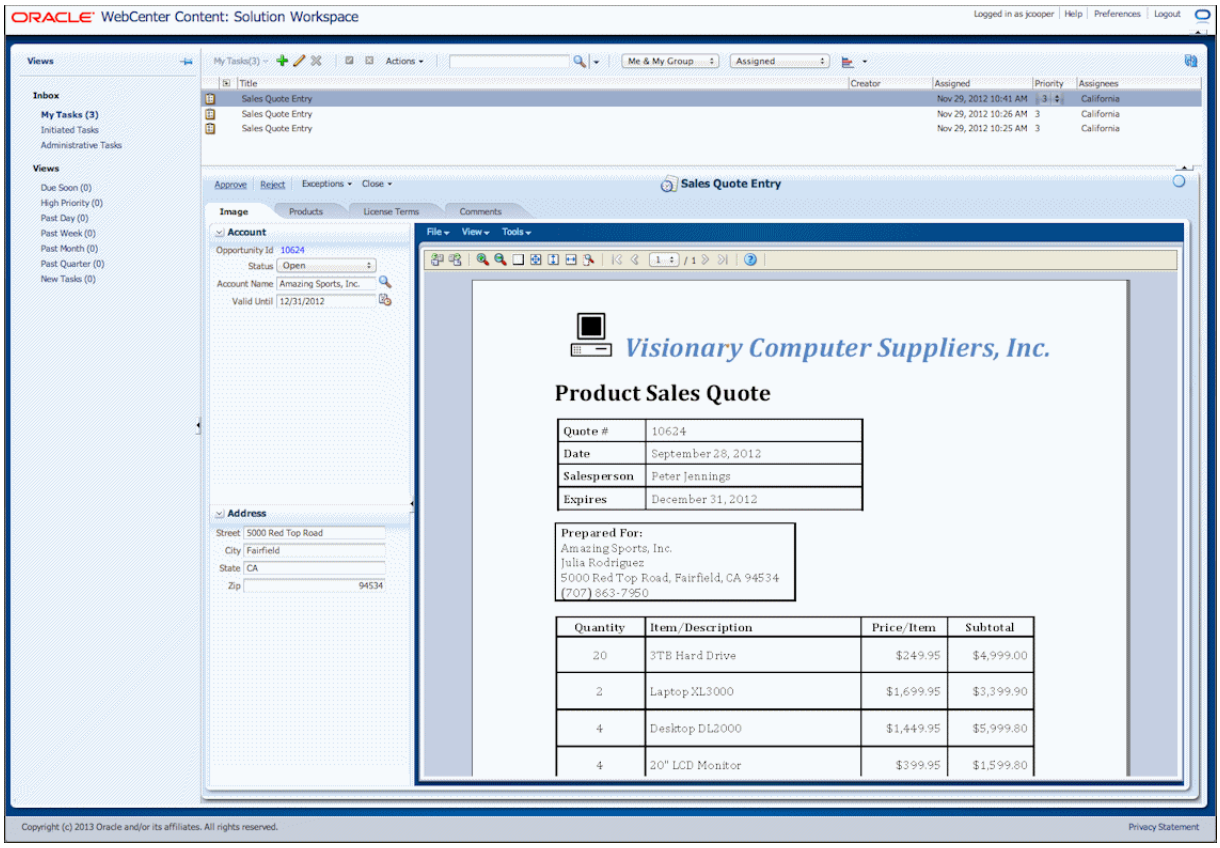
The AXF for BPM infrastructure provides a configurable user interface through its **solution workspace**. This workspace comprises the entire window shown in Figure 1–1, and surfaces a list of BPM tasks, views, and other interface components to allow users to manage and complete assigned tasks.

- The solution workspace provides the primary entry point into all solutions.

  Users typically access the workspace through their Line of Business integration (for example, through Oracle E-Business Suite or Oracle PeopleSoft), or through an email link sent from BPM.

- The solution workspace displays the human tasks assigned to the user in an interface that contains the following panes shown in Figure 1–1:
  - **Views**, which displays BPM views for the user's work context.
  - **Task List**, which displays a list of assigned BPM tasks from which the user selects. For configuration information, see Section 5.5.
  - **Solution Application**, which displays the user interface representing a specific human task's details. For more information, see Section 1.2.4.2.

Figure 1–1 Solution Workspace



1.2.4.2 About the Solution Application

The solution application uses a Dynamic Page Template common to all pages, that contains the elements described below and shown in Figure 1–2. The user interface and template functionality are driven by business rules contained in dictionaries.

- **Action control** (or menu bar), which extends the full width of the solution application at its top. It contains actions that users can perform on the selected task. For example, actions links might enable users to approve, reject, route, or close the task.

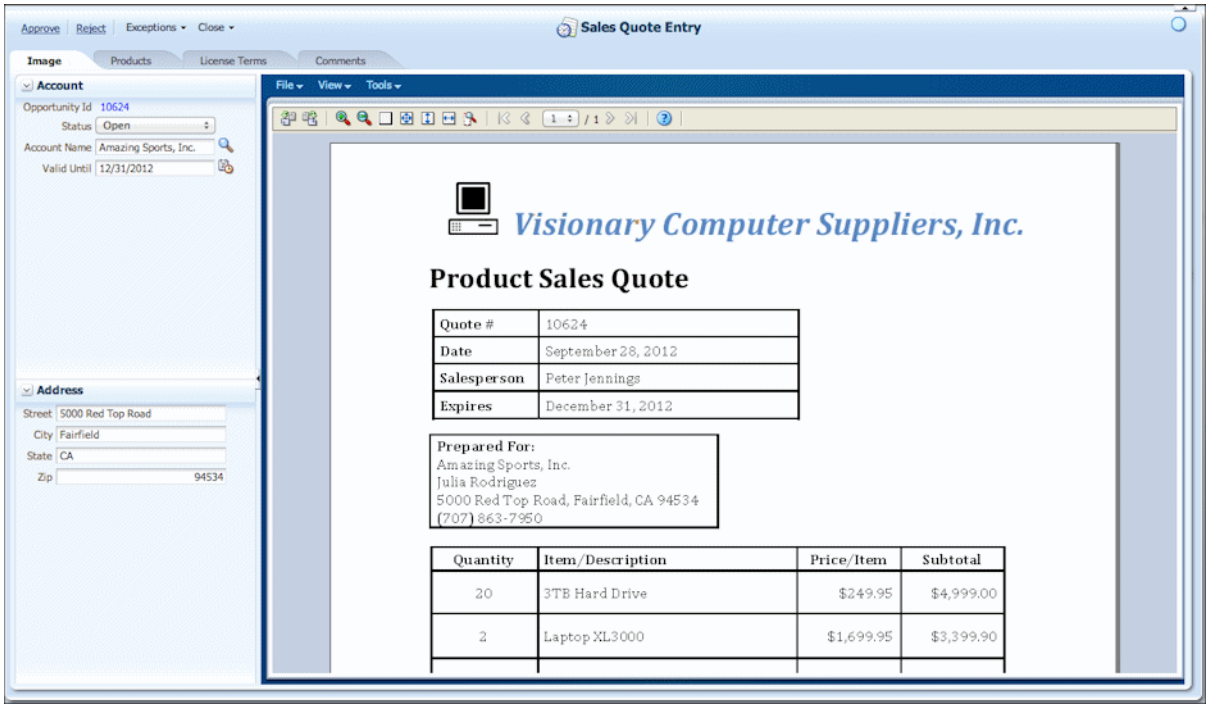
  The actions that display in this area depend on configuration, such as whether a **context** is used that displays certain actions to some user roles and not others.

- **Dynamic tabs**, which extend below the action control, and where business users perform the bulk of the tasks.

  The tabs that display in this area depend on configuration, such as whether a **context** is used and which **page components** are implemented.

Administrators edit the solution application using Oracle Business Rules, as described in Chapter 5. The solution's **functional blocks** define its functional units of work.

Figure 1–2 Solution Application



1.3 About the BPEL Imaging Solution

A workflow Imaging solution is an integration between a business application and BPEL (Oracle BPEL Process Manager), using Imaging as the imaging source. Through an AXF configuration, business users can process associated images and perform document-centric workflow tasks from their business application user interface.

This section covers the following topics:

- Section 1.3.1, "Business User View of the BPEL Imaging Solution"
- Section 1.3.2, "About AXF Commands Used by the BPEL Imaging Solution"
- Section 1.3.3, "About BPEL Imaging User Interface Components"

1.3.1 Business User View of the BPEL Imaging Solution

From a business user's perspective, the integration is virtually seamless. End-users use the Imaging solution to:

- View attached images and metadata values. Use Imaging's tools for viewing, annotating, and redacting images, as permissions allow.
- Key entries in the business application while viewing images and related values in the Imaging viewer.
- Perform actions related to the workflow task, such as routing, canceling, updating, and completing tasks.
- Scan or upload supporting documents for a selected business application record.
- View supporting images for a business application record without leaving the business application.

1.3.1.1 Imaging Scenario 1: Processing Invoices

A BPEL Imaging Solution configured for invoice processing might work as follows:

1. A workflow process automatically generates user tasks.

   An invoice is uploaded, metadata values are assigned, and a task for processing the invoice is generated. Typically, tasks are pooled into profiles from which groups of users select. A user may have access to tasks in multiple profiles.

2. From the business application, the user launches the Imaging Solution, by clicking a button or link or selecting a command called **Process Invoices**.

3. The user selects a task from those listed for a selected profile. Once a task is selected (acquired), it is no longer available to other users.

4. In the Task Viewer, users view the task's invoice image, key entries in the business application based on the image, and perform related commands.

   Additional action commands are typically provided in a side panel. Users might route the task to another user or user group for approval, add comments for others to view, skip the task, or re-scan or delete the task's document.

5. Users complete the task and begin another, if desired.

   Most often, changes users make in the business application are synchronized with Imaging, and vice versa.

> **Note:** For details about Imaging Solution user tasks, see the *Oracle WebCenter User's Guide for Application Adapters*.

1.3.1.2 Imaging Scenario 2: Capturing Supporting Employee Documents

An Imaging Solution configured for capturing supporting documents might work as follows:

1. From the business application, a user retrieves a record such as an employee record.

2. The user launches the document imaging solution, by clicking a link or button or selecting a command called **Scan Employee Document**.

3. Oracle Distributed Document Capture launches and automatically initiates a scan (if a scanner is attached to the desktop) or enables the user to upload electronic images from desktop.

4. The user enters index values (metadata) in Oracle Distributed Document Capture to store with the images.

5. The user clicks **Send**, which transmits the captured document images and their metadata from Oracle Distributed Document Capture to Imaging.

1.3.1.3 Imaging Scenario 3: Viewing Supporting Employee Documents

An Imaging Solution configured for viewing supporting documents might work as follows:

1. From the business application, a user retrieves a record such as an employee record.

2. A user launches the document imaging solution, by clicking a button or link or selecting a command called **View Employee Documents**.

3. From the list of documents associated with the employee record and their metadata values, the user selects a document.
4. The document displays in the Imaging viewer, where the user can view its images, and with appropriate permissions, apply annotations or redactions.

1.3.2 About AXF Commands Used by the BPEL Imaging Solution

The reusable AXF commands allow you to implement the functionality described below. For information about these commands, including their parameters and example implementations, see Section B.3.

BPEL AXF Command	Description
Open Task	Displays the AXF Task Viewer web page and claims a human workflow task. See Section B.3.1.
Autotask	Displays autotask mode, in which a new human workflow task is automatically claimed in the AXF Task Viewer without displaying the Task List. See Section B.3.2.
Release Task	Releases a human workflow task, and, if configured, executes additional actions. See Section B.3.3.
Complete Task	Completes a human workflow task, and, if configured, updates BPEL payload attribute values and executes additional actions. See Section B.3.4.
Redirect	Redirects the current AXF web page to any URL or AXF web page specified in the configuration. See Section B.3.5.
Terminate Conversation	Used by an external client to terminate a conversation with AXF. (This command does not include parameters.)
Update Task	Updates BPEL payload field values on a specified human task or values in the XML payload using XPATH. See Section B.3.6.
Update Task From Procedure	Calls a stored procedure using a specified data source and updates values in the BPEL payload using XPATH. See Section B.3.7.
Validate Task	Used to validate BPEL system attribute data or BPEL payload data using the Regular Expression language, and based on validation results, execute a subsequent command. See Section B.3.9.

Note: You can also deploy custom commands and chained commands to execute through AXF. See Section 6.4.8 and Section 6.4.9.

1.3.3 About BPEL Imaging User Interface Components

The BPEL Imaging Solution provides the following user interface components. These are web interface components displayed to users and configured through the AXF tables. Once populated, you can modify the components using Imaging's Solution Editor, as described in Section B.2.1.

- Section 1.3.3.1, "About the Task List"
- Section 1.3.3.2, "About the Task Viewer"
- Section 1.3.3.3, "About the Enumeration Picker"
- Section 1.3.3.4, "About the Identity Picker"

- Section 1.3.3.5, "About Comments"

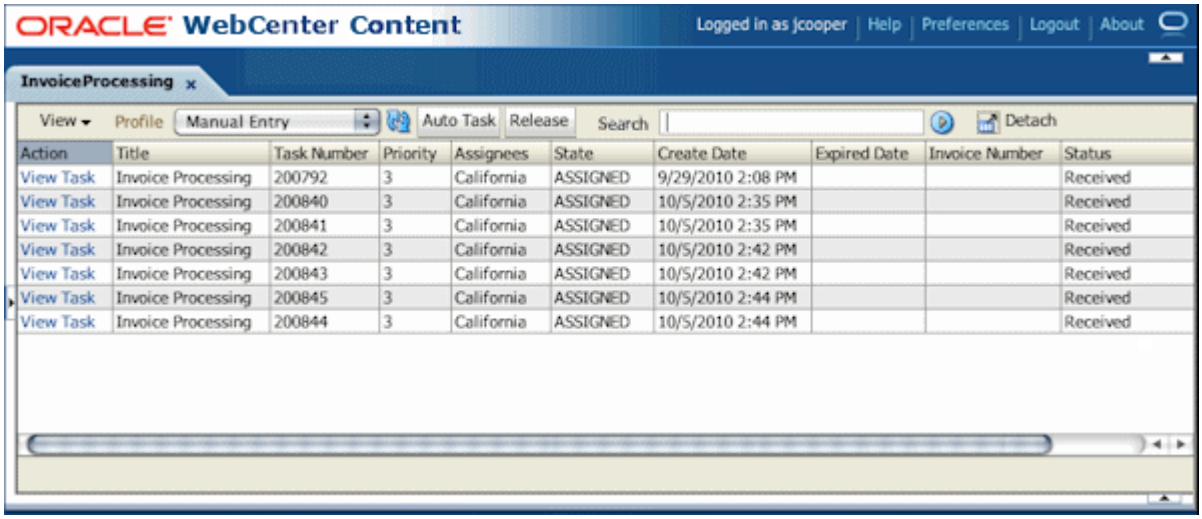
1.3.3.1 About the Task List

The Task List web page shown in Figure 1–3 displays a list of available tasks to users. It interacts with the AXF infrastructure and BPEL to display the list using views configured in the BPEL Worklist application.

> **Note:** Use the BPM Worklist application to create views and share them with other users or groups.

For configuration information, see Section B.2.2.

Figure 1–3 Task List Web Tool



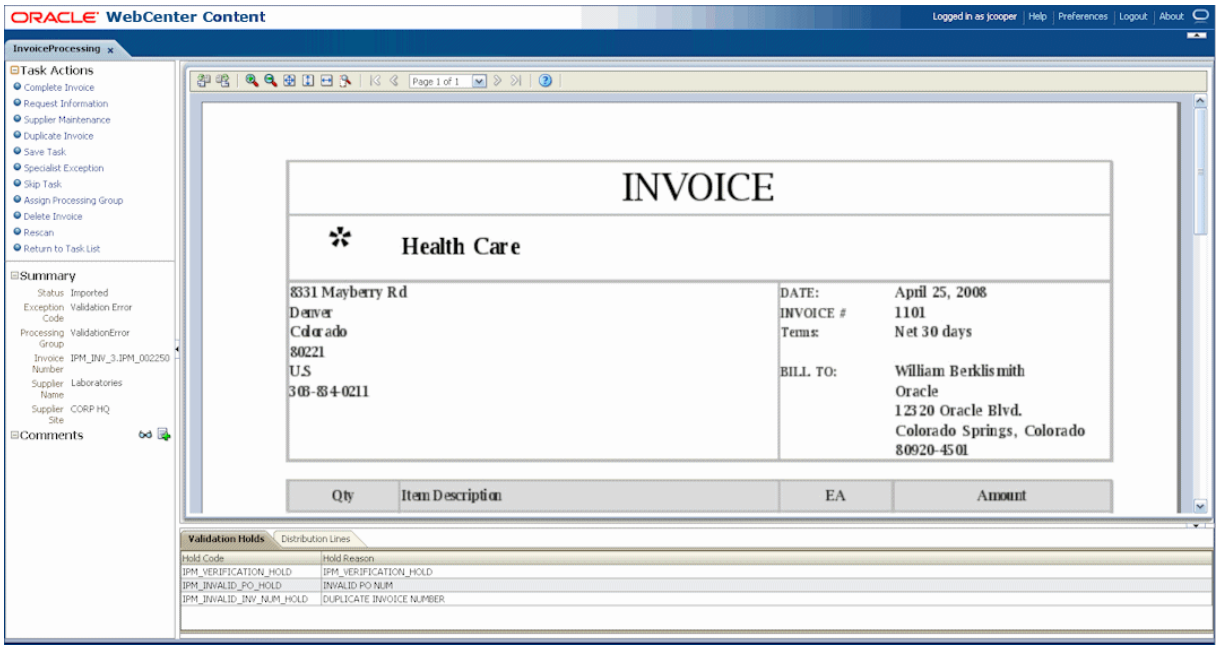
The Task List web tool can also display a list of AXF actions, using AXF action commands.

1.3.3.2 About the Task Viewer

The Task Viewer web page shown in Figure 1–4 displays images and metadata values through interaction with the AXF infrastructure, BPEL, Imaging, and the business application. It also typically displays a side menu containing AXF action commands. It may also include a summary section with metadata values, a comments section, and a dynamic data table.

For configuration information, see Section B.2.3.

Figure 1–4 Task Viewer Web Tool

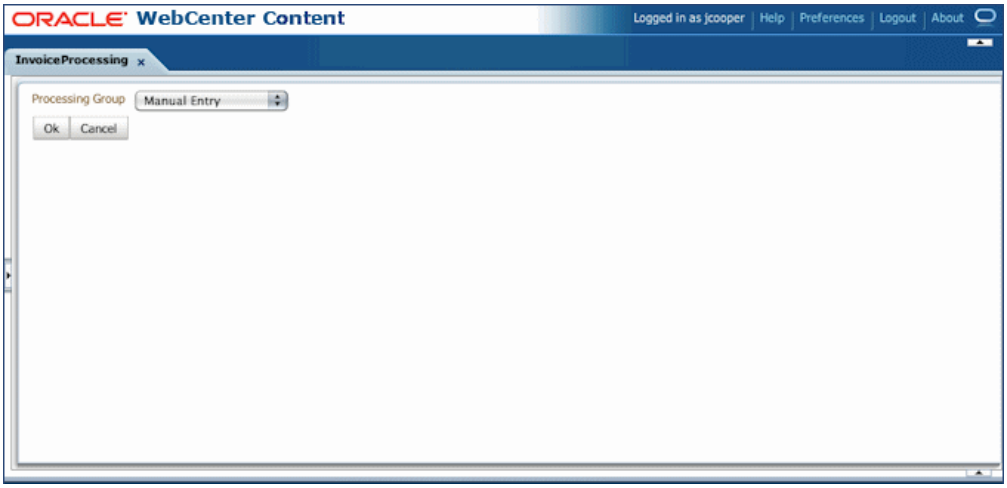


1.3.3.3 About the Enumeration Picker

The Enumeration Picker web page shown in Figure 1–5 enables users to select from a list of enumerated values.

For configuration information, see Section B.2.4.

Figure 1–5 Enumeration Picker Web Tool

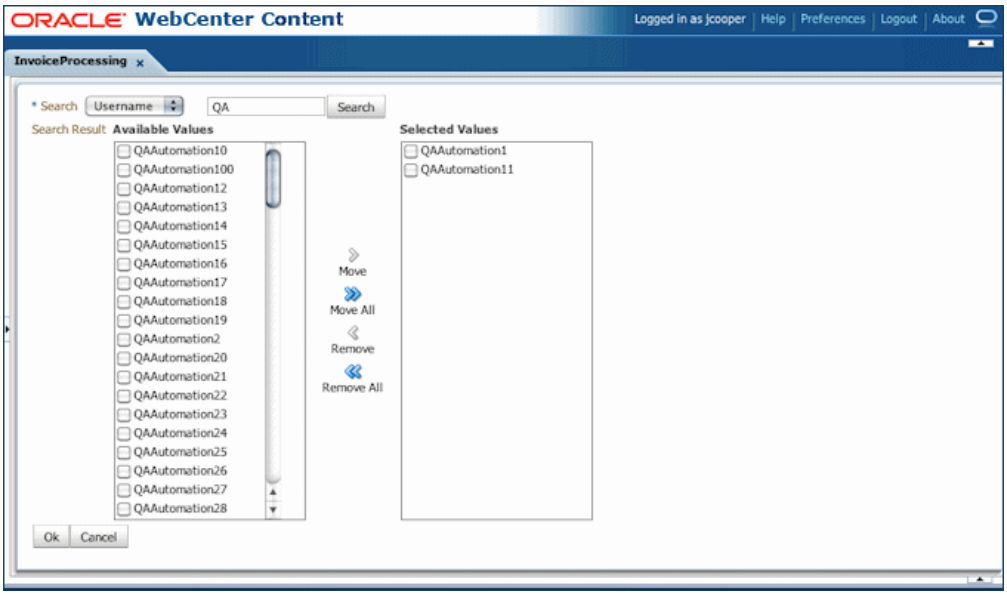


1.3.3.4 About the Identity Picker

The Identity Picker web page shown in Figure 1–6 enables users to select one or more users or groups from an identity store configured for BPEL. After the user chooses an identity, AXF typically performs a related, configurable action. Most likely, it assigns or delegates a task to the selected user or group of users. For example, a business user who encounters a problem with a transaction might select an exception handler to send the transaction to, after entering a comment that describes the problem.

For configuration information, see Section B.2.5.

Figure 1–6 Identity Picker Web Tool



1.3.3.5 About Comments

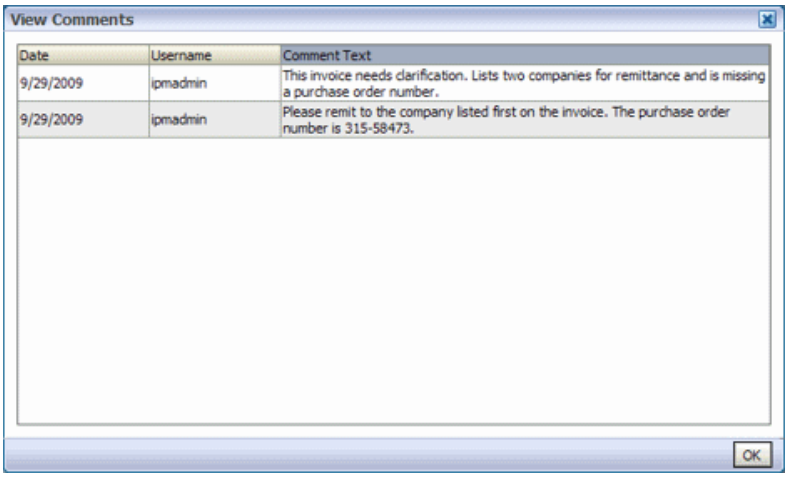
The Comments web page shown in Figure 1–8 enables users to enter comments related to the human task during the transaction's processing. Administrators can display comments on any AXF web page that shows data from a single task, such as the Task Viewer shown in Figure 1–4. Comments persist for the entire process, allowing users to view and add comments. BPEL workflow task saves comments using its native comments capabilities.

To view comments, users click the **View Comments** icon on the AXF web page, as shown below and in Figure 1–4. To add a comment, users click the **Add Comment** icon.

Figure 1–7 View Comments Icon (left), Add Comment Icon (right)

For configuration information, see Section B.2.3.3 and Section B.2.3.1.2.

Figure 1–8 Comments Web Tool



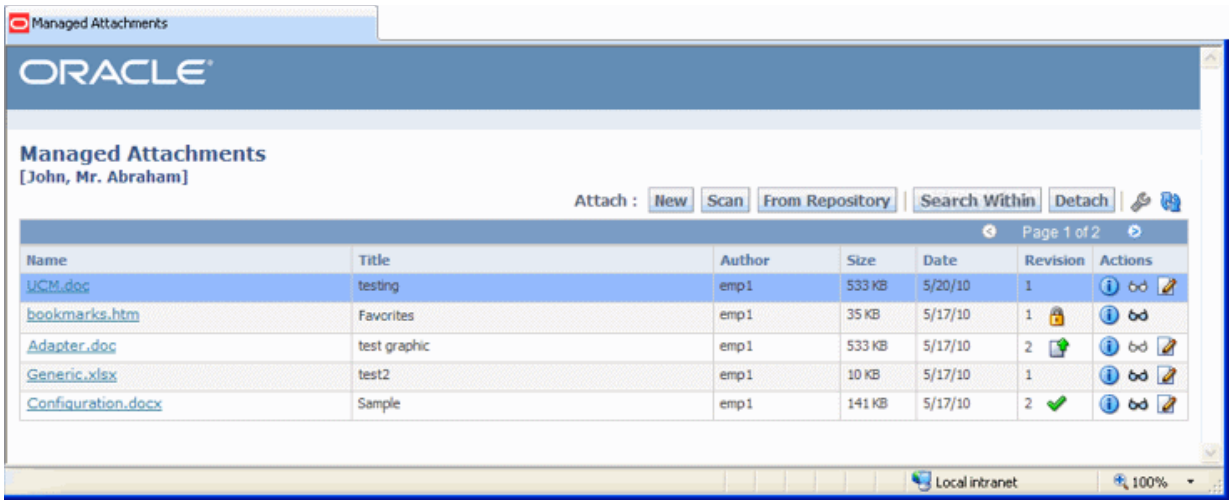
1.4 About the Managed Attachments Solution

With this solution, Content Server documents are listed as managed attachments to business application entities in a customizable window launched from a link, button, or menu, depending on business application configuration. For example, a business application user displays an Employee record, invokes the Managed Attachment functionality by clicking a button, and attaches a passport image and supporting identity documents.

> **Note:** In addition to applications built using Oracle E-Business Suite Forms, Oracle Applications Framework (OAF), and PeopleSoft, the Managed Attachments solution can be integrated with other business applications using the Enterprise Application Adapter Framework, as described in Section 7.4.

The Managed Attachments Solution, shown in Figure 1–9, enables business users to attach, scan, and retrieve attachments stored in a Content Server repository. An Oracle WebCenter Content Server repository enables users throughout an enterprise to view, collaborate on, and retire content, ensuring that content is secure, accurate, and up-to-date.

Figure 1–9 Managed Attachments Page



This section covers the following topics:

- Section 1.4.1, "Business User View of the Managed Attachments Solution"
- Section 1.4.2, "About the AXF Command and Service Used By the Attachments Solution"

1.4.1 Business User View of the Managed Attachments Solution

Business users can perform these tasks in the Managed Attachments solution, as an extension to their business application environment:

- Check in new documents to Content Server and attach them to the selected business application entity
- Scan and import documents using Oracle Distributed Document Capture, attaching them to the selected business application
- Open documents in their native application, Web-viewable format, or AutoVue viewer (if configured)
- Detach documents from the selected business application entity
- Search Content Server and attach documents to the selected business application entity from the Content Server repository
- Check out documents, locking them to changes by other users
- Modify a document's metadata values
- If configured, view a previous attached revision and update the attachment revision
- View an attached document's information

Note: For details about Managed Attachments Solution user tasks, see the *Oracle WebCenter User's Guide for Application Adapters*.

1.4.1.1 Attachments Scenario 1: Capturing Identity Documents

A Managed Attachments solution configured for capturing employee identity documents such as passports might work as follows:

1. From the business application, a user retrieves a record such as an employee record.
2. The user launches the Managed Attachments solution, by selecting a command or clicking a link or button called **Scan Employee Document**.
3. Oracle Distributed Document Capture launches and automatically initiates a scan (if a scanner is attached to the desktop) or enables the user to upload electronic images from desktop.
4. The user selects attachment settings, clicks **Scan Document**, then enters index values (metadata) in Oracle Distributed Document Capture to store with the images.
5. The user clicks **Send**, which transmits the captured document images and their metadata from Oracle Distributed Document Capture to Content Server.
6. The user clicks the **Refresh** button to see the newly scanned or imported document listed as attached to the selected employee record.

1.4.1.2 Attachments Scenario 2: Viewing and Editing Identity Documents

A Managed Attachments solution configured for viewing employee identity documents might work as follows:

1. From the business application, the user retrieves an employee record.
2. The user launches the Managed Attachments solution, by selecting a command or clicking a link or button called **View Employee Documents**.
3. The Managed Attachments page displays, listing attachments for the employee record.
4. An attachment may show its revision number, and if it was checked out by the user or another user. If revision-specific mode has been configured and an older revision is attached, the user can either view the older attached revision or view the newer revision and make it the new attached version.
5. The user selects an attached document's **Edit** icon. As the user makes updates to the document, it is checked out to the user and other users cannot edit it. When the user checks in the revised document, its revision number is incremented.
6. The document displays in Content Server (or the AutoVue viewer), where the user can view its images, and with appropriate permissions, apply annotations or redactions.

1.4.2 About the AXF Command and Service Used By the Attachments Solution

The Managed Attachments solution uses an AXF solution namespace called **UCM_Managed_Attachments** to orchestrate user requests from the business application to display the Managed Attachments user interface for the Content Server repository.

- When the business application user activates the Managed Attachments command from a business application entity, the adapter makes an AXF request. The solution supports a single AXF command namespace called **UCM_Managed_Attachments**, which invokes the AF_GRANT_ACCESS command.
- The **AF_GRANT_ACCESS** command implements the **AF_GRANT_ACCESS** service. This Content Server service temporarily grants a user who has logged into the business application access to all Content Server documents associated with the selected business application entity and to which the user has security group access. This service also returns an AXF response containing a Managed

Attachments URL to invoke the Content Server **attachments framework search**. This framework search lists all documents associated with the business application entity.

- With the returned URL, the adapter opens the Managed Attachments browser window for the business application user, also displaying key values (AFLabel) for the business application entity with which the attachment list is associated, and optionally passing in metadata values from the business application to Content Server.

How Content Server Access is Granted to the Business Application User

The adapter uses a temporary authorization mechanism for managed attachments access. Communicating through a trusted RIDC mechanism, AXF invokes the AF_GRANT_ACCESS service with the application entity and user information needing authorization. The AF_GRANT_ACCESS service grants access to the user for the specified period, then ends the user session.

1.5 About Adapter Components

This section describes the main components of the application adapters in the following topics:

- Section 1.5.1, "Business Application Plug-In"
- Section 1.5.2, "AXF Infrastructure"
- Section 1.5.3, "AXF User Interface Components"
- Section 1.5.4, "Content Server Components (Managed Attachments)"
- Section 1.5.5, "AXF Solution Accelerators"

1.5.1 Business Application Plug-In

This portion consists of a business application plug-in: Oracle E-Business Suite, Oracle PeopleSoft, both, or another business application.

- **Oracle E-Business Suite:** Configuring the Oracle E-Business Suite plug-in consists of running PL/SQL scripts against the Oracle E-Business Suite database that populate tables and configure functionality. See Chapter 2. AXF-related Oracle E-Business Suite tables enable AXF components on Oracle E-Business Suite forms or pages.
  - For **Forms** configurations, you also upload files to the Oracle E-Business Suite system that integrate custom actions with pre-existing Oracle E-Business Suite Forms.
  - For **OAF** configurations (available for Managed Attachments only), you import files to the MDS repository and add a Managed Attachments button via OAF page personalization.
- **PeopleSoft:** Configuring the Oracle PeopleSoft plug-in consists of importing the Oracle PeopleSoft project, configuring the Integration Broker to communicate with AXF, setting user roles, and enabling AXF functionality on Oracle PeopleSoft pages. See Chapter 3. AXF-related Oracle PeopleSoft tables enable AXF components on Oracle PeopleSoft pages.
- **Other business application:** A generic component called the Enterprise Application Adapter Framework allows you to customize the Managed

Attachments solution for use with a business application other than Oracle E-Business Suite or Oracle PeopleSoft. For more information, see Section 7.4.

1.5.2 AXF Infrastructure

The AXF integration infrastructure employs a service layer that handles requests from business application integration adapters. The infrastructure is included in Oracle WebCenter Content installation, and AXF tables are automatically created during installation.

The AXF infrastructure contains the following components:

- Solution Mediator
- AXF Request /Response
- Command Mediator
- AXF Commands
- Conversation

1.5.2.1 Solution Mediator

The solution mediator receives AXF requests from the business application integration adapters. It implements a simple AXF request/response protocol to delegate abstract functional requests from the business application integration adapters to AXF command components that implement business functionality. The solution mediator is responsible for facilitating those services that are common across all AXF request/response, including invoking the Command Mediator to service the AXF request, managing the Conversation, and general error handling.

1.5.2.2 AXF Request /Response

The AXF request provides a functional request, defined by a solution namespace (representing a work thread, such as DocumentProcessing) and a command namespace (representing a service, such as StartProcessing). The AXF request may include optional arguments. The AXF response contains instructions to be performed by the requester as well as the Conversation ID (CID) used to synchronize activity between the business application and AXF solution application. The business application integration adapter stores the returned CID and includes it in subsequent requests within the same solution namespace.

1.5.2.3 Command Mediator

The command mediator maps logical functional requests to AXF commands based on the AXF request's solution namespace and command namespace.

1.5.2.4 AXF Commands

AXF commands implement business functionality. Commands may consume arguments provided by the request and return instructions to the requester through the AXF response.

- The BPM Imaging solution uses several AXF commands, as described in Section 1.2.3.
- The BPEL Imaging solution uses multiple AXF commands, as described in Section 1.3.2.

- The Managed Attachments solution uses a single AXF command that implements Content Server services that temporarily display and provide access to documents associated with a business application entity. See Section 1.4.2.

1.5.2.5 Conversation

The conversation maintains the shared state between the business application and AXF solution applications. The conversation is referenced by both applications by the CID returned to the business application in the AXF response and provided as a parameter to the AXF solution application.

1.5.3 AXF User Interface Components

The adapter provides user interface components for the imaging solution, such as a task list and task detail (that includes subcomponents such as enumeration control, identity browser, metadata display, and comments).

- For the **BPM imaging solution**, user interface components (solution workspace and solution application) are provided through the User Interface Library. Business agility tier administrators customize the solution's behavior by modifying BPM business rules. See Section 1.2.2 and Section 1.2.4.
- For the **BPEL imaging solution**, user interface components (task list and task viewer) are configured through the AXF (WebCenter Imaging) database tables. Solution developers customize the solution using Imaging's Solution Editor or by modifying the AXF tables. See Section 1.3.3.

1.5.4 Content Server Components (Managed Attachments)

For the Managed Attachments solution, the adapter includes Content Server components bundled in ZIP files that you install and enable using Component Manager. They include the AppAdapter framework core files. See Section 7.3.

1.5.5 AXF Solution Accelerators

Oracle provides implementation accelerators for specific functions, such as automating invoice and receipt processing using workflows with associated approval rules, data entry forms, and reports.

> **Note:** To obtain a solution implementation accelerator, contact your systems integrator, Oracle Consulting, or Oracle Support.

1.6 About Adapter Security and Authentication

The adapters provide multiple security and authentication levels, which vary depending on the solution (BPM Imaging, BPEL Imaging, or Managed Attachments) and the business application, as identified in Table 1–1.

Table 1–1 Security Controls for the Adapter

Authentication/ Security Level	Business Applications	Supported Solutions	Description	Configuration Information
Web Service Authentication (WSS/SOAP)	Oracle E-Business Suite and Oracle PeopleSoft	BPM Imaging, BPEL Imaging, and Managed Attachments	Occurs when web service calls are made to AXF. The solutions support user authentication against the AXF solution mediator web services using a user name token security installed on the application server on which AXF resides. ▪ In Oracle E-Business Suite, set user name token authentication using an AXF_SOAP_POLICY key in the properties table (AXF_PROPERTIES or OAF_AXF_PROPERTIES). ▪ In Oracle PeopleSoft, configure the Integration Broker node that communicates to AXF with user name token authentication. For the BPM imaging solution, the *solution workspace* is secured via WebLogic Server security. You can avoid requiring multiple sign-ons between the business application and the *solution application* by enabling single sign-on (SSO).	For Oracle E-Business Suite, see Section 2.4.1. For Oracle PeopleSoft, see Section 3.2.2.1.
SSL	Oracle E-Business Suite and Oracle PeopleSoft	BPM Imaging, BPEL Imaging, and Managed Attachments	For Oracle E-Business Suite, configure SSL settings through the AXF_CONFIGS and AXF_PROPERTIES tables (Forms) or OAF_AXF_CONFIGS and OAF_AXF_PROPERTIES tables (OAF). For Oracle PeopleSoft, configure SSL settings via Integration Broker. For both Oracle E-Business Suite and Oracle PeopleSoft, configure web server-based digital certificates on the business application and Oracle WebCenter Content side.	For Oracle E-Business Suite, see Section 2.4.2. For Oracle PeopleSoft, see Section 3.2.1.
AXF to Content Server	Oracle E-Business Suite and Oracle PeopleSoft	Managed Attachments	Enable trusted communication between the host on which AXF is running and the Content Server.	See Section 7.7.4.
Document Security	Oracle E-Business Suite and Oracle PeopleSoft	Managed Attachments	Users specify private or shared access to a document upon attachment (check-in to Content Server). ▪ Users can access private documents only through the Managed Attachments page. ▪ Any Content Server user with the document's security group access can access shared documents.	See Section 7.7.3.

1.7 Adapter System Requirements

The adapter is composed of optional solutions installed over a base configuration of AXF, Imaging, and Content Server files standard to adapters that use AXF functionality. General requirements for the adapter are listed in this section, and supported configuration and certification information is available at:

http://www.oracle.com/technetwork/middleware/webcenter/content/documentation/documentation-155348.html

Notes: When beginning configuration, be sure to check MyOracleSupport for the latest patches to apply.

General requirements include:

- Oracle E-Business Suite
- Oracle PeopleSoft Enterprise
- SOA 11g (Imaging Solutions Only)
- Oracle BPM Suite (BPM Imaging Solution Only)
- Additional Managed Attachments Requirements

1.7.1 Oracle E-Business Suite

- A fully functioning Oracle E-Business Suite system. The Oracle E-Business Suite application adapter supports both Oracle E-Business Suite Forms (Imaging and Managed Attachments) and OAF web pages (Managed Attachments only). See the certification matrix for supported releases. Also see the Oracle E-Business Suite requirements in Section 7.1.
- For Forms applications, Oracle E-Business Suite Forms Builder is required for .PLL compilation. Specific releases are listed in the matrix.
- To avoid duplicate logins, Oracle Single Sign-On or Oracle Access Manager is required. For more information, see the *Oracle Fusion Middleware Application Security Guide*.

> **Note:** If using Oracle Single Sign-On, an Oracle Identity Management (OIM) / Oracle Internet Directory (OID) Server is required.

1.7.2 Oracle PeopleSoft Enterprise

- A fully functioning Oracle PeopleSoft system, including PeopleTools. See the certification matrix for supported releases.
- To avoid duplicate logins, Oracle Single Sign-On or Oracle Access Manager is required. For more information, see the *Oracle Fusion Middleware Application Security Guide*.

> **Note:** If using Oracle Single Sign-On, an Oracle Identity Management (OIM) / Oracle Internet Directory (OID) Server is required.

1.7.3 SOA 11g (Imaging Solutions Only)

For either imaging solution, SOA 11*g* is required, along with available patch sets. A BPEL server instance is also required. BPEL is part of SOA 11*g*.

> **Note:** SOA 11*g* is not needed if configuring the Managed Attachments Solution only.

Note: Specific version mapping between Oracle WebCenter Content (formerly Oracle Enterprise Content Management) and SOA Suite is required. Refer to the following certification information at:

http://www.oracle.com/technetwork/middleware/webcenter/content/documentation/documentation-155348.html

1.7.4 Oracle BPM Suite (BPM Imaging Solution Only)

If configuring AXF for BPM, Oracle Business Process Management (Oracle BPM) is required. Oracle BPM is part of BPM Suite. Oracle BPM Suite is layered on the Oracle SOA Suite.

1.7.5 Additional Managed Attachments Requirements

The Managed Attachments Solution has additional requirements, listed in Section 7.1.

The Managed Attachments solution is supported on WebSphere Application Server when Content Server is running on WebSphere Application Server and Application Extension Framework (Imaging Server) is running on Oracle WebLogic Server.

1.8 Adapter Configuration Roadmap

This section lists the major configuration steps for the adapter, after installation.

1. **If configuring the adapter for Oracle E-Business Suite, configure the Oracle E-Business Suite plug-in.**

Tasks	Refer To
Configure the Oracle E-Business Suite database by running PL/SQL scripts.	Section 2.1
Configure one or both Oracle E-Business Suite plug-ins: ■ For Forms configuration, compile Oracle E-Business Suite Forms by modifying PLL files. ■ For OAF configuration, add the Managed Attachments button to OAF pages via personalization.	■ Section 2.2 (Forms) ■ Section 2.3 (OAF)
Secure Oracle E-Business Suite to AXF communications via SOAP security and SSL.	Section 2.4
Configure Oracle E-Business Suite logging.	Section 2.5

2. **If configuring the adapter for Oracle PeopleSoft, configure the PeopleSoft plug-in.**

Tasks	Refer To
Import the Oracle PeopleSoft integration project.	Section 3.1
Configure Integration Broker to communicate with AXF securely and configure AXF access for Oracle PeopleSoft users.	Section 3.2
Place AXF components on Oracle PeopleSoft pages.	Section 3.3
Configure Oracle PeopleSoft logging.	Section 3.4

3. **Configure AXF settings.**

Tasks	Refer To
Configure AXF logging.	Section 4.1 or Section 4.2

4. If configuring the BPM Imaging solution, complete these steps.

Tasks	Refer To
Run the HelloBPM solution or apply a solution accelerator. Use the solution to verify BPM imaging functionality.	Section 5.2.1
Use Solution Administration links to configure the solution application. Manage business rules and contexts, and use the command driver to test and verify configurations such as task actions, dynamic tabs, contexts, and control pages.	Section 5.2
Use the HelloBPM solution and its use cases to learn business rule modification.	Section 5.3 and Section 5.4
Configure BPM views. (The HelloBPM solution provides BPM views configured for a single administrator user.)	Section 5.5
System administrators modify the configuration in Enterprise Manager, as needed.	Section 5.6

5. If configuring the BPEL Imaging solution, complete these steps.

Tasks	Refer To
Configure the BPEL connection.	Section 6.1
Configure the AXF tables or apply a solution implementation accelerator.	Section 6.2
After initial configuration, use Imaging's Solution Editor to edit user interface components such as the Task List and Task Viewer.	Section B.2.1
Test functionality using the AXF driver page.	Section 6.3
Customize Imaging functionality as needed.	Section 6.4

6. If configuring the Managed Attachments solution, complete these steps.

Tasks	Refer To
Ensure your configuration meets additional Managed Attachments system requirements.	Section 7.1
Run Managed Attachments installation scripts.	Section 7.2
Configure Content Server components.	Section 7.3
If configuring Managed Attachments for a business application other than Oracle E-Business Suite or Oracle PeopleSoft, configure the Enterprise Application Adapter Framework.	Section 7.4
Configure Managed Attachments variables.	Section 7.5
Customize Managed Attachments functionality as needed.	Section 7.6
Configure authentication and security.	Section 7.7
Configure document viewing with AutoVue (optional).	Section 7.8
Configure document scanning/importing with Oracle Distributed Document Capture (optional).	Section 7.9

2

Configuring the Oracle E-Business Suite Plug-In

This chapter describes how to configure the Oracle E-Business Suite plug-in for use by WebCenter adapter solutions, including configuring the Oracle E-Business Suite database, compiling Oracle E-Business Suite Forms, and configuring Oracle Application Framework (OAF) pages via personalization. It also addresses securing communications through SOAP security and SSL.

Configuring the Oracle E-Business Suite portion of AXF requires an active connection to the Oracle E-Business Suite database, general database experience, and knowledge of Oracle E-Business Suite Forms Builder and OAF page personalization.

> **Note:** If installing adapter solutions for Oracle PeopleSoft but not Oracle E-Business Suite, skip this chapter and complete the steps described in Chapter 3, "Configuring the Oracle PeopleSoft Plug-In."

This chapter covers the following topics:

- Section 2.1, "Configuring the Oracle E-Business Suite Database"
- Section 2.2, "Compiling Oracle E-Business Suite Forms"
- Section 2.3, "Configuring Oracle Application Framework (OAF) Pages Via Personalization"
- Section 2.4, "Securing Adapter Communications for Oracle E-Business Suite"
- Section 2.5, "Configuring Oracle E-Business Suite Logging"
- Section 2.6, "Setting User Locales"
- Section 2.7, "Configuring Solutions for Oracle E-Business Suite"
- Section 2.8, "Uninstalling AXF From Oracle E-Business Suite"

2.1 Configuring the Oracle E-Business Suite Database

The adapter uses an Oracle E-Business Suite database to store PL/SQL procedures and Oracle E-Business Suite configuration information. Consult your local DBA for assistance with these tasks. The instructions in this section assume the use of SQL*PLUS, but you can use any tool capable of querying the Oracle Database.

- Section 2.1.1, "Creating the AXF Oracle E-Business Suite Configuration Schema User (AXF)"
- Section 2.1.2, "PL/SQL Procedures Provided"

- Section 2.1.3, "Running the PL/SQL Scripts"

2.1.1 Creating the AXF Oracle E-Business Suite Configuration Schema User (AXF)

Follow these steps to create a database user for use by AXF within the Oracle E-Business Suite database.

> **Note:** If using Oracle E-Business Suite 11i and Oracle Database 9i, you must install the Oracle XML Database XDB Package into the Oracle E-Business Suite database. (This requirement applies only when using Oracle E-Business Suite 11i and Oracle Database 9i.) For information about installing the Oracle XML Database XDB Package, see the Oracle Database 9i documentation.

1. Create a user named `AXF`.

   A system account user name and password is required to create the user. For assistance creating the user, contact your DBA.

2. Assign the configuration schema user the following access privileges:
   - Create table
   - Create sequence
   - Create type
   - Create public synonym
   - Create session
   - Create procedure
   - Unlimited tablespace

2.1.2 PL/SQL Procedures Provided

Table 2–1 lists the PL/SQL scripts to run. In some cases, separate Oracle E-Business Suite release 11 and 12 scripts are provided. Scripts whose names begin with **OAF** apply to OAF configurations.

It is recommended that you run all scripts for your Oracle E-Business Suite release as described in Section 2.1.3 regardless of whether you are configuring Forms and/or OAF applications.

Table 2–1 PL/SQL Procedures Provided With Oracle E-Business Suite Plug-In

PL/SQL Procedure	Applies To	Description
AXF_CREATE_TABLES_SYNONYM OAF_AXF_CREATE_TABLES_SYNONYM	Forms OAF	These scripts create the tables and synonyms used by AXF for Forms and OAF applications. In addition, OAF_AXF_CREATE_TABLES_SYNONYM.sql populates the OAF_AXF_PROPERTIES table.
AXF_EBS_PROPERTIES_DATA	Forms	Populates the AXF_PROPERTIES table with security information for various Oracle E-Business Suite calls to AXF.
AXF_APPS_INIT	Forms	Required for Oracle E-Business Suite attachment for Forms applications.

Table 2–1 (Cont.) PL/SQL Procedures Provided With Oracle E-Business Suite Plug-In

PL/SQL Procedure	Applies To	Description
AXF_ADD_EBS_ATTACHMENT_PROC_R11 AXF_ADD_EBS_ATTACHMENT_PROC_R12	Forms	Creates a stored procedure that adds an attachment to a transaction in Oracle E-Business Suite Forms applications.
AXF_MANAGED_ATTACHMENT_DATA AXF_MANAGED_ATTACH_AVAIL AXF_MANAGED_ATTACH_VALUES	Forms	Required for document attachment, for Forms applications.
AXF_SOAP_CALL_PROC	Forms OAF	This script creates a stored procedure to make SOAP calls from PL/SQL.

2.1.3 Running the PL/SQL Scripts

Follow these steps to configure the Oracle E-Business Suite database.

1. Locate the scripts in the following folders. Separate folders are provided for Oracle E-Business Suite Forms releases 11 and 12.
   - For Oracle E-Business Suite Forms:

     MW_HOME/*ECM_HOME*/axf/adapters/ebs/R11/

     MW_HOME/*ECM_HOME*/axf/adapters/ebs/R12/
   - For Oracle E-Business Suite OAF:

     MW_HOME/*ECM_HOME*/axf/adapters/ebs/oaf/
2. Using SQL*PLUS, log in to the Oracle E-Business Suite database as the AXF Oracle E-Business Suite configuration schema user (AXF).

   This user was previously created, as described in Section 2.1.1.
3. As the AXF user, execute the **AXF_CREATE_TABLES_SYNONYM** script from the applicable Oracle E-Business Suite location.

   To execute the script, enter:

   ```
   @AXF_CREATE_TABLES_SYNONYM.sql
   ```

   Verify that the following tables were created: AXF_COMMAND_PARAMETERS, AXF_COMMANDS, AXF_CONFIGS, AXF_PROPERTIES, and AXF_FND_MAP.
4. As the AXF user, customize and execute the **OAF_AXF_CREATE_TABLES_SYNONYM** script from the applicable Oracle E-Business Suite location.

   Before executing the script, set the script's `AXF_OAF_REDIRECT_PAGE_URL` value so that it identifies the Oracle E-Business Suite system's hostname and port (typically 8000).

   To execute the script, enter:

   ```
   @OAF_AXF_CREATE_TABLES_SYNONYM.sql
   ```

   Verify that the following tables were created: OAF_AXF_CMD_PARAMS, OAF_AXF_CMDS, OAF_AXF_CONFIGS, and OAF_AXF_PROPERTIES.

> **Note:** The AXF_CREATE_TABLES_SYNONYM and OAF_AXF_CREATE_SYNONYM scripts insert rows into the AXF_PROPERTIES and OAF_AXF_PROPERTIES tables, respectively. By default, the AXF_SOAP_SECURITY security parameter is set to TRUE. For SOAP security to work, you must store the SOAP user's password in the FND_VAULT. See Section 2.4 for information on storing the SOAP user name and password in the Oracle E-Business Suite FND_VAULT.

5. As the AXF user, execute the **AXF_EBS_PROPERTIES_DATA** script from the applicable Oracle E-Business Suite location.

   To execute the script, enter:

   ```
   @AXF_EBS_PROPERTIES_DATA.sql
   ```

6. Log in as the APPS user.

7. As the APPS user, execute the **AXF_APPS_INIT** script from the applicable Oracle E-Business Suite location.

   Execute the script by entering:

   ```
   @AXF_APPS_INIT.sql
   ```

8. As the APPS user, execute the **AXF_ADD_EBS_ATTACHMENT_PROC_R12** or **AXF_ADD_EBS_ATTACHMENT_PROC_R11** script from the applicable Oracle E-Business Suite location.

   Execute the script by entering the command appropriate for your version:

   ```
   @AXF_ADD_EBS_ATTACHMENT_PROC_R12.sql
   ```

   ```
   @AXF_ADD_EBS_ATTACHMENT_PROC_R11.sql
   ```

   > **Note:** This compilation may result in warnings, which you can ignore.

9. As the APPS user, execute the **AXF_MANAGED_ATTACH_AVAIL**, **AXF_MANAGED_ATTACH_VALUES**, and **AXF_MANAGED ATTACHMENT_DATA** scripts from the applicable Oracle E-Business Suite location.

   Execute the scripts by entering:

   ```
   @AXF_MANAGED_ATTACH_AVAIL.sql
   ```

   ```
   @AXF_MANAGED_ATTACH_VALUES.sql
   ```

   ```
   @AXF_MANAGED_ATTACHMENT_DATA.sql
   ```

10. As the APPS user, execute the **AXF_SOAP_CALL_PROC** script from the applicable Oracle E-Business Suite location.

    Execute the script by entering:

    ```
    @AXF_SOAP_CALL_PROC.sql
    ```

    > **Note:** This compilation may result in warnings, which you can ignore.

2.2 Compiling Oracle E-Business Suite Forms

To enable a seamless integration of custom actions with existing Oracle E-Business Suite Forms, certain files are uploaded to the Oracle E-Business Suite system for the adapters. The adapters use user interface .PLL extension modules to access workflow tasks (Imaging Solution) and documents associated with business records (both solutions).

The *Custom.PLL* module is slightly modified during configuration to call AXF functions. It notifies AXF each time an Oracle E-Business Suite event occurs, allowing AXF to determine if it relates to AXF functionality.

The *AXF_Custom.PLL* component performs the following functions:

- Calls out to a web service to execute an AXF Command
- Responds to the following AXF response commands:
  - Open Browser
  - Terminate Conversation
- Renders menus to expose AXF functionality based on the Oracle E-Business Suite configuration database.

Follow these steps to copy the AXF_CUSTOM.pld file, convert it to an AXF_CUSTOM.pll file, make modifications, and then compile it to an AXF_CUSTOM.plx file.

Note: To compile AXF_CUSTOM.pll using an alternative command line method (applicable for Oracle E-Business Suite 12 Forms use only), see Section 2.2.1.

Note: For information on using Oracle Forms Builder, see its Oracle E-Business Suite documentation.

---

1. For the applicable version listed below, copy the **AXF_CUSTOM.pld** file to the E-Business Server (to **FORMS_PATH** for Oracle E-Business Suite 12, or **FORMS60_PATH** for Oracle E-Business Suite 11).

   Oracle E-Business Suite 12: *MW_HOME/ECM_HOME*`/axf/adapters/ebs/R12/AXF_CUSTOM.pld`

   Oracle E-Business Suite 11: *MW_HOME/ECM_HOME*`/axf/adapters/ebs/R11/AXF_CUSTOM.pld`

   Note: If you are using a Linux/UNIX system and copied the .PLD from a Windows system, issue the `dos2unix` command before converting it below.

   ---

2. Open Oracle Forms Builder and connect to the Oracle E-Business Suite database as the APPS user. Forms Builder is typically located in the `/bin/` subdirectory of your database's Oracle home.

Note: Be sure to connect to the Oracle E-Business Suite database. If you fail to connect, verify the tnsnames.ora file.

3. In Forms Builder, convert `AXF_CUSTOM.pld` to `AXF_CUSTOM.pll`.
   - In Oracle E-Business Suite 12, select **File**, then **Convert**.
   - In Oracle E-Business Suite 11, select **File**, then **Administration**, then **Convert**.

   Select **PL/SQL libraries** and **Text to binary** while converting the file.

   Note: If the following error displays during conversion of AXF_CUSTOM.pld to AXF_CUSTOM.pll, repeat this step until the file successfully converts.

   PDE-PLI038 - Can not open file as a PL/SQL Library

   Note: If the following error displays during conversion, click OK repeatedly until the file successfully converts.

   PDE-PLI018 - Could not find library AXF_CUSTOM

4. From the File menu, open `AXF_CUSTOM.pll`.
   - In Oracle E-Business Suite 12, select **Program**, then **Compile pl/sql**, then **All**.
   - In Oracle E-Business Suite 11, select **Program**, then **Compile**, then **All**.
5. Compile `AXF_CUSTOM` into a module (`.plx`).
   - In Oracle E-Business Suite 12, select **Program**, then **Compile Module**.
   - In Oracle E-Business Suite 11, select **File**, then **Administration**, then **Compile File**.

   Notes:
   - You must compile AXF_CUSTOM using the APPS schema user ID.
   - If you encounter the following identifier or other errors referencing objects in APPCORE.pll while compiling, this indicates that the `APPCORE.pll` file must be attached to your form:

     `'APP_SPECIAL.ENABLE' must be declared` (a).

6. Select **File** then **Connect** and ensure that you are connected to the database as the APPS user.
7. Back up the `CUSTOM.pll` file.

   WARNING: Modifications to CUSTOM.pll are modifications to the Oracle E-Business Suite infrastructure. Ensure that you have appropriately backed up this file before making changes.

8. Open CUSTOM.pll by selecting **File**, then **Open** and selecting **PL/SQL Libraries** (*.pll) in the Files of Type field. After opening the file and expanding Program Units, right-click the custom package body of CUSTOM.pll and select pl/sql editor.

9. In the body text of CUSTOM.pll, modify the following text formatted in bold italics for the solutions you are configuring. If the file contains other customizations, place the following modification after the existing code inside each function/procedure.

   - *For Managed Attachments Only or Both Solutions:*

     ```
     function zoom_available return boolean is
     begin

     -- Required for ALL integrations
     return true;
     end zoom_available;
     ```

   - *For Imaging Solution Only:*

     ```
     function zoom_available return boolean is
     begin

     -- Required for ALL integrations
     return AXF_CUSTOM.zoom_available();
     end zoom_available;
     ```

     Important: Be sure to modify the body text of the pll, NOT its header. Scroll down until you reach the following comment header:

     ```
     - -Real code starts here
     ```

10. In the body text of CUSTOM.pll, modify the following text formatted in bold italics. If the file contains other customizations, place the following modification after the existing code inside each function/procedure.

    ```
    procedure event(event_name varchar2) is
    begin

    -- Required for AXF integrations
    AXF_CUSTOM.event(event_name);
    null;

    end event;
    ```

    Note: Do not remove the `null;` line.

11. Save CUSTOM.pll by selecting **File**, then **Save**.

12. With CUSTOM.pll open, determine if AXF_CUSTOM is listed as an attached library.

    - If it is listed, highlight AXF_CUSTOM and click the minus (-) symbol to detach it. Then reattach AXF_CUSTOM by highlighting Attached Libraries under CUSTOM and clicking the plus (+) symbol; browse to AXF_CUSTOM.pll and select it.

- If it is not listed, attach AXF_CUSTOM by highlighting Attached Libraries under CUSTOM and clicking the plus (+) symbol; browse to AXF_CUSTOM.pll and select it.

When prompted to remove the path, click **Yes**.

13. With `CUSTOM.pll` open, verify that APPCORE and APPCORE2 are listed as attached libraries to AXF_CUSTOM.pll. If not listed, attach them. If listed, detach and attach them.
14. With `CUSTOM.pll` open, select **Program**, then **Compile pl/sql,** then **All** (Oracle E-Business Suite 12) or **Program**, then **Compile**, then **All** (Oracle E-Business Suite 11).
15. Compile CUSTOM into a module (.plx) by selecting **Program**, then **Compile Module** (Oracle E-Business Suite 12) or **File**, then **Administration**, then **Compile File** (Oracle E-Business Suite 11).
16. Save all before exiting Forms Builder. Verify that the Zoom menu command displays in the appropriate Oracle E-Business Suite forms.

2.2.1 Compiling AXF_CUSTOM.pll Using the Command Line

This section describes an alternative command line method for compiling Oracle E-Business Suite Forms for adapter use. It applies only to Oracle E-Business Suite Suite 12 Forms use only, and does not apply to Oracle E-Business Suite 11.

Follow the steps below to copy the AXF_CUSTOM.pld file, convert it to an AXF_CUSTOM.pll file, make modifications, and then compile it to an AXF_CUSTOM.plx file. Then follow the steps in Section 2.2 as indicated to modify the other file, CUSTOM.pll, and complete the Forms configuration.

1. Copy the AXF_CUSTOM.pld file to a UNIX system and run the **dos2unix** command to convert the file to UNIX format.

   As an alternative, open the file in vi editor and manually remove the ^M characters from the file.
2. Enter the following command line to convert AXF_CUSTOM.pld to AXF_CUSTOM.pll:

   ```
   $APPL_TOP/au/12.0.0/resource>frmcmp_batch.sh module=AXF_CUSTOM.pld
   userid=apps/apps output_file=AXF_CUSTOM.pll module_type=library parse=yes
   ```
3. Enter the following command line to convert AXF_CUSTOM.pll to AXF_CUSTOM.plx:

   ```
   $APPL_TOP/au/12.0.0/resource>frmcmp_batch.sh module=AXF_CUSTOM.pll
   userid=apps/apps output_file=AXF_CUSTOM.plx module_type=library compile_
   all=special
   ```
4. Modify the CUSTOM.pll file by following the steps in Section 2.2, *starting with step 6*. Complete the remaining steps.

2.3 Configuring Oracle Application Framework (OAF) Pages Via Personalization

Managed Attachments functionality for OAF pages works as follows:

- You manually add a Managed Attachments button to the OAF page using OAF personalization, positioning the button in a custom flex region imported into the MDS repository.
- The button is associated with a generic extended controller class which redirects the call from Oracle E-Business Suite to a custom intermediate OAF page, passing the required Oracle E-Business Suite mapping entities.
- The controller class analyzes which OAF page called it. The OAF page's key identifiers (name, view object, primary key attributes, and labels) are stored in the database tables. Based on the OAF page calling the controller, the solution fetches the appropriate rows from the database tables along with the required business entity values, which it adds as parameters to the custom intermediate OAF page's URL.
- When the user clicks the Managed Attachments button, the custom intermediate OAF page opens, its ProcessRequest method is called, and the SOAP packet is created. The SOAP call fetches the Content Server URL, and upon loading, the page is redirected to the Content Server URL.

> **Note:** In addition to configuring OAF pages for Managed Attachments functionality, you can also provide imaging access on OAF pages, as described in Section 2.3.3.

To configure the Oracle E-Business Suite OAF plug-in, follow the instructions in these sections:

- Section 2.3.1, "Compiling Java Files, Copying Class Files, and Importing XML Files"
- Section 2.3.2, "Personalizing OAF Pages to Add Managed Attachments"
- Section 2.3.3, "Configuring Other Services on OAF Pages (Optional)"

2.3.1 Compiling Java Files, Copying Class Files, and Importing XML Files

Follow these steps to compile Java files, copy class files, and import XML files to the MDS repository, to enable adding the Managed Attachments button to OAF pages. In Section 2.3.2, you will add the button to a custom flex region from the MDS repository.

> **Note:** This procedure assumes you are familiar with using JDeveloper to compile Java files. For basic information, see the section on building your first program with JDeveloper 10g at:
>
> `http://download.oracle.com/otndocs/tech/ias/portal/files/RG/OBE/FirstStep/FirstStep.htm`

1. Verify that you are using the correct JDeveloper version for your Oracle E-Business Suite release. At `http://support.oracle.com`, search for the following articles to verify that your JDeveloper version includes the needed OAF patch. If you do not have the correct version, download and unzip the appropriate JDeveloper OAF patch file.
   - Doc ID 416708.1, How to find the correct version of JDeveloper to use with Oracle E-Business Suite 11i or Release 12.x
   - Doc ID 416708.1, Troubleshooting JDeveloper setup for Oracle Applications

2. Start JDeveloper, and create an application. Within the new application, create a project.
3. In the following location, locate the java files **GenericUcmButtonCO.java** and **RedirectToAxfCO.java** and add them to the project:

   $MiddlewareHome/OracleHome/axf/adapters/ebs/oaf/src/oracle/apps/ak/ucm/axf/webui

   Choose **Add Project Content**, then click the **Add** button and select the folder containing the Java files.
4. In the project properties, add the following OAF libraries:
   - BC4J Client
   - BC4J Runtime (Note: Do NOT add BC4J *EJB* Runtime)
   - Cabo
   - FND Libraries
5. Compile the Java files in the project (right-click the project and choose **Rebuild**).
6. Copy the class files (find **Output Directory**, which has the generated classes, via project properties) to the specified Oracle E-Business Suite mid tier $JAVA_TOP directory, creating the required directory structure.
   - Class files to copy: **GenericUcmButtonCO.class** and **RedirectToAxfCO.class**
   - Oracle E-Business Suite Server directory to copy class files to:

     $JAVA_TOP/oracle/apps/ak/ucm/axf/webui
7. Import the XML file for the Generic Flex Layout Region into the MDS Repository.
   - XML file to import: **ManagedAttachmentButtonRN.xml**
   - Directory from which to execute the command (might be $JDEV_USER_HOME, then up two levels to jdevbin):

     $JdevHome/jdevbin/oaext/bin
   - ManagedAttachmentButtonRN import command to execute (from the command prompt, replacing italicized parameters):

     ./import *Full path to ManagedAttachmentButtonRN.xml* -username *appsUsername* -password *appsPassword* -dbconnection "(description = (address_list = (address = (community = tcp.world)(protocol = tcp)(host = *DB Hostname*)(port = *DB Port*)))(connect_data = (sid = *DB SID*)))" -rootdir *path to directory where ManagedAttachmentButtonRN.xml is located* -rootPackage /oracle/apps/ak/ucm/axf/webui
   - Example ManagedAttachmentButtonRN import command (where the XML file is in the $JAVA_TOP directory):

     ./import /$JAVA_TOP/ManagedAttachmentButtonRN.xml -username apps -password password -dbconnection "(description = (address_list = (address = (community = tcp.world)(protocol = tcp)(host = *DB Hostname*)(port = *DB Port*)))(connect_data = (sid = *DB SID*)))" -rootdir /$JAVA_TOP -rootPackage /oracle/apps/ak/ucm/axf/webui

Note: If the import fails and returns an error that a jar file is missing, add the full path for the **oamdsdt.jar file** to the CLASSPATH entry in the import script (import file on Linux, import.bat on Windows). The full path for oamdsdt.jar is:

```
$JdevHome/jdevbin/oaext/lib/oamdsdt.jar
```

For example, the import file typically contains a CLASSPATH entry of the form:

```
set CLASSPATH=$JRAD_ROOT/jdev/appslibrt/uix2.jar
```

Change it to add the required jar, like this:

```
set CLASSPATH=$JRAD_
ROOT/jdev/appslibrt/uix2.jar:/oracle/r12/JDeveloper/jdevbin/
oaext/lib/oamdsdt.jar
```

8. Import the XML file for the Custom OAF page used for redirection into the MDS Repository.
   - XML file to import: **RedirectToAxfPG.xml**
   - Directory from which to execute the command:

     ```
     $JdevHome/jdevbin/oaext/bin
     ```

   - RedirectToAxfPG import command to execute (from the command prompt, replacing italicized parameters):

     ```
     ./import Full path to RedirectToAxfPG.xml -username appsUsername -password
     appsPassword -dbconnection "(description = (address_list = (address =
     (community = tcp.world)(protocol = tcp)(host = DB Hostname)(port = DB
     Port)))(connect_data = (sid = DB SID)))" -rootdir Path to directory where
     RedirectToAxfPG.xml is located -rootPackage /oracle/apps/ak/ucm/axf/webui
     ```

     Example RedirectToAxfPG import command (where the XML file is in the $JAVA_TOP directory):

     ```
     ./import /$JAVA_TOP/RedirectToAxfPG.xml -username apps -password password
     -dbconnection "(description = (address_list = (address = (community =
     tcp.world)(protocol = tcp)(host = DB Hostname)(port = DB Port)))(connect_
     data = (sid = DB SID)))" -rootdir /$JAVA_TOP -rootPackage
     /oracle/apps/ak/ucm/axf/webui
     ```

     Note: If the import fails and returns an error that a jar file is missing, see the note in step 7.

9. Verify that the import was successful, by running the following commands via SQL prompt as the Oracle E-Business Suite APPS user:

   ```
   SET SERVEROUTPUT ON
   BEGIN
   JDR_UTILS.PRINTDOCUMENT('/oracle/apps/ak/ucm/axf/webui/RedirectToAxfPG');
   JDR_
   UTILS.PRINTDOCUMENT('/oracle/apps/ak/ucm/axf/webui/ManagedAttachmentButtonRN');
   END;
   /
   ```

10. Load the **oaf_axf_messages.ldt** message bundle using FNDLOAD. This file contains the translatable messages used by the controller classes.

- Verify that the LD_LIBRARY_PATH variable is set to `$ORACLE_HOME/lib`
- As the Oracle E-Business Suite APPS user, execute the following command from $FND_TOP/bin:

```
FNDLOAD apps/$CLIENT_APPS_PWD 0 Y UPLOAD $FND_
TOP/patch/115/import/afmdmsg.lct Full path to oaf_axf_messages.ldt
```

- When prompted, enter the APPS user password.

11. Restart the Oracle E-Business Suite instance.

2.3.2 Personalizing OAF Pages to Add Managed Attachments

Perform these steps for each OAF page on which you want to display a Managed Attachment button. You create a flexible layout via personalization and add the Managed Attachment button file you previously imported into the MDS repository as flexible content.

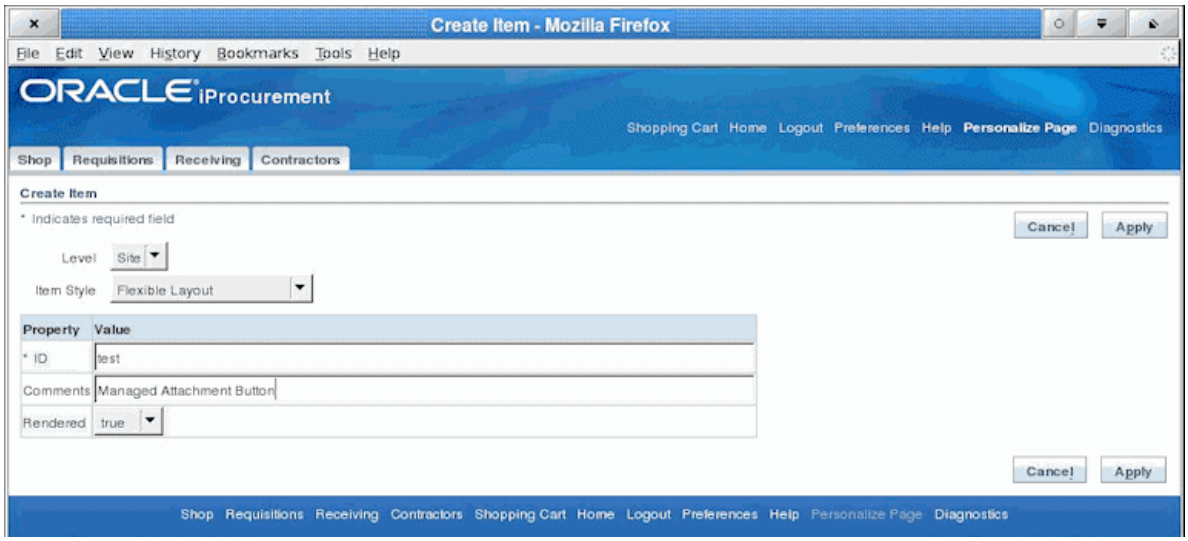
Notes: Add one Managed Attachments button per OAF page only; including multiple buttons causes the same controller code to execute multiple times.

The Managed Attachments functionality cannot be implemented for attachments for line items on OAF pages.

---

1. Log in to Oracle E-Business Suite as the OPERATIONS user.
2. Display the OAF page on which to add a Managed Attachment button, and click the **Personalize the page** link at the top of the page.
3. Under **Personalization Structure** on the Personalize Page page, select the **Complete View** option, and click the **Expand All** link. A full hierarchical view of all regions on the OAF page displays.
4. Create and configure the flexible layout.
   a. Scroll down the hierarchical listing below the region where the button will display on the OAF page and click the **Create Item** icon. A Create Item page displays, as shown in Figure 2–1.
   b. In the **Level** field, select **Site**.
   c. In the **Item Style** field, select **Flexible Layout**.
   d. In the **ID** field, enter any value (`test`, for example).
   e. Click **Apply**.

Figure 2–1 Creating the Flexible Layout



5. Create and configure the flexible content.

   a. In the hierarchical list, locate the flexible layout you just added and click its **Create Item** icon. A Create Item page displays, as shown in Figure 2–2.

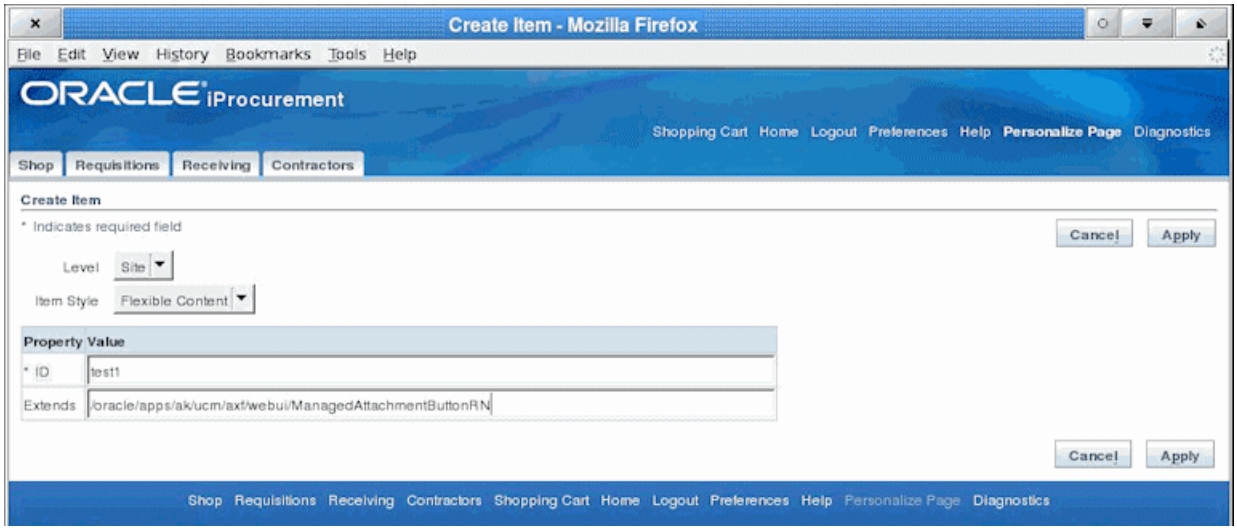
   b. In the **Item Style** field, select **Flexible Content**.

   c. In the **ID** field, enter any value (`test1`, for example).

   d. In the **Extends** field, enter the package for the flex region you imported into the MDS repository (in Section 2.3.1):

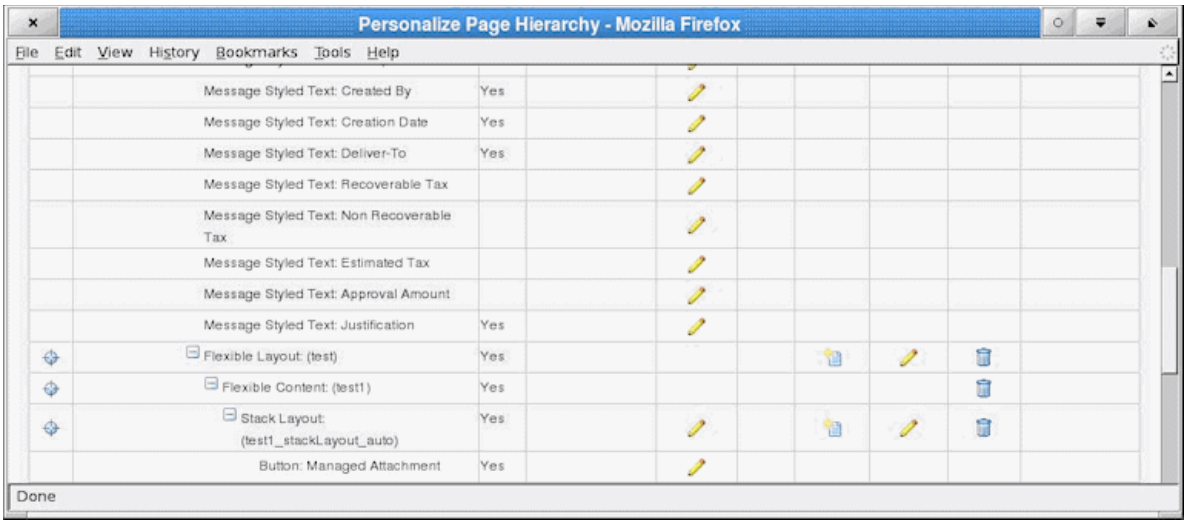
`/oracle/apps/ak/ucm/axf/webui/ManagedAttachmentButtonRN`

Figure 2–2 Creating the Flexible Content



6. Click the **Apply** button.

Figure 2–3 Page Hierarchy After Adding Managed Attachments Button Via Personalization



2.3.3 Configuring Other Services on OAF Pages (Optional)

You can provide access to the Imaging solution instead of the Managed Attachments solution on OAF pages. For example, you might include a button that allows users to access the supplier maintenance functionality available in the Imaging solution.

The procedure below provides several steps to follow to add a custom button (instead of the Managed Attachment button) and to identify the button's action in the database table as Imaging instead of Managed Attachments.

1. Complete all steps except step 7 in Section 2.3.1. (Step 7, which imports the ManagedAttachmentButtonRN.xml file, is not needed.)
2. Customize the sample XML file provided (SampleAXFButtonRN.xml), modifying the following button ID and button prompt (title) values.

   ```
   <oa:button id="AXFButton" prompt="Sample AXF Button"/>
   ```

   For example, you might change the button ID to `IPM_BUTTON` and button prompt (title) to `Supplier Maintenance`.
3. Import the XML file into the MDS Repository.
   - XML file to import: **SampleAXFButtonRN.xml** (or modified version)
   - Directory from which to execute the command:

     ```
     $JdevHome/jdevbin/oaext/bin
     ```

   - Import command to execute (from the command prompt, replacing italicized parameters):

     ```
     ./import  Full path to XML file -username apps -password appsPassword
     -dbconnection "(description = (address_list = (address = (community =
     tcp.world)(protocol = tcp)(host = DB Hostname)(port = DB Port)))(connect_
     data = (sid = DB SID)))" -rootdir path to directory where XML file is
     located -rootPackage /oracle/apps/ak/ucm/axf/webui
     ```

   - Example import command (where the XML file is in the $JAVA_TOP directory):

     ```
     ./import /$JAVA_TOP/SampleAXFButtonRN.xml  -username appsUsername -password
     appsPassword -dbconnection "(description = (address_list = (address =
     ```

```
(cUommunity = tcp.world)(protocol = tcp)(host = DB Hostname)(port = DB
Port)))(connect_data = (sid = DB SID)))" -rootdir /$JAVA_TOP -rootPackage
/oracle/apps/ak/ucm/axf/webui
```

4. Use this button ID for the command in the OAF_AXF_COMMANDS Table (Oracle E-Business Suite OAF). The new button ID will correspond to AXF_EVENT_ID. An example is shown in Section C.3.3.2.
5. In the OAF_AXF_CMD_PARAMS table, add required command parameters, if any, for the AXF_EVENT_ID.
6. Follow the steps in Section 2.3.2 to add the button imported above via personalization. In step 5 of Section 2.3.2, be sure to enter the name of the package you imported into the MDS repository in step 3 of this section. In the example provided, this name is */oracle/apps/ak/ucm/axf/webui/SampleAXFButtonRN*.

2.4 Securing Adapter Communications for Oracle E-Business Suite

This section covers the following topics:

- Section 2.4.1, "Securing Communications Through SOAP Security"
- Section 2.4.2, "Securing Communications Through SSL (Oracle E-Business Suite)"

2.4.1 Securing Communications Through SOAP Security

Follow these steps to configure SOAP security, in which the application sends the SOAP user and password in the header for authentication.

Note: Configuring SOAP security is required.

1. Ensure that SOAP security is enabled, by verifying that it is set to `TRUE` for the AXF_SOAP_SECURITY property in the AXF_PROPERTIES Table (Oracle E-Business Suite) and the OAF_AXF_PROPERTIES Table (Oracle E-Business Suite OAF).
2. Set the AXF_SOAP_POLICY property to `USER_NAME_TOKEN`.
3. Set the AXF_SOAP_USER value (for example, to `weblogic`). By default, this setting might be `AXF` and should be changed. The AXF_SOAP_USER refers to a valid Oracle WebLogic Server administrator.
4. Store the SOAP password in the database vault by executing the following command as APPS schema:

   ```
   execute fnd_vault.put('AXF','AXF_SOAP_USER','SOAP_PASSWORD');
   ```

   Where *AXF_SOAP_USER* is the SOAP user ID used in the SOAP header for authentication, and *SOAP_PASSWORD* is the SOAP password. (The first item, `'AXF'`, is an identifier that needs to be included as is.)

   For example:

   ```
   execute fnd_vault.put('AXF','weblogic','examplepassword');
   ```

5. Verify the previous command with this statement:

   ```
   select fnd_vault.get ('AXF','AXF_SOAP_USER') from dual;
   ```

   For example:

```
select fnd_vault.get ('AXF','weblogic') from dual;
```

2.4.2 Securing Communications Through SSL (Oracle E-Business Suite)

Perform the following procedures to specify Oracle E-Business Suite system settings for SSL configuration for AXF.

- Section 2.4.2.1, "Configuring the Oracle Wallet"
- Section 2.4.2.2, "Configuring the Integration for SSL"

2.4.2.1 Configuring the Oracle Wallet

1. Run Oracle Wallet Manager. In Linux, the owm executable is located at:

   ```
   //ORACLE_HOME/bin/owm
   ```

2. Create a wallet. Using the wizard, enter the required information to create a certificate. Export to a file.
3. Submit the certificate request to CA (Certificate Authority, such as Verisign) to purchase a new certificate.
4. Save the SSL certificates with a .cer extension. Most likely, the CA provided an SSL certificate, an Intermediate certificate, and a Trusted Root certificate through e-mail.
5. Import the Root and Intermediate certificates into the Oracle Wallet by right-clicking **Trusted Certificates** and importing.
6. Import the SSL certificate into the wallet manager.

   If the process is successfully completed, a Ready status displays.

   > **Tip:** To import the SSL certificate, right- click the Certificate Requested tree item if needed.

7. Save the wallet in one of the folders defined in the FORMS_PATH. (It is saved with the name *ewallet.p12*).

2.4.2.2 Configuring the Integration for SSL

For more information, see "Configuring SSL" in *Oracle Fusion Middleware Securing Oracle WebLogic Server*.

1. On the Oracle E-Business Suite database, run the SQL statements listed below.

   a. For Oracle E-Business Suite Forms applications:

   ```
   update AXF_PROPERTIES set propvalue='file:walletpath' where propname =
   'AXFWalletPath';
   update AXF_PROPERTIES set propvalue='ON/OFF' where propname = 'SecureMode';
   commit;
   ```

   b. For Oracle E-Business Suite OAF applications:

   ```
   update OAF_AXF_PROPERTIES set propvalue='file:walletpath' where propname =
   'AXFWalletPath';
   update OAF_AXF_PROPERTIES set propvalue='ON/OFF' where propname =
   'SecureMode';
   commit;
   ```

2. Update tables to reflect the SecureMode setting and secure port. When SecureMode is on, the integration attempts to connect to the AXF application using SSL (https).

   a. In the Oracle E-Business Suite AXF_CONFIGS table, update the SOLUTIONENDPOINT value to begin with `https://` if SecureMode is on, or begin with `http://` if SecureMode is off. Change *`Port`* to match the secure port on the Oracle WebCenter Content side.

   b. In the Oracle E-Business Suite OAF_AXF_CONFIGS table, update the SOLUTIONENDPOINT value to begin with `https://` if SecureMode is on, or begin with `http://` if SecureMode is off. Change *`Port`* to match the secure port on the Oracle WebCenter Content side.

   For an Imaging configuration, see Section B.4.2. For a Managed Attachments Forms configuration, see Section C.2.1. For a Managed Attachments OAF configuration, see Section C.3.1.

3. Store the Wallet password in the database vault by executing the following command as APPS schema:

   ```
   execute fnd_vault.put('AXF','AXFWalletKey','WalletPassword');
   ```

   Where *WalletPassword* is the wallet password.

4. Verify the previous command with this statement:

   ```
   select fnd_vault.get ('AXF','AXFWalletKey') from dual;
   ```

5. Import the AXF certificate into the wallet.

6. Enable SSL on Oracle WebLogic Server on the Oracle WebCenter Content side.

   For information, see "Configuring SSL for Oracle WebCenter Content Applications" in *Oracle WebCenter Content Installation Guide*.

7. Exchange server certificates between the web servers.

   This step involves exporting the certificate from Oracle Wallet and importing it into the Oracle WebLogic Server keystore on the Oracle WebCenter Content side of the integration. This keystore must be enabled for both the Administration and Imaging servers. In addition, you must export the server certificate from the Oracle Weblogic Server and import it into the Oracle Wallet.

2.5 Configuring Oracle E-Business Suite Logging

- Section 2.5.1, "Configuring Logging for Forms Applications"
- Section 2.5.2, "Configuring Logging for OAF Applications"

2.5.1 Configuring Logging for Forms Applications

Enable logging for specific forms in the AXF_CONFIGS table. For the Imaging solution, see Section B.4.2; for the Managed Attachments solution, see Section C.2.1.

To enable logging for a particular Form function, set the **LOGENABLED** field to either `1`, `YES` or `TRUE` to create the file in the UTL_FILE_DIR folder. Consult with your DBA to verify that the UTL_FILE_DIR folder is available and accessible. Log files are named *Username_MASTER_LOG.txt*, and continue to grow as items are appended.

2.5.2 Configuring Logging for OAF Applications

Follow these steps to enable diagnostics on a specific OAF page.

1. Click the **Diagnostics** link at the top of the OAF page.
2. On the Diagnostics page, select the following settings:
   a. In the **Diagnostic** field, select **Show Log on page.**
   b. In the **Log Level** field, select **Statement (1)**.
3. Click **Go**.
4. Navigate to the page with an error. Search for the string `Debug-->` on the page.

2.6 Setting User Locales

To prevent issues with different locales when invoking AXF, Oracle E-Business Suite users should set the same values for their user locale preference and their browser locale. If using the Managed Attachments Solution, set the same value for the Content Server locale.

2.7 Configuring Solutions for Oracle E-Business Suite

See these sections listed below for additional tasks to configure each solution.

For the Imaging Solution

Configure the Imaging solution as described in Chapter 6. Configure AXF-related Oracle E-Business Suite tables as described in Section B.4 or apply a solution accelerator as described in Section 6.2.

> **Note:** The Oracle E-Business Suite imaging solution is supported for Forms-based applications only, not OAF applications.

For the Managed Attachments Solution

1. Configure the Managed Attachments solution as described in Chapter 7. In addition to configuring Content Server components and running the AXF_ATTACHMENTS_CONFIG.sql script, which configures AXF elements such as solutions and commands, you must run scripts to populate the tables for Forms and OAF applications:
   a. For Forms applications, running the AXF_EBS_ATTACHMENTS_CONFIG.sql configures the Zoom menu to enable the Managed Attachments solution. For details, see Section 7.2.2.
   b. For OAF applications, running the OAF_AXF_EBS_ATTACHMENTS_CONFIG.sql script populates the tables listed in Section C.4 for the Managed Attachments solution for selected OAF pages, which you can modify for your OAF pages. For details, see Section 7.2.3.

2.8 Uninstalling AXF From Oracle E-Business Suite

Follow the steps that apply to your Oracle E-Business Suite configuration to uninstall AXF from Oracle E-Business Suite.

- Section 2.8.1, "Uninstalling From Oracle E-Business Suite Forms"

- Section 2.8.2, "Uninstalling From Oracle E-Business Suite OAF"

2.8.1 Uninstalling From Oracle E-Business Suite Forms

1. Assign the AXF configuration schema user the following privileges:
   - Drop table
   - Drop sequence
   - Drop public synonym
2. As the AXF user, execute the **AXF_DROP_TABLES_SYNONYM** script for your Oracle E-Business Suite version, from the applicable location listed below. This script drops all tables, synonyms, and sequences created by the AXF_CREATE_TABLES_SYNONYM script run during installation.

   Oracle E-Business Suite 12: *MW_HOME/ECM_HOME*/axf/adapters/ebs/R12/AXF_DROP_TABLES_SYNONYM.sql

   Oracle E-Business Suite 11: *MW_HOME/ECM_HOME*/axf/adapters/ebs/R11/AXF_DROP_TABLES_SYNONYM.sql

   Execute the script by entering:

   ```
   @AXF_DROP_TABLES_SYNONYM.sql
   ```
3. Remove the AXF database schema user.
4. Remove AXF_CUSTOM.* (AXF_CUSTOM.pll, AXF_CUSTOM.pld, and AXF_CUSTOM.plx) from `FORMS_PATH` (or `FORMS60_PATH` on Oracle E-Business Suite 11 systems).
5. Restore the CUSTOM.pll file you backed up in step 7 in Section 2.2.
6. Open Oracle Forms Builder and connect to the Oracle E-Business Suite database as the APPS user. Forms Builder is typically located in the `/bin/` subdirectory of your database's Oracle home.
7. Open the restored `CUSTOM.pll` by selecting **File**, then **Open** and selecting **PL/SQL Libraries** (*.pll) in the Files of Type field.
8. With `CUSTOM.pll` open, select **Program**, then **Compile pl/sql**, then **All** (Oracle E-Business Suite 12) or **Program**, then **Compile**, then **All** (Oracle E-Business Suite 11).
9. Compile CUSTOM into a module (.plx) by selecting **Program**, then **Compile Module** (Oracle E-Business Suite 12) or **File**, then **Administration**, then **Compile File** (Oracle E-Business Suite 11).
10. Save all before exiting Forms Builder.

2.8.2 Uninstalling From Oracle E-Business Suite OAF

1. Assign the AXF configuration schema user the following privileges:
   - Drop table
   - Drop sequence
   - Drop public synonym
2. As the AXF user, execute the **OAF_AXF_DROP_TABLES_SYNONYM** script from the location listed below. This script drops all tables, synonyms, and sequences

created by the OAF_AXF_CREATE_TABLES_SYNONYM script run during installation.

MW_HOME/ECM_HOME/axf/adapters/ebs/oaf/OAF_AXF_DROP_TABLES_SYNONYM.sql

Execute the script by entering:

```
@OAF_AXF_DROP_TABLES_SYNONYM.sql
```

3. Remove the AXF database schema user.

3

Configuring the Oracle PeopleSoft Plug-In

This chapter describes how to configure the Oracle PeopleSoft plug-in for use by one or more WebCenter application adapter solutions. It covers importing the Oracle PeopleSoft project, configuring via Integration Broker, securing communications through SSL, and placing AXF components on Oracle PeopleSoft pages.

This chapter covers the following topics:

- Section 3.1, "Importing the Oracle PeopleSoft Project"
- Section 3.2, "Configuring and Securing Adapter Communications for Oracle PeopleSoft"
- Section 3.3, "Placing AXF Components on Oracle PeopleSoft Pages"
- Section 3.4, "Configuring Oracle PeopleSoft Logging"
- Section 3.5, "Configuring Solutions for Oracle PeopleSoft"
- Section 3.6, "Uninstalling AXF From Oracle PeopleSoft"

> **Note:** If installing adapter solutions for Oracle E-Business Suite but not Oracle PeopleSoft, skip this chapter after completing the steps described in Chapter 2, "Configuring the Oracle E-Business Suite Plug-In."

3.1 Importing the Oracle PeopleSoft Project

Follow these steps to import the Oracle PeopleSoft project, which creates and populates the AXF-related Oracle PeopleSoft tables and enables AXF configuration on Oracle PeopleSoft pages:

1. As a user with administrative privileges, create a tablespace called `AXF` in the database containing the Oracle PeopleSoft schemas.

> **Note:** Oracle PeopleSoft integration does not require an additional database user, as all necessary tables are created when importing a project.

> **Note:** SQL Server automatically creates the tablespace upon importing the project. If your Oracle PeopleSoft installation uses a SQL Server, this step is not required.

2. Start the Oracle PeopleSoft Application Designer in two-tier mode and log in as the user with administrative privileges.

   Note: If your Oracle PeopleSoft system already has an AXF_PS_INTEGRATION project installed, delete the existing project in Oracle PeopleSoft Application Designer before proceeding to the next step. To delete it, choose **File**, then **Delete**. Enter AXF_PS_INTEGRATION for the project name and delete it. Then click **Save All** in Oracle PeopleSoft Application Designer.

3. From the menu, choose **Tools**, then **Copy Project**, then **From File**. The Copy From File dialog box displays.
4. Locate the following directory.

   MW_HOME/ECM_HOME/axf/adapters/psft/

5. Select the **psft** directory, then select the **AXF_PS_INTEGRATION** directory and click **Open**.

   The second AXF_PS_INTEGRATION directory contains the project XML files, but you must select its parent directory (of the same name) to access them from the Oracle PeopleSoft Application Designer. The AXF_PS_INTEGRATION project displays in the lower pane of the page.

6. Click **Select**, select all definition types, and click **Copy**. The project files begin copying.
7. After all files have copied, choose **Build**, then **Project** from the menu. The Build page displays.
8. Select the **Create Tables** and **Execute SQL Now** options, and click **Build**. You can monitor the build status as the files are imported into Oracle PeopleSoft.
9. Once done, check the PSBUILD.LOG file to verify that the project imported successfully.
10. If the log file shows *tablespace 'AXF' does not exist* errors, the tablespace was not created properly. Manually create a tablespace called `AXF` in the same database you logged into when starting the Oracle PeopleSoft Application Designer, then return to step 5 to rebuild the project.
11. Click **Save All** in Oracle PeopleSoft Application Designer.

3.2 Configuring and Securing Adapter Communications for Oracle PeopleSoft

Oracle PeopleSoft adapter security is configured through Integration Broker, where you select user name token authentication and then grant AXF access for Oracle PeopleSoft users.

This section covers the following topics:

- Section 3.2.1, "Securing Communications Through SSL (Oracle PeopleSoft)"
- Section 3.2.2, "Configuring Integration Broker to Communicate With AXF"
- Section 3.2.3, "Configuring AXF Access For Oracle PeopleSoft Users"

3.2.1 Securing Communications Through SSL (Oracle PeopleSoft)

Follow these main steps to implement web server SSL encryption for the Oracle PeopleSoft adapter:

1. Install web server-based digital certificates.

   Refer to the Oracle PeopleSoft documentation for an overview of securing integration environments, and outbound PeopleSoft Integration Broker security processing. In the Enterprise PeopleTools PeopleBook, Integration Broker Administration Guide, see the section on installing web server-based digital certificates.

2. Enable SSL on Oracle WebLogic Server on the Oracle WebCenter Content side.

   For information, see "Configuring SSL for Oracle WebCenter Content Applications" in *Oracle WebCenter Content Installation Guide*.

3. Exchange server certificates between the web servers.

   For information, see "exportKeyStoreObject" in *Oracle Fusion Middleware Administrator's Guide*.

3.2.2 Configuring Integration Broker to Communicate With AXF

Importing the AXF_PS_INTEGRATION project into Oracle PeopleSoft also imports the Integration Broker connection information. These components contain information needed to connect from Oracle PeopleSoft to the AXF Server. This section describes how to access this information in the Oracle PeopleSoft web client and edit it to point to the AXF Server.

For the appropriate AXF Server version, follow the steps listed to configure communication between the Integration Broker and AXF:

- Section 3.2.2.1, "Configuring Communication With an 11g AXF Server"
- Section 3.2.2.2, "Configuring Communication With a 10g AXF Server"

After configuring communication for the appropriate AXF Server version, complete these steps:

- Section 3.2.2.3, "Encrypting the Gateway Password"
- Section 3.2.2.4, "Setting Up the Service Operation Routings"
- Section 3.2.2.5, "Validating Domain Status"

> **Note:** An Oracle PeopleSoft local integration gateway is required for Oracle PeopleSoft external communications. See the Oracle PeopleSoft Enterprise documentation for information about setting up gateways. You can then proceed with configuring the Oracle PeopleSoft service operation to communicate with an AXF server.

3.2.2.1 Configuring Communication With an 11g AXF Server

1. In Oracle PeopleSoft Server, open **PeopleTools**, then **Integration Broker**, then **Integration Setup**. The Integration setup menu expands to show available options.
2. Click **Nodes**.
3. On the Find an Existing Value tab, select **Node Name** in the **Search By** field, enter `AXF` in the **Begins With** field, and click **Search**.

4. From the search results, click the **AXF_SOLUTION_MEDIATOR_11G** link.
5. On the Node Definitions tab, select the **Active Node** field. In the **External User ID** and **External Password** fields, enter the user name and password for a valid Oracle WebLogic Server administrator.

   This user authenticates against the Solution Mediator web services installed on the application server on which AXF is installed.
6. Click the Connectors tab.
7. In the **Property Value** column, set **Host** to the AXF Server name or IP address.
8. In the **Property Value** column, set the **URL**.
   - If using SSL, set it to the following URL, using `https` instead of `http`. Note that `port_number` must match the secure port on the Oracle WebCenter Content side.

     `https://`*`AXF_Server name or IP address:port_number`*`/axf-ws/AxfSolutionMediatorService`
   - If not using SSL, set it to the following URL:

     `http://`*`AXF_Server name or IP address:port_number`*`/axf-ws/AxfSolutionMediatorService`
9. Click **Save**.
10. Click **Ping Node** to verify that the node is configured properly.
    - If configured properly, the word *Success* displays in the message text area of the Ping NodeResults page. Proceed to Section 3.2.2.4 and Section 3.2.2.5.
    - If *Success* is not displayed, return to the Node Configuration page to reenter values until you can ping the node successfully.
11. Click the WS Security tab.
12. Select **Username Token** in the Authentication Token Type field and select the **Use External User ID** field.
13. Click **Save**.

3.2.2.2 Configuring Communication With a 10g AXF Server

1. In Oracle PeopleSoft Server, open **PeopleTools**, then **Integration Broker**, then **Integration Setup**. The Integration setup menu expands to show the available options.
2. Click **Nodes**.
3. On the Find an Existing Value tab, select **Node Name** in the Search By field, enter `AXF` in the Node Name field, and click **Search**.
4. From the search results, click the **AXF_SOLUTION_MEDIATOR_10G** link.
5. Click the Connectors tab.
6. In the **Property Name** column, set **Host** to the AXF Server name or IP address.
7. In the **Property Name** column, set **URL** to the following URL:

   `http://`*`AXF_Server name or IP address`*`:port_number/imaging-bai-axf/AxfSolutionMediator`
8. Click **Save**.

9. Click **Ping Node** to verify that the node is configured properly.
   - If configured properly, the word *Success* displays in the message text area of the Ping NodeResults page. Proceed to Section 3.2.2.4 and Section 3.2.2.5.
   - If *Success* is not displayed, see Section 3.2.2.3. Return to the Node Configuration page to reenter values until you can ping the node successfully.
10. Click the **WS Security** tab and verify that the **Authentication Token Type** field is set to None.

3.2.2.3 Encrypting the Gateway Password

If *Success* is not displayed after pinging the node, encrypt the secureFileKeystorePasswd value under Gateway Properties, as described in the following steps.

1. In Oracle PeopleSoft Server, open **PeopleTools**, then **Integration Broker**, then **Configuration**, then **Gateways**.
2. Search for and open the LOCAL gateway, then click the **Gateway Setup Properties** link.
3. Sign on to access the integrationGateway.properties file.
4. Click the **Advanced Properties Page** link.
5. Use the Password Encryption Utility on the Gateway Properties page to encrypt the secureFileKeystorePasswd value.
6. On the Gateway Properties page, paste the encrypted password after **secureFileKeystorePasswd=** and click **OK**.

3.2.2.4 Setting Up the Service Operation Routings

1. From the Integration Setup options, select **Service Operations**.
2. On the Find Service Operation tab, enter `AXF` in the Service Operation field and click **Search**. A single results listing is returned.
3. Click the **AXF_EXECUTE** link.
4. Click the Routings tab.
5. Verify status.
   - If communicating with an **11g AXF Server**, verify that the AXF_SOLUTION_MEDIATOR_ROUTING routing definition's status is Active. If it is not, choose its Selected field, click the **Activate Selected Routings** button, and ensure that all other routing definition are set to **Inactive**.
   - If communicating with a **10g AXF Server**, verify that the AXF_SM_ROUTING_10G routing definition's status is Active. If it is not, choose its Selected field, click the **Activate Selected Routings** button, and ensure that all other routing definition are set to **Inactive**.
6. Click the **Save** button.

3.2.2.5 Validating Domain Status

Once you have configured the Integration Broker to communicate with AXF, follow these steps to verify its status.

1. Navigate to the Service Operation Monitor page by selecting **PeopleTools,** then **Integration Broker,** and then **Service Operations Monitor**. The Service Operations Monitor page displays.
2. Click **Domain Status** under the **Administration** section. The Domain Status page displays.
3. Under Domains, verify that the domain status is listed as **Active** next to the Oracle PeopleSoft Server. If not, select **Active** from the Domain Status field and click **Update**.

3.2.3 Configuring AXF Access For Oracle PeopleSoft Users

Follow the steps in these sections to grant AXF access to Oracle PeopleSoft users:

- Section 3.2.3.1, "Verifying the AXF_ROLE for Oracle PeopleSoft Users"
- Section 3.2.3.2, "Assigning Users the AXF_ROLE"

3.2.3.1 Verifying the AXF_ROLE for Oracle PeopleSoft Users

During Oracle PeopleSoft project import, the AXF_ROLE is automatically configured. Follow these steps to verify that appropriate Oracle PeopleSoft users have access to AXF functionality. If needed, follow the steps listed to manually add the AXF_ROLE and permissions to Oracle PeopleSoft.

1. In Oracle PeopleSoft Server, open **PeopleTools**, then **Security,** then **Permissions & Roles**. The Permissions & Roles menu expands to show available options.
2. Click **Roles**. The Roles page displays. The AXF_ROLE should be displayed. If it is not, follow the steps below to add it.

Manually Adding the AXF_ROLE to Oracle PeopleSoft

Follow these steps only if the AXF_ROLE was not successfully verified in the previous section.

1. Click the **Add a New Value** tab.
2. Enter `AXF_ROLE` in the Role Name field and click **Add**. The Role Name field is cleared and the AXF_ROLE is added. Enter a description if needed and click **Save**.
3. Click **Roles**. Click the **Find an Existing Value** tab and find AXF_ROLE from the listing of available roles. Click AXF_ROLE, and its details are displayed.
4. Click the **Permission Lists** tab, enter `AXF_PERMS` in the Permission List field, and click **Save**.

3.2.3.2 Assigning Users the AXF_ROLE

After creating the AXF_ROLE in Oracle PeopleSoft, follow these steps to assign the role to all users that require access to AXF functionality.

Note: You must assign the AXF_ROLE in Oracle PeopleSoft to all users needing access to AXF functionality for AXF calls to process correctly from Oracle PeopleSoft.

---

1. In Oracle PeopleSoft Server, open **PeopleTools**, then **Security**, then **User Profiles**. The User Profiles menu expands to show available options.
2. Click **User Profiles**. The User Profiles page displays.

3. Select the criteria by which to search, enter it in the search field, and click **Search**. A listing of users displays.
4. Click the user to which to assign the role. A detailed page of user information displays.
5. Click the **Roles** tab. A listing of roles assigned to the user displays.
6. Enter `AXF_ROLE` in a blank field (clicking the + icon to create a blank row if needed), or click the magnifying glass icon to search for the role.
7. Click **Save**. The user can now access AXF functionality on Oracle PeopleSoft pages.

3.3 Placing AXF Components on Oracle PeopleSoft Pages

You configure AXF components on Oracle PeopleSoft pages using the Oracle PeopleSoft Application Designer and the AXF tables in Oracle PeopleSoft.

Determining Primary Keys and Business Objects for Oracle PeopleSoft Pages

If needed, follow these steps.

1. In the Oracle PeopleSoft application, navigate to the page to enable.
2. Press **Ctrl+J**. In the info page, note the **Page** and **Component** entries for the selected page.
3. In the Application Designer, select **File/Open** and choose **Page** from the Definition field.
4. In the **Name** field, enter the page name you noted in step 2 and click **Open**.
5. Select the **Order** tab at the top.
6. From the set of fields listed, identify the field that is the most likely unique ID (for example, VOUCHER_ID for Invoice Entry) and note the record name.

   The record name is used for the business object value in the PS_AXF_CMD_PARAMS Table (Managed Attachments solution) and PS_AXF_COMMAND_PARAMS Table (Imaging solution).
7. From the **Insert** menu, choose the **Definitions into Project** command, and select **Records** from the Definition Type field.
8. In the **Name** field, enter the record name you noted in step 6 and click **Insert**.
9. Select the record in the list and click **Insert**. The record is inserted into the Project Tree on the left.
10. Expand the Records tree on the left, then expand the record you added. The primary keys are listed at the top, with a gold key icon to the left of the key name. Note the primary keys.

Sample Primary Keys

Table 3–1 Sample Primary Keys

Pages	Page Name	Record Name	Primary Key
Maintenance Management > Work Order Management > Work Order	WM_WO_HDR	WM_WO_HDR	BUSINESS_UNIT
Maintenance Management > Work Order Management > Work Order	WM_WO_HDR	WM_WO_HDR	WO_ID
Customer Contracts > Create and Amend > General Information	CA_HDR_PNL	CA_CONTR_HDR	BUSINESS_UNIT
Customer Contracts > Create and Amend > General Information	CA_HDR_PNL	CA_CONTR_HDR	CONTRACT_NUM
Purchasing > Requisitions > Add/Update Requisitions	REQ_FORM	REQ_HDR	BUSINESS_UNIT
Purchasing > Requisitions > Add/Update Requisitions	REQ_FORM	REQ_HDR	REQ_ID

About Placing AXF Components on Oracle PeopleSoft Pages

There are two categories of AXF components for use on Oracle PeopleSoft pages:

- **User interface items** refer to AXF buttons, links, and menus you place on an Oracle PeopleSoft page. Five of each are provided (for example, AXF_BUTTON_1 through _5). The Managed Attachments solution uses these user interface items only.

  A user interface item becomes active when positioned on a page. If AXF_BUTTON_1 is placed on multiple pages in the same Oracle PeopleSoft component, it shares the same configuration across the pages. To configure different actions for each page in the same component, use a different user interface item (for example, AXF_BUTTON_2). Note that you can reuse user interface items on different Oracle PeopleSoft components without conflicts.

- **System event components** refer to pre- and post-save items you place on an Oracle PeopleSoft page that trigger a pre- or post-save system event. The Imaging solution uses both user interface and system event items.

  System events, like user interface items, are configured at the Oracle PeopleSoft component level. However, because they are not visible to the user, they are triggered when appropriate regardless of the page on which they are placed. For example, the AXF_POST_SAVE_SBP and the AXF_PRE_SAVE_SBP items are placed on specific pages; if a Save event occurs on a page in an Oracle PeopleSoft component that is being saved, the configured command executes.

  – The AXF_POST_SAVE_SBP item captures SAVE_POST_CHANGE system events, which execute a configured command *after* an Oracle PeopleSoft component is saved.

    For example, you might configure a SaveInvoice command to invoke during the SAVE_POST_CHANGE event, so that whenever an action inserts a new Oracle PeopleSoft transaction record, the SaveInvoice command automatically performs a save.

  – The AXF_PRE_SAVE_SBP item captures SAVE_PRE_CHANGE system events, which execute a configured command *before* an Oracle PeopleSoft component is saved.

Steps For Placing AXF Components on Oracle PeopleSoft Pages

1. In the Oracle PeopleSoft Application Designer, open the AXF_PS_Integration Project that you imported in step 5 of Section 3.1.
2. Open an Oracle PeopleSoft page to enable. For example, open VCHR_HEADER_QV2.
3. From the Pages folder, select a button or link (subpage) and drag it onto the Oracle PeopleSoft page in an appropriate location. For example, drag the link called AXF_LINK1_SBP or the button called AXF_BTN1_SBP. This subpage link or button corresponds to AXF event settings in Table C–33 in Section C.4.2 (see Table C–34 example).
4. Save the page.

3.4 Configuring Oracle PeopleSoft Logging

The AXF Oracle PeopleSoft adapter uses standard apache log4j logging. You can activate and manage logging through the log4j.properties file. This file is located in the *PeopleSoft Installation Directory*/class directory. Set the following options:

```
# A1 is set to be a ConsoleAppender which outputs to System.out.
log4j.appender.A1=org.apache.log4j.DailyRollingFileAppender
log4j.appender.A1.File=./LOGS/PS_AXF.log

# A1 uses PatternLayout.
log4j.appender.A1.layout=org.apache.log4j.PatternLayout
log4j.appender.A1.DatePattern=.yyyy-MM-dd
log4j.appender.A1.layout.ConversionPattern=%{dDATE} [%t] %-5p %c %x - %m%n

log4j.category.com.oracle.axf=DEBUG, A1
```

At a minimum, define an appender with a valid file location, and the level of messages to report (see above). Valid levels include the following in order of decreasing detail, where DEBUG displays all messages and detail:

- DEBUG
- INFO
- WARN
- ERROR
- FATAL

Note: For detailed information about log4j, see the log4j documentation.

3.5 Configuring Solutions for Oracle PeopleSoft

See these sections listed below for additional tasks to configure each solution. After placing a component on an Oracle PeopleSoft page, a key task is to configure the AXF-related Oracle PeopleSoft tables to associate a command with the newly added button, link, or event.

For the Imaging Solution

1. Configure the Imaging solution as described in Chapter 6. Configure AXF-related Oracle PeopleSoft tables as described in Section B.5 or apply a solution accelerator as described in Section 6.2.
2. Specify an Oracle PeopleSoft role in the PS_AXF_COMMANDS Table to either grant users permission to use a specified user interface item or allow a specified system event to trigger for a user. See Section B.5.3.

For the Managed Attachments Solution

1. Configure the Managed Attachments solution as described in Chapter 7.
2. Make entries or changes to the PeopleSoft AXF tables listed in Section C.4. Running the sample script described in Section 7.2.4 populates the tables with sample values that you can modify. You must enter this information:
   a. Enter the primary page keys you noted in Section 3.3.
   b. Specify an Oracle PeopleSoft role in the PS_AXF_COMMANDS Table to grant users permission to use a specified user interface item. See Section C.4.2.

3.6 Uninstalling AXF From Oracle PeopleSoft

No resources are being used if no AXF solutions are implemented, so there is no harm in leaving AXF objects in the system. If no AXF solutions are implemented, you can remove all AXF objects using Application Designer.

4

Configuring AXF Settings

Most AXF configuration is completed as part of Oracle WebCenter Content installation. This chapter describes how to configure AXF logging.

This chapter covers the following topic:

- Section 4.1, "Configuring Logging for AXF for BPM"
- Section 4.2, "Configuring Logging for AXF for BPEL"

4.1 Configuring Logging for AXF for BPM

This section describes AXF for BPM logging through Enterprise Manager and configuring DMS logging.

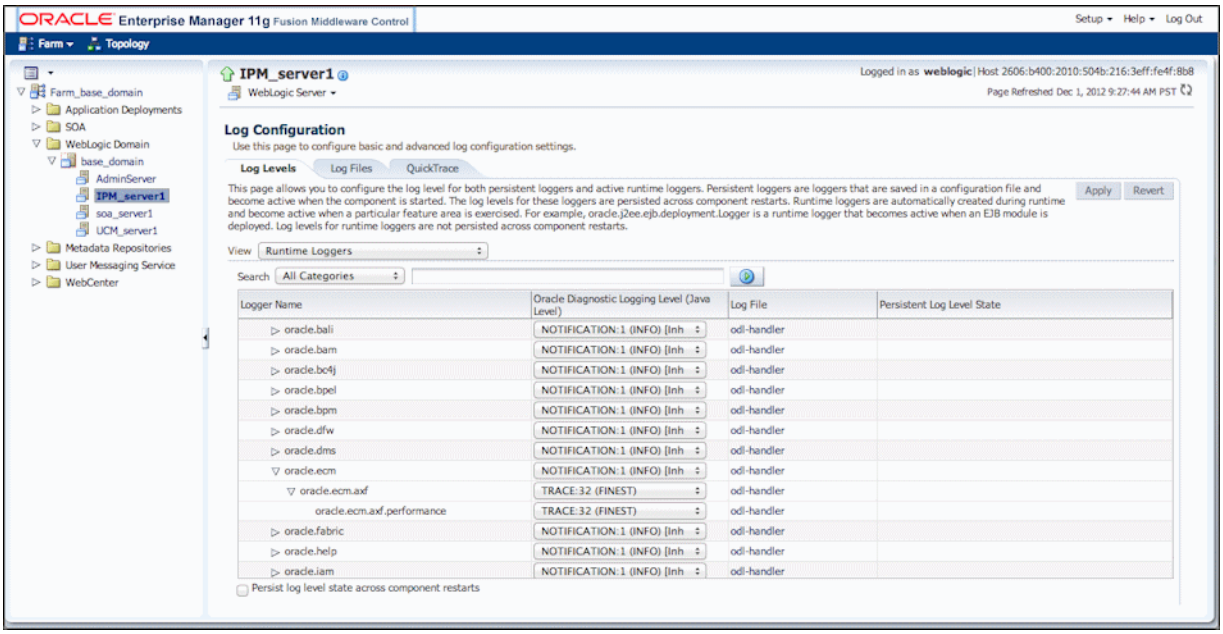
Table 4–1 lists the loggers you can configure, which include base, customization, and performance loggers. For advanced configuration, see Section 4.1.1.

Table 4–1 AXF Loggers

Logger	Description
oracle.ecm.axf	This base AXF logger provides a highly detailed log for AXF system diagnostics. It also includes information for tracing execution and exceptions for the product.
oracle.ecm.axf.performance	This AXF performance logger provides a detailed log that corresponds to DMS metrics, for determining the overall performance of the system or isolating specific performance based test cases. See Section 4.1.2.

The base AXF logger inherits its logging level from the Oracle parent logger, which by default is configured to use the NOTIFICATION level of logging. The customization and performance loggers inherit their logging level from the base AXF logger. The NOTIFICATION configuration logs warning and error messages out to the console only. (See Table 4–4 for additional information.) To increase the level of logging and increase the console logging detail, adjust the level through the Enterprise Manager console, shown in Figure 4–1.

Figure 4–1 Enterprise Manager Console Log Configuration



4.1.1 Advanced Configuration

To examine specific behavior on an AXF for BPM system, you may need to capture more detailed information than that provided by logging to the console. This section describes steps for advanced configuration, which involves defining one or more log handlers and loggers through the logging.xml files on the server. Each server has a logging.xml file set up under the Oracle domain, located in the following directory:

```
Oracle/Middleware/user_projects/domains/<domain
name>/config/fmwconfig/servers/<server> directory
```

1. Edit the logging.xml file and define two log handlers (log_handler) and two loggers (logger). This enables you to handle the output from the two loggers independently and use the corresponding log handler to route the output to a separate and unique log file. Define the log handlers under the `log_handlers` tag, using the properties listed in Table 4–2 and referring to Example 4–1.

Table 4–2 Log Handler Properties

Property	Description
path	Defines the server location at which to store the logging.
maxFileSize	Specifies the maximum size in bytes for each log file. When the main log file reaches the given size, it triggers a log rotation, where the main log file is archived and a new log file is created.
maxLogSize	Specifies the maximum size in bytes for the entire log. Older archive files are deleted to keep the total log size under the given limit.

Example 4–1 Logging.xml File With Log Handler Modifications

```
<log_handler name='axf-handler' class='oracle.core.ojdl.logging.ODLHandlerFactory' level='ALL'>
   <property name='path' value='${domain.home}/servers/${weblogic.Name}/logs/axf.log'/>
   <property name='maxFileSize' value='5485760'/>
   <property name='maxLogSize' value='54857600'/>
```

```
  </log_handler>
<log_handler name='axf-handler-performance' class='oracle.core.ojdl.logging.ODLHandlerFactory'
level='ALL'>
   <property name='path' value='${domain.home}/servers/${weblogic.Name}/logs/axf_performance.log'/>
   <property name='maxFileSize' value='5485760'/>
   <property name='maxLogSize' value='54857600'/>
  </log_handler>
```

2. Similar to defining log handlers, define three loggers (logger), one for each logger defined in the AXF application and visible through the Enterprise Manager console. This enables you to configure the level of logging for each of the loggers independently and specify which log handler to use for output. Define the loggers under the `loggers` tag, using the properties listed in Table 4–3 and referring to Example 4–2.

Table 4–3 Logger Properties

Property	Description
name	The logger name defined in the application. AXF include these loggers: oracle.ecm.axf, oracle.ecm.axf.performance, and oracle.ecm.axf.customization.
level	The level at which to log information. See Table 4–4 below for additional information.
useParentHandlers	Set to `true` to inherit level from parent and `false` to use the specified level.

Example 4–2 Logging.xml File With Logger Modifications

```
<logger name='oracle.ecm.axf' level='TRACE:32' useParentHandlers='false'>
 <handler name='axf-handler'/>
 <handler name='console-handler'/>
</logger>
<logger name='oracle.ecm.axf.performance' level='TRACE:32' useParentHandlers='false'>
 <handler name='axf-handler-performance'/>
 <handler name='console-handler'/>
</logger>
```

3. After defining and saving the loggers and log handlers, log files automatically begin filling with logging as defined in the logger level attribute. You can also change the levels through the Enterprise Manager console to isolate a particular log where detail is required and minimize logging to the rest to reduce server performance impacts. Acceptable values for logging levels are listed in Table 4–4.

Table 4–4 Logging Levels

Message Type/Logger Level	Description
ERROR	A serious problem that requires immediate attention from the System Administrator.
WARNING	A potential problem that the System Administrator should review.
NOTIFICATION	A major lifecycle event.
TRACE	Trace or debug information.
UNKNOWN	May be used when the type is unknown.

4.1.2 Configuring DMS Logging

The Oracle Dynamic Monitoring Service (DMS) is the Oracle standard for diagnosing application performance. As part of the Oracle standard, the DMS logging should always be on. AXF handles DMS logging with these goals in mind:

- The need to properly gauge performance of AXF requires the ability to break out time spent in external product.
- The need to have a log for correlating DMS metrics to a timing call stack metric in order to facilitate breaking down a performance issue to a specific area of the product.

The metric tables listed in Table 4–5 are used to help break down performance based on the type of actions being performed.

Table 4–5 DMS Logging

Metric Table	Description
AXF	Records timing statistics on AXF API operations. Statistics are recorded on a per service operation basis.
AXF_Business Rules	Records timing statistics for interaction with SOA Business Rules.
AXF_HumanTasks	Records timing statistics for interaction with SOA Human Tasks.

As part of the instrumentation of the AXF product, a special logger, oracle.ecm.axf.performance, facilitates the correlation of DMS data and general execution stack tracing. The AXF_Performance.log is configurable based on standard Java and Oracle practices through Enterprise Manager. See Section 4.1.1 for an example on configuring the logs.

4.1.2.1 Clearing DMS Logging

When isolating particular use cases, it is helpful to clear the DMS logging. Clear DMS logging from the AXF Command Driver page by clicking the **Reset DMS Metrics** button.

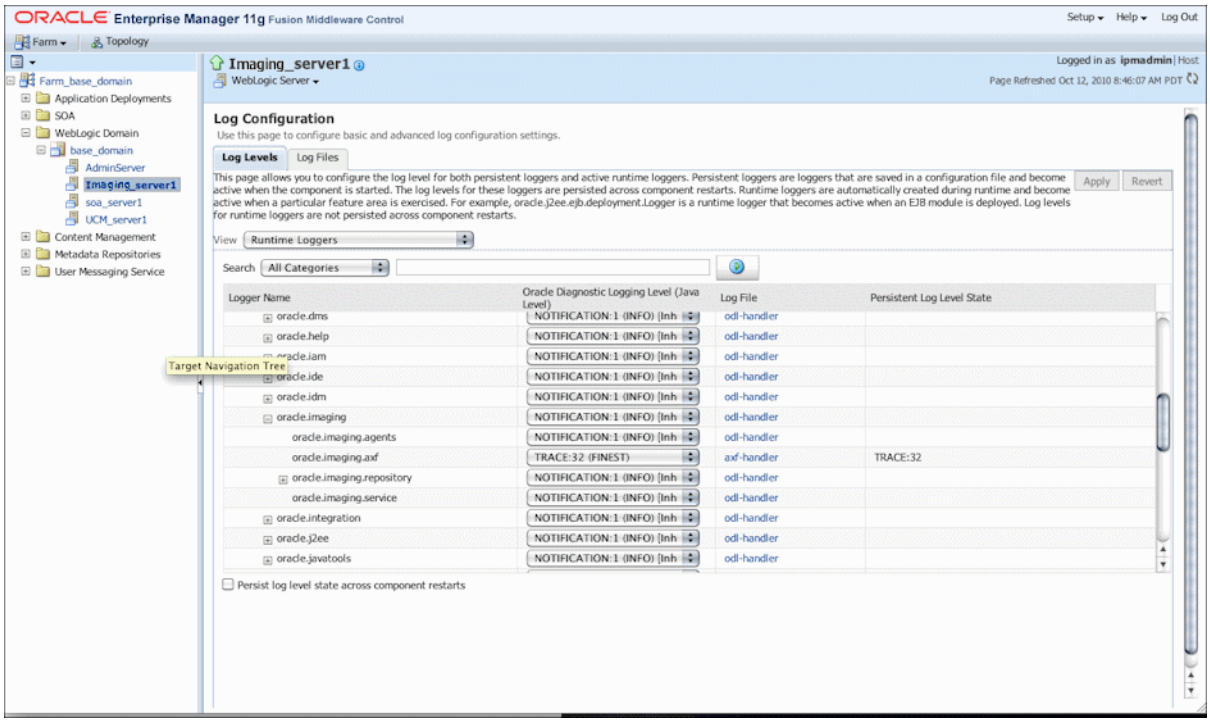
4.2 Configuring Logging for AXF for BPEL

You can configure logging using either of these methods:

- Section 4.2.1, "Using Enterprise Manager"
- Section 4.2.2, "Using Application Server"

4.2.1 Using Enterprise Manager

To use the web interface to set up loggers, navigate to the Log Configuration page in Enterprise Manager. For example, select the server in the side pane, right-click, and choose **Logging**, then **Configure Logging**. For more information, see the Oracle Enterprise Manager documentation.



4.2.2 Using Application Server

Use the AXF logs to isolate issues in solution configuration. By default, some AXF logging automatically occurs as part of Application Server logging. Follow these steps to configure more detailed and separate AXF logging.

1. Add a log handler to the Application Server configuration. Add the handler inside the <log_handlers> tag in the logging.xml file, at the following location:

   DOMAIN/config/fmwconfig/servers/*SERVER*/logging.xml

   An example location follows:

   base_domain/config/fmwconfig/servers/IPM_Server1/logging.xml

```
<log_handler name='axf-handler' level='ALL'
class='oracle.core.ojdl.logging.ODLHandlerFactory'>
   <property name='path'
value='${domain.home}/servers/${weblogic.Name}/logs/axf.log'/>
   <property name='maxFileSize' value='5485760'/>
   <property name='maxLogSize' value='54857600'/>
   <property name='encoding' value='UTF-8'/>
</log_handler>
```

2. Add a logger to the logging.xml file and set the level from the Log Levels (ODL Message Types) listed in Table 4–6. You can set the logging level in the XML file or using Enterprise Manager.

```
<logger name='oracle.imaging.axf' level='TRACE:32' useParentHandlers='false'>
     <handler name='axf-handler'/>
     <handler name='console-handler'/>
</logger>
```

> **Note:** Remove the console-handler tag to omit logging on the terminal.

Table 4–6 Available Logging Levels

Log Type	Description	Log Level (ODL Message Type)
NULL	The logger inherits the log level set for its parent.	n/a
SEVERE	Log system errors requiring attention from the system administrator.	ERROR:1
WARNING	Log actions or conditions discovered that should be reviewed and may require action before an error occurs.	WARNING:1
INFO	Log normal actions or events. This could be a user operation, such as login completed, or an automatic operation, such as a log file rotation.	NOTIFICATION:1
CONFIG	Log configuration-related messages or problems.	NOTIFICATION:16
FINE	Log trace or debug messages used for debugging or performance monitoring. Typically contains detailed event data.	TRACE:1
FINER	Log fairly detailed trace or debug messages.	TRACE:16
FINEST	Log highly detailed trace or debug messages.	TRACE:32

3. If using Windows, restart Administration Server if it is running. The logger displays in Enterprise Manager. You can change the logging level at run time.

5

Configuring the BPM Imaging Solution

This chapter describes how to configure BPM imaging solution components and options, based on your administrator role. For more information about roles and the types of solution changes administrators can make with each role, see Section 1.2.2.

This chapter covers the following topics:

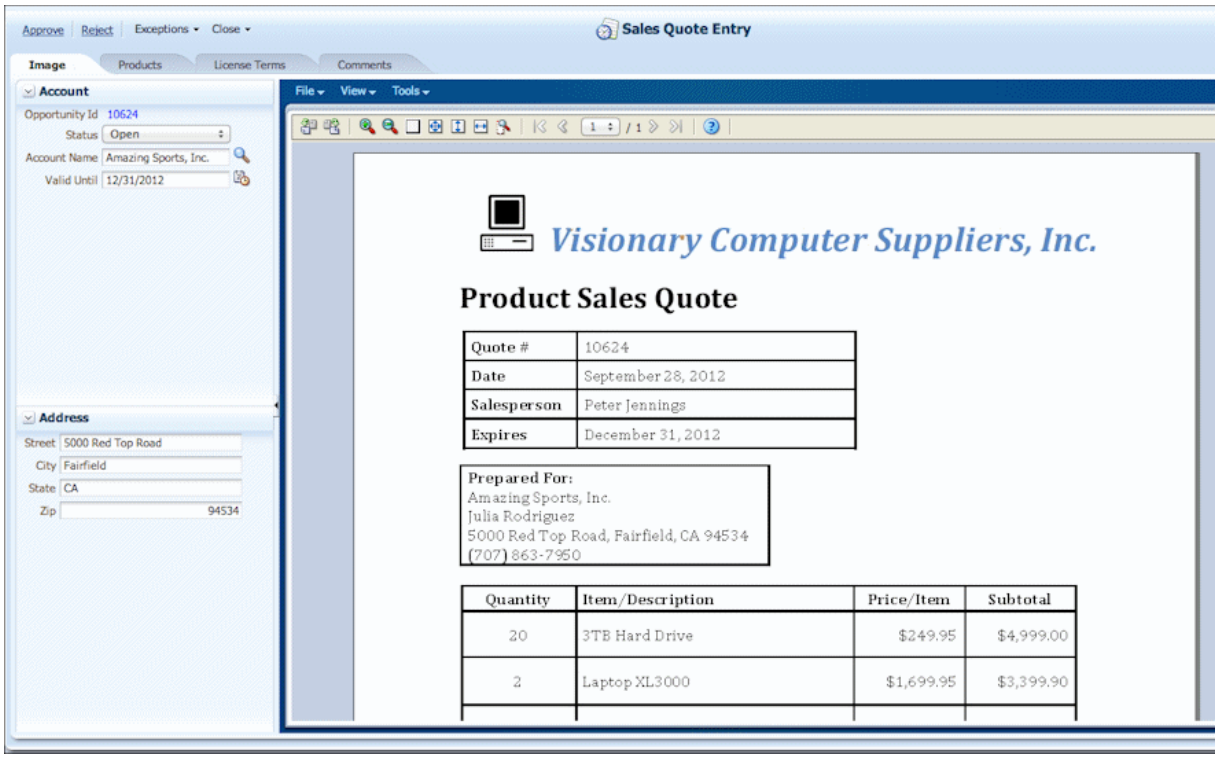
- Section 5.1, "Understanding Solution Application Administration"
- Section 5.2, "Getting Started Configuring the Solution Application"
- Section 5.3, "Understanding the HelloBPM Solution"
- Section 5.4, "HelloBPM Solution Use Cases"
- Section 5.5, "Configuring BPM Views"
- Section 5.6, "Using Oracle Fusion Middleware Control to Manage AXF for BPM"

5.1 Understanding Solution Application Administration

Section 1.2.4 describes how the solution application portion of the solution workspace (shown in Figure 5–1) provides a highly configurable human task area where users complete selected tasks. This section describes the concepts behind editing the solution application's underlying business rules. In addition, the HelloBPM solution installed with AXF for BPM provides an example configuration you can use to examine and practice solution application configuration.

Note: This section assumes a general knowledge of business rule editing, including rulesets, decision tables, and dictionaries. Business rule information is covered extensively in the *Oracle Fusion Middleware Business Process Composer User's Guide for Oracle Business Process Management*.

Figure 5–1 Solution Application Driven by Business Rules



This section covers the following topics:

- Section 5.1.1, "About the Dynamic Page Template and Page Components"
- Section 5.1.2, "About Business Rules and Dictionaries"
- Section 5.1.3, "About Functional Blocks"
- Section 5.1.4, "About Control Pages"
- Section 5.1.5, "About Prompts and Validations"
- Section 5.1.6, "About Packages"
- Section 5.1.7, "About Contexts"
- Section 5.1.8, "About Data Fields, Tables, and Lookups"
- Section 5.1.9, "About Disabling or Discarding Business Rule Changes"

5.1.1 About the Dynamic Page Template and Page Components

In AXF for BPM, every solution application page a user sees is built on a single **dynamic page template**. This template includes menu (action control) and tab (dynamic tabs) components, as shown in Figure 5–1. Although every page the user navigates to is the same page, business rules define how the page is rendered. For example, depending on context, different tabs may display to some users and not others and the contents of the tabs may differ.

- Section 5.1.1.1, "About the Action Control"
- Section 5.1.1.2, "About the Update Status Icon"
- Section 5.1.1.3, "About Dynamic Tabs"
- Section 5.1.1.4, "About Page Components"

5.1.1.1 About the Action Control

The action control is a business rule-driven component that defines actions that call AXF functionality. The functionality that actions call is specified by functional blocks, which are described in Section 5.1.3. For example, the HelloBPM solution action control contains **Approve** and **Reject** actions displayed as links. This configuration can be used for more commonly used actions. You can also configure the action control to display actions grouped in dropdown menus. For example, the **Account Manager** action and **Save and Close** action are organized under the **Exceptions** and **Close** dropdown menus.

Use the **Action** rule dictionary to configure the layout and behavior of the actions displayed in the action control, and the **FunctionalBlock** dictionary to configure the functionality the actions invoke. For example, use business rules to:

- Define actions and sections (for example, several close action links might display within a Close dropdown menu).

  Note that actions not assigned to a section display left-justified in the Action menu as links, allowing users to easily access commonly used actions. Sections display as dropdown menus with their actions listed as menu items.

- Add new actions to the control. After adding an action, define the functional block or functional block sequence to invoke for the action. For example, clicking on an Approve action might navigate to an Identity Browser control page.

- Exclude or disable actions or sections based on rule outcome. For example, an Approve action might be disabled or hidden when a required data field is blank or set to 0.

5.1.1.2 About the Update Status Icon

To the right of the action control is the **Update Status** icon shown below.

This icon changes (appears selected) to indicate to the user that an update to the internal, underlying data has occurred. For example, it changes when a value is updated in the business application, but not when users change data values on dynamic tabs. Once changed, the user clicks the icon to acknowledge the data update, which resets the icon to its deselected state.

5.1.1.3 About Dynamic Tabs

The dynamic tabs control is a business rule-driven component that defines the tabs that display within the solution application and their page component contents. The default dynamic tabs in the HelloBPM solution shown in Figure 5–1 include Image Data, Master Detail for Products, Master Detail for License Terms, and Comments.

Use the **DynamicTab** dictionary to configure a tab's contents and behavior. See Section 5.1.2.

5.1.1.4 About Page Components

Page components display contents such as images, data fields, and/or tables on a dynamic tab. The dynamic tab control hosts a page component, as described in more

detail in Section A.1.1. Page components implement the main functionality displayed in the task details area.

- A page component can be comprised of other page components, resulting in a composite page component. For example, if a dynamic tab hosts the Data page component, the user role sees data only. But if the tab also hosts the Image or TableData page components, it displays an image or a data table, respectively.
- The Image Data and Master Detail page components are examples of page components made from other page components. The Image Data page component contains a Data and Image page component while the Master Detail page component contains a Data and Table Data page component.

5.1.2 About Business Rules and Dictionaries

Business rules provide the means of configuring the solution application to meet the custom needs of business application users. In AXF for BPM solution administration, business rules are standard SOA business rules that you modify to change the AXF solution's behavior. Using the solution administration area, you can load, edit, and save business rules, test the results of your modifications, and if needed, restore the business rules to their product state.

Rules are stored in a separate AXF MDS repository that uses the same schema but is separate from the SOA repository. The initial product rules are deployed as part of the solution application.

A **dictionary** is a container for business rules grouped for a solution application feature. For example, the Action dictionary defines actions and sections (dropdown menus) in the action control portion of the solution application. The dictionaries provided with the AXF for BPM infrastructure are listed in Table 5–1. A solution's dictionaries are grouped into a **package**, as described in Section 5.1.6. For example, all dictionaries in the HelloBPM solution are contained in the oracle.axf.solution.hellobpm package.

For the most part, rule dictionaries are separate from one another, although some have entity relationships, as highlighted in Section A.1.9. All rules tie to the solution application, not to the BPM processes.

Table 5–1 Dictionaries Provided with the AXF for BPM Infrastructure

Dictionary	Description
Action	This dictionary defines the layout and behavior of the action control area, in which task actions display. You can optionally define actions as conditionally displayed or disabled. For example, you could edit business rules so that an Approve button displays as disabled to certain user roles when a specific data value (such as Amount) exceeds a specified amount. For reference information, see Section A.1.3.
Data	This dictionary defines the data (metadata) and items displayed within the Data page component. Data values can be defined in a variety of ways, including editable or read-only. For reference information, see Section A.1.4.
DynamicTab	This dictionary defines the layout and behavior of the tabs displayed in the solution application. For example, you might use this dictionary to display a Comments tab and require that users enter a comment before routing a task. For reference information, see Section A.1.5.

Table 5–1 (Cont.) Dictionaries Provided with the AXF for BPM Infrastructure

Dictionary	Description
FunctionalBlock	This dictionary defines solution application functions and function sequences for actions, such as the actions that occur when a user selects an action link. For example, you can use functional blocks to set an outcome to a human task or to navigate to a predefined control page such as the identity browser control page. For reference information, see Section A.1.6.
LookupAndValidator	This dictionary defines lookups (user selection of data values from a selection list) for the Data and TableData controls. Lookups can be either static (fixed list) or dynamic (SQL query from a database), and dynamic lookup fields can be dependent (linked). In addition to lookups, this dictionary also defines validators for validation configuration. For reference information, see Section A.1.7.
TableData	This dictionary defines how data is displayed in table format. A single table can be defined that contains multiple column definitions. Tables are dynamically rendered at runtime based on rule configuration. For reference information, see Section A.1.8.

5.1.2.1 About Editing Dictionaries

When you edit a dictionary's business rules, you edit its **custom rules**, but you can return the dictionary's rules back to **product rules** if needed. For more information about editing dictionaries, see Section 5.2.5.

A **context** is a specialized version of one or more dictionaries. Typically, contexts are used to provide additional functionality to certain tasks. For example, the HelloBPM solution includes a SalesQuoteEntry context that displays data fields and tables when a sales quote entry task is selected. For more information, see Section 5.2.6.

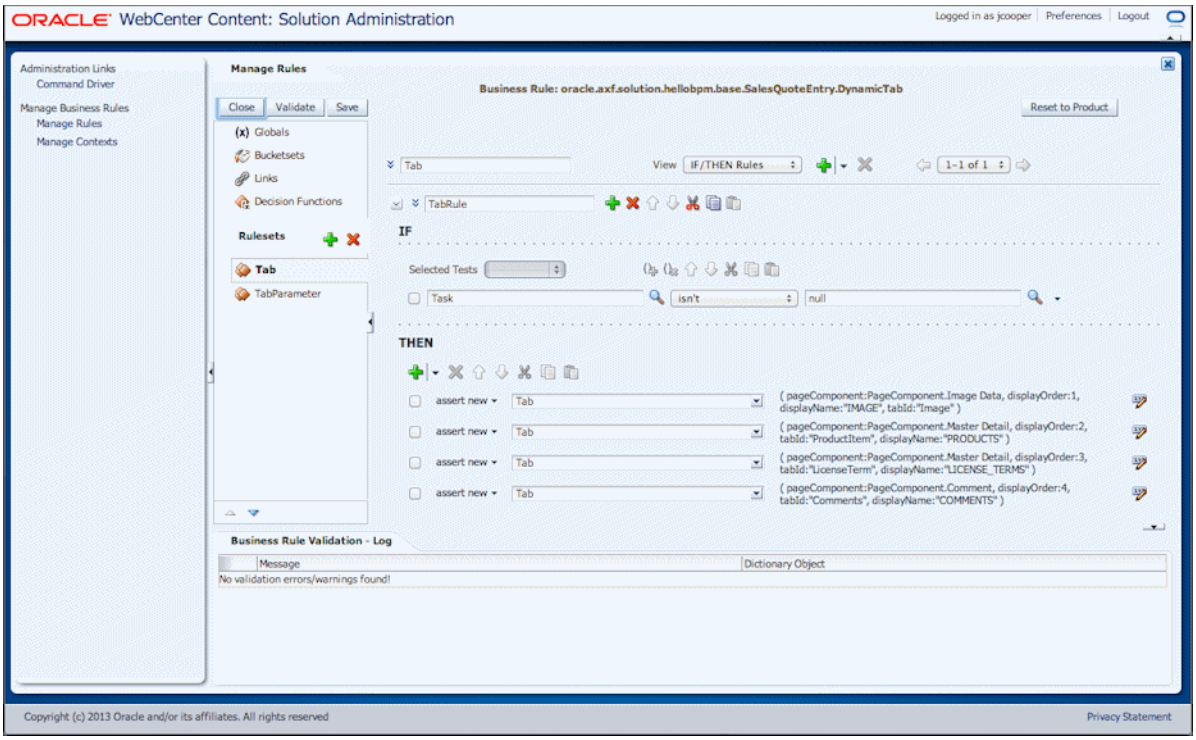
5.1.2.2 About Rulesets and Bucketsets

In a business rule dictionary, related rules are grouped into **rulesets** that define specific aspects of the dictionary. For example, with the DynamicTab dictionary open as shown in Figure 5–2, the right portion of the pane lists individual rules defined for the ruleset selected in the adjacent pane (under **Rulesets**).

Note that each ruleset is prepopulated as much as possible, enabling you to add new functionality by adapting previous settings or changing the `placeholder` value.

Bucketsets define constraints, such as a list of values or a range of values for a specified fact type. For example, the DynamicTab dictionary's bucketset, PageComponent, defines the page components that you can configure (Image, Image Data, Data, Table Data, Master Detail, Comment, and Identity Browser).

Figure 5–2 Editing Business Rules in the DynamicTab Dictionary



5.1.2.3 About Inputs

Each user interface control's rulesets have access to the input variables listed in Table 5–2, which are sent to the decision point. See the specific control's section in Section A.1 for information about outputs generated by rulesets.

Table 5–2 Inputs Used by AXF for BPM Rulesets

Input	Description
Task	The human task
Payload	The payload of the human task
Context	The current context, which can be used for conditional rules

5.1.3 About Functional Blocks

In AXF for BPM, **functional blocks** represent the raw functional building blocks for things you can do in a solution application. For example, you use functional blocks to set an outcome to a human task, such as approving a task, or to navigate to another page such as a control page.

- A **named functional block** is a defined instance of a functional block, including parameters. For example, a functional block might navigate to a control page, but the named functional block specifies the control page to which to navigate and which parameters to use.
- A **named functional block sequence** represents a sequence of named functional blocks. For example, a sequence might navigate to an identity browser control page, and then set the human task's outcome to Approve.

You define named functional blocks and their sequences in the **FunctionalBlock** rule dictionary. Typically, all functional blocks are defined in the base context, and then referenced by other rules in other contexts.

You can use functional blocks for the types of actions listed in Table 5–3. See Section 5.3.6.1 for examples of their use.

Table 5–3 Actions Performed by Functional Blocks

Action	Description
Set Outcome	Sets a specified Human Task outcome, such as Approve or Reject.
Navigate Dynamic Page	Navigate to a specified dynamic page.
Navigate Control Page	Navigate to a specified control page, such as the Identity Browser, Comments or Data control page.
Save	Performs an explicit save.
Revert	Loads the original payload, providing a cancel function for changes made to the task since it was last opened. Note: Comments added to the task are not rolled back, even with the Revert Functional Block.
Close Task	Tasks close automatically when a user selects another from the task list, but this functional block provides an explicit close.
SystemActionSuspend	BPM action that suspends a specified action. See the Oracle BPM documentation for more information.
SystemActionWithdraw	BPM action that withdraws a specified action. See the Oracle BPM documentation for more information.
SystemActionSkipCurrentAssignment	BPM action that skips an assignment. See the Oracle BPM documentation for more information.
SystemActionEscalate	BPM action that escalates a specified action. See the Oracle BPM documentation for more information.

5.1.4 About Control Pages

Control pages are predefined dynamic tab pages that contain one or more tabs and their associated page components for use in a solution. They enable you to quickly incorporate commonly used functions. For example, the Identity Browser Control Page displays an interface from which users can select a user or group, such as for approval purposes.

The AXF for BPM infrastructure provides the following control pages:

- Data (see Section 5.1.4.1)
- Identity Browser (see Section 5.1.4.2)
- Comment (see Section 5.1.4.3)

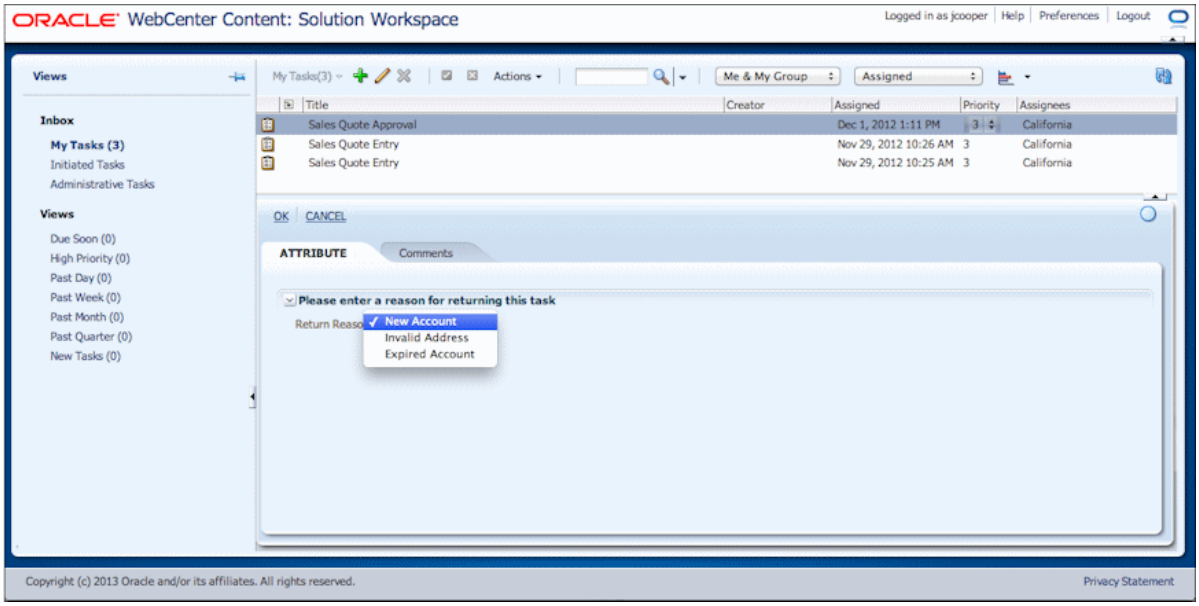
Each control page has a fixed structure with configurable content based on business rule configurations.

5.1.4.1 About the Data Control Page

The Data control page enables you to specify data fields to users for display or edit, as shown in Figure 5–3. Each data field displayed on the control page is bound to a data field defined in the Data dictionary. Data control page parameters are listed in Table A–7.

In addition to specifying data fields, you can use this control page to configure tables of data fields and configure fixed and SQL-based lookups, as described in Section 5.1.8. You can also group data items in sections.

Figure 5–3 Data Control Page



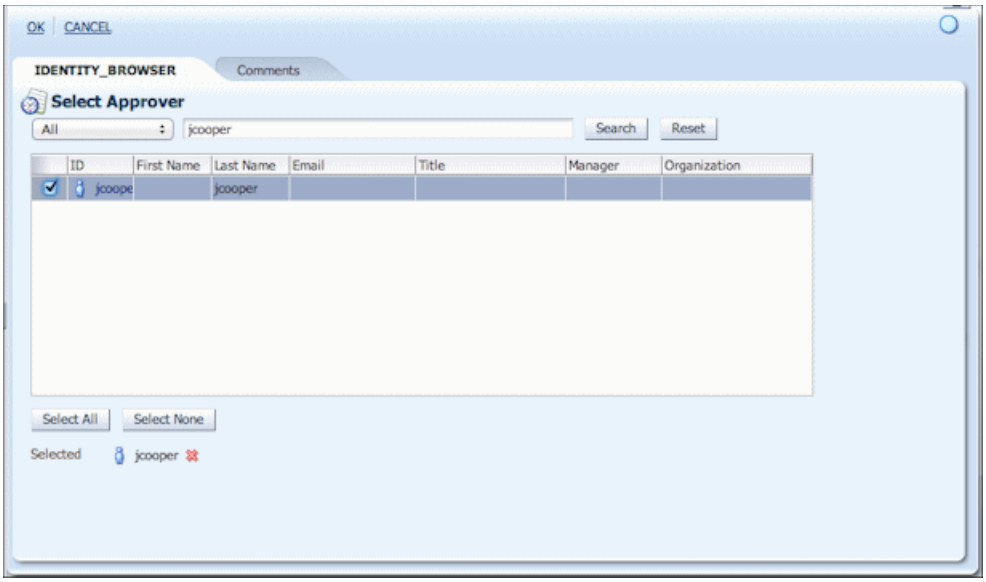
5.1.4.2 About the Identity Browser Control Page

The Identity Browser control page shown in Figure 5–4 enables users to search for users, groups, or application roles and to update the task payload, typically for use in task assignments later in the process. Identity Browser control page parameters are listed in Table A–8. In the HelloBPM example, a sales quote entry user selects the Approve action link, which displays an Identity Browser tab prompting users to select an approver role.

The identity browser control page:

- Is typically used as part of a sequence of Named Functional Blocks that collects assignment information then completes the task.
- Allows the task assignment to be updated for future tasks within the process flow.

Figure 5–4 Identity Browser Control Page

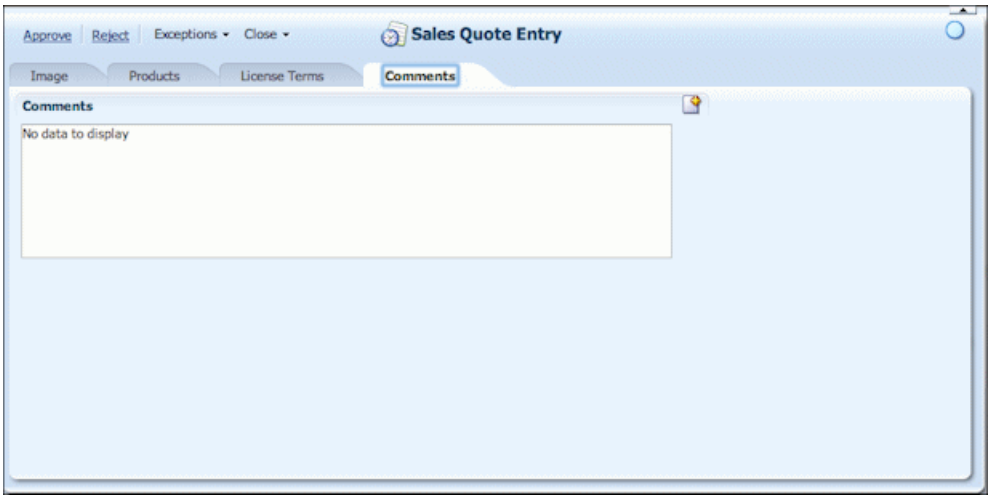


5.1.4.3 About the Comments Control Page

The Comments control page enables users to view and add comments to a task, as shown in Figure 5–5. Comments control page parameters are listed in Table A–9.

You can require users to enter comments. For example, an Approve action link might require users to enter a comment before the approval task can be completed.

Figure 5–5 Comments Control Page



5.1.5 About Prompts and Validations

As part of solution application configuration, you can include prompts and validations in human task flows.

About Prompts

A prompt displays a specified message to users as part of a functional block. Prompts can occur before or after validation and allow users to cancel/continue an action or revert changes and continue an action.

You configure prompts in the NamedFunctionalBlockPrompt Properties of the FunctionalBlock Dictionary, specifying settings such as the following:

- The functional block at which the prompt occurs
- The point at which the prompt displays (before or after validation)
- The prompt message to display to users
- The prompt type (for example, OK or Cancel)

For example, you might define a prompt that occurs at the Reject functional block (when the user clicks the **Reject** link) that prompts users after validation occurs and displays OK/Cancel buttons along with the following prompt message: "Are you sure you would like to reject this request?" For example prompt configurations, see Table 5–24.

About Validation

You can configure validations in a variety of ways:

- A validation can occur immediately or via a functional block.

  An immediate validation might occur when the user completes a data field and clicks away from the field.

  Functional block validation occurs when a user clicks an action, such as an OK or Cancel button on the page.

- Validations are performed by validators defined in the LookupAndValidator Dictionary.

  A basic validation might check for a null data value when the user clicks **Approve**, and if found, display a message reminding a user to do something first.

  A more complex validation might run a specified validator on a data field, such as the DateTimeRangeValidator, which validates that a date field's entry falls within a specified range. See Table A–33 for a list of validators.

- A functional block validation can have a specific result (continue, abort, or prompt) if an error occurs.

  The result occurs when the functional block executes, enabling you to tailor results by functional block.

Depending on validation type, you configure validations using the following dictionaries:

- LookupAndValidator Dictionary: In Validator Properties and ValidatorParameter Properties, configure settings such as the validator class and its parameters.
- Data Dictionary: If specifying validation on a data item, identify the field on which to run the validation (in DataValidator Properties) and whether the validation occurs immediately or via a functional block.
- TableData Dictionary: If specifying validation on a table column, identify the column on which to run the validation (in ColumnValidator Properties) and whether the validation occurs immediately or via a functional block.

5.1.6 About Packages

A business rule **package** provides a container for a solution application's set of dictionaries. It includes all business rules in the solution application, including its product rules, edited rules, and all dictionaries defined in all contexts for the solution.

(Contexts are described in Section 5.1.7.) For example, the HelloBPM solution is contained in a package called *oracle.axf.solution.hellobpm*, and contains all dictionaries defined in all contexts.

5.1.7 About Contexts

A **context** provides a specialized version of one or more dictionaries in the package, separate from the **base** context. In other words, a context specifies which business rules to execute. Contexts allow you to modify solution application behavior based on human task definition or payload content.

In the HelloBPM example solution, context is based on human task, where the user interface shown depends on the selected task. When users select Sales Quote Entry tasks from the task list, they see additional tabs and actions not shown after selecting Sales Quote Approval tasks.

The HelloBPM solution uses multiple contexts to provide different functionality to three user tasks (sales quote entry users, account managers, and approvers). The base context defines the base set of functionality (functional blocks, lookups, and validators), and the contexts layer on additional functionality to the base (actions and tabs containing data fields and tables), as described in Section 5.3.4.

Contexts form a period-delimited, hierarchical string in decreasing order of specialization, as shown in Example 5–1. Based on runtime context, the AXF for BPM infrastructure traverses down the string until it either finds a rule in one of the contexts or reaches the base context.

Example 5–1 Context Hierarchical String Structure

```
a.b.c
a.b
a
base
```

As a HelloBPM example, you might define the contexts listed in Example 5–2. When the runtime context is SalesQuoteEntry.Specialist, the rules in the SalesQuoteEntry.Specialist context execute. When the runtime context is SalesQuoteEntry, the rules within the SalesQuoteEntry context execute.

Example 5–2 HelloBPM Context Hierarchy

```
SalesQuoteEntry.Specialist
SalesQuoteEntry
base
```

You can create, update, and delete contexts, as described in Section 5.2.6.

- Creating and implementing a context involves identifying one or more of the package's dictionaries to include in the context, then modifying the context's dictionaries.
- When creating contexts, use period delimiters in their names to specify their hierarchy, as shown in this section's examples.
- Deleting a context permanently removes its modified rules, even if the context is in use.

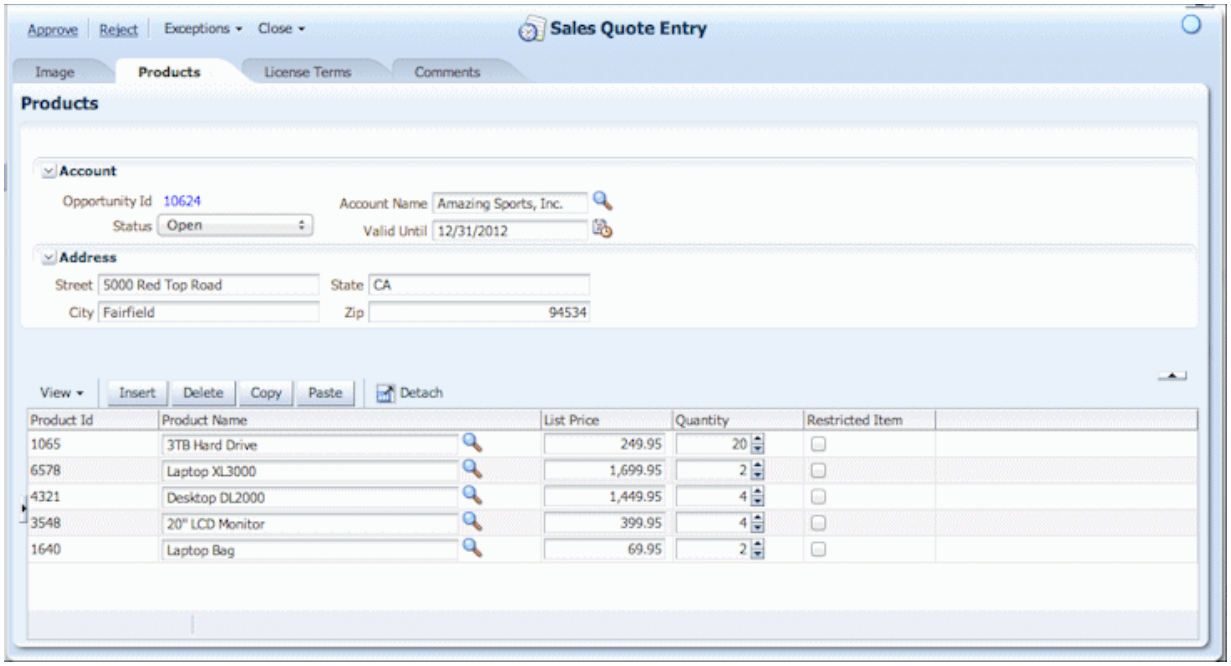
5.1.8 About Data Fields, Tables, and Lookups

Most BPM imaging solutions include business data items that users can view or edit. Using business rules, you can customize how data items display and behave in the solution application. In each case, the data source is the human task payload.

The Master Detail page component displays data items in the solution application's dynamic tabs. You can include and configure the following data controls, which are shown in Figure 5–6 and used in the HelloBPM solution (Section 5.3):

- **Data fields**, which are individual data items for display or editing. See Section 5.1.8.1.
- **Tables**, which are data items displayed as columns. See Section 5.1.8.2.
- **Lookups**, which provide a static or dynamic list of choices; dynamic lookups can also return database values. See Section 5.1.8.3.

Figure 5–6 Data Controls Configured in HelloBPM Solution



5.1.8.1 About Data Fields

Table 5–4 lists the type of data fields you can display to users, with examples shown in Figure 5–6.

Table 5–4 Data and Table Field Types

Type	Appearance	Description
Input text	80920	These fields can accept text and numbers. The **Street, City, State**, and **Zip** fields are input text fields.
Output text	101	These fields, such as **Opportunity Id**, are read-only.
Input date		These fields, such as **Valid Until**, display a **Select Date** icon users click to select a date from a calendar that displays.

Table 5–4 (Cont.) Data and Table Field Types

Type	Appearance	Description
Input number		These fields, such as **Quantity**, display a number spinbox on which users click the upper triangle to increment the number by one, or the lower triangle to decrement the number by one.
Static dropdown		Users click the triangle in these dropdown fields, such as the **Status** field, to choose a selection from the choice list that displays. Static dropdowns are used for static lookups.
Dynamic search		Users click the **Search** icon for these fields, such as the **Account Name** field, and choose a selection from search results that display. If needed, users can specify additional search criteria to narrow the search results. Dynamic search is used for SQL-based lookups.

You can group data fields in sections. Figure 5–6 shows data fields grouped in **Account** and **Address** sections.

Specifying a data field involves adding a new data item to the Data Dictionary for the context in which the field will display, then specifying its properties, including:

- display order and section (if any)
- binding name, which identifies where the value is read and written
- special attributes, such as color
- other special options, such as lookups

See Data Dictionary for configuration options for the Data component.

5.1.8.2 About Tables

Tables are data fields displayed in a column format instead of as single items. For example, the **Product** table shown in Figure 5–6 includes multiple data items arranged in columns and rows, with standard options for inserting, deleting, copying, and pasting table rows, which are useful when users need to duplicate and slightly modify multiple line items.

Tables can contain the data item types described in Table 5–4. You can include the table controls described in Table 5–5.

Table 5–5 Available Table Controls

Control	Description
View	Users select table view options from this menu. ▪ **Columns:** Displays or hides selected columns in the table, when users either select them in the list or choose **Columns**, then **Manage Columns**. ▪ **Detach:** Magnifies the table as described under the Detach element below.
Insert	Enables users to add a new row to the bottom of the table.
Delete	Enables users to remove the selected rows from the table.

Table 5–5 (Cont.) Available Table Controls

Control	Description
Copy	Enables users to copy one or more columns from the selected row for pasting. Select each column to copy or All.
Paste	Enables users to replace selected rows with columns from previously copied rows.
	Enables users to magnify the table so that it fills the Task Detail pane. Users can select the Detach icon again to return the table display to its original view.

Specifying a table involves adding a new table in the TableData Dictionary for the context in which the table will display, then specifying its properties, including:

- binding, which defines where the table rows and columns load
- enabling or disabling Copy, Paste, Delete, and Insert buttons for users to work with rows
- setting options such as column stretching
- setting the columns, including their binding names (which identifies where the values are read and written), display order, column width, and optional lookup

> **Note:** You can specify one table definition in the TableData dictionary.

5.1.8.3 About Lookups

Lookups are data items (individual fields or table columns) linked to a lookup list. For example, Figure 5–6 shows **Status** and **Account Name** as data field lookups and **Product Name** as a table lookup. Lookups work the same way in data fields and tables.

Lookups include the following types:

- **Static**, which link to a fixed list, and display as a dropdown choice list. For example, the **Status** field is a fixed lookup from which users select a value of Open, Pending Approval, or Closed. These lookup values are defined in the StaticLookup ruleset of the LookupAndValidator Dictionary.
- **Dynamic**, which link to a SQL database, and display with a search icon that users click to query. For example, the **Account Name** and **Product Name** fields are dynamic lookups. You can also configure dynamic lookups to autopopulate other data fields when users select an item from a dynamic lookup, as the HelloBPM solution's AccountName lookup populates address fields from the AccountName lookup.

Specifying a lookup involves adding a new lookup in the LookupAndValidator Dictionary for use in one or more contexts. Main steps include:

- For a static lookup, specifying the values and their order.
- For a dynamic lookup, creating the lookup's database configuration in Enterprise Manager, then specifying a SQL query to execute against the specified database connection. In addition, you must specify a binding location where the selected value is stored when users make a selection in the lookup.
- In the Data or TableData dictionary, referencing the new lookup for the data or table field.

5.1.9 About Disabling or Discarding Business Rule Changes

As you edit custom business rules, you may decide to temporarily disable or permanently discard your changes. You might discard changes, for example, if you cannot get the business rule to validate on the Manage Rules tab.

- To view product rules for reference at any point, select **Product Rules** from the **Rule Type** field on the Manage Rules tab. When selected, the **Edit** link changes to **View**, as product rules are read-only.
- To permanently reset a dictionary's custom rules to its default product settings using the **Reset to Product** button, see Section 5.2.7.
- To run a specified solution in product mode, which temporarily suppresses all business rule modifications, see Section 5.6.2.

5.2 Getting Started Configuring the Solution Application

This section covers the following topics:

- Section 5.2.1, "Running or Applying a BPM Imaging Solution"
- Section 5.2.2, "Starting Up Solution Administration"
- Section 5.2.3, "Testing the Solution as a User Through the Command Driver Page"
- Section 5.2.4, "Injecting Tasks for Use in the HelloBPM Solution"
- Section 5.2.5, "Managing Business Rules"
- Section 5.2.6, "Managing Contexts"
- Section 5.2.7, "Resetting a Dictionary's Custom Rules to Product Settings"

5.2.1 Running or Applying a BPM Imaging Solution

After completing installation of the BPM imaging solution, complete one of the following steps for implementation:

- Use the HelloBPM imaging solution, which is automatically configured during installation and described in Section 5.3. Use this solution to verify BPM imaging functionality and to customize for use in exploring and practicing making configuration changes.

  OR

- Apply a solution accelerator. To obtain an accelerator, contact your systems integrator or Oracle Consulting.

5.2.2 Starting Up Solution Administration

In order to access and edit business rules using the Manage Rules link in Solution Administration, you must be assigned to the **axfadmin** group in Oracle WebLogic Server.

To access administration functions for the solution application:

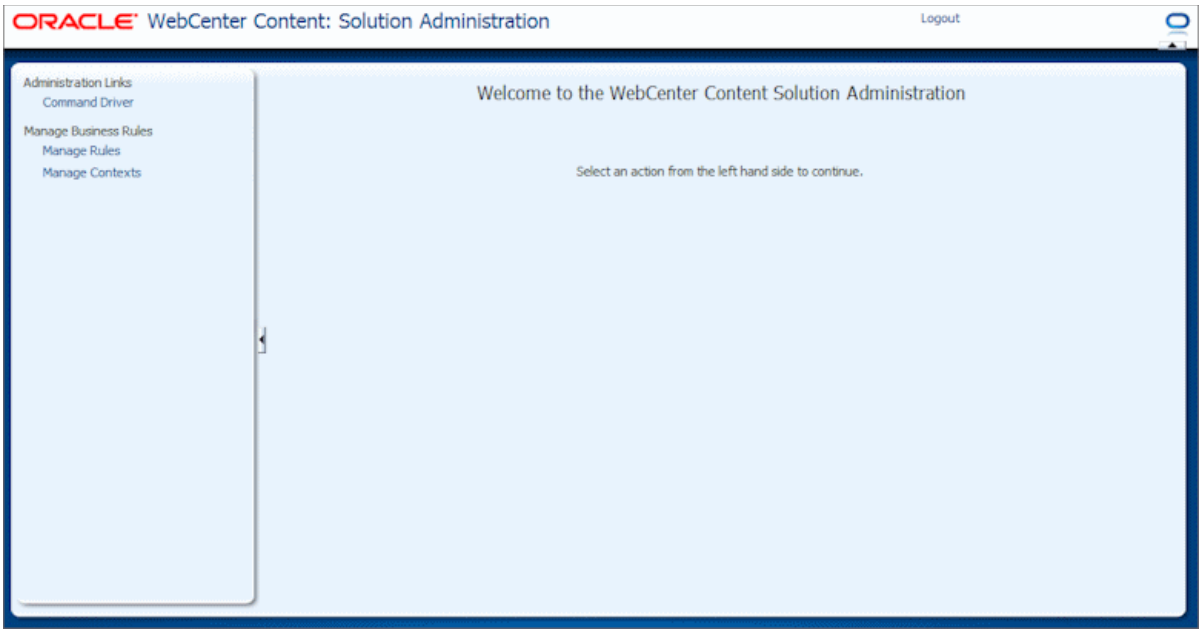
1. Open the Solution Administration page, which uses the following format:

   `http://hostname:port/axf/faces/pages/axfadmin.jspx`

2. Enter your user ID (Oracle WebLogic Server) and password, and click **Sign In**.

   The Solution Administration window displays, as shown in Figure 5–7.

Figure 5–7 Solution Administration Links



3. Click an administration link in the left pane.
   - Click **Command Driver** to run the solution as a user, as described in Section 5.2.3.
   - Click **Manage Rules** to view or update solution dictionaries and their business rules, as described in Section 5.2.5.
   - Click **Manage Contexts** to create, update, or delete variations of business rules for runtime contexts such as specific human tasks, as described in Section 5.2.6.

5.2.3 Testing the Solution as a User Through the Command Driver Page

The Command Driver page allows you to run the AXF application as an end user would, in order to explore functionality or test updates made to business rules and contexts, without needing to launch AXF from a business application or an email link.

Follow these steps to use the command driver:

1. On the Solution Administration page (see Section 5.2.2), click the **Command Driver** link under **Administration Links**.

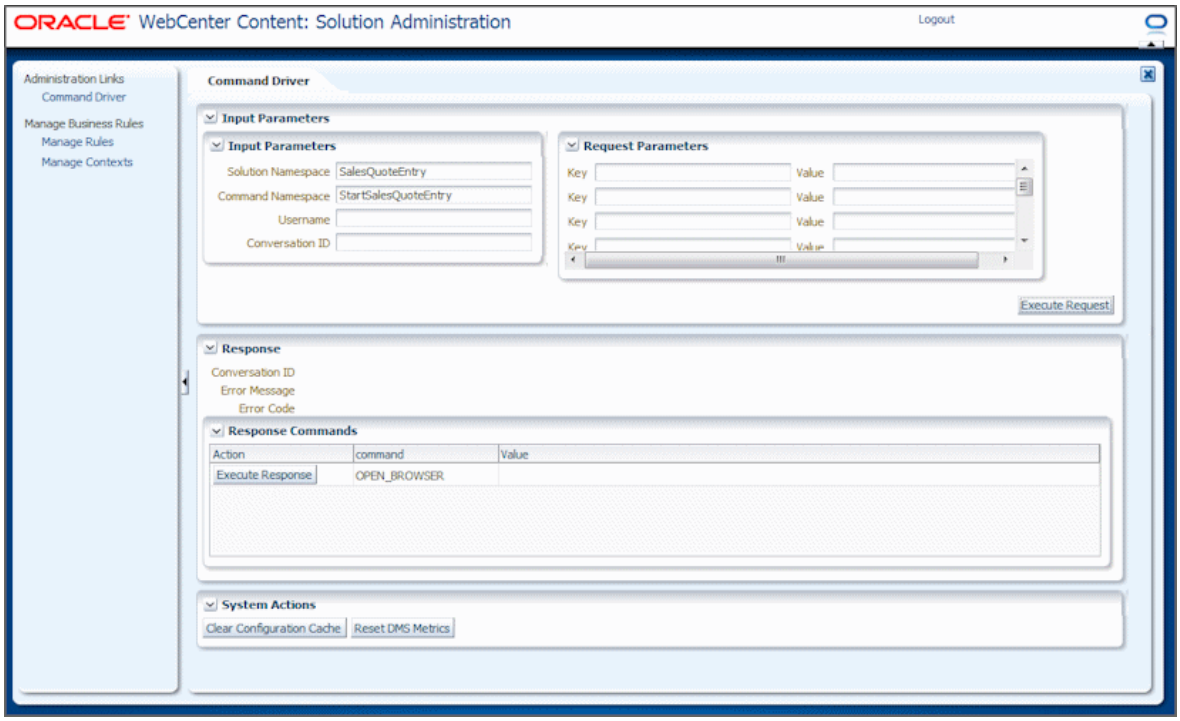
   Command Driver options display.

2. Under **Input Parameters**, enter **`SalesQuoteEntry`** in the **solutionNamespace** field and **`StartSalesQuoteEntry`** in the **commandNamespace** field.

   For the HelloBPM solution, the default context is based on the human task. The human task you select determines the user interface (tables and tabs) that display.

3. Click the **Execute Request** button.

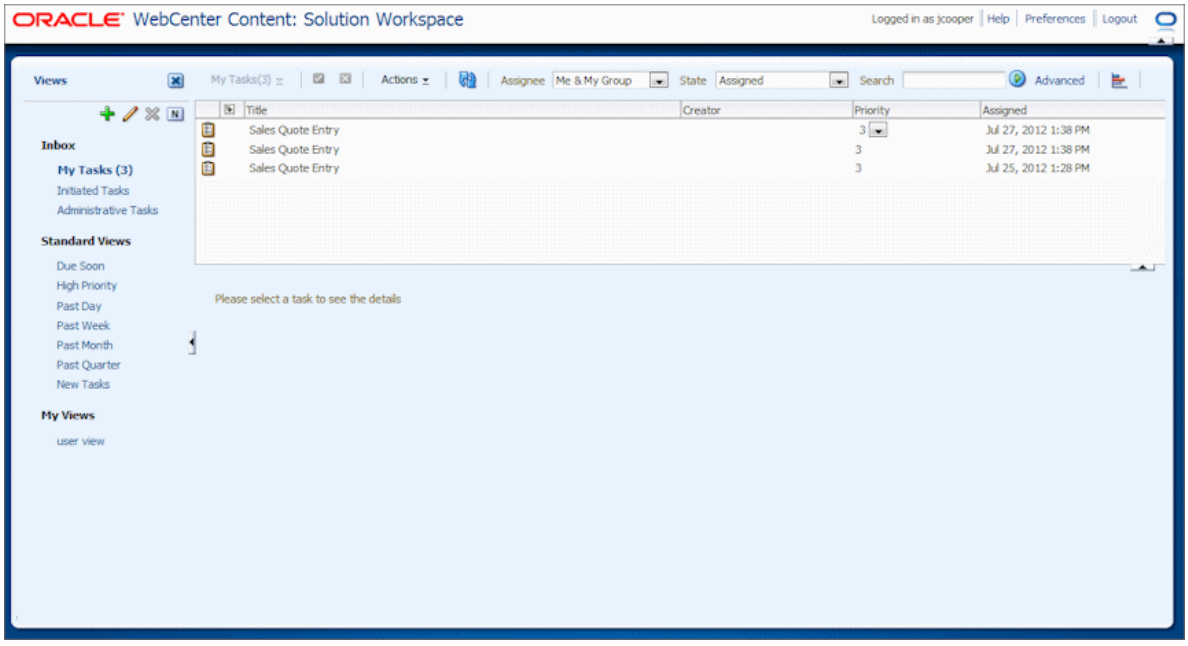
   A conversation starts, as indicated by entries in the **conversationId** and other response fields.



4. Under **Response Commands**, click the **Execute Response** button.

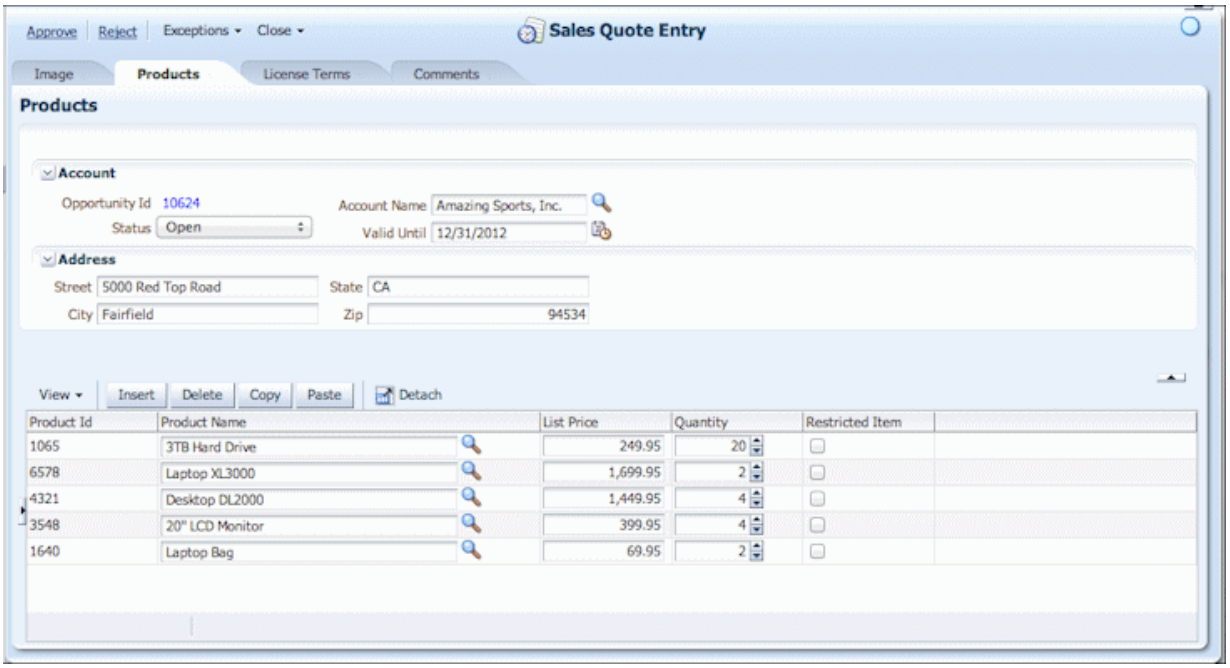
   A new browser window opens and displays the Solution Workspace page.

   In the HelloBPM solution, BPM views are configured for a single administrator user, where all tasks are assigned to a single default view. To inject tasks, see Section 5.2.4.



5. Select a task from the list to view its details in the solution application in the lower right pane.

Note that you can enlarge the solution application view by hiding the Views and Task List panes, or by pinning/unpinning the Views pane, as described in the *Oracle WebCenter User's Guide for Application Adapters*.



6. You can easily switch between the solution application user view and Solution Administration (Section 5.2.2) to make business rule changes and then view their results, after re-selecting a task.

5.2.3.1 Using the Command Executor

The command executor enables you to define a URL link that represents the same information entered in the command driver page. For example, the command executor URL is used for email links to users.

As an example command executor, use the following URL to display the solution workspace with the HelloBPM solution:

```
http://hostname:port/axf/faces/command/CommandExecutor.jspx?sol=SalesQuote
Entry&cmd=StartSalesQuoteEntry
```

After you sign in, the solution workspace you identified displays.

5.2.4 Injecting Tasks for Use in the HelloBPM Solution

Along with the HelloBPM solution automatically installed with the AXF for BPM infrastructure are content input files you can use to inject tasks into the solution. Injecting tasks using Imaging's input agent allows you to test solution application changes. If needed, you can modify the input files to match the HelloBPM workflows described in Section 5.3. For initial configuration details, see "Verifying the AXF for BPM Installation and Configuration with HelloBPM" in *Oracle WebCenter Content Installation Guide*.

The content files are located in the following directory along with HelloBPM.xml, the Imaging application definition.

```
$ECM_ORACLE_HOME/axf_bpm/ipm
```

The sample input files include:

- TestSalesQuote.pdf, which contains the sample image
- TestSalesQuote.txt, which contains the input file
- TestSalesQuote.xml, which contains the document's supporting content

This section describes how tasks are injected, assuming that the default directories configured during adapter installation are being used and that the input agent has been configured to look for files in the input directory listed above. For more information, see "Enabling Input Agent" in *Oracle Fusion Middleware Administrator's Guide for Oracle Imaging and Process Management*.

Within the time interval specified for the input agent (by default, 15 minutes), the input agent picks up the input files and creates a document with the data values contained in the text input file, the image contained in the PDF, and the supporting content contained in the XML file.

Based on the workflow configuration in place with the HelloBPM solution, a task is created for the document that displays in the BPM task list.

1. In the task list, click the newly injected task to view its details in the solution application.
2. As needed, modify the input file's data values before injecting the input files again. For example, you might inject a task with missing account information to work with its human task flow.

5.2.5 Managing Business Rules

This section describes the basis of solution application administration, which is customizing the solution application's user interface for each context, by editing the solution's business rules. For the most part, this process involves opening a context's dictionaries, and selecting or adding rules to rulesets and specifying their properties. For background information about managing business rules, see Section 5.1.2.

1. On the Solution Administration page (see Section 5.2.2), click the **Manage Rules** link under **Manage Business Rules**.

   Manage Rules options display.

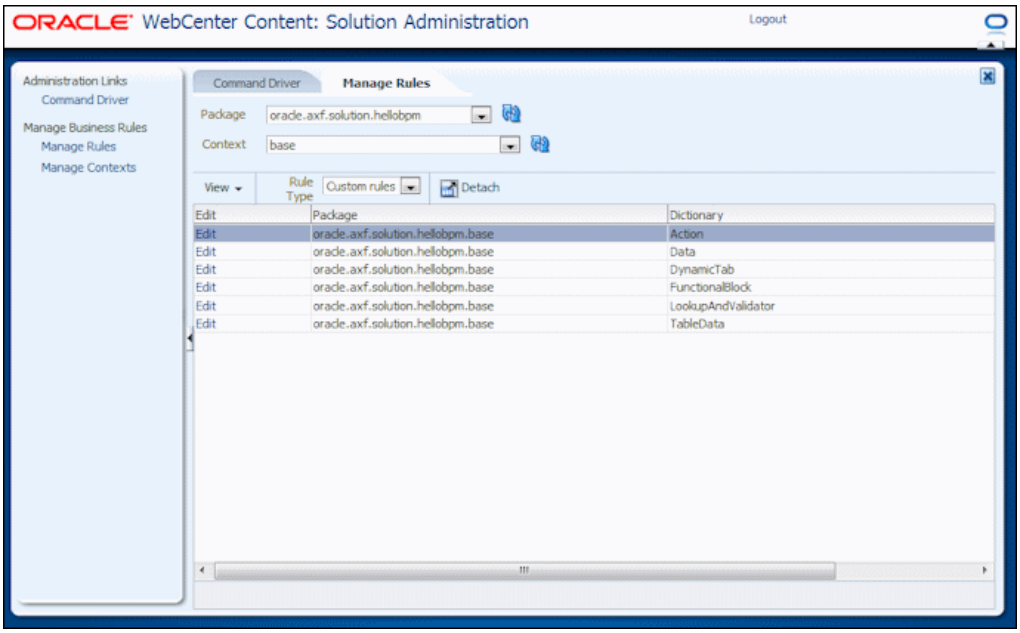
2. In the **Package** field, select a solution package. If needed, click **Refresh** to reload the list of packages.

   For example, choose **oracle.axf.solution.hellobpm** to display business rules for the HelloBPM solution.

3. In the **Context** field, select **base** to edit rules in the base dictionary or a previously created context to edit rules for this context.

   Dictionaries defined for the selected package display in the table, as shown in Figure 5–8.

Figure 5–8 Manage Rules Window, Dictionaries for Base Context



4. Notice that **Custom Rules** is selected in the **Rule Type** field.

   Unlike read-only product rules, you can edit custom rules. You can also revert modified custom rules back to their product defaults, as described in Section 5.2.7.

5. Click the **Edit** link corresponding to the dictionary to edit.

   - Figure 5–9 shows the Manage Rules window that displays when you click the **Edit** link for the DynamicTab dictionary.
   - Table 5–6 describes tools for adding, editing, or deleting rules in the Manage Rules window.

Figure 5–9 Manage Rules Window, Editing DynamicTab Dictionary

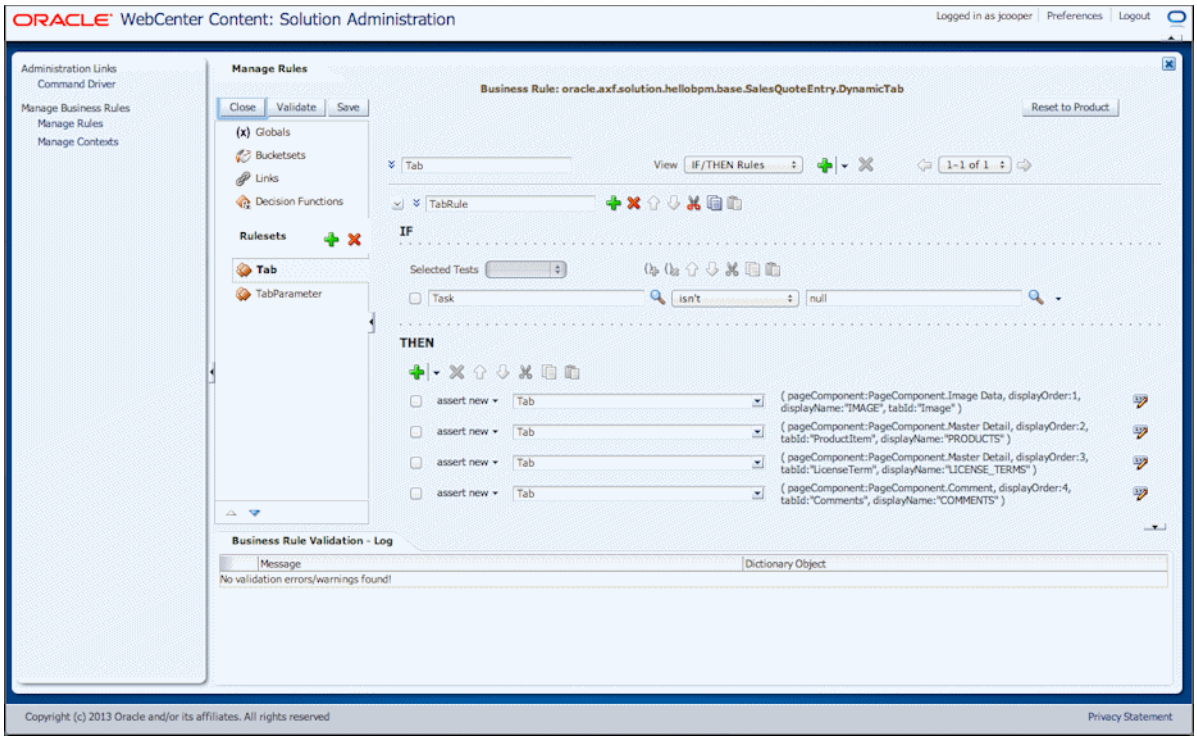


Table 5–6 Manage Rules Window Tools

Elements	Description
Business Rule	This center heading lists the package, context and dictionary you are currently editing. For example, the business rule shown in Figure 5–9 indicates: ■ package (oracle.axf.solution.hellobpm) ■ context (base) ■ dictionary (DynamicTab)
Rulesets	Lists rulesets you can edit for the dictionary. A selected ruleset's rules display at right for editing. Note that rulesets not used contain ruleset placeholders to guide you in adding new items.
Close	Click this button to close the rule without saving business rule changes.
Validate	Click to validate the rule. Results display in the Business Rule Validation log section.
Save	Click to validate and save changes made to the business rule. If a validation error is found, a warning displays.
Reset to Product	Click to discard all changes made to this dictionary's business rules and return the dictionary to its product rule settings. **Note:** You cannot undo this action.
	Click the **Add Action** icon to add a new rule to the selected ruleset.
	Click the **Edit Properties** icon to edit the selected rule's properties.
	Click the **Delete** icon to remove the selected rule from the ruleset.

Table 5–6 (Cont.) Manage Rules Window Tools

Elements	Description
Business Rule Validation Log	Displays validation results for the dictionary after clicking the **Validate** button.

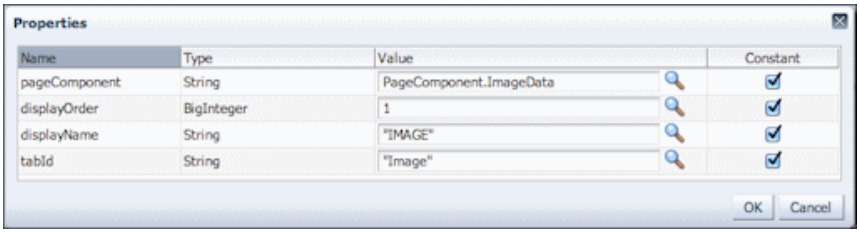
6. To edit a rule, select its ruleset under **Rulesets**. From the THEN items listed, click the rule's **Edit Properties** icon.

Edit the rule's properties, as shown in Figure 5–10. When applicable, clicking the **Value** (magnifying glass) icon displays possible values for the property. For example, clicking the pageComponent property's Value icon in a Tab rule displays available page component values. To guide you in specifying properties, see the dictionary properties reference in Section A.1 and the example settings selected for the HelloBPM solution in Section 5.3.6 and Section 5.3.5.

Notes: Quotation marks must be included when indicated for the property type. Check the **Constant** field to automatically add quotation marks.

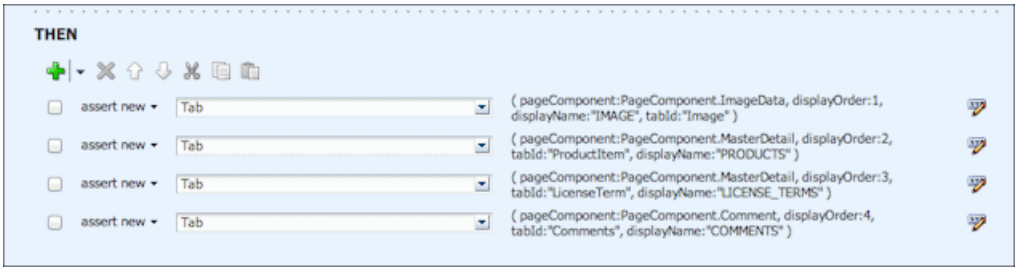
When entering properties, case matters. For example, "salesRepName" and "SalesRepName" are treated as separate entries.

Figure 5–10 Editing a Rule's Properties (DynamicTab Dictionary's Tab Properties)



7. To add a rule, select a ruleset under **Rulesets**. In the section showing THEN items, click the inverted triangle in the **Add Action** icon (shown below) and choose **assert new**, which adds a new item at the bottom of the list of rules.

From the field adjacent to **assert new**, select the item to add. Typically, you select the same type assigned to previous items (for example, add a new tab to the Tab ruleset).



Click the **Edit Properties** icon shown below and edit the rule's properties, as shown in Figure 5–10, and described in step 6. Checking the **Constant** box for each property adds quotation marks only if they are needed.

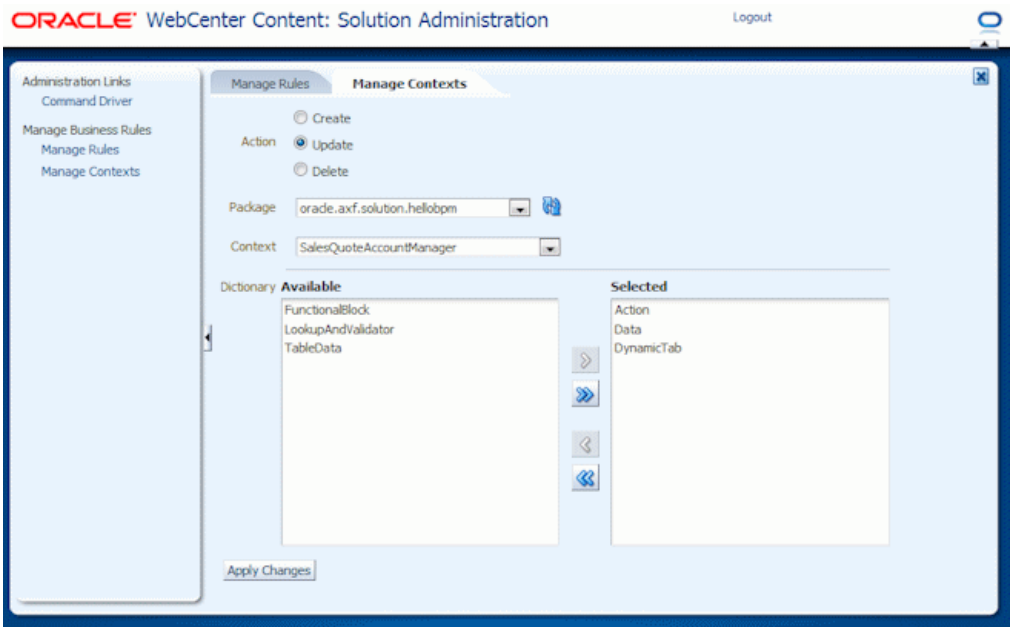
8. Validate changes or additions you made to the rule, then save and close the dictionary.
   - Click the **Save** button, which validates first, warning you of any errors encountered before saving. In the lower portion of the window, a validation log displays errors or warnings found.
   - Click the **Close** button. If you close the dictionary without saving, changes are not saved.
9. Use the command driver page (Section 5.2.3) to view the results.

   By keeping the solution application user view open in another browser window, you can quickly switch between editing and viewing results. Note that to see the results of dictionary changes, you must close and reopen a task.

5.2.6 Managing Contexts

See the following sections to manage contexts. For background information on contexts, see Section 5.1.7.

- Section 5.2.6.1, "Creating a Context"
- Section 5.2.6.2, "Updating a Context"
- Section 5.2.6.3, "Deleting a Context"



5.2.6.1 Creating a Context

Create a context to provide specialization of the base rules.

1. On the Solution Administration page (see Section 5.2.2), click the **Manage Contexts** link under **Manage Business Rules**.

2. From the **Action** choices, select **Create**.
3. In the **Package** field, select the package to which to add a context.
4. In the **Context** field, enter a name for the new context.

   If needed, use a period delimiter in the context name to create a context hierarchy, as described in Section 5.1.7.
5. Move dictionaries to include in the context to the **Selected** field.
6. Click the **Apply Changes** button.
7. Edit the new context's dictionaries according to the business user's role, as described in Section 5.2.5.

5.2.6.2 Updating a Context

Update a context to include or exclude selected dictionaries from its user role.

1. On the Solution Administration page (see Section 5.2.2), click the **Manage Contexts** link under **Manage Business Rules**.
2. From the **Action** choices, select **Update**.
3. In the **Package** field, select the package containing the context to edit.
4. In the **Context** field, select the context to update.
5. Move dictionaries to include to the **Selected** field and move dictionaries to exclude to the **Available** field.
6. Click the **Apply Changes** button.
7. Edit its dictionaries according to the business user's role, as described in Section 5.2.5.

5.2.6.3 Deleting a Context

Deleting a context deletes its selected dictionaries and their business rules, even if in use by a solution.

1. On the Solution Administration page (see Section 5.2.2), click the **Manage Contexts** link under **Manage Business Rules**.
2. From the **Action** choices, select **Delete**.
3. In the **Package** field, select the package containing the context to delete.
4. In the **Context** field, select the context to delete.
5. Click the **Apply Changes** button. If deleting, confirm the deletion.

5.2.7 Resetting a Dictionary's Custom Rules to Product Settings

Resetting a dictionary's custom business rules to product settings permanently removes all business rule modifications you have made and restores the custom rules to product rules. This can be useful if you are receiving errors when running the solution or a customization will not validate. Note that resetting to product rules removes all changes made to the selected dictionary. Note that you must reset each dictionary in a solution separately.

As an alternative, you can temporarily remove all business rule modifications using the RunAsProduct MBean settings, as described in Section 5.6.2.

You can reset dictionary rules for any context, including the base context. Resetting for a context affects the dictionary only in that context.

To reset a dictionary's custom rules to product rules:

1. On the Manage Rules tab, select a package and context.
2. Click the **Edit** link for the rule dictionary. The dictionary opens.
3. Click the **Reset to Product** button. You are prompted to confirm permanently deleting the business rule changes.
4. Click **OK**. A message displays that the product dictionary rules successfully replaced the custom rules.

   When you click a task in the solution application, previous business rule changes no longer display.

5.3 Understanding the HelloBPM Solution

This section covers the following topics:

- Section 5.3.1, "What is the HelloBPM Solution?"
- Section 5.3.2, "Overview of the HelloBPM Sales Quote System"
- Section 5.3.3, "HelloBPM User Roles and Workflow"
- Section 5.3.4, "How the HelloBPM Solution Uses Contexts"
- Section 5.3.5, "Business Rules for the Sales Quote Entry Context"
- Section 5.3.6, "Business Rules for the Base Context"

5.3.1 What is the HelloBPM Solution?

The HelloBPM solution is an example AXF for BPM solution that is automatically installed and configured with the AXF for BPM infrastructure. It provides components to:

1. Verify and learn AXF for BPM functionality as a user.

   Start up the HelloBPM solution as described in Section 5.2.2 and run it as a user as described in Section 5.2.3.

2. Learn to modify business rules, by examining the HelloBPM example and using the use cases.

   Begin solution application configuration as an administrator through Manage Business Rules functions (shown in Figure 5–7) and test solution functionality as a user (shown in Figure 5–11).

   The use cases provide examples of modifying dictionaries and contexts for the HelloBPM solution, while testing and viewing changes as a user.

Note: At any point in modifying the HelloBPM solution's rules, you can reset dictionary rules to their original product settings. See Section 5.2.7.

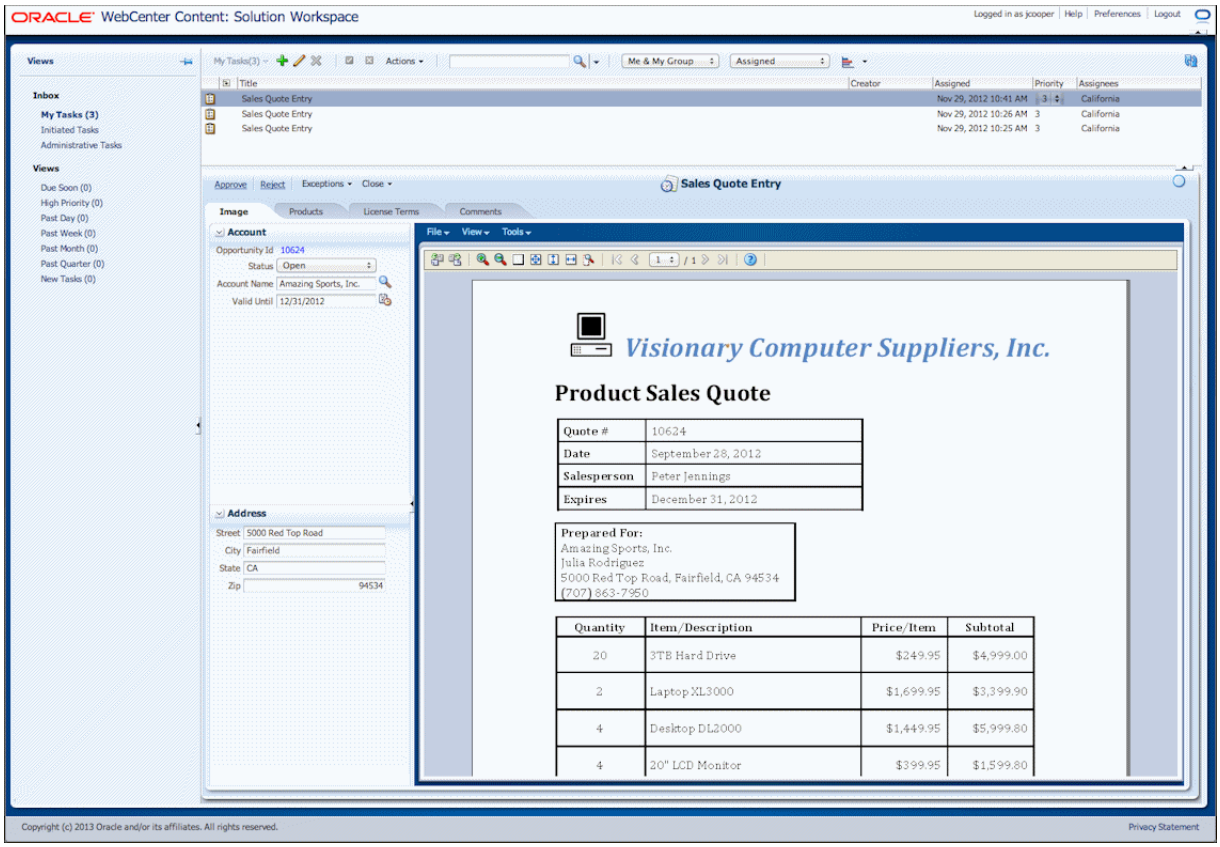
5.3.2 Overview of the HelloBPM Sales Quote System

The HelloBPM solution demonstrates a fictitious sales quote system. The solution includes the following BPM human tasks, which correspond to the three user contexts and roles:

- Sales Quote Entry
- Sales Quote Account Manager
- Sales Quote Approval

Figure 5–11 illustrates the solution application (task detail) that the main user, the Sales Quote Entry user, sees.

Figure 5–11 Viewing HelloBPM Solution as a Sales Quote Entry User



5.3.3 HelloBPM User Roles and Workflow

The HelloBPM solution's basic workflow functions as follows.

1. A sales quote entry task flows into the system. It displays in the task list view as a Sales Entry Quote task.

   A **sales quote entry** user (entry user) selects the task and reviews and enters fields, such as the product and license tables shown in Figure 5–11. This entry user has access to the following action links:

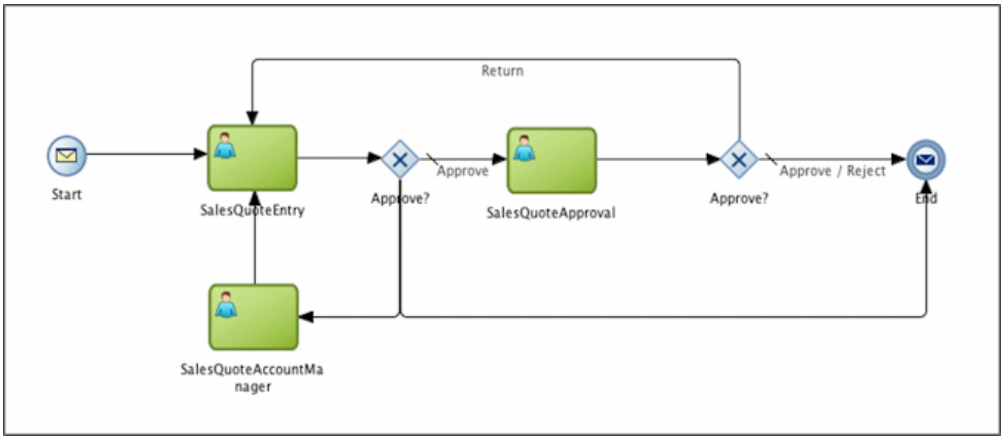
   – **Approve:** When a sales quote entry user clicks this link, the Identity Browser control page displays and the entry user selects an approver. Upon clicking the Identity Browser control page OK action, the task is completed and routed to the SalesQuoteApprover human task.

- **Reject:** When an entry user clicks this link, a prompt displays, asking if the user is sure about the rejection.
- **Account Manager** (in Exceptions section): This link is active to entry users only when there is no entry in the Account Name field. When an entry user clicks it, a Select Account Manager Reason screen where the user selects a reason (for example, new account, invalid address, or expired account).

2. A **sales quote account manager** (account manager) enters the solution workspace and selects a Sales Quote Account Manager task. He or she completes account management updates, saves and closes the task.
3. A **sales quote approver** (approver) enters the solution workspace and selects a Sales Quote Approval task. He or she either approves or rejects the task. If approved, the task is automatically marked as completed and removed from the task list.

The workflow is illustrated in Figure 5–12.

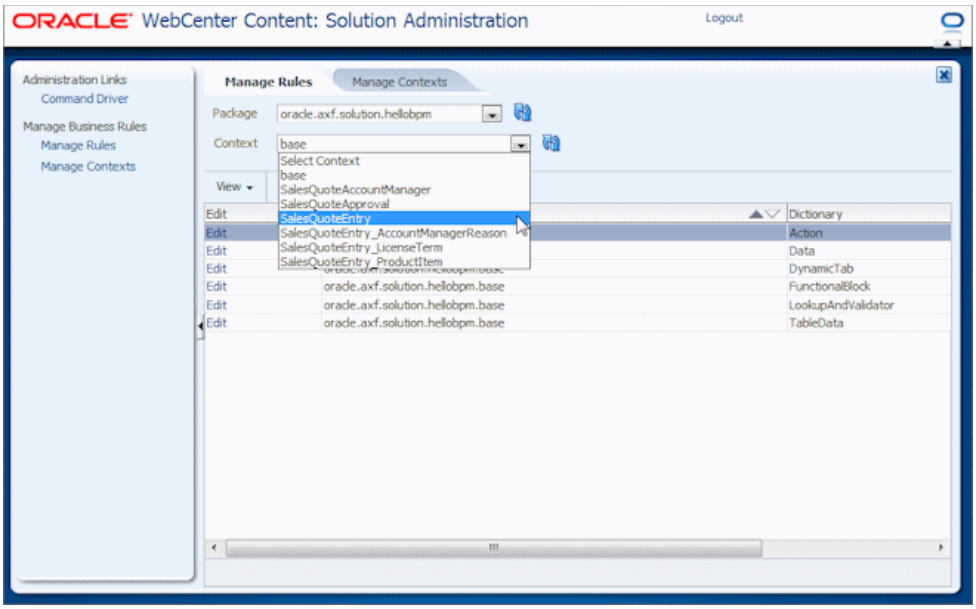
Figure 5–12 Basic Workflow of HelloBPM Solution



5.3.4 How the HelloBPM Solution Uses Contexts

Because each of the three human tasks included in the HelloBPM solution needs access to different data and actions, multiple contexts are configured to display a different and customized user interface depending on the type of task selected. For example, when a user selects a SalesQuoteEntry task, the user sees additional tabs and actions for completing the task.

Figure 5–13 Contexts in HelloBPM Solution



The solution's contexts include the following:

- When users select a **Sales Quote Approval** task, they do not see the Products or License Terms tabs, but have access to Approve and Reject commands and a Comments tab for documenting approval/rejection decisions. The approver context includes the Action, DynamicTab, and Data dictionaries only for defining the actions, tabs, and data fields for this type of task.
- When users select a **Sales Quote Account Manager** task, they see limited functions for setting up or modifying sales accounts. This task can change an organization's contact information and route the task for further data entry or approval. Like approval tasks, account manager context includes only the Action, DynamicTab, and Data dictionaries for defining the actions, tabs, and data fields seen with this type of task.
- When users select a **Sales Quote Entry** task, they see the most specialized task, with additional Products and License Terms tabs and an Exceptions section with an Account Manager action link. This context contains Action, DynamicTab, and Data dictionaries to display the additional tabs, actions, and data fields.
- The **SalesQuoteEntry_ProductItem** and **SalesQuoteEntry_LicenseTerm** contexts contain the TableData dictionary only, in order to display the Products and License Terms tables, respectively.
- The **SalesQuoteEntry_AccountManagerReason** context contains a Data dictionary in order to display a reason lookup data field when a user selects the Account Manager action link.

5.3.5 Business Rules for the Sales Quote Entry Context

- Action Rulesets for the SalesQuoteEntry Context
- Data Rulesets for the SalesQuoteEntry Context
- DynamicTab Rulesets for the SalesQuoteEntry Context
- Data Rulesets for the SalesQuoteEntry_AccountManagerReason Context

- Table Rulesets for the SalesQuoteEntry_ProductItem Context
- Table Rulesets for the SalesQuoteEntry_LicenseTerm Context

5.3.5.1 Action Rulesets for the SalesQuoteEntry Context

Because the Action dictionary is included in the SalesQuoteEntry context, its definitions are used instead of those in the base context. It defines the following actions:

- The **Approve** link, which uses a functional block named SendSalesQuoteEntryToSalesQuoteApproval to perform the action.
- The **Account Manager** command, contained in an action dropdown menu called **Exceptions**. The Account Manager command is disabled unless the Account Name field is empty.
- An action dropdown menu called **Close**, with **Save and Close** and **Revert and Close** commands.

Table 5–7 Action Properties

actionId	displayName	display Order	enableMode	namedFunctionalBlockId	sectionId
"Approve"	"Approve"	1	ENABLE	"SendSalesQuoteEntryToSales QuoteApproval"	
"Reject"	"Reject"	2	ENABLE	"Reject"	
"AccountManager"	"Account Manager"	1	ENABLE	"SendSalesQuoteEntryToSales QuoteAccountManager"	"Exceptions"
"CloseTask"	"Save and Close"	1	ENABLE	"CloseTask"	"Close"
"RevertCloseTask"	"Revert Changes and Close"	2	ENABLE	"RevertCloseTask"	"Close"

Table 5–8 Section Properties

disclosed	displayName	displayOrder	enableMode	sectionId
true	"Exceptions"	1	ENABLE	"Exceptions"
true	"Close"	2	ENABLE	"Close"

5.3.5.2 Data Rulesets for the SalesQuoteEntry Context

The Data dictionary is included in the SalesQuoteEntry context, allowing it to provide the following data fields and sections. (In Table 5–9, the componentClass field contains no values and is not shown.)

Table 5–9 Data Properties

displayName	bindingName	display Order	lookupId	dataId	enableMode	sectionId	validator
"OPPORTUNITY_ID"	"OpportunityID"	1		"OpportunityID"	DISABLE	"Account"	
"STATUS"	"QuoteRequestStatus"	2	"QuoteRequestStatus"	"QuoteRequestStatus"	ENABLE	"Account"	
"ACCOUNT_NAME"	"Account Name"	3	"AccountName"	"AccountName"	ENABLE	"Account"	"Account Name"
"VALID_UNTIL"	"Valid Until"	4		"ValidUntil"	ENABLE	"Account"	
"STREET"	"Street"	1		"Street"	ENABLE	"Address"	

Table 5–9 (Cont.) Data Properties

displayName	bindingName	display Order	lookupId	dataId	enableMode	sectionId	validator
"CITY"	"City"	2		"City"	ENABLE	"Address"	
"STATE"	"State"	3		"State"	ENABLE	"Address"	

Table 5–10 Section Properties

disclosed	displayName	displayOrder	enableMode	sectionId
true	"ACCOUNT"	1	ENABLE	"Account"
true	"ADDRESS"	2	ENABLE	"Address"

Table 5–11 DataComponentAttribute Properties

attributeName	attributeValue	dataId
"contentStyle"	""color:blue"	"OpportunityID"
ComponentAttribute.Converter.Pattern	"MMM dd, yyyy"	"ValidUntil"

5.3.5.3 DynamicTab Rulesets for the SalesQuoteEntry Context

The DynamicTab dictionary is included in the SalesQuoteEntry context, allowing it to provide additional tabs. It defines the following tabs and parameters:

- It adds the Products and License Terms tabs.
- In addition to setting the title and positioning, the parameters ruleset sets data and table context for the new tabs. For example, it sets the Product Items tab to use the SalesQuoteEntry_ProductItem context for table settings but the SalesQuoteEntry context for the actual data fields.

Table 5–12 Tab Properties

pageComponent	display Order	tabId	displayName
PageComponent.Image Data	1	"Image"	"Image"
PageComponent.Master Detail	2	"ProductItem"	"Products"
PageComponent.Master Detail	3	"LicenseTerm"	"License Terms"
PageComponent.Comment	4	"Comments"	"Comments"

Table 5–13 TabParameter Properties

tabId	parameterKey	parameterValue
"Image"	"HideData"	"false"
"Image"	"SplitterPosition"	"250"
"ProductItem"	"TableDataContext"	"SalesQuoteEntry_ProductItem"
"ProductItem"	"DataContext"	"SalesQuoteEntry"
"ProductItem"	"PageComponentTitle"	"Products"
"LicenseTerm"	"TableDataContext"	"SalesQuoteEntry_LicenseTerm"
"LicenseTerm"	"DataContext"	"SalesQuoteEntry"
"LicenseTerm"	"PageComponentTitle"	"License Terms"

5.3.5.4 Data Rulesets for the SalesQuoteEntry_AccountManagerReason Context

The Data dictionary is included in the SalesQuoteEntry_AccountManagerReason context to provide a data lookup field from which the entry user selects an account manager reason. (In Table 5–14, the componentClass and Validator fields contain no values and are not shown.)

Table 5–14 Data Properties

displayName	bindingName	display Order	lookupId	dataId	sectionId	enableMode
"Account Manager Reason"	"NewCustomer"	1	"AccountManager Reason"	"AccountManager Reason"	"AccountManager Reason"	ENABLE

Table 5–15 Section Properties

disclosed	displayName	displayOrder	enableMode	sectionId
true	"Select Account Manager Reason"	1	ENABLE	"AccountManagerReason"

5.3.5.5 Table Rulesets for the SalesQuoteEntry_ProductItem Context

The TableData dictionary is included in the SalesQuoteEntry_ProductItem context to display the Product table to sales quote entry users but not other contexts.

Table 5–16 Table Properties

bindingName	enableCopyPaste	enableDelete	enableInsert	tableId	tableStretchMode
"ProductItem"	ENABLE	HIDE	ENABLE	"ProductItem"	TableStretchMode.Blank

(In Table 5–17, the Validator field contains no values and is not shown.)

Table 5–17 Column Properties

bindingName	columnId	componentClass	displayName	display Order	enableMode	column Width	lookupId
"ProductID"	"ProductID"		"Product ID"	1	DISABLE		
"ProductName"	"ProductName"		"Product Name"	2	ENABLE	300	"ProductName"
"ListPrice"	"ListPrice"		"PRICE"	3	ENABLE		
"Quantity"	"Quantity"	ComponentClass. RichInputNumber Spinbox	"Quantity"	4	ENABLE		
"RestrictedItem"	"RestrictedItem"	ComponentClass. RichSelectBoolean Checkbox	"Restricted Item"	5	ENABLE		

Table 5–18 ColumnComponentAttribute Properties

attributeName	columnId	attributeValue
ColumnComponentAttribute.pattern	"ListPrice"	"#{bindings.ListPrice.format}"

5.3.5.6 Table Rulesets for the SalesQuoteEntry_LicenseTerm Context

The TableData dictionary is included in the SalesQuoteEntry_LicenseTerm context to display the License Terms table to sales quote entry users but not other contexts.

Table 5–19 Table Properties

bindingName	enableCopyPaste	enableDelete	enableInsert	tableId	tableStretch
"LicenseTerm"	ENABLE	ENABLE	ENABLE	"LicenseTerm"	TableStretchMode.Last

Table 5–20 Column Properties

bindingName	columnId	component Class	displayName	display Order	enableMode	column Width	lookupId
"Category"	"Category"		"Category"	1	ENABLE	300	"LicenseTermCategory"
"Type"	"Type"		"Type"	2	ENABLE	300	"LicenseTermType"
"Description"	"Description"		"Description"	3	ENABLE		

5.3.6 Business Rules for the Base Context

The **base** context of the HelloBPM solution provides a base layer of functionality upon which the other contexts layer additional functionality and reference its functional blocks and lookups.

- FunctionalBlock Rulesets for the Base Context
- LookupAndValidator Rulesets for the Base Context

5.3.6.1 FunctionalBlock Rulesets for the Base Context

The following rulesets define the functional blocks, sequences, and parameters for use in the solution.

NamedFunctionalBlockRuleset sets outcome, task detail dynamic page, and ability to navigate to the identity browser, data, and comment control pages.

Table 5–21 NamedFunctionalBlock Properties

functionalBlock	namedFunctionalBlockId
FunctionalBlock.Set Outcome	"Approve"
FunctionalBlock.Set Outcome	"Reject"
FunctionalBlock.Set Outcome	"AccountManager"
FunctionalBlock.Close Task	"RevertCloseTask"
FunctionalBlock.Navigate Control Page	"SelectSalesQuoteApprover"
FunctionalBlock.Navigate Control Page	"SelectAccountManagerReason"
FunctionalBlock.Navigate Dynamic Page	TaskDetail"
FunctionalBlock.Navigate Control Page	"IdentityBrowserControlPage"
FunctionalBlock.Navigate Control Page	"DataControlPage"
FunctionalBlock.Navigate Control Page	"CommentControlPage"
FunctionalBlock.Save	"Save"
FunctionalBlock.Close Task	"CloseTask"
FunctionalBlock.Revert	"Revert"

Table 5–22 NamedFunctionalBlockSequence Properties

namedFunctionalBlockId	functionalBlockSequenceId	order
"SelectSalesQuoteApprover"	"SendSalesOrderEntryToSalesOrderApproval"	1

Table 5–22 (Cont.) NamedFunctionalBlockSequence Properties

namedFunctionalBlockId	functionalBlockSequenceId	order
"Approve"	"SendSalesOrderEntryToSalesOrderApproval"	2
"SelectAccountManagerReason"	"SendSalesQuoteEntryToSalesQuoteAccountManager"	1
"AccountManager"	"SendSalesQuoteEntryToSalesQuoteAccountManager"	2

Table 5–23 NamedFunctionalBlockParameter Properties

namedFunctionalBlockId	parameterKey	parameterValue
"Approve"	"Outcome"	"APPROVE"
"Reject"	"Outcome"	"REJECT"
"AccountManager"	"Outcome"	"ACCOUNT_MANAGER"
"SelectSalesQuoteApprover"	"ControlPageName"	"IdentityBrowserControlPage"
"SelectSalesQuoteApprover"	"ControlPageTitle"	"Select Approver"
"SelectSalesQuoteApprover"	"UserBindingName"	"UserBindingName"
"SelectSalesQuoteApprover"	"GroupBindingName"	"GroupBindingName"
"SelectSalesQuoteApprover"	"CommentRequired"	"true"
"SelectAccountManagerReason"	"ControlPageName"	"DataControlPage"
"SelectAccountManagerReason"	"DataContext"	"SalesQuoteEntry_AccountManagerReason"
"TaskDetail"	"pageTitle"	SolutionApplicationProperty.TaskTitle
"DataControlPage"	"ControlPageName"	"DataControlPage"
"DataControlPage"	"ControlPageTitle"	
"IdentityBrowserControlPage"	"ControlPageName"	"IdentityBrowserControlPage"
"IdentityBrowserControlPage"	"ControlPageTitle"	
"IdentityBrowserControlPage"	"UserBindingName"	"UserBindingName"
"IdentityBrowserControlPage"	"GroupBindingName"	"GroupBindingName"
"CommentControlPage"	"ControlPageName"	"CommentControlPage"
"CommentControlPage"	"ControlPageTitle"	

Table 5–24 NamedFunctionalBlockPrompt Properties

namedFunctionalBlockId	order	promptMessage	promptMode	promptType
"Reject"	1	"Are you sure you would like to reject this request?"	PromptModeType.AFTER_VALIDATION	PromptType.OK_CANCEL_FUNCTIONAL_BLOCK
"RevertCloseTask"	1	"Are you sure you would like to revert any changes before closing?"	PromptModeType.BEFORE_VALIDATION	PromptType.ROLLBACK_SAVED_CHANGES

Table 5–25 NamedFunctionalBlockValidator Properties

namedFunctionalBlockId	validatorMode	validatorCategoryId
"Reject"	ValidatorMode.CONTINUE	"Account"
"RevertCloseTask"	ValidatorMode.CONTINUE	"Account"
"CloseTask"	ValidatorMode.CONTINUE	"Account"

5.3.6.2 LookupAndValidator Rulesets for the Base Context

The following rulesets define lookups, prompts, and validations used elsewhere in the solution.

Table 5–26 StaticLookup Properties

displayName	lookupId	value	display Order
"Open"	"QuoteRequestStatus"	"OPEN"	1
"Pending Approval"	"QuoteRequestStatus"	"PENDING_APPROVAL"	2
"Closed	"QuoteRequestStatus"	"CLOSED"	3
"Price Hold Options"	"LicenseTermCategory"	"PriceHoldOptions"	1
"Non-Standard Pricing And Currency Options"	"LicenseTermCategory"	"NonStandardPricingAndCurrencyOptions"	2
"License Management Services Options"	"LicenseTermCategory"	"LicenseManagementServicesOptions"	3
"Non-Standard Licensing Options"	"LicenseTermCategory"	"NonStandardLicensingOptions"	4
"Future Program Price Holds"	"LicenseTermType"	"FutureProgramPriceHolds"	1
"Future Discount Provisions"	"LicenseTermType"	"FutureDiscountProvisions"	2
"Price Hold On Entire Price List"	"LicenseTermType"	"PriceHoldOnEntirePriceList"	3
"Modifications To Contractual Pricing"	"LicenseTermType"	"Modifications ToContractualPricing"	4
"Currency"	"LicenseTermType"	"Currency"	5
"Preferred Customer Provisions"	"LicenseTermType"	"PreferredCustomerProvisions"	6
"Waiving Or Lowering Purchase Minimums"	"LicenseTermType"	"WaivingOrLoweringPurchaseMinimums"	7
"Audit Waivers"	"LicenseTermType"	"AuditWaivers"	8
"Cancel And Replace"	"LicenseTermType"	"CancelAndReplace"	9
""Direct Order From Competitors"	"LicenseTermType"	"DirectOrderFromCompetitors"	10
"Reciprocal Transaction"	"LicenseTermType"	"ReciprocalTransaction"	11
"Hosting"	"LicenseTermType"	"Hosting"	12
"Hosting Rights"	"LicenseTermType"	"HostingRights"	13
"NDA"	"LicenseTermType"	"NDA"	14
"New Account"	"AccountManagerReason"	"NEW_ACCOUNT"	1
"Invalid Address"	"AccountManagerReason"	"INVALID_ADDRESS"	2
"Expired Account"	"AccountManagerReason"	"EXPIRED_ACCOUNT"	3

Table 5–27 DynamicLookup Properties

sqlColumn	dataSource	lookupId	lookupType	sqlQuery
"NAME"	"/jdbc/IPMDS"	"AccountName"	LookupType.DYNAMIC_SEARCH	"select ACCOUNT_NAME as Name, ADDRESS_STREET as Street, ADDRESS_CITY as City, ADDRESS_STATE as State, ADDRESS_ZIP as Zip, ADDRESS_COUNTRY as Country from AXF_HELLOBPM_ACCOUNTS"
"PRODUCT"	"jdbc/IPMDS"	"ProductName"	LookupType.DYNAMIC_SEARCH	"select PRODUCT_ID as ID, PRODUCT_NAME as Product, LIST_PRICE as Price from AXF_HELLOBPM_PRODUCTS"

Table 5–28 DynamicLookupRelatedBinding Properties

bindingName	lookupId	sqlColumn
"Street"	"AccountName"	"STREET"
"City"	"AccountName"	"CITY"
"State"	"AccountName"	"STATE"
"Zip"	"AccountName"	"ZIP"
"Country"	"AccountName"	"COUNTRY"
"ListPrice"	"ProductName"	"PRICE"
"ProductID"	"ProductName"	"ID"

Table 5–29 Validator Properties

validatorClass	validatorId
ValidatorClass."RegExpValidator"	"AccountName"
ValidatorClass."RegExpValidator"	"State"

Table 5–30 ValidatorParameter Properties

parameterKey	parameterValue	validatorId
"Pattern"	"\\S"	"AccountName"
"MessageDetailNoMatch"	"You must enter an Account Name before requesting approval"	"AccountName"
"Pattern"	"AL | AK | AS | AZ | AR | CA | CO | CT | DE | DC | FM | FL | GA | GU | HI | ID | IL | IN | IA | KS | KY | LA | ME | MH | MD | MA | MI | MN | MS | MO | MT | NE | NV | NH | NJ | NM | NY | NC | ND | MP | OH | OK | OR | PW | PA | PR | RI | SC | SD | TN | TX | UT | VT | VI | VA | WA | WV | WI | WY"	"State"
"MessageDetailNoMatch"	"You must enter a valid state abbreviation"	"State"

5.4 HelloBPM Solution Use Cases

The use case examples provided in this section are intended to help you learn how to implement common scenarios by example.

> **Note:** The HelloBPM solution is installed during AXF installation. For more information, see "Verifying the AXF for BPM Installation and Configuration with HelloBPM" in *Oracle WebCenter Content Installation Guide*.

Action and Functional Block Use Cases

- Section 5.4.1, "Use Case: Add and Configure an Action Link"
- Section 5.4.2, "Use Case: Navigate to a Data Control Page"
- Section 5.4.3, "Use Case: Add a New Context"

Data and Table Use Cases

- Section 5.4.4, "Use Case: Add a Data Field in a New Section"
- Section 5.4.5, "Use Case: Set a Data Field's Background Color"

- Section 5.4.6, "Use Case: Change the Display Order of Data Items or Sections"
- Section 5.4.7, "Use Case: Set a Section to Initially Display Closed"
- Section 5.4.8, "Use Case: Hide all Table Row Buttons"

Lookup and Validator Use Cases

- Section 5.4.9, "Use Case: View a Static Lookup's Configuration"
- Section 5.4.10, "Use Case: View a Dynamic Lookup's Configuration"

5.4.1 Use Case: Add and Configure an Action Link

This exercise leads you through adding a link called **Return** to the action menu to complete the sales quote approval task and return it to the sales quote entry task queue. You will add the action, then configure functional block settings for it.

1. Under Manage Rules, select the HelloBPM package (oracle.axf.solution.hellobpm) in the **Package** field.
2. Select the **SalesQuoteApproval** context and edit its Action dictionary.
3. In the **Action** ruleset, add an action by adding an **assert new Action** and assigning it the following properties:

Property	Value
actionId	"Return"
displayName	"Return"
displayOrder	4
enableMode	EnableMode.ENABLE
namedFunctionalBlockId	"Return"
sectionId	

4. Validate, save, and close the dictionary.

   The next step is to configure the functional blocks behind the new action, specifically to create a call to BPM to complete the task.
5. Under Manage Rules, select the **base** context and edit its FunctionalBlock dictionary.
6. Under **Rulesets**, choose **NamedFunctionalBlock**. Add an **assert new NamedFunctionalBlock** and assign it the following properties:

Property	Value
functionalBlock	FunctionalBlock.Set Outcome
namedFunctionalBlockId	"Return"

7. Under **Rulesets**, choose **NamedFunctionalBlockParameter**. Add an **assert new NamedFunctionalBlockParameter** and assign it the following properties:

Property	Value
namedFunctionalBlockId	"Return"

Property	Value
parameterKey	"Outcome"
parameterValue	"RETURN"

8. Validate, save, and close the dictionary.
9. Test the configuration so far.
   a. From the command driver page, execute the solution as SalesQuoteEntry/StartSalesQuoteEntry.
   b. In the task list, select a Sales Quote Approval task. (In order for a task to appear as a Sales Quote Approval task, you must approve a Sales Quote Entry task.)
   c. From the action menu, click the new **Return** link you added. The task is removed from the task list, indicating that it was returned to Sales Quote Entry.

5.4.2 Use Case: Navigate to a Data Control Page

This exercise leads you through configuring a means for users to select a reason for returning the sales quote request. Clicking the Return link will navigate to a data control page (predefined page that displays data) that will display reason codes from which the user selects.

1. From the HelloBPM package, select the **base** context and edit its FunctionalBlock dictionary.

   First specify navigation to a control page.

2. In the **NamedFunctionalBlock** ruleset, add an **assert new NamedFunctionalBlock** and assign it the following properties:

Property	Value
functionalBlock	FunctionalBlock.Navigate Control Page
namedFunctionalBlockId	"ReturnReason"

   Then specify the control page to navigate to.

3. Under **Rulesets**, choose **NamedFunctionalBlockParameter**. Add an **assert new NamedFunctionalBlockParameter** and assign it the following properties:

Property	Value
namedFunctionalBlockId	"ReturnReason"
parameterKey	"ControlPageName"
parameterValue	"DataControlPage"

   Next define a sequence of calls, with the first to navigate to the Return Reason page.

4. Under **Rulesets**, choose **NamedFunctionalBlockSequence**. Add an **assert new NamedFunctionalBlockSequence** and assign it the following properties:

Property	Value
namedFunctionalBlockId	"ReturnReason"
namedFunctionalBlockSequenceID	"ReturnToSalesQuoteEntry"
order	1

In the next call, define the return to complete the task.

5. Add another **assert new NamedFunctionalBlockSequence** and assign it the following properties:

Property	Value
namedFunctionalBlockId	"Return"
namedFunctionalBlockSequenceID	"ReturnToSalesQuoteEntry"
order	2

6. Validate, save, and close the dictionary.

   The last step is to update the action to call the new sequence you defined.

7. Select the **SalesQuoteApproval** context and edit its Action dictionary.
8. Update the action you created in step 3 of Section 5.4.1, by changing the namedFunctionalBlock from "Return" to "ReturnToSalesQuoteEntry".

Property	Value
actionId	"Return"
displayName	"Return"
displayOrder	4
enableMode	EnableMode.ENABLE
namedFunctionalBlockId	**"ReturnToSalesQuoteEntry"**
sectionId	

9. Validate, save, and close the dictionary.
10. Test the configuration.
    a. Return to the solution application and select a Sales Quote Approval task from the task list.
    b. From the action menu, click the **Return** link. The new return reason data page displays according to the navigation you configured.

       The return reason data page displays the same data as displayed on other pages. Follow the steps in Section 5.4.3 to change its data display.
    c. Click the **Cancel** link.

5.4.3 Use Case: Add a New Context

This exercise leads you through adding a context for the return reason data page in order to display the appropriate data (return reason codes). First you create the

context, then configure the data field to display along with its choices, and finally configure the data page to use the new context.

1. From the left-most menu, choose **Manage Contexts**.
2. In the **Package** field, select the HelloBPM solution (oracle.axf.solution.hellobpm).
3. In the **Context** field, enter a name of **`SalesQuoteApproval_ReturnReason`**.
4. Move the Data dictionary to the **Selected** field, and click **Apply Changes**.

   Now you define the attributes of the data to display for the return reason data page. You will add a data field called **Return Reason** and a section that instructs the user to select a reason code from the field.
5. Choose **Manage Rules**. Click the **Refresh** icon adjacent to the **Context** field, then select the new context, **SalesQuoteApproval_ReturnReason**, you created. Edit its Data dictionary.
6. Under the **Data** ruleset, edit the placeholder rule and assign it the following properties:

Property	Value
displayName	"Return Reason"
bindingName	"NewCustomer"
displayOrder	1
lookupId	"ReturnReason"
dataId	"ReturnReason"
sectionId	"ReturnReason"
enableMode	EnableMode.ENABLE
componentClass	

7. Select the **Section** ruleset, edit the placeholder rule and assign it the following properties:

Property	Value
disclosed	true
displayName	"Select the return reason:"
displayOrder	1
enableMode	EnableMode.ENABLE
sectionId	"ReturnReason"

8. Validate, save, and close the dictionary.

   The next step is to configure the reason codes for the Return Reason field, which you will define as static lookups.
9. Select the **base** context and edit its LookupAndValidator dictionary.
10. Under the **StaticLookup** ruleset, add an **assert new StaticLookup** rule and assign it the following properties:

Property	Value
displayName	"Invalid License Terms"
displayOrder	1
lookupId	"ReturnReason"
value	"INVALID_LICENSE_TERMS"

11. Add another StaticLookup rule and assign it the following properties:

Property	Value
displayName	"Invalid Product Discount"
displayOrder	2
lookupId	"ReturnReason"
value	"INVALID_PRODUCT_DISCOUNT"

12. Validate, save, and close the dictionary.
13. Open the **base** context's FunctionalBlock dictionary.

    The last step is to identify the new context, SalesQuoteApproval_ReturnReason, as the context to use for the return reason data page.
14. Under **Rulesets**, select **NamedFunctionalBlockParameter**. Add an **assert new NamedFunctionalBlockParameter** and assign it the following properties:

Property	Value
namedFunctionalBlockId	"ReturnReason"
parameterKey	"DataContext"
parameterValue	"SalesQuoteApproval_ReturnReason"

15. Validate, save, and close the dictionary.
16. Test the final configuration.
    a. Return to the solution application and select a Sales Quote Approval task from the task list.
    b. From the action menu, click the **Return** link.
    c. From the data control page that displays, choose a reason from the **Return Reason** field, and click **OK**. The task is returned and removed.

5.4.4 Use Case: Add a Data Field in a New Section

This exercise leads you through adding a data item (read-only input text field) called **Sales Rep Name** to the Data dictionary to display within a section called **Sales Info**. The steps to create other data field types, such as an output (editable) text field, date input field, or number input field are nearly the same, as shown in the variations at the end of this section.

1. First add the text field. Under Manage Rules, select the **SalesQuoteApproval** context in the **oracle.axf.solution.hellobpm** package.
2. Edit the **Data** dictionary.

3. Under Rulesets, select **Data**.
4. Add an **assert new Data** with the following values:

Property	Value
bindingName	"SalesRepName"
componentClass	ComponentClass.RichInputText
displayName	"Sales Rep Name"
displayOrder	
lookupId	
dataId	"salesRepName"
enableMode	EnableMode.ENABLE
sectionId	"sales"

5. Now add a section for the text field. Under Rulesets, select **Section**.
6. Add an **assert new Section** with the following values:

Property	Value
disclosed	true
displayName	"Sales Info"
displayOrder	3
enableMode	EnableMode.ENABLE
sectionId	"sales"

7. Validate and save the dictionary.
8. Now view the text field. From the Command Driver page, execute SalesQuoteEntry/StartSalesQuoteEntry, and select a Sales Quote Approval task.

   Result: On the Image tab, a new data section called **Sales Info** displays that contains the new input text **Sales Rep Name** field.

Variations

The data item's **componentClass** property defines its type. In step 4 above, specifying **ComponentClass.RichInputText** defines an input text field.

- To define a date input field, specify **ComponentClass.RichInputDate** for the componentClass property.
- To define a number input field, specify **ComponentClass.RichInputNumberSpinBox** for the componentClass property.

A data item or section's **enableMode** property defines its enable mode. In steps 4 and 6 above, specifying **EnableMode.ENABLE** enables the field and section, respectively.

- To display a field or section as disabled, specify **EnableMode.DISABLE** for the enableMode property.
- To hide a field or section, specify **EnableMode.HIDE** for the enableMode property.

5.4.5 Use Case: Set a Data Field's Background Color

This use case sets a background color to the Sales Rep Name data field you added in Section 5.4.4.

1. Open the **SalesQuoteApproval** context in the **oracle.axf.solution.hellobpm** package.
2. Edit the **Data** dictionary.
3. Under **Rulesets**, select **DataComponentAttribute**.
4. Add an **assert new DataComponentAttribute** with the following values:

Property	Value
attributeName	"contentStyle"
attributeValue	"background-color:lightblue"
dataId	"salesRepName"

5. Validate, save, and close the dictionary.
6. From the Command Driver page, execute SalesQuoteEntry/StartSalesQuoteEntry.
7. Select a Sales Quote Approval task.

   Result: On the Image tab, the Sales Rep Name field displays with a light blue background.

Variations

In the use case attribute value above, you specified **`"background-color:lightblue"`** for a light blue field background. To specify a field entry's color, specify an attribute value such as one of the following:

`"color:red"`

`"color:blue"`

5.4.6 Use Case: Change the Display Order of Data Items or Sections

This use case shows you how to reverse the order of data sections and data items within a section.

1. Under Manage Rules, open the **SalesQuoteApproval** context in the **oracle.axf.solution.hellobpm** package.
2. Edit the **Data** dictionary.
3. Under **Rulesets**, select **Data**.
4. Change the displayOrder property for each data item with a sectionId of Account so the data items are displayed in reverse order.
5. Under **Rulesets**, select **Section**.
6. Change the displayOrder property of each section so the sections are displayed in reverse order.
7. Validate, save, and close the dictionary.
8. From the Command Driver page, execute SalesQuoteEntry/StartSalesQuoteEntry.
9. Select a Sales Quote Approval task.

Result: On the Image tab, the sections display in reverse order and the Account section's data fields display in reverse order.

5.4.7 Use Case: Set a Section to Initially Display Closed

This exercise sets a section to initially display collapsed to users.

1. Under Manage Rules, open the **SalesQuoteApproval** context in the **oracle.axf.solution.hellobpm** package.
2. Open the **Data** dictionary.
3. Under Rulesets, select **Section**, and edit the Sales Info section by changing the disclosed property to **false**.
4. Validate and save the dictionary.
5. From the Command Driver page, execute SalesQuoteEntry/StartSalesQuoteEntry.
6. Select a Sales Quote Approval task.

   Result: The Sales Info section displays in a collapsed state.

> **Note:** If a user manually uncollapses a section, it stays uncollapsed in subsequent page loads due to personalization.

5.4.8 Use Case: Hide all Table Row Buttons

This exercise suppresses display of the Insert, Delete, Copy, and Paste buttons that when enabled allow users to edit table rows.

1. Under Manage Rules, open the **SalesQuoteEntry_LicenseTerm** context in the oracle.axf.solution.hellobpm package.

   This use case uses the license term table, but you can edit any table as described.
2. Edit the **TableData** dictionary.
3. Under **Rulesets**, select **Table**.
4. Edit the existing Table item so its properties are as follows:

Property	Value
bindingName	"LicenseTerm"
enableCopyPaste	**EnableMode.HIDE**
enableDelete	**EnableMode.DISABLE**
enableInsert	**EnableMode.HIDE**
tableId	"LicenseTerm"
tableStretch	TableStretchMode.Last

5. Validate and save the dictionary.
6. In the solution application, select a Sales Quote Entry task and select the License Term tab.

   Result: The table on the License Terms tab displays a disabled Delete button with the Copy, Paste, and Insert buttons hidden.

5.4.9 Use Case: View a Static Lookup's Configuration

This use case familiarizes you with configuring a static lookup and using one as a user.

1. Under Manage Rules, open the **base** context in the oracle.axf.solution.hellobpm package.
2. Edit the **LookupAndValidator** dictionary.
3. Under **Rulesets**, select **StaticLookup**.

   Notice that three entries contain a lookupId of **QuoteRequestStatus**.
4. From the Command Driver page, execute SalesQuoteEntry/StartSalesQuoteEntry.
5. Select a Sales Quote Entry task.

   Result: The Status field displays as a dropdown with three values.
6. Select **Pending Approval**.
7. Select the **Save and Close** action from the Close menu.
8. Select the same task again.

   Result: The Status field displays as a dropdown with three values and **Pending Approval** is selected.

5.4.10 Use Case: View a Dynamic Lookup's Configuration

Configuring a dynamic lookup requires specifying a datasource configured in Oracle WebLogic Server (for example, */jdbc/IPMDS*, where IPMDS was automatically configured upon installation and creation of the IPM_server domain needed to host AXF). Follow the steps below to view how a dynamic lookup is configured in the LookupAndValidator dictionary, and referenced in the Data or TableData dictionary for the context.

1. Under Manage Rules, open the **base** context in the oracle.axf.solution.hellobpm package.
2. Edit the **LookupAndValidator** dictionary.
3. Under **Rulesets**, select **DynamicLookup**.

   Notice that each of the DynamicLookup entries uses **/jdbc/IPMDS** as its datasource. Examine the entry with **lookupId** set to **AccountName**, and notice that it queries a sample accounts table. This will return a list of account names.
4. Close the LookupAndValidator dictionary, select the **SalesQuoteEntry** context and open its Data dictionary. In the **Data** ruleset, find the Account Name data item and view its properties. Notice that "AccountName" is specified for its LookupId property.
5. From the Command Driver page, execute SalesQuoteEntry/StartSalesQuoteEntry.
6. Select a Sales Quote Entry task.

   Result: Notice that the Account Name field displays with a Search icon (magnifying glass) next to it.
7. Select the **Search** icon next to the **Account Name** field. Select an account from the list and select OK.

   Result: The Account Name field is now populated with the account name you chose.
8. Select the **Save and Close** task action.

9. Select the same task again.

   Result: Notice that the Account Name field remains populated with the account name you chose.

5.5 Configuring BPM Views

The solution workspace makes use of standard BPM views and task list functionality for displaying human workflow tasks controlled by an AXF solution. Use the BPM Worklist application to create BPM views and share them with other users or groups, as described in *Oracle Fusion Middleware Developer's Guide for Oracle SOA Suite*.

Users can perform actions in the task list, such as sorting tasks by clicking a column heading and searching for specific tasks using the **Search** field. For details, see the *Oracle WebCenter User's Guide for Application Adapters*.

5.6 Using Oracle Fusion Middleware Control to Manage AXF for BPM

System administrators use Oracle Fusion Middleware Control (also referred to as Enterprise Manager) to manage certain aspects of AXF for BPM functionality. To perform these tasks, you must be a system administrator with Enterprise Manager access.

This section covers the following topics:

- Section 5.6.1, "Redeploying AXF for BPM Libraries and Applications"
- Section 5.6.2, "Temporarily Turning Off Business Rule Modifications"
- Section 5.6.3, "Moving from a Test to a Production AXF for BPM Environment"

You can also set AXF for BPM logging functions in Enterprise Manager. See Section 4.1.

5.6.1 Redeploying AXF for BPM Libraries and Applications

AXF for BPM is deployed as applications and libraries to Weblogic Server. The solution workspace and solution application are deployed to the IPM Managed Server.

Table 5–31 describes the libraries and applications deployed with AXF for BPM. For more information about deploying, see *Oracle WebCenter Content Installation Guide*.

Table 5–31 Applications and Libraries Deployed With AXF for BPM

Item Name	Type	WLS Managed Server	Description
axf-solutionWorkspace.ear	Application	IPM	Contains the solution workspace application, which displays a list of tasks users can select to bring up a solution application.
axf-infrastructure.ear	Application	IPM	Contains the infrastructure code which includes the Solution Mediator, Command Mediator, Commands, Business Rule Framework, Solution Workspace ADF Shared Library, Solution Application ADF Shared Library, Web Services, and other common infrastructure code.
axf-common-lib.war	Library	IPM	Contains common code used across all other Application and Libraries. In addition to general common code, this WAR contains code for the System MBean configuration, Solution Mediator Interface, Command Mediator Interface, Command Interface, Logging, and Conversation Management.

5.6.2 Temporarily Turning Off Business Rule Modifications

Suppressing business rule modifications is useful when troubleshooting solution application configurations. The MBeans described below allow you to run a specified solution application with product business rules, rather than modified custom rules applied, and then reapply the modified rules when needed.

MBean Setting	Description
RunAsProduct	When set to true, this MBean disables all workspace customizations for the application specified in the RunAsProductSolutionApplicationValue MBean, running the specified AXF application in its initial deployed state. Change this setting to either disable customizations (true) or reinstate modified business rules and customizations (false).
RunAsProductSolutionApplicationValue	Specifies the AXF solution application to disable or enable, according to the RunAsProduct MBean setting. This value is the solution deployment name found in Console Deployments. For the HelloBPM solution, this is **axf-solutionApplication-HelloBPM**. Change this setting to disable customizations in a different solution application.

To change the RunAsProduct MBeans:

1. Open the AXF for BPM MBean in the Enterprise Manager System MBean browser.
   a. In Enterprise Manager, expand **Weblogic Domain**.
   b. Right-click **Base Domain**, and select **System MBean Browser**.
   c. Under Application Defined MBeans, expand **oracle.ecm.axf**, expand the **IPM Server** entry, and display the **config** MBean.
2. Change **RunAsProduct** and **RunAsProductSolutionApplicationValue**, as needed.

5.6.3 Moving from a Test to a Production AXF for BPM Environment

After moving the BPM Imaging solution from a test to a production environment, as described in "Moving from a Test to a Production Environment" in *Oracle Fusion Middleware Administrator's Guide*, perform the following steps. These steps update the foreign JNDI of the production environment, to ensure that communication with the production environment is not linked to the test environment.

1. At left in the Weblogic Server Administration Console, select **Domain Structure**, then **Services**, then **Foreign JNDI Providers**.
2. Click the Foreign JNDI provider named **ForeignJNDIProvider-SOA** that was copied as part of the test to production process.
3. Update the **Provider URL** field to identify the production Oracle SOA Suite.
4. Update the **User** and **Password** fields as they apply to the production environment.
5. Click **Save**.

6

Configuring the BPEL Imaging Solution

This chapter describes how to configure Imaging solution components and options, including configuring the BPEL connection using a Credential Store Framework (CSF) alias, configuring AXF tables, and using a solution accelerator. It also describes options such as clustering and alternate authentication and login.

This chapter covers the following topics:

- Section 6.1, "Configuring the BPEL Connection"
- Section 6.2, "Configuring the AXF Tables or Applying a Solution Accelerator"
- Section 6.3, "Testing Functionality Using the AXF Driver Page"
- Section 6.4, "Configuring Imaging Solution Options"

Note: You can provide access to the Imaging solution from Oracle E-Business Suite Oracle Application Framework (OAF) applications. For information, see Section 2.3.3.

Note: To make certain modifications to Imaging solutions, such as changing the profiles displayed to users in the Task List and task actions and metadata displayed to users in the Task Viewer, you can use the Imaging Solution Editor. For information about the Solution Editor, see "Solutions Editor Page" in *Oracle Fusion Middleware Administrator's Guide for Oracle Imaging and Process Management*.

6.1 Configuring the BPEL Connection

Configuring the BPEL connection for use by an AXF solution involves the following tasks:

- Section 6.1.1, "Creating a CSF Credential Alias"
- Section 6.1.2, "Creating a Connection in Imaging Connections"
- Section 6.1.3, "Referencing the Connection in the AXF_SOLUTION_ATTRIBUTES Table"

6.1.1 Creating a CSF Credential Alias

The Credential Store Framework (CSF) enables you to create a user name/password alias for use in an Imaging connection configuration. With a CSF alias, you supply a

key instead of a user name and password, and use this key in creating an Imaging connection. (You can use one CSF key for multiple imaging connections.)

For information about creating keys and aliases, see the *Oracle Fusion Middleware Administrator's Guide*.

Note: When implementing an AXF solution using MS LDAP as a user repository, the credential user must be the administrator belonging to the Administrators group within the Builtin CN in MS LDAP. The AXF uses this credential user when making certain calls (for example, when retrieving a task list from SOA). In addition, you must configure the Group Base DN (in WLS Console, Provider Specific) to import CN=Builtin.

6.1.2 Creating a Connection in Imaging Connections

Follow these steps to create a connection and specify the CSF alias key, BPEL server name, and port.

1. Log in to the Imaging system as an administrator.
2. From Manage Connections in the side pane, click the + (plus) sign document icon for creating a BPEL connection.
3. Enter a name for the connection, and click **Next**.

   This name is referenced in the AXF_SOLUTION_ATTRIBUTES table to establish the connection.
4. On the BPEL Settings step, enter BPEL connection settings.
   - HTTP Front End Address: http://hostname:BPEL server port
   - Credential Alias (previously created, as described in Section 6.1.1)
   - Provider: t3://hostname:BPEL server port

   For example:
   - HTTP Front End Address: `http://hostname:port`
   - Credential Alias: `axf.credential`
   - Provider: `t3://hostname:port`
5. Click **Next**, then **Submit**.

6.1.3 Referencing the Connection in the AXF_SOLUTION_ATTRIBUTES Table

Follow this step to identify the Imaging connection to the AXF solution, as described in Section 6.1.2. Run the command from SQL Developer (or other suitable tool that can connect to the imaging database schema).

Note: If using an implementation accelerator (including the HelloBPEL solution), this step is not needed.

---

1. Run the configuration row specified below, where:

```
Insert into AXF_SOLUTION_ATTRIBUTES (SOLUTION_NAMESPACE,PARAMETER_KEY,PARAMETER_VALUE) values \
('InvoiceProcessing','BPEL_CONNECTION','axfconnection');
```

6.2 Configuring the AXF Tables or Applying a Solution Accelerator

After completing installation and configuration of the Imaging solution, complete one of the following steps for implementation:

- Configure the AXF tables and AXF-related Oracle E-Business Suite or Oracle PeopleSoft tables. Table descriptions and example implementations are provided in Appendix B, "BPEL Imaging Solution Reference."

  OR

- Apply a solution implementation accelerator. To obtain an accelerator, contact your systems integrator, Oracle Consulting, or Oracle Support.

6.3 Testing Functionality Using the AXF Driver Page

Access the driver page of the AXF web application to verify functionality. For more information about the driver page, see "Verifying the AXF Installation with HelloWorld" in *Oracle WebCenter Content Installation Guide*.

6.4 Configuring Imaging Solution Options

This section describes the following optional configurations for the Imaging solution:

- Section 6.4.1, "Configuring Automatic Imaging Viewer Login"
- Section 6.4.2, "Allowing Oracle WebLogic Server/Business Application Users to Perform Tasks"
- Section 6.4.3, "Configuring Clustering for AXF in Imaging Managed Servers"
- Section 6.4.4, "Configuring Autotask Locking"
- Section 6.4.5, "Updating the Task Payload Using XPATH"
- Section 6.4.6, "Targeting a Single Conversation With Multiple Window Instances Displayed"
- Section 6.4.7, "Adding a Validation"
- Section 6.4.8, "Deploying Custom Commands"
- Section 6.4.9, "Configuring Chained Commands and User Interface Components"
- Section 6.4.10, "Configuring or Disabling a Dynamic Data Table"
- Section 6.4.11, "Reenabling Paperclip Attachments (Oracle E-Business Suite and Imaging Only)"
- Section 6.4.12, "Resizing Oracle PeopleSoft Popup Solution Windows"

6.4.1 Configuring Automatic Imaging Viewer Login

Follow the steps in this section to prevent users from having to log in to Imaging the first time they access the Imaging viewer per session. You set the front end HTTP host and port so that the Imaging hostname and the AXF server hostname match.

Follow these steps to set the front end HTTP host and port:

1. Open the Oracle WebLogic Server Administration Console.
2. On the Home Page, click **Servers** under the Environment heading.
3. Click the Imaging server from the servers listed in the **Name** column.

4. Click the Protocols tab, then the HTTP tab.
5. Make changes in the **Frontend Host** field and appropriate frontend port field. (If using SSL, specify a value in the **Frontend HTTPS Port** field. If not using SSL, specify a value in the **Frontend HTTP Port** field.)
6. Click **Save**.

6.4.2 Allowing Oracle WebLogic Server/Business Application Users to Perform Tasks

You can allow the authenticated Oracle WebLogic Server user to perform tasks, such as BPEL tasks, rather than the user passed from Oracle E-Business Suite or Oracle PeopleSoft. This allows you to use different user stores for Oracle E-Business Suite/Oracle PeopleSoft and the Oracle WebLogic Server.

The related AXF_SOLUTION_ATTRIBUTES table configuration is shown in Table 6–1; also see Section B.1.2.

Table 6–1 AXF_SOLUTION_ATTRIBUTES Table, USERNAME_PASS_THROUGH Configuration

SOLUTION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
AXF	USERNAME_PASS_THROUGH	*TRUE* or *FALSE*

where:

- TRUE (default): Uses the passed-in user from Oracle E-Business Suite or Oracle PeopleSoft to perform tasks.
- FALSE: Uses the authenticated Oracle WebLogic Server user to perform tasks.

Note: For Managed Attachments configurations, the USERNAME_PASS_THROUGH parameter should be set to TRUE. The AXF_ATTACHMENTS_CONFIG.SQL script for Managed Attachments (described in Section 7.2.1) sets this value to TRUE. If no value is provided, the value defaults to TRUE. Also see Section 7.6.19.

Note: After changing the value of USERNAME_PASS_THROUGH in the database, stop and start the Imaging application in the Oracle WebLogic Server Administration Console.

6.4.3 Configuring Clustering for AXF in Imaging Managed Servers

For Oracle Application Extension Framework (AXF) in Imaging Managed Servers that run in a cluster, you need to configure a Java Object Cache (JOC) to be distributed to all of the Managed Servers. See "Configuring the Java Object Cache for AXF in Distributed Imaging Managed Servers" in *Oracle WebCenter Content Installation Guide*.

6.4.4 Configuring Autotask Locking

In AXF configurations with multiple simultaneous users, collisions may occur when end users attempt to acquire tasks in Autotask mode. (For details about Autotask mode, see Section B.3.2.) To prevent collisions, enable autotask locking for each named BPEL connection in the AXF database. When locking is enabled, only one user may automatically acquire a task at a given time.

Enabling the lock functionality prevents an error from appearing on the Task List if two users acquire a task simultaneously, and is the recommended setting. In situations where simultaneous acquisition is unlikely, disabling the lock functionality may increase performance.

The setting is configured in the AXF_SOLUTION_ATTRIBUTES Table by inserting the following row:

SOLUTION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
BPEL.default	USE_AUTOTASK_LOCKING	TRUE

6.4.5 Updating the Task Payload Using XPATH

The Update Task From Procedure command calls a stored pl/sql procedure using a specified data source and updates the task payload using XPATH, as described in Section B.3.7.

6.4.5.1 Example PL/SQL Procedure For Updating the Task Payload

The pl/sql procedure that follows loads the xml into the DOM, retrieves the invoice ID, queries for the invoice amount for that transaction, and based on that amount, returns a set of users.

To use this example, modify this procedure to retrieve the specific pieces of data from the payload you would like. The only requirement is that the pl/sql function you create must take a VARCHAR2, set up two VARCHAR2 out parameters, and return an INTEGER. The name of the function is in the AXF configuration.

```
create or replace FUNCTION axfretrieveuserlist(xmlPayload IN VARCHAR2, pDataString OUT VARCHAR2,
pMsgString OUT VARCHAR2) RETURN INTEGER IS

    v_node    xmldom.DOMNode;
    v_node2   xmldom.DOMNode;
    v_nl      xmldom.DOMNodeList;
    v_doc     xmldom.DOMDocument;
    v_elem    xmldom.DOMElement;
    v_parser  xmlparser.Parser;
    invoiceID     VARCHAR2(256);
    invoiceAmount NUMBER(8,2);
    userList      VARCHAR2(256);
    rtnCode      INTEGER;

BEGIN

    rtnCode := 0;  -- execute CMD_NO_DATA
    -- rtnCode := -1; -- execute CMD_ERROR
    pmsgstring := ''; -- empty msg string (task is updated with this value)
    v_parser := xmlparser.newParser;
    xmlparser.parseBuffer(v_parser, xmlPayload);
    v_doc := xmlparser.getDocument(v_parser);
    xmlparser.freeParser(v_parser);

    -- Retrieve the invoice ID
    v_nl := xmldom.getElementsByTagName(v_doc, 'invoiceID');
    v_node := xmldom.item(v_nl, 0);
    v_node2 := xmldom.getFirstChild(v_node);
    invoiceID := xmldom.getNodeValue(v_node2);

    -- Retrieve Invoice Amount for given invoice id
```

```
    select INVOICE_AMOUNT into invoiceAmount from ap_invoices_all where INVOICE_ID = invoiceid;

    if invoiceamount > 10000 then
      pdatastring := 'jlondon';
      rtnCode := 1;  -- execute CMD_DATA
    else
      pdatastring := 'jcooper,mtwain';
      rtnCode := 1;  -- execute CMD_DATA
      pmsgstring := '';
    end if;

    RETURN rtnCode;

END;
```

6.4.6 Targeting a Single Conversation With Multiple Window Instances Displayed

You can configure the Oracle E-Business Suite adapter to target the same instance/conversation when multiple instances of a particular Oracle E-Business Suite window are displayed. Enable this feature using the **AXF_MULTINSTANCE** parameter in the AXF_PROPERTIES Table (Oracle E-Business Suite).

- If AXF_MULTINSTANCE is set to TRUE, each Oracle E-Business Suite window is treated as a separate conversation and a separate Imaging or AXF window is targeted.
- If AXF_MULTINSTANCE is set to FALSE (default), each Oracle E-Business Suite instance window is treated as the same conversation and uses the same Imaging or AXF target window.

6.4.7 Adding a Validation

The Validate Task command validates BPEL system attribute or BPEL payload data, and based on validation results, executes a subsequent command, as described in Section B.3.9.

The following example and corresponding steps add a validation that verifies that a Transaction ID is present before allowing a task to complete. This example assumes that you have installed the Invoice Processing solution implementation accelerator data.

> **Note:** Apply this configuration change only in use cases where users must create the business application invoice before completing the task. This configuration would not apply in use cases where users may not create an invoice before completing the task (typically, for example, when the task is being completed with an outcome of SupplierMaintenance).

1. Add the following row to the AXF_COMMANDS table:

Table 6–2 Example AXF_COMMANDS Table

SOLUTION_ NAMESPACE	COMMAND_CLASS	COMMAND_ NAMESPACE
InvoiceProcessing	oracle.imaging.axf.commands.bpel.ValidateTaskCommand	ValidateTransactionID

2. Add the rows shown in Table 6–3 to the AXF_SOLUTION_PARAMETERS table.

The following configuration validates that the invoice has been saved (Invoice Transaction ID is not 0). If it is 0, the command reports the error message specified in the FAIL_MESSAGE parameter.

Fields not shown: SOLUTION_NAMESPACE=InvoiceProcessing

Table 6–3 Example ValidateTask Command in AXF_SOLUTION_PARAMETERS Table

COMMAND_ NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_ KEY	PARAMETER_VALUE
ValidateTransactionID	oracle.imaging.axf.commands.bpel. ValidateTaskCommand	ATTRIBUTE_TO_ VALIDATE	XPATH:InvoiceProcessing_TransactionID
ValidateTransactionID	oracle.imaging.axf.commands.bpel. ValidateTaskCommand	CMD_ON_PASS	CompleteInvoice
ValidateTransactionID	oracle.imaging.axf.commands.bpel. ValidateTaskCommand	REGULAR_ EXPRESSION	[^0]
ValidateTransactionID	oracle.imaging.axf.commands.bpel. ValidateTaskCommand	FAIL_MESSAGE	Please save the transaction before completing the task.

3. In the AXF_ACTIONS Table, edit the row in which the Complete Task is configured, replacing the Complete action's COMMAND_NAMESPACE column with the ValidateTransactionID's command namespace.

Table 6–4 AXF_ACTIONS Table

ACTION_ID	VIEW_ID	DISPLAY_NAME	COMMAND_NAMESPACE	MENU_ORDER
CompleteInvoice	/TaskViewer.jspx	Complete Invoice	ValidateTransactionID	3

6.4.8 Deploying Custom Commands

You can also deploy custom commands to work within the AXF infrastructure. Custom commands must implement the oracle.imaging.axf.commands.AxfCommand interface. The execute(AxfRequest) method is invoked by the infrastructure. Configure the implementation to execute in the AXF configuration database.

In addition, commands may implement the oracle.imaging.axf.commands.ValidatableCommand interface, which provides a way for the AXF infrastructure to validate the configuration and operation of a command without executing it to provide a system command status.

6.4.9 Configuring Chained Commands and User Interface Components

Some AXF commands have parameter keys that specify what occurs after the command completes, allowing you to chain them. For example, Table 6–5 shows a portion of the AXF_SOLUTION_PARAMETERS table. After the CompleteTask command executes, additional AXF commands are executed (StartInvoiceProcessing and AutoOpenTask, based on program logic).

Table 6–5 Example AXF_SOLUTION_PARAMETERS Table for CompleteTask Command (InvoiceProcessing Solution)

COMMAND_ NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
DuplicateInvoice	oracle.imaging.axf.commands.bpel.CompleteTaskCommand	CMD_AUTOTASK_OFF	StartInvoiceProcessing
DuplicateInvoice	oracle.imaging.axf.commands.bpel.CompleteTaskCommand	CMD_AUTOTASK_ON	AutoOpenTask
DuplicateInvoice	oracle.imaging.axf.commands.bpel.CompleteTaskCommand	OUTCOME	DUPLICATE_INVOICE

6.4.10 Configuring or Disabling a Dynamic Data Table

In the Task Viewer, you can display a table of dynamic data from the BPEL payload XML, such as General Ledger lines for an invoice processing solution, as shown in the bottom tabs in Figure 1–4. You configure the table in the AXF_METADATA_BLOCKS Table and its data lines in the AXF_METADATA_ATTRIBUTES Table. For information on formatting XML data in the BPEL payload, see Section 6.4.10.1; also see Section B.2.3.

The bottom row of Table 6–6 shows an example dynamic data table called *GL Lines* configured.

Table 6–6 Example AXF_METADATA_BLOCKS Table

BLOCK_ ID	BLOCK_LOCATION	LABEL	DISPLAY _ORDER	TASK_FLOW_ID	SOLUTION_ NAMESPACE	BLOCK_ TYPE	METADATA _STYLE
1	LEFT_SIDEBAR	Summary	1	axf-taskviewer-tfd	InvoiceProcessing	METADATA	null
2	LEFT_SIDEBAR	Comments	2	axf-taskviewer-tfd	InvoiceProcessing	COMMENT	null
3	BOTTOM_PANEL	GL Lines	3	axf-taskviewer-tfd	InvoiceProcessing	METADATA	TABLE

Table 6–7 shows the GL Lines table's data lines configured. This example results in three data columns in the table. It assumes that the XPATH attributes exist in the AXF_XPATH_ATTRIBUTES Table.

Fields not shown include: DATA_TYPE=String

Table 6–7 Example AXF_METADATA_ATTRIBUTES Table for Dynamic Data Table

BLOCK_ID	ATTRIBUTE_ID	LABEL	ATTRIBUTE_KEY	IS_XPATH	DISPLAY _ORDER
2	8	Line Number	DistributionLines_LineNumber	TRUE	0
2	9	Dist Account	DistributionLines_DistributionAccount	TRUE	1
2	10	Amount	DistributionLines_Amount	TRUE	2

6.4.10.1 Formatting XML Data For a Dynamic Data Table

After adding the table in the AXF_METADATA_BLOCKS Table and configuring its data lines in the AXF_METADATA_ATTRIBUTES Table, follow the guidelines below to ensure that the XML data in the BPEL payload is correctly formatted for display in the table.

Below is an XML sample for display at any level within the XML payload. The *First column XPATH* retrieves the parent and its peer elements (*collectionItem*). Each of the configured XPATHs point to an *itemValue* element used to retrieve the cell values for the table from each collectionItem.

Note: All columns must display within the same direct parent element.

```
<rootElement>
  <collectionContainerElement>
    <collectionItem>                                    <--First row for table
      <itemValue1>value1</itemValue1>                   <--First column XPATH
      <itemValue2>value2</itemValue2>
      <itemValue3>value3</itemValue3>
    </collectionItem>
    <collectionItem>                                    <--Second row
      <itemValue1>value1</itemValue1>
      <itemValue2>value2</itemValue2>
      <itemValue3>value3</itemValue3>
    </collectionItem>
    <collectionItem>
      <itemValue1>value1</itemValue1>
      <itemValue2>value2</itemValue2>
      <itemValue3>value3</itemValue3>
    </collectionItem>
  </collectionContainerElement>
</rootElement>
```

The first column XPATH for the above XML should be similar to the following:

```
/task:payload/task:rootElement/collectionContainerElement/collectionItem/itemValue1
```

6.4.10.2 Hiding the Bottom Panel in the Task Viewer

If your imaging solution does not utilize the bottom panel on the Task Viewer, you can disable it in the AXF solution. For example, for an invoice processing solution, you may want to hide the bottom panel displaying General Ledger lines. Hiding the panel increases performance speed and page space.

Follow these steps to hide the bottom panel:

1. Using a tool such as Oracle SQL Developer, open a connection to the AXF database.
2. Run the following query, and note the BLOCK_IDs in use for the bottom panel:

   ```
   SELECT BLOCK_ID FROM AXF_METADATA_BLOCKS WHERE BLOCK_LOCATION =
   'BOTTOM_PANEL'
   ```

3. Run the following command to delete lines from the AXF_METADATA_ ATTRIBUTES table, replacing *BLOCK_ID* with the ID you identified in step 2:

   ```
   DELETE FROM AXF_METADATA_ATTRIBUTES where BLOCK_ID = BLOCK_ID;insert
   BLOCK_ID from previous step;
   ```

4. Run the following command to delete lines from the AXF_METADATA_BLOCKS table.

   ```
   DELETE FROM AXF_METADATA_BLOCKS WHERE BLOCK_LOCATION = 'BOTTOM_PANEL'
   ```

5. Click the **Clear Configuration Cache** button on the driver page.

6.4.11 Reenabling Paperclip Attachments (Oracle E-Business Suite and Imaging Only)

Installing and configuring the Managed Attachments solution automatically disables the Oracle E-Business Suite attachments paperclip icon and functionality. To reenable

the paperclip functionality for an Imaging Solution only configuration, follow these steps to disable the Managed Attachments solution:

> **Note:** This section applies to Oracle E-Business Suite use only.

1. Open the AXF_CONFIGS table (Oracle E-Business Suite) table.
2. In the FORMFUNCTION field, rename the `AXF_MANAGED_ATTACHMENTS` entry. For details, see Section B.4.2.2.

   For example, rename the entry as follows:

   `AXF_MANAGED_ATTACHMENTS-DISABLED`

   > **Note:** To reenable the Managed Attachments solution, change the FORMFUNCTION field back to the following entry:
   >
   > `AXF_MANAGED_ATTACHMENTS`

3. Verify that the AXF_PAPERCLIP property in the AXF_PROPERTIES table is set to TRUE. For more information, see Section B.4.5.

Table 6–8 AXF_PROPERTIES Values For PaperClip Use

PROPNAME	PROPVALUE
AXF_PAPERCLIP	Set to `TRUE` to enable the paperclip option, or `FALSE` to disable it. (Value must be in uppercase characters.)

6.4.12 Resizing Oracle PeopleSoft Popup Solution Windows

You can change the default system-wide size for all Oracle PeopleSoft popup windows that display as part of solutions. Note the following:

- The default popup window size is 600 pixels high by 800 wide, set in the PS_AXF_PROPERTIES Table. When a size is set in this table, users cannot set and save popup window size themselves.
- To change window size, enter new values in the table, as shown in Section B.5.5.
- To remove the popup window size (and allow individual users to change and save popup window size using standard browser controls), remove the WINDOW_HEIGHT and WINDOW_WIDTH rows from the PS_AXF_PROPERTIES Table.

7

Configuring the Managed Attachments Solution

This chapter describes how to configure the Managed Attachments solution for Oracle E-Business Suite, Oracle PeopleSoft, or another business application. It describes how to run configuration scripts, configure Content Server components, customize the solution, integrate Oracle AutoVue and Oracle Distributed Document Capture, and configure authentication and security.

> **Note:** For background information on the Managed Attachments solution, see Section 1.4.

This chapter covers the following topics:

- Section 7.1, "System Requirements for the Managed Attachments Solution"
- Section 7.2, "Running Configuration Scripts to Enable Managed Attachments"
- Section 7.3, "Configuring the Content Server Components"
- Section 7.4, "Configuring the Enterprise Application Adapter Framework"
- Section 7.5, "Configuring Managed Attachments Settings"
- Section 7.6, "Customizing the Managed Attachments Solution"
- Section 7.7, "Configuring Authentication and Security"
- Section 7.8, "Configuring Oracle AutoVue Document Viewing"
- Section 7.9, "Configuring Oracle Distributed Document Capture"
- Section 7.10, "Testing the Managed Attachments Solution"
- Section 7.11, "Configuring Content Server Logging"
- Section 7.12, "Uninstalling Adapter Components on Content Server"

7.1 System Requirements for the Managed Attachments Solution

In addition to all other requirements listed in Section 1.7, the Managed Attachments solution includes additional requirements. General Managed Attachments requirements are listed in this section, and supported configuration and certification information, such as browsers, is available at:

`http://www.oracle.com/technetwork/middleware/webcenter/content/documentation/documentation-155348.html`

Oracle WebCenter Content: Imaging 11g Release 1 (11.1.1 or higher)

Installation of Imaging 11g is required, even if only configuring the Managed Attachments solution. This is because AXF is part of the Imaging deployment.

> **Note:** The Managed Attachments solution is supported on WebSphere Application Server when Content Server is running on WebSphere Application Server and Application Extension Framework (Imaging Server) is running on Oracle WebLogic Server.

Oracle WebCenter Content Server 11g Release 1 (11.1.1 or higher)

The Managed Attachments solution supports the following Content Server search engines.

- Metadata Only Search
- Database – Full Text Search
- Oracle Text – Full Text Search

> **Note:** The Folders_g component by default sets a variable called **AuthorDelete=true**, which allows users without delete privileges on a security group to delete documents if they are the author. If using the Folders_g Content Server component with the Managed Attachments solution, be aware that this variable's setting may override Managed Attachments dPrivilege or AppAdapterGrantPrivilege deletion settings.

Oracle E-Business Suite

- For Managed Attachments with Oracle E-Business Suite **Forms-based entities**, Managed Attachments requires that the EBS FND attachments framework be configured for the specific entity. Although Managed Attachments is an alternative attachments framework to FND attachments, Managed Attachments leverages the underlying FND attachments infrastructure.
- For Managed Attachments with Oracle E-Business **OAF-based entities**, Managed Attachments requires that the OAF page associated with the entity have a *Personalize Page* option with the *About this page* link. The OAF page should support site level personalization so that the Managed Attachment button can be added manually. The *About this page* link obtains the OAF entity details, such as the primary key. OAF-based pages without this option are not supported for Managed Attachments.

7.2 Running Configuration Scripts to Enable Managed Attachments

This section describes running the configuration scripts listed in Table 7–1, following the steps described in the sections below. The first script configures AXF elements, and is required for all Managed Attachments configurations, including the Enterprise Application Adapter Framework. The remaining scripts apply to specific business applications as noted in Table 7–1.

- Section 7.2.1, "Configuring AXF Elements for Managed Attachments"
- Section 7.2.2, "Enabling Managed Attachments for Oracle E-Business Suite Forms Applications"

- Section 7.2.3, "Enabling Managed Attachments for Oracle E-Business Suite OAF Applications"
- Section 7.2.4, "Enabling Managed Attachments for Oracle PeopleSoft"

Table 7–1 SQL Scripts for Configuring Managed Attachments

SQL Script	Applies To	Description
AXF_ATTACHMENTS_CONFIG.sql	▪ Oracle E-Business Suite Forms ▪ Oracle E-Business Suite OAF ▪ Oracle PeopleSoft ▪ Enterprise Application Adapter Framework (see Section 7.4)	AXF script that configures AXF elements such as solutions and commands. See Section 7.2.1. Run this script on the Oracle WebCenter Content middle tier as the user who owns the Imaging schema.
AXF_EBS_ATTACHMENTS_CONFIG.sql	Oracle E-Business Suite Forms only	Populates the Oracle E-Business Suite Forms tables used by AXF for Managed Attachment access via the Zoom menu within the Oracle E-Business Suite Forms application. See Section 7.2.2. Run this script as the Oracle E-Business Suite plug-in schema user (AXF).
OAF_AXF_EBS_ATTACHMENTS_CONFIG.sql	Oracle E-Business Suite OAF only	Provides a sample script that populates the Oracle E-Business Suite OAF tables used by AXF for Managed Attachments access. See Section 7.2.3. Run this script as the Oracle E-Business Suite plug-in schema user (AXF).
AXF_PSFT_ATTACHMENTS_CONFIG.sql	Oracle PeopleSoft only	Provides a sample script that populates the PeopleSoft tables used by AXF for Managed Attachments access. See Section 7.2.4. Run this script as an Oracle PeopleSoft user with administrative privileges.

7.2.1 Configuring AXF Elements for Managed Attachments

Follow these steps to run AXF_ATTACHMENTS_CONFIG.sql, an AXF script that configures AXF elements such as solutions and commands. This script is required for Managed Attachments for all business applications.

1. On the Fusion Middleware Home, locate the script in the folders specified below for your business application.
   - For Oracle E-Business Suite Forms:

     MW_HOME/*ECM_HOME*/axf/adapters/ebs/R11/

     MW_HOME/*ECM_HOME*/axf/adapters/ebs/R12/
   - For Oracle E-Business Suite OAF:

     MW_HOME/*ECM_HOME*/axf/adapters/ebs/oaf/
   - For Oracle PeopleSoft:

     MW_HOME/*ECM_HOME*/axf/adapters/psft/
   - For Enterprise Application Adapter Framework:

     MW_HOME/*ECM_HOME*/axf/adapters/framework/
2. Modify parameters in the **AXF_ATTACHMENTS_CONFIG.sql** script to match your environment.
   a. Change the system name and port for the RIDC_CONNECTION_STR parameter to match your Content Server installation. The port number should correspond to that of the Content Server's Intradoc server port that receives

RIDC calls. (To find the value for the Content Server port, locate the IntradocServerPort config value in config.cfg.)

b. Change the system name and port for the UCM_CONNECTION_STR parameter to match your Content Server installation. The port number should correspond to that of the Content Server's web server port.

c. Change the [Content ServerProfile] and [ResultTemplate] values in the UCM_CONNECTION_STR parameter for your configuration. These values are used in the Managed Attachments URL that invokes the Content Server attachments framework search, as described in Section C.1.4.

- Profile values include: *EBSProfile* (default), *PSFTProfile*, and *UniversalProfile* (for the Enterprise Application Adapter Framework). For details about customizing profiles, see Section 7.6.5.
- Result template values include: *EBS_LIST* (adapter_list_results_template.htm), *PSFT_LIST* (ps_adapter_list_results_template.htm), and *Universal_LIST* (universal_adapter_list_results_template.htm, for the Enterprise Application Adapter Framework). For details about customizing results templates, see Section 7.6.13.

Note: If you are using the AXF for BPM infrastructure, open the AXF_ATTACHMENTS_CONFIG.sql file in any editor. Replace `.imaging.` with `.ecm.` and save the file.

---

3. On the Oracle WebCenter Content middle tier, log in to the AXF Configuration Database *as the user who owns the Imaging schema* and run the AXF_ATTACHMENTS_CONFIG.sql script.

To execute the script, enter:

```
@AXF_ATTACHMENTS_CONFIG.sql
```

Note: For information about creating schemas before or after Oracle WebCenter Content installation, see "Creating Oracle WebCenter Content Schemas with the Repository Creation Utility" in *Oracle WebCenter Content Installation Guide*.

7.2.2 Enabling Managed Attachments for Oracle E-Business Suite Forms Applications

If configuring the adapter for Oracle E-Business Suite Forms, follow these steps to run AXF_EBS_ATTACHMENTS_CONFIG.sql, an Oracle E-Business Suite plug-in script that configures the Zoom menu to enable the Managed Attachments solution within the Oracle E-Business Suite system.

1. Locate the script corresponding to your release in the folder specified below.

MW_HOME/ECM_HOME/axf/adapters/ebs/R11/

MW_HOME/ECM_HOME/axf/adapters/ebs/R12/

2. In the **AXF_EBS_ATTACHMENTS_CONFIG.sql** script, change the **host name** and **port** to match your Imaging server.

3. Log in to the Oracle E-Business Suite database *as the* Oracle E-Business Suite *plug-in schema user (AXF)* and run the appropriate script.

To execute the script, enter:

```
@AXF_EBS_ATTACHMENTS_CONFIG.sql
```

4. Open the AXF_FND_MAP table and delete unneeded rows.

   See Section C.2.3. The script inserts multiple rows, allowing you to keep the rows that apply to your particular configuration.

7.2.3 Enabling Managed Attachments for Oracle E-Business Suite OAF Applications

In order to link OAF page details with documents checked into Content Server, the OAF page's business entity and primary key values must be stored in the Oracle E-Business Suite OAF AXF database tables described in Section C.3. This configuration involves these steps:

- Populating the database via the OAF_AXF_EBS_ATTACHMENTS_CONFIG.sql script, which populates the tables with values for certain sample OAF pages, as described in Section 7.2.3.1.
- Modify the sample key values for your OAF pages, using values you identify in Section 7.2.3.2.

7.2.3.1 Populating the OAF AXF Database Tables With Sample Key Values

Table 7–2 lists the sample key values populated by the OAF_AXF_EBS_ATTACHMENTS_CONFIG.sql script for the specified OAF pages.

- The OAF page location is used as the AXF_OAF_PAGE value in the OAF_AXF_CONFIGS Table (Oracle E-Business Suite OAF). See the example table in Section C.3.1.2.
- The following key values are used in the OAF_AXF_CMD_PARAMS Table (Oracle E-Business Suite OAF). See the example table in Section C.3.4.2.
  - Entity (businessObjectType); for example, REQ_HEADERS
  - Primary Key (businessObjectKey1); for example, REQUISITION_HEADER_ID
  - View Object (businessObjectValue1); for example, ReqHeadersVO
  - View Attribute (businessObjectValue1 or labelValue1); for example, RequisitionHeaderId

Table 7–2 OAF Pages Configured By Sample OAF_AXF_EBS_ATTACHMENTS_CONFIG.SQL Script

OAF Page	Navigation	Sample Key Values
iProcurement Requisitions	Choose iProcurement, then the Requisitions tab, then click **Requisition Description**	OAF Page = /oracle/apps/icx/por/reqmgmt/webui/ReqDetailsPG ▪ Entity = REQ_HEADERS ▪ Primary Key = REQUISITION_HEADER_ID ▪ View Object = ReqHeadersVO ▪ View Attribute = RequisitionHeaderId
iProcurement Purchase Orders	Choose iProcurement, then the Requisitions tab, then click **Order Number**	OAF Page = /oracle/apps/pos/changeorder/webui/PosViewPOPG ▪ Entity = PO_HEADERS ▪ Primary Key = PO_HEADER_ID ▪ View Object = PosViewHeadersVO ▪ View Attribute = PO_HEADER_ID
Purchase Orders	Choose Purchasing, then Buyer Work Center, then Orders, then click the **Order** number	OAF Page = /oracle/apps/po/document/order/webui/OrderPG ▪ Entity = PO_HEADERS_MERGE_V ▪ Primary Key = PO_HEADER_ID ▪ View Object = PoHeaderMergeVO ▪ View Attribute = PoHeaderId

Table 7–2 (Cont.) OAF Pages Configured By Sample OAF_AXF_EBS_ATTACHMENTS_CONFIG.SQL Script

OAF Page	Navigation	Sample Key Values
Suppliers	Choose Purchasing, then Supply Base, then Suppliers. Search by clicking **Go**, then click the **Update** icon	OAF Page = /oracle/apps/pos/supplier/webui/QuickUpdatePG ■ Entity = PO_VENDORS ■ Primary Key = VENDOR_ID ■ View Object = SupplierVO ■ View Attribute = VendorId
Request for Quote (RFQ) Header	Choose Purchasing, then Buyer Work Center, then Negotiations, then RFQ, then **Header**	OAF Page = /oracle/apps/pon/negotiation/creation/webui/ponDefineHdrPG ■ Entity = PON_AUCTION_HEADERS_ALL ■ Primary Key = AUCTION_HEADER_ID ■ View Object = AuctionHeadersAllVO ■ View Attribute = AuctionHeaderId
Request for Quote (RFQ) Line	Choose Purchasing, then Buyer Work Center, then Negotiations, then RFQ, then **Lines**. Click the **GO** button.	OAF Page = /oracle/apps/pon/negotiation/creation/webui/ponItemHeaderPG ■ Entity = PON_AUCTION_ITEM_PRICES_ALL ■ Primary Key = AUCTION_HEADER_ID & LINE_NUMBER ■ View Object = AuctionItemPricesAllVO ■ View Attributes = AuctionHeaderId & LineNumber
Projects	Choose Project Manager, then Projects: Delivery, then Project List. Click Project name, then the **Attachments** tab	OAF Page = /oracle/apps/pa/project/webui/ProjAttachmentPG ■ Entity = PA_PROJECTS ■ Primary Key = PROJECT_ID ■ View Object = ProjAttachmentsVO ■ View Attribute = PROJECT_ID

Follow these steps to run the script.

1. Locate the OAF_AXF_EBS_ATTACHMENTS_CONFIG.sql script in this folder:

   MW_HOME`/`*ECM_HOME*`/axf/adapters/ebs/oaf/`

2. Log in to the Oracle E-Business Suite database *as the* Oracle E-Business Suite *plug-in schema user (AXF)*.

3. In the script, replace the host name and port to match your Imaging server host name and port.

4. Run the OAF_AXF_EBS_ATTACHMENTS_CONFIG.sql script.

   To execute the script, enter:

   ```
   @OAF_AXF_EBS_ATTACHMENTS_CONFIG.sql
   ```

7.2.3.2 Identifying Key Values for Your OAF Pages

This section provides steps for identifying the primary field(s) to link the OAF page with Managed Attachments functionality, and identifying the page name and fields to use as labels.

How you identify the needed information depends on the OAF page's contents, as described in the following scenarios:

- Section 7.2.3.2.1, "With Both Attachment Information and Primary Keys Present"
- Section 7.2.3.2.2, "With Attachment Information Present, But Not Primary Keys"
- Section 7.2.3.2.3, "With Neither Attachment Information Nor Primary Keys Present"

7.2.3.2.1 With Both Attachment Information and Primary Keys Present Ideally, the needed Oracle E-Business Suite attachment and primary key details are present in the OAF page's XML file. In this case, follow these steps to locate required mapping information.

1. Navigate to the OAF page and click the **About this page** link at the bottom.

   For example, navigate to Employee Self Service, then the Appraisals OAF page, and click the **About this page** link.

2. Note the full path of the page's XML file. For example:

   ```
   /oracle/apps/per/selfservice/appraisals/webui/EmpMyAppraisalsPG
   ```

3. Search the XML file for an **<oa:attachment>** XML tag that contains view object name and primary key details for the business entity associated with the view.

   For example, the following <oa:attachment> section provides these key values, highlighted in bold characters.

   ```
   <oa:attachmentImage id="EMPInprogressJournal" prompt="Journal"
   viewName="EmpMyAppraisalsInProgressVO" autoCommitEnabled="true"
   enableDocumentCatalog="false">
      <oa:entityMappings>
      <oa:entityMap id="APPRAISEEentityMap" entityId="APPRAISEE_JOURNAL"
   showAll="false">
      <oa:primaryKeys>
      <oa:primaryKey id="primaryKey1" viewAttr="AppraisalId"/>
      <oa:primaryKey id="primaryKey2" viewAttr="AppraiseePersonId"/>
      </oa:primaryKeys>
      <oa:categoryMappings>
      <oa:categoryMap id="categoryMap1" categoryName="MISC"/>
      </oa:categoryMappings>
      </oa:entityMap>
      </oa:entityMappings>
      </oa:attachmentImage>
   ```

4. Note the identifying values. You use them in Section 7.2.3 to map OAF page details with Managed Attachments.

   In the example, the following values were located:

   - Entity = **APPRAISEE_JOURNAL**
   - Primary Key = **APPRAISAL_ID** and **APPRAISEE_PERSON_ID**
   - View Object = **EmpMyAppraisalsInProgressVO**
   - View Attribute = **AppraisalId** and **AppraiseePersonId**

7.2.3.2.2 With Attachment Information Present, But Not Primary Keys In this case, the OAF page's XML file contains attachment object information (view object and entity), but not the corresponding primary key. Follow these steps to determine the view object and entity name from the XML file, then query the FND_DOCUMENT_ENTITIES table to determine the primary key.

1. Navigate to the OAF page and click the **About this page** link at the bottom.

   For example, navigate to iProcurement, then Requisitions, then the Requisition Details OAF page, and click the **About this page** link.

2. Note the full path of the page's XML file. For example:

   ```
   /oracle/apps/icx/por/reqmgmt/webui/ReqDetailsPG
   ```

3. From the APPS schema, run the following query against the FND_DOCUMENT_ENTITIES table, replacing *EntityName* with the entity identified in the XML file.

   ```
   SELECT * FROM FND_DOCUMENT_ENTITIES WHERE ENTITY_NAME=EntityName
   ```

   For example, the following xml file's <oa:attachment> section provides these key values, highlighted in bold characters:

   ```
   <oa:attachmentLink id="Attachment" viewName="ReqHeaderVO" prompt="Attachment">
      <oa:entityMappings>
      <oa:entityMap entityId="REQ_HEADERS" id="entityMap1" insertAllowed="false"
   updateAllowed="false" deleteAllowed="false">
      <oa:categoryMappings>
      <oa:categoryMap categoryName="REQ Internal" id="ReqInternal"/>
      <oa:categoryMap categoryName="Approver" id="ToApprover"/>
      <oa:categoryMap categoryName="Buyer" id="ToBuyer"/>
      <oa:categoryMap categoryName="Receiver" id="ToReceiver"/>
      <oa:categoryMap categoryName="Vendor" id="ToSupplier"/>
      </oa:categoryMappings>
      </oa:entityMap>
      </oa:entityMappings>
      </oa:attachmentLink>
   ```

   The following information is known:

   – Entity = **REQ_HEADERS**

   – View Object = **ReqHeaderVO**

4. Based on the primary key column, determine its corresponding view attribute.

   a. Run the FND_DOCUMENT_ENTITIES query for the REQ_HEADERS entity.

      ```
      SELECT * FROM FND_DOCUMENT_ENTITIES WHERE ENTITY_NAME='REQ_HEADERS'
      ```

      The query returns the primary key column name: REQ_HEADER_ID.

   b. Click the **About this page** link, select **View Object** and find the corresponding attribute name, which matches the column name: RequisitionHeaderId.

      – Primary Key = **REQUISITION_HEADER_ID**

      – View Attribute = **RequisitionHeaderId**

5. Note the identifying values. You use them in Section 7.2.3 to map OAF page details with Managed Attachments.

   In the example, the following values were located:

   - Entity = **REQ_HEADERS**
   - Primary Key = **REQUISITION_HEADER_ID**
   - View Object = **ReqHeadersVO**
   - View Attribute = **RequisitionHeaderId**

7.2.3.2.3 With Neither Attachment Information Nor Primary Keys Present In this case, the XML file associated with the page does not contain any attachment information. Follow these steps to determine the needed information.

1. Navigate to the OAF page and click the **About this page** link at the bottom.

   For example, navigate to Purchasing, then Buyer Work Center, then Orders, then click the Order Number. Click the **About this page** link.

Note the full path of the page's XML file. For example:

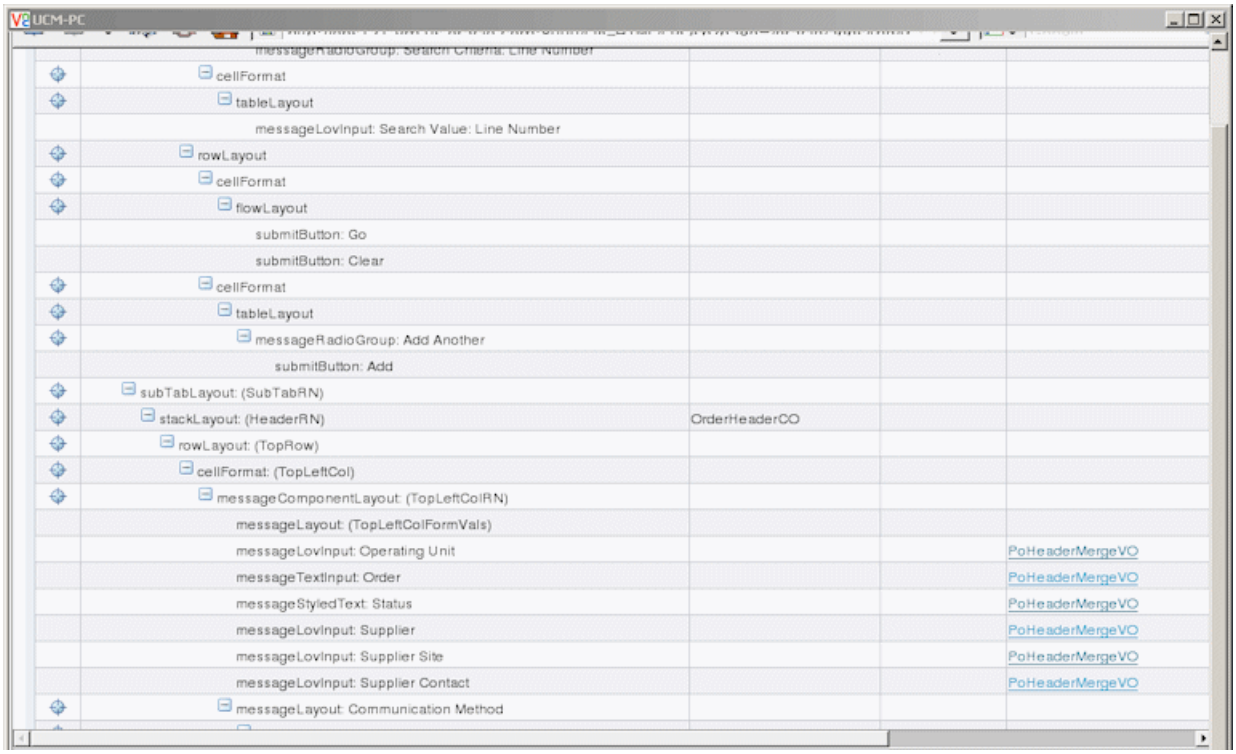
```
/oracle/apps/po/document/order/webui/OrderPG
```

The page's XML file does not contain any Oracle E-Business Suite attachment information.

2. Select **Page Definition**, then click the **Expand All** link.
3. Determine the view object.

   In the **View Object** column, locate the view object listed the maximum number of times. The view object used in the top-level region is the driving view object for the page. In the example shown in Figure 7–1, the view object is: **PoHeaderMergeVO**.

Figure 7–1 Determining the View Object



4. Determine the entity (driving table).

   Click the view object link to display details for the view object. For example, the entity (driving table) for the PoHeaderMergerVO view object is **PO_HEADERS_MERGE_V**.

5. Determine the entity's primary keys.

   a. See the *Oracle E-Business Suite Electronic Technical Reference Manual* at the following location:

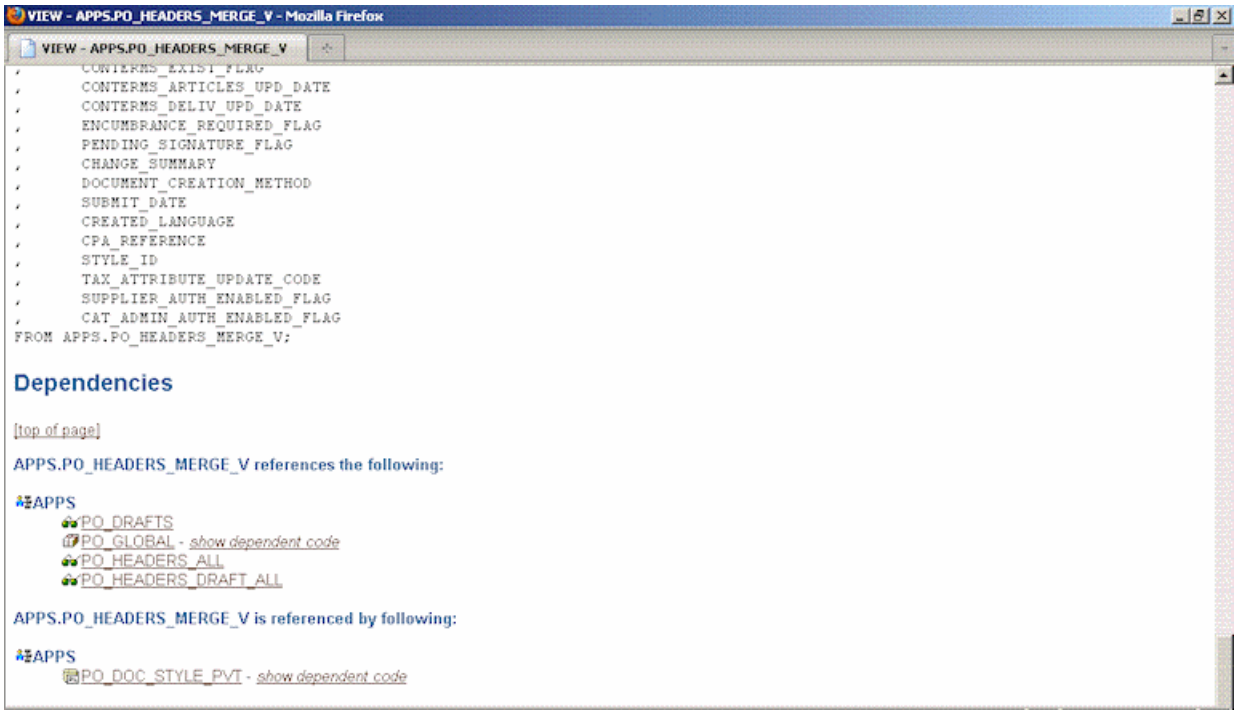
      http://etrm.oracle.com/pls/etrm/etrm_search.search

   b. Enter a search that includes the entity name to obtain the table details. For example, for the PO_HEADERS_MERGE_V entity, enter:

      http://etrm.oracle.com/pls/etrm/etrm_pnav.show_object?c_name=PO_HEADERS_MERGE_V&c_owner=APPS&c_type=VIEW

c. From the search results, display the view or table details. The example shown in Figure 7–2 provides view details.

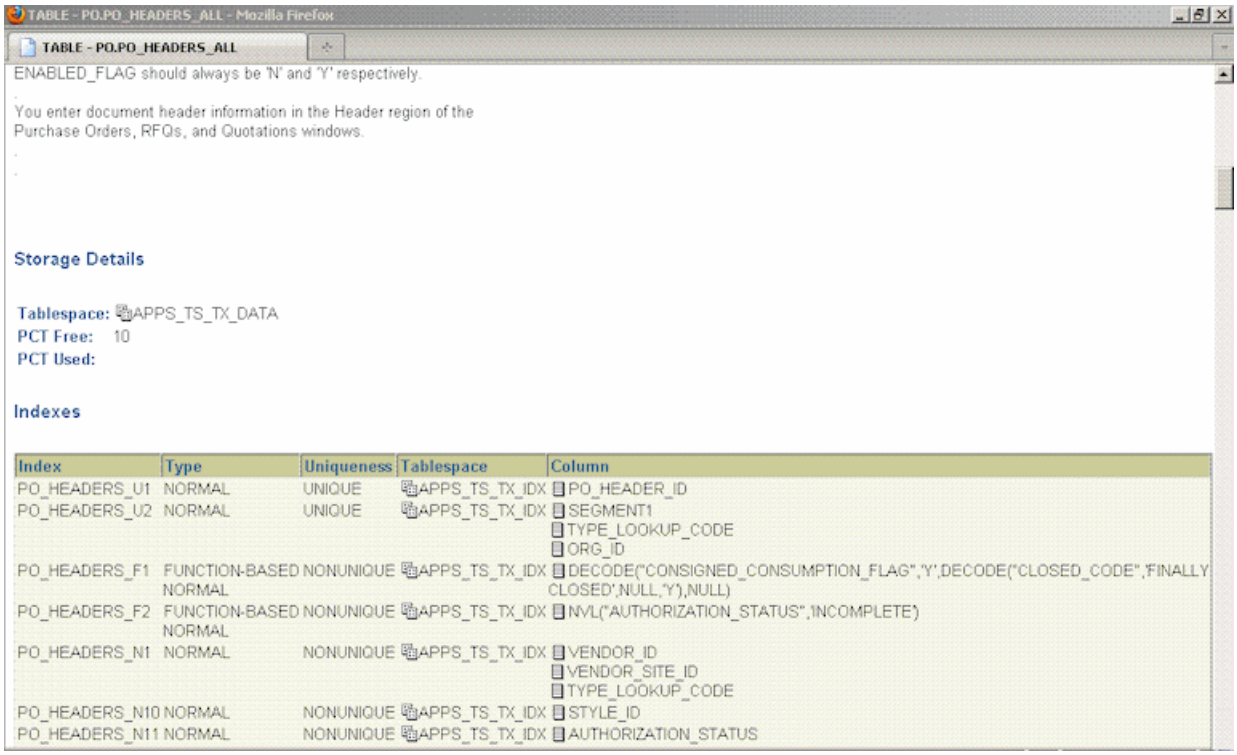
Figure 7–2 Displaying View Details



d. Select the table that best describes the selected view (in this example, **PO_HEADERS_ALL**).

e. Figure 7–3 shows the primary key details associated with the table. Determine its primary key (in this example, **PO_HEADER_ID**).

Figure 7–3 Displaying Table Details



6. Based on the primary key, obtain its corresponding view attribute.

   Return to the OAF page, clicking the **About this page** link and clicking the **Expand All** link. Select the view object that displays most often (PoHeaderMergeVO, for example). From the **Attributes** list that displays, determine the attribute name, which typically varies only slightly from the primary key name (for example, PO_HEADER_ID versus PoHeaderId).

7. Note the identifying values. You use them in Section 7.2.3 to map OAF page details with Managed Attachments.

   In the example, the following values were located:

   – Entity = **PO_HEADERS_MERGE_V**

   – Primary Key = **PO_HEADER_ID**

   – View Object = **PoHeaderMergeVO**

   – View Attribute = **PoHeaderId**

7.2.4 Enabling Managed Attachments for Oracle PeopleSoft

The main configuration steps include:

- Configuring each Oracle PeopleSoft entity's primary key values in the Oracle PeopleSoft AXF database tables, described in Section C.4. The AXF_PSFT_ATTACHMENTS_CONFIG.sql script is provided as a starting point. It contains sample values for PeopleSoft entities such as work order, sales contract, and purchase requisition. You can modify the sample key values to reflect your Oracle PeopleSoft pages, using the primary keys you noted in Section 3.3.
- Specifying an Oracle PeopleSoft role to grant users permission.

Follow the steps in this section to enable Managed Attachments for Oracle PeopleSoft.

1. Locate the AXF_PSFT_ATTACHMENTS_CONFIG.sql sample script in this folder:

   MW_HOME/ECM_HOME/axf/adapters/psft/

2. Log in to the Oracle PeopleSoft database as a user with administrative privileges.

3. Modify the sample script to include the values you identified for your Oracle PeopleSoft entities in Section 3.3, then run the script.

   To execute the script, enter:

   ```
   @AXF_PSFT_ATTACHMENTS_CONFIG.sql
   ```

4. Specify an Oracle PeopleSoft role in the PS_AXF_COMMANDS Table to grant users permission to use a specified user interface item. For example, you might change the `Employee` value to **`PeopleSoft User`** or an appropriate Oracle PeopleSoft user role. See Section C.4.2.2.

7.3 Configuring the Content Server Components

Follow the sections listed below to configure Content Server components for the Managed Attachments solution for Oracle E-Business Suite or Oracle PeopleSoft, and test the completed solution.

> **Note:** To configure the Managed Attachments solution for another business application, see the instructions provided in Section 7.4.

- Section 7.3.1, "Enabling Content Server Components"
- Section 7.3.2, "Verifying Required Content Server Components and Configuration"
- Section 7.3.3, "Testing the Managed Attachments Solution"

7.3.1 Enabling Content Server Components

Table 7–3 lists Content Server components for the Managed Attachments solution.

Table 7–3 Content Server Components for Managed Attachments

Content Server Component	Description
AppAdapterCore.zip	Required for all Managed Attachments configurations
AppAdapterEBS.zip	Required for Oracle E-Business Suite Forms and OAF integrations
AppAdapterPSFT.zip	Required for Oracle PeopleSoft integrations
AppAdapterUniversal.zip	Required for other business application integrations, as described in Section 7.4.

Follow these steps to enable the Content Server components using Component Manager.

1. If upgrading from an earlier installation of the adapter, back up previous customizations, as described in Section 7.3.1.1.

2. Log in to Content Server as an administrator.

3. Choose **Admin Server** from the Administration menu.

The Component Manager page displays.

4. Choose **Integration**.

   The Content Server components described in Table 7–3 are listed as Integration options.

5. Select the **AppAdapterCore** component to enable it.

6. Select the appropriate business application component:

   - **AppAdapterEBS.zip** for Oracle E-Business Suite
   - **AppAdapterPSFT.zip** for Oracle PeopleSoft
   - **AppAdapterUniversal.zip** for other business applications, as described in Section 7.4.

7. Click **Update**.

8. Restart Content Server.

9. Check the Content Server and database logs to ensure that no application adapter install errors occurred.

7.3.1.1 Backing Up Customizations Before Upgrading

If upgrading from a previous version of the adapter (Managed Attachments solution), perform these additional upgrade steps if needed before installing Content Server components, as indicated in step 1 in Section 7.3.1.

1. Back up Content Server rule and profile customizations.

   Installing the new adapter overwrites existing rules and profiles. If your previous installation includes customizations to Content Server rules and profiles, back up these rules and profiles using a CMU bundle before installing Content Server components. After installation, create new rules and profiles, and manually apply any customizations.

2. Back up the config.cfg file.

   This adapter release stores environment variables using a preferences.hda file instead of config.cfg. After installation, manually copy configuration preferences from config.cfg to preferences.hda.

3. Back up any other template or resource include customizations and modifications.

   Some resource includes and templates have changed in this release. Manually set these customizations after installation.

7.3.2 Verifying Required Content Server Components and Configuration

Follow these steps to verify that required Content Server components are enabled.

1. Log in to Content Server as an administrator.

2. Click the **Configuration for [Instance]** link in the content server Administration tray.

3. In the Features And Components section, click **Enabled Component Details**.

4. From the details shown, verify that the following components are enabled. If a component is not listed, install and enable it according to its installation documentation.

The following components must be at or above the version level included with the installed Content Server 11*g* release:

- YahooUserInterfaceLibrary
- CheckoutAndOpenInNative
- CoreWebdav
- NativeOsUtils
- ContentFolios

5. In the User Admin applet, ensure that the roles below have the correct access level to the corresponding security groups.

   In Content Server, choose the Administration option from the main menu, choose **Admin Applets**, then **User Admin**. Select **Security**, then **Permissions by Role**.

   - AFAdmin: AFDocuments (RWDA) and no permissions on any other groups
   - AFDelete: AFDocuments (RWD) and no permissions on any other groups
   - AFWrite: AFDocuments (RW) and no permissions on any other groups
   - AFRead: AFDocuments (R) and no permissions on any other groups

7.3.3 Testing the Managed Attachments Solution

Test the configuration for your selected business application by following the steps listed in the *Oracle WebCenter User's Guide for Application Adapters*.

7.4 Configuring the Enterprise Application Adapter Framework

In addition to Oracle E-Business Suite and Oracle PeopleSoft components, the adapter provides a generic component, called **Enterprise Application Adapter Framework**, for customizing the Managed Attachments solution for use with another business application. This component contains similar settings to the Oracle E-Business Suite and Oracle PeopleSoft component zip files.

In Section 7.2.1, you ran the AXF_ATTACHMENTS_CONFIG.sql script, which configured AXF tables for the UCM_Managed_Attachments solution. Follow the steps listed in these sections to configure the business application/Managed Attachments integration.

- Section 7.4.1, "Deploying the AppAdapterUniversal Component on Content Server"
- Section 7.4.2, "Configuring the SOAP Service Call and Managed Attachments URL Display"

7.4.1 Deploying the AppAdapterUniversal Component on Content Server

Before deploying the AppAdapterUniversal.zip component, you can optionally rename the component files and customize the framework layout, images, and profile, as described later in this section.

From Component Manager, install the component, enable it, and restart the Content Server. For more information, see Section 7.3.1 or "Creating Components Using the Component Wizard" in *Oracle Fusion Middleware Administering Oracle WebCenter Content*.

7.4.1.1 Optionally Renaming Component Files

If you opt to rename files, follow these steps to unzip the component, rename certain files for your business application, and zip the files into a newly named file. You will need an uppercase business application name (for example, BIZAPP), a lowercase name (for example, `bizapp`), and a long name (for example, `ExampleBusinessApp`).

1. Back up the AppAdapterUniversal.zip file.

   The zip file is located at:

   ECM_HOME`/ucm/Distribution/AppAdapter`

2. Unzip the AppAdapterUniversal.zip file.
3. Replace the word `Universal` in the file name with the business application name, matching case (for example, replace `Universal` with `BIZAPP`).

   Note that this string is used in the ResultTemplate name used in the Managed Attachments URL (for example, `ResultTemplate=BIZAPP_LIST`). For details about the result template, see Section 7.6.13.
4. Locate the universal_AppAdapter_strings.htm file and replace all occurrences of the string `Application` with your business application long name (for example, replace `Application` with `ExampleBusinessApp`).
5. Search each file for instances of the string `Universal` and replace them with the name you specified in step 3, matching case (for example, `BIZAPP`).
6. Search each file for instances of the string `universal` and replace them with your business application's lowercase name (for example, `bizapp`).
7. Zip the files into the newly named file (for example, `AppAdapterBuzzApp.zip`).

7.4.1.2 Optionally Customizing the Framework Layout, Images, and Profile

- To change the look-and-feel of the result template file, customize the result template file (originally named Universal_LIST, universal_adapter_list_results_template.htm). See Section 7.6.13.
- To customize images, replace the images in the resources\publish\images folder.
- To customize the Content Server profile used, change the UniversalProfile file. See Section 7.6.5.

7.4.2 Configuring the SOAP Service Call and Managed Attachments URL Display

Follow the steps in this section to configure the business application to call AXF and receive the Managed Attachments URL.

1. Configure the business application to provide an application entity context link, button, or menu item to call AXF via the SOAP service and to pass the Managed Attachments attributes listed in Table 7–4.

   - Example 7–1 provides a sample SOAP XML file for calling AXF and Table 7–5 describes fields in the AXFRequest.
   - Example 7–2 provides a sample AXFResponse and Table 7–6 describes fields in the AXFResponse.

     The SOAP response contains the Managed Attachments page's URL. If an error occurs during the grant call, an AXF Error Page URL is returned. The WSDL URL is:

     `http://`*`Imaging Hostname:Imaging Port`*`/axf-ws/AxfSolutionMediatorService?wsdl`

Table 7–4 Attributes to Pass Via SOAP Call to AXF

Managed Attachments Attribute Key	Description	Example Value
application (required)	Specifies the business application instance name, providing a unique identifier to differentiate between instances.	BIZAPP_instanceA
businessObjectType (required)	Specifies the business application's object type, typically the name of the record where the business object data is stored.	REQ_HEADERS
businessObjectKey1...businessObjectKey5 (one pair required)	Specifies the ObjectKey portion of key-value pairs that uniquely map business application record fields to Content Server. One key-value pair is required, and up to 5 pairs are supported.	REQUISITION_HEADER_ID
businessObjectValue1...businessObjectValue5 (one pair required)	Specifies the ObjectValue portion of key-value pairs that uniquely map business application record fields to Content Server. One key-value pair is required, and up to 5 pairs are supported.	1234
labelValue1...labelValue5 (optional)	Specifies labels to be displayed in the Managed Attachments page's title. Use these labelValue fields to pass text constants and record data that will allow the end user to see, via the user interface, the specific application entity record to which the Managed Attachments apply. Labels are optional, and up to 5 label values are supported.	Business Unit US001 Requisition ID 1234
extra*ParamName* (optional; for example, extraxReqNum)	Extra parameter to be passed as metadata from the business application to Content Server in check-in or scan forms. (For details, see Section 7.6.16.1.)	1234

Example 7–1 Sample SOAP XML File for Calling AXF

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ser="http://service.axf.imaging.oracle/">
      <soapenv:Header>
            <Security
xmlns="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
            <UsernameToken>
                  <Username>weblogic</Username>
                  <Password>welcome1</Password>
            </UsernameToken>
            </Security>
      </soapenv:Header>
      <soapenv:Body>
            <ser:execute>
                  <request>
                        <solutionNamespace>UCM_Managed_Attachments</solutionNamespace>
                        <commandNamespace>UCM_Managed_Attachments</commandNamespace>
                        <conversationId></conversationId>
                        <systemName></systemName> <!-- future use -->
                        <role></role> <!-- future use -->
                        <userContext></userContext> <!-- future user -->
                        <username>jcooper</username>
                        <requestParameters>
                              <entry>
                                    <key>application</key>
                                    <value>BIZAPP_instanceA</value>
                              </entry>
                              <entry>
```

```
                        <key>businessObjectType</key>
                        <value>REQ_HEADERS</value>
                    </entry>
                    <entry>
                        <key>businessObjectKey1</key>
                        <value>BUSINESS_UNIT</value>
                    </entry>
                    <entry>
                        <key>businessObjectValue1</key>
                        <value>US001</value>
                    </entry>
                    <entry>
                        <key>businessObjectKey2</key>
                        <value>REQUISITION_HEADER_ID</value>
                    </entry>
                    <entry>
                        <key>businessObjectValue2</key>
                        <value>1234</value>
                    </entry>
                    <entry>
                        <key>extraxReqNum</key>
                        <value>1234</value>
                    </entry>
                </requestParameters>
            </request>
        </ser:execute>
    </soapenv:Body>
</soapenv:Envelope>
```

Table 7–5 AXFRequest Attributes

AXFRequest Attributes	Description	Example
solutionNamespace	SolutionNamespace in which to execute the command.	UCM_Managed_Attachments
commandNamespace	Command to execute within the specified SolutionNamespace.	UCM_Managed_Attachments
conversationID	Currently opened conversation created by AXF. This attribute is blank if it is the first call and is not used by the Managed Attachments solution.	
systemName	Reserved for future use.	
role	Reserved for future use.	
userContext	Reserved for future use.	
userName	Business application user name. (Note that for the Managed Attachments solution, the USERNAME_PASS_THROUGH parameter must be set to TRUE if used with AXF for BPEL. For more information about USERNAME_PASS_THROUGH, see Section 7.6.19.)	jcooper

requestParameters	A collection of key/value pairs providing the command parameters.	`<entry>`
		`<key>application</key>`
		`<value>BIZAPP_instanceA</value>`
		`</entry>`

Example 7–2 AXFResponse

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
      <S:Body>
            <ns2:executeResponse xmlns:ns2="http://service.axf.imaging.oracle/">
                  <response>
                        <conversationId>b80ce79b-6b46-4302-ab59-ddb91e30692b</conversationId>
                        <responseCommands>
                              <command>OPEN_BROWSER</command>

                              <value>http://imaging.example.com:16200/cs/idcplg/-UniversalProfile?Id
cService=GET_SEARCH_RESULTS_FORCELOGIN&ResultTemplate=Universal_
LIST&afGuid=00000132-b540-810e-63f4-00e94577f990</value>
                        </responseCommands>
                  </response>
            </ns2:executeResponse>
      </S:Body>
</S:Envelope>
```

Table 7–6 AXFResponse Attributes

AXFResponse Attributes	Description
conversationId	Current conversation created by AXF to help maintain state/context for the transaction.
responseCommands	A collection of key/value pairs providing a command for the business application to execute with a single parameter. In most cases, it is the OPEN_BROWSER command, which provides a URL that the business application must open.

2. Display the URL returned by the AXF SOAP call in a new browser window or iFrame.
3. Optionally, configure SOAP over SSL security for the call, as described in Section 7.4.2.1.

7.4.2.1 Securing Communications Through SSL (Other Business Applications)

Follow these main steps to implement web server SSL encryption for the Enterprise Application Adapter Framework:

1. Install web server-based digital certificates.

   In your business application documentation, refer to information about configuring certificates on the business application's web server environment.
2. Enable SSL on Oracle WebLogic Server on the Oracle WebCenter Content side.

   For information, see "Configuring SSL for Oracle WebCenter Content Applications" in *Oracle WebCenter Content Installation Guide*.
3. Exchange server certificates between the web servers.

   For information, see "exportKeyStoreObject" in *Oracle Fusion Middleware Administrator's Guide*.

7.5 Configuring Managed Attachments Settings

After configuring the Content Server components as described in Section 7.3, use the Managed Attachments administration interface to configure its settings. This section provides general configuration information, and Section 7.6 provides detailed customization instruction and examples.

Settings Order of Precedence

The order of precedence of the configuration settings is as follows.

- role specific (highest)
- business context specific
- application entity specific
- application instance specific
- global (lowest)

Keep this order of precedence in mind when configuring attachment settings. For example, configuring a role that enables scanning results in scanning disabled for all other roles.

> **Note:** This section describes how to edit Managed Attachments settings in the administration interface. As an alternative, you can customize settings by directly editing the **preferences.hda** file configuration file where the settings are stored, as described in Section 7.5.2.

1. On Content Server, choose **Managed Attachments** from the **Administration** menu.

   The Configure Managed Attachments page displays, as shown in Figure 7–4.

Figure 7–4 Configure Managed Attachments Page

2. Edit **Global Configuration** variables if needed, and click **Update**.

Use these settings to configure Content Server environment variables for the adapter, as described in Section 7.5.1 and Section 7.6.

3. Add, edit, or delete **Attachment Relationship Attributes**.

   Use this area to configure custom relationships between attached documents and business entities, as described in Section 7.6.10. For example, enable users to print document attachments for certain business entities but not for others.

4. Add, edit, or delete **Entity Specific Configurations**.

   Use this area to override global settings for a specified business entity or application, as described in Section 7.6.3. For example, override a global setting that displays the From Repository button and hide the button for a specified business entity.

5. Add, edit, or delete **Role Specific Configurations** or **Business Context Specific Configurations**.

   Use these areas to configure which Managed Attachments features and buttons are displayed and available, based on either:

   - the user's assigned role, as described in Section 7.6.1. For example, configure all users granted a read role to have read-only Managed Attachments access.
   - the user's business application context, as described in Section 7.6.2. For example, display and enable the Scan button only when the businessObjectContext passed from the business application=ScanContext.

7.5.1 Working With Global Configuration Variables

Use the **Global Configuration** section of the Configure Managed Attachments page to change global environment variables from their default values.

Global configuration variables are grouped on the page as follows:

- Check-In variables (Table 7–7)
- Scanning variables (Table 7–8)
- Button variables (Table 7–9)
- General variables (Table 7–10)

Table 7–7 Check-in Variables

Field	Default Value	Description
AppAdapterCheckinPrivate	true	If set to true, the **Share Document** field on the Content Check-In and Scan Document pages is deselected by default. This setting configures documents for private check-in, although users can select the field for shared check-in. See Section 7.7.3.
AppAdapterExtraCheckinMetadata	null	Specify a comma-separated list of parameters to optionally pass on the AXF grant call to be prepopulated on the Checkin page (leaving no space after a comma before the next value). For more information, see Section 7.6.16.

Table 7–8 Scanning Variables

Field	Default Value	Description
AppAdapterExtraScanMetadata	null	Specify a comma-separated list of parameters to optionally pass on the AXF grant call to be prepopulated on the Scan Document page (leaving no space after a comma before the next value). For more information, see Section 7.6.16.

ODDCURLPath		Enables document attachment scanning and importing through Oracle Distributed Document Capture. For configuration instructions, see Section 7.9. This entry specifies the web address to Oracle Distributed Document Capture.
		Format:
		`ODDCURLPath`
		`http://`*`ODDC_host/ODDC_webcapture_address`*
		Example:
		`ODDCURLPath`
		`http://xyz/webcapture.asp`

ODDCScanAction	1	Use to configure document attachment scanning and importing through Oracle Distributed Document Capture, as described in Section 7.9. In this entry, specify the scan action to perform, where: ■ 1 = Scan ■ 2 = Import

Table 7–9 Button Variables

Field	Default Value	Description
AppAdapterDetachBtnVisible	true	Specifies if the **Detach** button displays on the Managed Attachments page.
AppAdapterRefreshBtnVisible	true	Specifies if the **Refresh** button displays on the Managed Attachments page.
AppAdapterCheckinNewBtnVisible	true	Specifies if the **New** button displays on the Managed Attachments page.
AppAdapterRepoAttachBtnVisible	true	Specifies if the **From Repository** button displays on the Managed Attachments page.
AppAdapterConfigureBtnVisible	true	Specifies if the **Configure** button displays on the Managed Attachments page.
AppAdapterSearchWithinBtnVisible	true	Specifies if the **Search Within** button that enables users to search within the attachments list displays on the Managed Attachments page.
AppAdapterScanBtnVisible	false	Specifies if the **Scan** button displays on the Managed Attachments page.
AppAdapterHelpIconVisible	true	Specifies if the **Help** icon displays on the Managed Attachments page.
AppAdapterAutoVueIconVisible	false	Specifies if the **AutoVue** icon displays in the Actions column on the Managed Attachments page. For more information, see Section 7.8.

Table 7–10 General Variables

Field	Default Value	Description
AppAdapterGrantPrivilege	W	Specifies the access level to grant to all users if the AF_GRANT_ACCESS service's *dPrivilege* parameter is not specified. Valid values include: R (read), W (write), D (delete) and A (admin). For more information, see Section C.5.1.
AppAdapterGrantHours	0.5	Specifies the time in hours for which the user session remains available. The solution uses this value only if the AF_GRANT_ACCESS service's *numHours* parameter is not specified. Valid values are numbers with optional decimal. For more information, see Section C.5.1.
AppAdapterMaxGrantHours	24	Specifies the maximum time in hours to grant the user access to the Managed Attachments page. Valid values are numbers with decimal. For example, suppose AppAdapterGrantHours is set to 1 hour and AppAdapterMaxGrantHours is set to 24 hours. If the user accesses the Managed Attachments page from the business application at 12:00 (noon), dExpirationDate in the AFGrants Table is set to 13:00 and dMaxExpiration is set to 12:00 (noon) the next day. If at 12:30, the user performs an action (such as checking in a document), dExpirationDate changes to 13:30, and so on. The result is that the user can have access to the Managed Attachments page up to 24 hours if at any given time the gap between two requests is less than one hour. Regardless, when 24 hours is reached, access is denied. This setting prevents a user from keeping access open for very long periods of time without user action, and prevents access from expiring if a user is actively using the system.
AppAdapterKeyDelimiter	|	Specifies the internal delimiter used to concatenate primary keys and values passed as parameters. For more information, see Section C.5.1. The default delimiter is a | (pipe) character.

AutoVueURLPath		Enables AutoVue viewing of attached documents through Oracle AutoVue Web Version. For configuration instructions, see Section 7.8. This entry specifies the AutoVue servlet to run.
		Format:
		`AutoVueURLPath`
		`http://ucm_host name:port/web_root/idcplg?IdcService=VIEW_IN_AUTOVUE`
		Example:
		`AutoVueURLPath`
		`http://ucm1.example.com/cs/idcplg?IdcService=VIEW_IN_AUTOVUE`

AppAdapterFrameLessWindowRequired	true	Specifies if standard browser menu options are hidden in browser windows that open from the attachments display (true) or displayed (false).
AppAdapterViewUnreleased	false	If set to true, displays documents in the attachments list before they are released, without links to the web files.
ResultCount	20	Specifies the number of attachments to display per page on the Managed Attachments page. If the number of results exceeds ResultCount, previous and next page controls display on the page.
SearchEngineName	DATABASE.METADATA.AFLIST	Specifies the search engine to use. You can choose: ▪ DATABASE.METADATA.AFLIST.STANDARD: Displays released documents only. ▪ DATABASE.METADATA.AFLIST: Displays released and unreleased documents. **Note:** If Database Full Text Search or Oracle Full Text Search is configured in Content Server, then the specified search engine is used only when users click the **From Repository** button.

7.5.2 Alternatively Editing Settings in the Preferences.hda File

The Managed Attachments Administration settings are stored in a configuration file called preferences.hda.

You can edit settings using the Managed Attachments Administration interface, as described in Section 7.5. Alternatively, you can follow the steps in this section to directly modify the preferences.hda file.

Notes: You must restart the Content Server after modifying the preferences.hda file.

You can add or delete application and business entity configuration entries as needed, but do NOT delete global configuration entries from the preferences.hda file.

---

1. Open the preferences.hda file in the following directory:

   UCM_Instance/data/appadapter

   This file provides Managed Attachments configuration options in the ResultSet sections described in Table 7–11.

Table 7–11 ResultSet Sections in Preferences.hda File

Section	Description
AFEnvironmentConfiguration	Use to configure Content Server environment variables for the adapter. For more information, see Table 7–7 and Section 7.6.
AFRelationshipAttributes	Use to configure custom relationships between attached documents and business entities. For more information, see Section 7.6.10.
AFRevisionSpecificEntities	Use to enable users to access the document version that was attached, even if the document was later updated. For more information, see Section 7.6.11.

2. Scroll to the AFEnvironmentConfiguration ResultSet section, and edit the entries described in Section 7.5.1 as needed. For more information about customizing the Managed Attachments solution using these variables, see Section 7.6.

7.6 Customizing the Managed Attachments Solution

This section describes methods of customizing the Managed Attachments solution. It covers the following topics:

- Section 7.6.1, "Customizing Settings Based on Users' Assigned Roles"
- Section 7.6.2, "Customizing Settings Based on Users' Business Context"
- Section 7.6.3, "Customizing Settings at Business Entity or Application Levels"
- Section 7.6.4, "Customizing the Document Type Used For New Attachments"
- Section 7.6.5, "Customizing the Content Server Profile Used"
- Section 7.6.6, "Hiding or Displaying Managed Attachments Page Buttons"
- Section 7.6.7, "Customizing Scanning"

- Section 7.6.8, "Customizing Managed Attachments Page Images"
- Section 7.6.9, "Customizing the Style Sheet (CSS) Used"
- Section 7.6.10, "Configuring Attachment Relationship Attributes"
- Section 7.6.11, "Configuring Revision-Specific Mode"
- Section 7.6.12, "Changing the Result Count"
- Section 7.6.13, "Changing the Result Template"
- Section 7.6.14, "Setting Default Columns"
- Section 7.6.15, "Enabling the Paperclip with Attachments (Oracle E-Business Suite Forms Only)"
- Section 7.6.16, "Passing Extra Metadata Values From a Business Application to Content Server"
- Section 7.6.17, "Resizing Oracle PeopleSoft Popup Solution Windows"
- Section 7.6.18, "Configuring an Entity's Private Attachments Security Group"
- Section 7.6.19, "Restoring Business Application User Pass Through for Managed Attachments"
- Section 7.6.20, "Renaming the Managed Attachments Button For OAF Entities"

7.6.1 Customizing Settings Based on Users' Assigned Roles

You can customize settings based on user role using Managed Attachments Administration. (For general information, see Section 7.5.) This enables you to retain a global value while customizing settings for specific user roles. For example, you might use this feature to limit Managed Attachments access to read-only for Content Server users who are assigned a read-only role.

1. On Content Server, choose **Managed Attachments** from the **Administration** menu.

   The Configure Managed Attachments page displays.

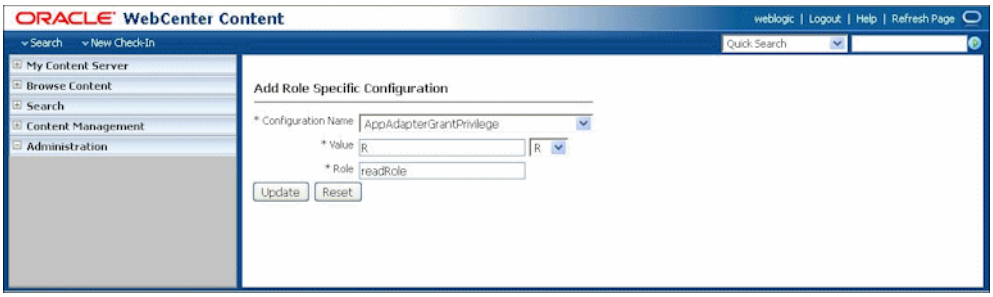
2. Scroll to the **Role Specific Configuration** settings in the lower portion of the page and click **Add**.

3. In the Add Role Specific Configuration page shown in Figure 7–5, enter fields to customize a Managed Attachments variable based on user role and click **Update**. This example specifies the following settings:

   - Configuration Name: AppAdapterGrantPrivilege
   - Value: R
   - Role: readRole

   The AppAdapterGrantPrivilege variable specifies the access level to grant to users. This example assumes that you have configured and assigned the user role (readRole, in this example) using User Admin functions in Content Server.

Figure 7–5 Example Role Specific Configuration



7.6.2 Customizing Settings Based on Users' Business Context

You can customize settings based on user's business context using Managed Attachments Administration. (For general information, see Section 7.5.) This enables you to retain the global value while customizing it for specific business contexts. For example, you might use this feature to display and enable the Scan button only when the user's business context (businessObjectContext passed from the business application) is a certain value (such as ScanContext).

1. On Content Server, choose **Managed Attachments** from the **Administration** menu.

   The Configure Managed Attachments page displays.

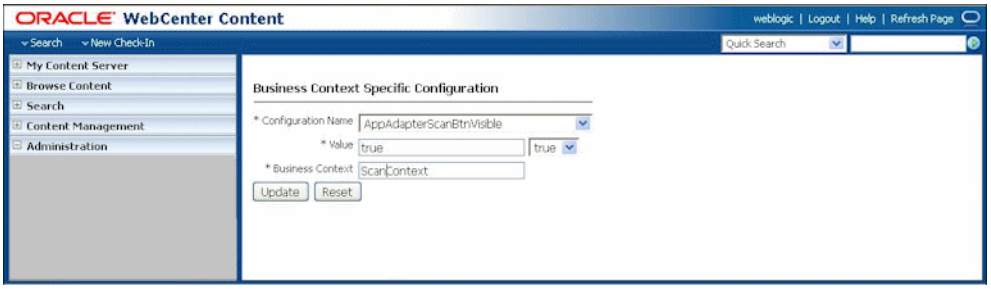
2. Scroll to the **Business Context Specific Configuration** settings in the lower portion of the page and click **Add**.

3. In the Business Context Specific Configuration page shown in Figure 7–6, enter fields to customize a Managed Attachments variable based on business context, and click **Update**.

   The example shown in Figure 7–6 configures the Scan button to display when the businessObjectContext passed from the business application equals ScanContext. This example specifies the following settings:

   - Configuration Name: AppAdapterScanBtnVisible
   - Value: true
   - Role: ScanContext

Figure 7–6 Example Business Context Specific Configuration



4. Configure the business application table to pass in the ScanContext value from the business application, just as you would configure key labels/values (labelValues).

   - For each Oracle PeopleSoft page to be configured, insert a row in the PS_AXF_CMD_PARAMS table that passes ScanContext. In Table C–36, see the row in

which AXF_PARAM_NAME=businessObjectContext and AXF_CONSTANT_VALUE=ScanContext.

- For each Oracle E-Business Suite OAF page, insert a row in the OAF_AXF_CMD_PARAMS table that passes ScanContext. In Table C–30, see the row in which AXF_PARAM_NAME=businessObjectContext and AXF_CONSTANT_VALUE=ScanContext.

7.6.3 Customizing Settings at Business Entity or Application Levels

You can customize settings at application and business entity levels using Managed Attachments Administration. (For general information, see Section 7.5.) This enables you to retain the global value while customizing it for specific business entities. For example, you might enable the Scan button to permit scanning globally, but disable it for a specific business entity by hiding the button.

1. On Content Server, choose **Managed Attachments** from the **Administration** menu.

   The Configure Managed Attachments page displays. Currently selected global settings are displayed under the Global Configuration heading.

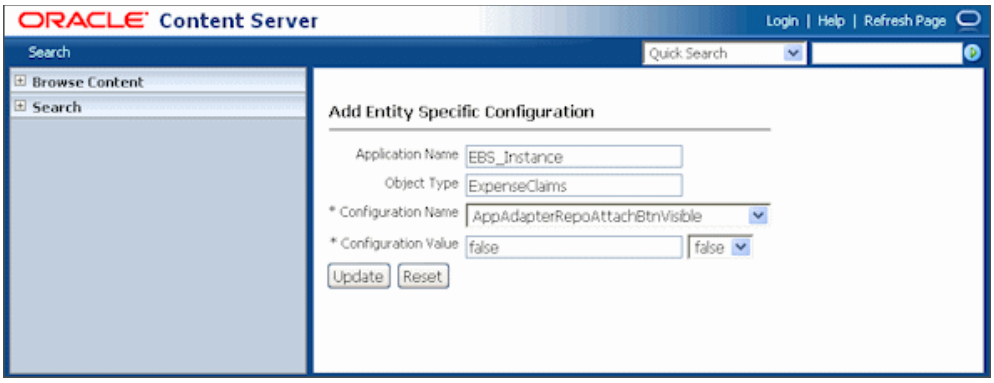
2. Scroll to the **Entity Specific Configuration** settings in the lower portion of the page and click **Add**.

3. In the Add Entity Specific Configuration page shown in Figure 7–7, enter fields to override the global setting for an entity or application.

   - To override at the business entity level, complete all fields. The example values below configure the **From Repository** button to display for the ExpenseClaims business entity only, regardless of the global setting.
   - To override at the application level, leave the **Object Name** field blank. Doing so in the example below would display the **From Repository** button for all business entities of the EBS_Instance application, regardless of the global setting.

Field	Description	Example
Application Name	Identifies the business application	EBS_Instance
Object Name	Identifies the business entity	ExpenseClaims
Configuration Name	Global configuration variable to set for business entity	AppAdapterRepoAttachButtonVisible
Configuration Value	Variable value to set for business entity	true

Figure 7–7 Example Entity Specific Configuration



Note: You must complete either the Application Name or Object Type field. Entering an object type without an application name applies the setting to the business entity of all applications with that object type.

7.6.4 Customizing the Document Type Used For New Attachments

A Content Server document type is selected by default when users click the **New** button to attach and check in a document. Using preferences.hda settings, you can specify a different default document type. In addition, you can specify document types on a global, application, or business entity level. For example, you might want to check in all new documents for an expenses business entity using a custom document type geared toward expense attachments.

Follow these steps.

1. On Content Server, create or modify a document type.

   For more information about document types, see the Content Server documentation.

2. In the preferences.hda file, add a new row in the AFEnvironmentConfiguration resultset. For general information, see Section 7.5.2.

3. Insert an entry that identifies the document type to use as the default type for a specified business entity, where:

   - name: afDocType
   - value: document type key
   - dAFApplication: Application to which to assign this document type
   - dAFBusinessObjectType: Business entity within application to which to assign document type

   Example 7–3 sets the document type *ExpenseClaimsType* as the default type when users attach documents to the ExpenseClaims business entity (within the PSFT_Instance application).

Example 7–3 Custom Default Document Type For Expenses Business Entity

```
afDocType
ExpenseClaimsType
PSFT_Instance
```

```
ExpenseClaims
```

4. Restart the Content Server.

7.6.5 Customizing the Content Server Profile Used

The adapter provides a default Content Server profile for each business application:

- *EBSProfile* for Oracle E-Business Suite
- *PSFTProfile* for Oracle PeopleSoft
- *UniversalProfile* for the Enterprise Application Adapter Framework

You set the profile when running the AXF_ATTACHMENTS_CONFIG.sql script in Section 7.2.1. You can copy and customize the profile, then assign it using one of the following methods, depending on customization needs. For example, you might want to add metadata fields to the profile.

> **Note:** If you use custom profiles, generic error messages display by default instead of application-specific error messages. To display custom error messages, override the resource include - af_custom_error_msg_display. In the overridden resource include, set the error message to **afErrorMsg**.

7.6.5.1 Changing the Profile Globally

Follow these steps to change the global profile used.

1. From the Content Server Configuration Manager, copy the default profile provided (EBSProfile, PSFTProfile, or UniversalProfile). To view these profiles, navigate to the **Profiles** tab in Configuration Manager.
2. Add new rules as needed. For example, add new metadata fields.
3. In the UCM_CONNECTION_STR parameter (trigger value) of the AXF_SOLUTION_PARAMETERS table, modify the URL to point to the new profile.

   For more information, see Section C.1.4.

7.6.5.2 Specifying a Profile For a Specific Business Entity or Application

Follow these steps to modify the profile for a specific business entity or application.

1. From the Content Server Configuration Manager, copy the default profile provided (EBSProfile, PSFTProfile, or UniversalProfile). To view these profiles, navigate to the **Profiles** tab in Configuration Manager.
2. Add new rules as needed. For example, add new metadata fields.
3. In the preferences.hda file, add a new row in the AFEnvironmentConfiguration resultset. For general information, see Section 7.5.2.
4. Insert an entry that identifies the Content Server profile to use for a specified business entity, where:
   - name: dpTriggerValue
   - value: Key of new Content Server profile
   - dAFApplication: Application to which to assign this profile

- dAFBusinessObjectType: Business entity within application to which to assign the profile

Example 7–4 sets a profile called *OrderPayableProfile* to use when users attach or scan documents in the OrderPayables business entity within the EBS_Instance application.

Example 7–4 Custom Content Server Profile For Expenses Business Entity

```
dpTriggerValue
OrderPayableProfile
EBS_Instance
OrderPayables
```

5. Restart the Content Server.

7.6.6 Hiding or Displaying Managed Attachments Page Buttons

You can choose to enable (display) or disable (hide) the buttons listed in Table 7–12 to users on the Managed Attachments page, by changing their variable setting to true or false, respectively. For details about Managed Attachments variables, see Section 7.5.1.

Table 7–12 Variables For Displaying or Hiding Managed Attachments Buttons

Button Variables	Description
AppAdapterRepoAttachBtnVisible	Displays or hides the **From Repository** button
AppAdapterRefreshBtnVisible	Displays or hides the **Refresh** button
AppAdapterConfigureBtnVisible	Displays or hides the **Configure** button
AppAdapterDetachBtnVisible	Displays or hides the **Detach** button
AppAdapterCheckinNewBtnVisible	Displays or hides the **New** button
AppAdapterScanBtnVisible	Displays or hides the **Scan** button
AppAdapterAutoVueIconVisible	Displays or hides the **AutoVue** button in the Actions column
AppAdapterSearchWithinBtnVisible	Displays or hides the **Search Within** button
AppAdapterHelpIconVisible	Displays or hides the **Help** icon

Follow these steps to hide or display a button globally, or for a specific business entity or application.

1. On Content Server, choose **Managed Attachments** from the Administration menu.
2. Under Global Configuration, change a button's global setting to **false** to hide it, or **true** to display it.
3. To configure a button's display at a business entity level, create an entity-specific configuration by clicking Add under **Entity Specific Configurations** and completing fields as described in Section 7.6.3.

7.6.7 Customizing Scanning

This section describes methods of customizing scanning functionality using Oracle Distributed Document Capture. To customize scanning, you must also configure standard scanning options, as described in Section 7.9. It includes the following topics:

- Section 7.6.7.1, "Enabling or Disabling Scanning"

- Section 7.6.7.2, "Configuring an Alternate Scan Profile For a Business Entity"
- Section 7.6.7.3, "Modifying the Document Classification Used"

7.6.7.1 Enabling or Disabling Scanning

You can enable or disable scanning at the global, application, or business entity level. For example, you might disable scanning globally, then add an entity specific configuration to enable scanning for an OrderPayables business entity only, using the following Entity Specific Configuration settings:

- Application Name: EBS_Instance
- Object Type: OrderPayables
- Configuration Name: AppAdapterScanBtnVisible
- Configuration Value: true

7.6.7.2 Configuring an Alternate Scan Profile For a Business Entity

The adapter provides these default business application scan profiles. Each points to an application-specific field:

- *EBSScanProfile* references the EBSAppScanProfiles application field
- *PSFTScanProfile* references the PSFTAppScanProfiles application field
- *UniversalScanProfile* references the UniversalAppScanProfiles application field

You can customize the scan profile used, by copying the profile appropriate for your business application, modifying it to reference another application field, and configuring it for one or more business entities, as described in the following steps.

1. Copy the scan profile that applies to your business application, saving it with a new name.
2. Create all required schema elements, including the AppScanProfiles application field, table, and view.

   To change the application field reference, log in to Content Server. In Configuration Manager, navigate to the Rules tab. Select **PSFTScanRule**, **EBSScanRule**, or **UniversalScanRule**, click **Edit** on the Side Effects tab, and update the ScanProfileKey field to point to another application field.
3. In the preferences.hda file, add a new row in the AFEnvironmentConfiguration resultset.
4. Insert an entry that includes the afScanProfile setting, identifies the scan profile to use, and the application and business entity in which it applies.

   Example 7–5 shows a scan profile called *MyPIMScanProfile* enabled for the PIM business entity only.

Example 7–5 Enabling Another Scan Profile for a Business Entity

```
afScanProfile
MyPIMScanProfile
PSFT_Instance
PIM
```

7.6.7.3 Modifying the Document Classification Used

The adapter provides the default business application document classifications for scanning listed in Table 7–13. Each classification includes a set of Content Server application fields, a table, and a view.

Table 7–13 Default Documentation Classifications Provided

Business Application	Content Server Fields	Table	View
Oracle E-Business Suite	EBSAppScanProfiles	EBSDocClass	EBSDocClassView
Oracle PeopleSoft	PSFTAppScanProfiles	PSFTDocClass	PSFTDocClassView
Enterprise Application Adapter Framework	UniversalAppScanProfiles	UniversalDocClass	UniversalDocClassView

To modify the classification:

1. Log in to Content Server. In Configuration Manager, navigate to the EBSDocClassView, PSFTDocClassView, or UniversalDocClassView.
2. Edit the document classification and scan type values. For more information, see Section 7.9.4.

7.6.8 Customizing Managed Attachments Page Images

The adapter provides standard images for Managed Attachments display, which you can customize. For example, you might replace the Oracle blue waves image with one specific for your organization. To change images, override the resource include *af_ebs_define_image_locations*. For sample implementations, see *AppAdapter_JS_resource.htm* for Oracle E-Business Suite or *ps_AppAdapter_Js_resource.htm* for Oracle PeopleSoft.

Note: Customizing images for Managed Attachments requires advanced knowledge of Content Server configuration and customization. Be sure to back up the appropriate files before making modifications.

7.6.9 Customizing the Style Sheet (CSS) Used

The adapter provides a standard style sheet for the Managed Attachments page. You can change it by overriding resource includes. For sample implementations, see *AppAdapter_CSS_resource.htm* for Oracle E-Business Suite or *ps_AppAdapter_CSS_resource.htm* for Oracle PeopleSoft.

Note: Customizing style sheets for Managed Attachments requires advanced knowledge of Content Server configuration and customization. Be sure to back up the appropriate files before making modifications.

7.6.10 Configuring Attachment Relationship Attributes

Attachment relationship attributes allow you to store relationship metadata (information about relationships between a document and its business entity) upon attaching documents in Managed Attachments.

For example, suppose you have one document attached to two work order entities (WO1 and WO2), and want to allow users to print the document from WO1, but not

from WO2. In this case, you would create a print attribute as an application field on Content Server, and then specify a print relationship attribute for the WO1 business entity in Managed Attachments Administration (or in the preferences.hda file). The print attribute would be displayed on the Update page to end-users for the WO1 entity only, allowing them to print the attached document.

Important Points About Attachment Relationship Attributes

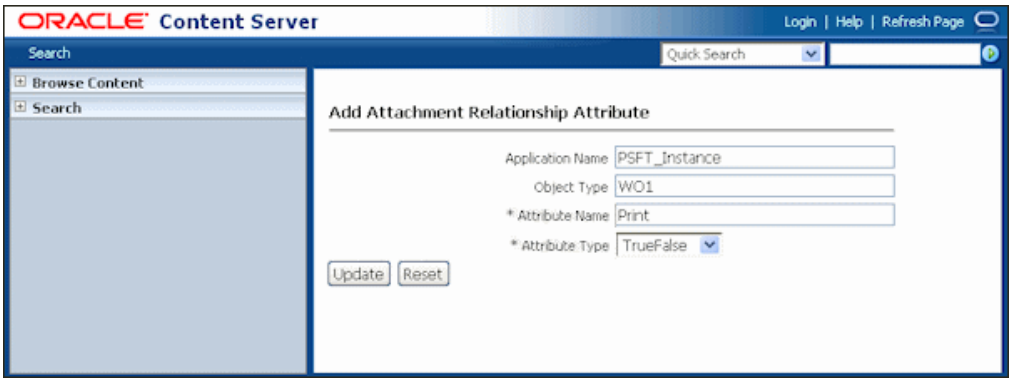
- You can configure attachment relationship attribute for new document check-in, but not for documents attached through scanning.
- Adding a relationship attribute creates a new Application field in Content Server for the relationship attribute if one does not exist. You can update the Application field using Content Server Admin Applets - Configuration Manager. Adding a relationship attribute also adds an entry in the AFRelationshipAttributes section of the preferences.hda file described in Section 7.5.2.
- At run-time, attachment relationship attribute values are stored in the AFRelationshipAttributes table; see Section C.5.3.

Follow these steps to configure attachment relationship attributes.

1. On Content Server, open Configuration Manager and navigate to the **Application Fields** tab.
2. Add one or more application fields.
3. Navigate to the Rules tab.
4. Edit the AFRelationshipAttributes rule. Add the application fields you added in step 2 to this rule.
5. On Content Server, choose **Managed Attachments** from the **Administration** menu.
6. On the Configure Managed Attachments page, scroll to the **Attachment Relationship Attributes** settings in the lower portion of the page, and click **Add**.

   The Add Attachment Relationship Attribute page displays.

Figure 7–8 Add Attachment Relationship Attribute Page



7. Complete fields for the attribute and click **Update**. The example in Figure 7–8 shows a relationship attribute (Print) configured for an WO1 business object for an Oracle PeopleSoft application.

Element	Description	Example
Application Name	Business application for which the relationship attribute is configured.	PSFT_Instance
Object Type	Business object for which to enable the relationship attribute	WO1
Attribute Name	Relationship attribute	Print
Attribute Type	Attribute's type. Available types include: Text, Long Text, Date, Memo, Integer, and TrueFalse.	TrueFalse

7.6.11 Configuring Revision-Specific Mode

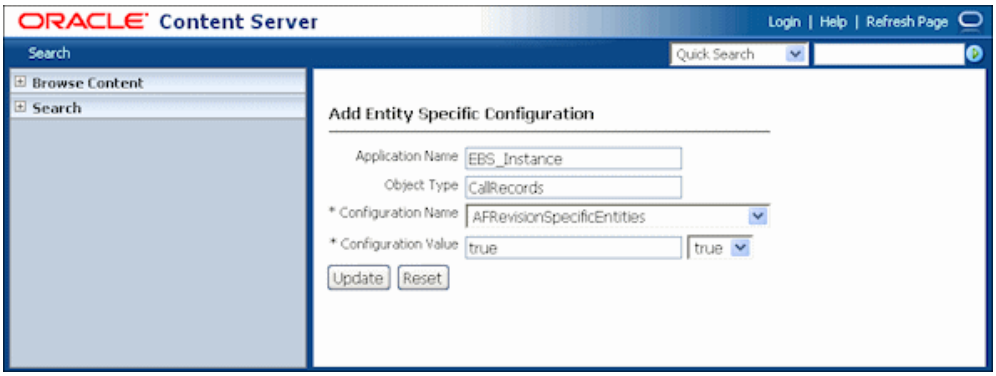
By default, each attachment on the Managed Attachments page provides access to the *most recent revision* of the document. In certain cases, however, you may want to configure one or more business entities to provide access to the revision that was *attached*, even if the document was later updated on Content Server. In this mode, users can see that a more recent revision exists, and if needed, view it and make it the attached revision.

Follow these steps to configure a business entity to display the attached revision instead of the most recent revision.

1. On Content Server, choose **Managed Attachments** from the Administration menu.
2. On the Configure Managed Attachments page, scroll to the **Entity Specific Configuration** settings in the lower portion of the page, and click **Add**.

   The Add Entity Specific Configuration page displays.

Figure 7–9 Configuring Revision Specific Mode in the Add Entity Specific Configuration Page



3. Complete fields for revision specific mode, choosing AFRevisionSpecificEntities for the **Configuration Name** field, and click **Update**. The example in Figure 7–9 shows revision specific mode configured for a CallRecords business object for an Oracle E-Business Suite application.

Element	Description	Example
Application Name	Business application for which the relationship attribute is configured.	EBS_Instance
Object Type	Business object for which to enable the relationship attribute.	CallRecords

Element	Description	Example
Configuration Name	To configure revision specific mode, choose **AFRevisionSpecificEntities**.	Print
Field Type	To configure revision specific mode, choose true.	true

Alternatively, follow these steps to configure revision-specific mode using the preferences.hda file.

1. Open the preferences.hda file, and add a new row in the AFRevisionSpecificEntities resultset. (For general information, see Section 7.5.2.)

   Each row in the resultset represents one business entity for which to enable revision specific attachments, where:.

   - *dAFApplication* refers to the business application for which to enable the revision feature.
   - *dAFBusinessObjectType* refers to the business object for which to enable the revision feature.

   Example 7–6 shows revision-specific mode configured for a CallRecords business object for an Oracle E-Business Suite application.

Example 7–6 AFRevisionSpecificEntities Customization

```
@ResultSet AFRevisionSpecificEntities
2
dAFApplication
dAFBusinessObjectType
EBS_Instance
CallRecords
@end
```

7.6.12 Changing the Result Count

The Managed Attachments page displays a specific number of attachment results per page (referred to as *ResultCount*). If the number of results exceeds ResultCount, previous and next page controls display on the page. You can change it to another number (such as 20) by specifying a new ResultCount value in Managed Attachments configuration. See Section 7.5.1.

7.6.13 Changing the Result Template

The Managed Attachments page uses a result template to determine its display. The Content Server result template specifies the Managed Attachments user interface. The adapter provides these result templates:

- The Oracle E-Business Suite template, *adapter_list_results_template.htm*, is referred to as **EBS_LIST**.
- The Oracle PeopleSoft template, *ps_adapter_list_results_template.htm*, is referred to as **PSFT_LIST**.
- The Enterprise Application Adapter Framework template, *universal_adapter_list_results_template.htm*, is referred to as **Universal_LIST**. See Section 7.4.1.2.

Follow these steps to identify an alternate result template for an application or business entity.

Note: Customizing the result template requires advanced knowledge of Content Server administration and customization.

1. Create a copy of the appropriate default result template and register the template with a new name, preferably in a new component.
   - The Oracle E-Business Suite template is located at:

     UCM_HOME/custom/AppAdapterEBS/adapter_list_results_template.htm
   - The Oracle PeopleSoft template is located at:

     UCM_HOME/custom/AppAdapterPSFT/ps_adapter_list_results_template.htm
   - The Enterprise Application Adapter Framework template is located at:

     UCM_HOME/custom/AppAdapterUniversal/universal_adapter_list_results_template.htm
2. In the preferences.hda file, add an entry in the AFEnvironmentConfiguration section as follows:

   ```
   ResultTemplate
   Template Reference
   Application Name
   Business Entity
   ```

   In Example 7–7, all requests to Managed Attachments for the WorkOrder business entity would use the *EBS_WORK_ORDER_LIST* custom result template instead of the default Oracle E-Business Suite template.

Example 7–7 Preferences.hda Entry For Specifying a Custom Result Template

```
ResultTemplate
EBS_WORK_ORDER_LIST
EBS_Instance
WorkOrder
```

7.6.14 Setting Default Columns

You can specify the default columns to display to users in the Managed Attachments page. (When the user clicks **Reset** in the Configure Fields for Display page, default columns are listed in the Main Information section of the page.)

The column defaults are title, type, author, date, and revision as follows:

dDocTitle,dDocType,dDocAuthor,dInDate,dRevision

To set alternate default columns for an application or business entity, follow these steps.

1. In the preferences.hda file, add an entry in the AFEnvironmentConfiguration section.
2. Set the value of the *AppAdapterDefaultDisplayColumns* configuration variable as a comma-delimited list containing Content Server metadata field names.

   In Example 7–8, the first entry sets document title, type, author, and date as the default columns for the EBS_Instance application. The second entry overrides this setting for the CallRecords business entity by adding the revision column (dRevision) as a default column.

Example 7–8 Preferences.hda Entries For Specifying Default Columns

```
AppAdapterDefaultDisplayColumns
dDocTitle,dDocType,dDocAuthor,dInDate
EBS_Instance

AppAdapterDefaultDisplayColumns
dDocTitle,dDocType,dDocAuthor,dInDate,dRevision
EBS_Instance
CallRecords
```

7.6.15 Enabling the Paperclip with Attachments (Oracle E-Business Suite Forms Only)

By default, the Oracle E-Business Suite paperclip attachment option is disabled as part of Managed Attachments configuration. To use this native Oracle E-Business Suite attachments feature in coexistence with the Managed Attachments solution, you must reenable the paperclip icon as described as follows.

Note: This section applies to Oracle E-Business Suite Forms use only.

1. Add the following row to the AXF_PROPERTIES table:

Table 7–14 AXF_PROPERTIES Values For PaperClip Use

PROPNAME	PROPVALUE
AXF_PAPERCLIP	Set to `TRUE` to enable the paperclip option, or `FALSE` (default) to disable it. (The value must be in uppercase characters.)

For more information, see Section C.2.2.

7.6.16 Passing Extra Metadata Values From a Business Application to Content Server

You can configure the Managed Attachments solution to pass extra metadata values from the business application to Content Server to store with attached documents. For example, you might pass a claim number from a Claims application so that when users attach or scan documents, a Claim Number field on the Content Check In and Scan Document pages is prepopulated with the value.

This feature works as follows:

- The business application field whose metadata value is to be passed to Content Server must exist in the business application. For example, you might pass a business record's service request number or primary key.
- Using Content Server's Managed Attachments administration variables (Table 7–7 and Table 7–8), you specify the field whose value will be passed when users either check in or scan a document. This prepopulates the specified field in the Content Check In or Scan Document page for users.
- In the business application's command parameters table, you identify the extra metadata field to be passed.

Follow these steps to configure passing extra metadata values.

1. From Content Server, choose **Managed Attachments** from the Administration menu.
2. Under **Global Configuration**, enter one or more field names to pass, using a comma to separate multiple names.

- Enter field names in the **AppAdapterExtraCheckinMetadata** field to pass their values directly to the Content Check In page.
- Enter field names in the **AppAdapterExtraScanMetadata** field to pass their values directly to the Scan Document page.

3. Access the command parameters table for your business application.
   - For an Oracle E-Business Suite Forms configuration, access the AXF_COMMAND_PARAMETERS Table (Oracle E-Business Suite Forms).
   - For an Oracle E-Business Suite OAF configuration, access the OAF_AXF_CMD_PARAMS Table (Oracle E-Business Suite OAF).
   - For a PeopleSoft configuration, access the PS_AXF_CMD_PARAMS Table.
   - For an Enterprise Application Adapter Framework configuration, see Section 7.6.16.1.
4. Add entries in the table for the extra values to be passed to Content Server, similar to adding labelValue rows. See the example command shown in Example 7–9 and the example table shown in Table 7–15.

   Add a row for each metadata value, prepending the Content Server field name with **`extra`**. For example, `xReqNum` becomes `extraxReqNum` and `dDocTitle` becomes `extradDocTitle`.

> **Note:** Typically, Content Server extra information fields have an x character in front of them and standard ones have a d character in front of them. For more information, refer to the section on standard metadata fields in the *Oracle Fusion Middleware Managing Oracle WebCenter Content*.

Example 7–9 Inserting Extra Metadata Field into OAF Command Parameters Table

```
Insert into OAF_AXF_CMD_PARAMS (AXF_CMD_PARAMS_ID, AXF_CMDS_ID, AXF_PARAM_NAME,
AXF_DATASOURCE,AXF_VIEW_OBJ_NAME, AXF_ATTRIBUTE_NAME,AXF_CONSTANT_VALUE) values
(OAF_AXF_CMD_PARAMS_SEQ.NEXTVAL,v_cmdId,'extraxReqNum','DATA','ReqHeaderVO',
'ReqNum',null);
```

Table 7–15 Example OAF_AXF_CMD_PARAMS Table With Extra Metadata Field Specified

AXF_CMD_PARAMS_ID	AXF_CMDS_ID	AXF_PARAM_NAME	AXF_DATASOURCE	AXF_VIEW_OBJ_NAME	AXF_ATTRIBUTE_NAME	AXF_CONSTANT_VALUE
1	1	application	CONSTANT			EBS_instanceA
2	1	businessObjectType	CONSTANT			REQ_HEADERS
3	1	businessObjectKey1	CONSTANT			REQUISITION_HEADER_ID
4	1	businessObjectValue1	DATA	ReqHeaderVO	RequisitionHeaderId	
5	1	labelValue1	DATA	ReqHeaderVO	PreparerName	
6	1	labelValue2	DATA	ReqHeaderVO	ReqNum	
7	1	**extraxReqNum**	DATA	ReqHeaderVO	ReqNum	

7.6.16.1 Passing Values Via Enterprise Application Adapter Framework

For the Enterprise Application Adapter Framework, pass the extra metadata values as part of the request parameters section of the SOAP envelope body, prepending the

Content Server field name with **extra** (xReqNum becomes extraxReqNum and dDocTitle becomes extradDocTitle). For example:

```
<entry><key>extraxReqNum</key><value>123456</value></entry>
```

7.6.17 Resizing Oracle PeopleSoft Popup Solution Windows

You can change the default system-wide size for all Oracle PeopleSoft popup windows that display as part of solutions. Note the following:

- The default popup window size is 600 pixels high by 800 wide, set in the PS_AXF_PROPERTIES Table. When a size is set in this table, users cannot set and save popup window size themselves.
- To change window size, enter new values in the table, as shown in Section C.4.4.
- To remove the popup window size and allow individual users to change and save popup window size using standard browser controls, remove the WINDOW_HEIGHT and WINDOW_WIDTH rows from the PS_AXF_PROPERTIES Table.

7.6.18 Configuring an Entity's Private Attachments Security Group

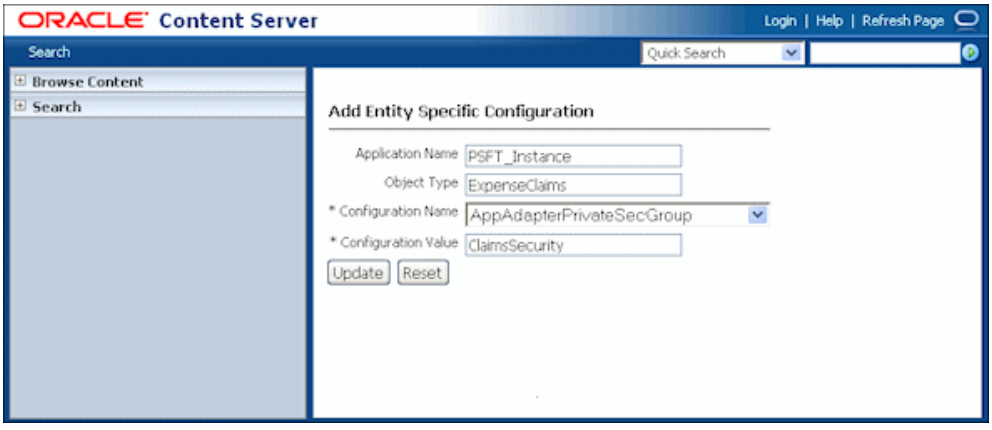
When users check in documents through Managed Attachments, they can choose private or shared access for them, as described in Section 7.7.3. Private documents are automatically assigned to a special Content Server security group called *AFDocuments*.

On a per entity level, you can optionally override the security group for private attachments, and assign documents to an alternate security group, by following these steps.

1. On Content Server, choose **Managed Attachments** from the Administration menu.
2. On the Configure Managed Attachments page, scroll to the **Entity Specific Configuration** settings in the lower portion of the page, and click **Add**.

   The Add Entity Specific Configuration page displays.

Figure 7–10 Configuring Entity Level Private Attachments Security Group



3. In the Add Entity Specific Configuration page shown in Figure 7–10, complete the fields, identifying the *AppAdapterPrivateSecGroup* variable and the alternate security group to use for a specified business entity, where:
   - Application Name: Application for which to assign alternate security group
   - Object Type: Business entity for which to assign alternate security group

- Configuration Name: AppAdapterPrivateSecGroup
- Configuration Value: Alternate security group

The example values shown in Figure 7–10 set *ClaimsSecurity* as the security group when users attach private documents to the ExpenseClaims business entity (within the PSFT_Instance application).

4. Assign the following roles to the specified security group (must be the same as those assigned to the AFDocuments security group): AFRead, AFWrite, AFDelete, and AFAdmin.
5. In the config.cfg file, add the alternate security group to the **SpecialAuthGroups** configuration variable.

7.6.19 Restoring Business Application User Pass Through for Managed Attachments

Note: This section applies only if using Managed Attachments with the AXF for BPEL infrastructure. It does not apply if using the AXF for BPM infrastructure.

As described in Section 6.4.2, the BPEL Imaging solution allows you to configure the authenticated Oracle WebLogic Server user to perform tasks, rather than the user passed from Oracle E-Business Suite or Oracle PeopleSoft. This option is provided through a USERNAME_PASS_THROUGH parameter in the AXF_SOLUTION_ATTRIBUTES Table, where:

- TRUE (default): Uses the passed-in user from Oracle E-Business Suite or Oracle PeopleSoft to perform tasks.
- FALSE: Uses the authenticated Oracle WebLogic Server user to perform tasks.

Note: If no value is provided, the value defaults to FALSE.

The Managed Attachments solution requires that the passed-in user from Oracle E-Business Suite or Oracle PeopleSoft perform tasks. The AXF_ATTACHMENTS_CONFIG.SQL script that you ran in Section 7.2.1 for Managed Attachments sets the USERNAME_PASS_THROUGH value to TRUE. If needed at any point, restore the parameter's value to TRUE, as shown in Table 7–16 and Section C.1.2.

Table 7–16 AXF_SOLUTION_ATTRIBUTES Table, USERNAME_PASS_THROUGH Configuration for Managed Attachments

SOLUTION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
AXF	USERNAME_PASS_THROUGH	TRUE

7.6.20 Renaming the Managed Attachments Button For OAF Entities

To rename the Managed Attachments button for Oracle E-Business Suite OAF entities, such as for localization purposes:

1. Open the *ManagedAttachmentButtonRN.xml* file, which was imported into the MDS repository on the Oracle E-Business Suite OAF side during setup.
2. Locate the following line and replace `Managed Attachments` with a localized name.

```
<oa:button id="MAButton" prompt="Managed Attachments"/>
```

3. Using the import command, reimport the file into MDS.

7.7 Configuring Authentication and Security

This section provides instructions on managing authentication and security for the Managed Attachments solution. It covers the following topics:

- Section 7.7.1, "Configuring a CSF Credential for Managed Attachments"
- Section 7.7.2, "Configuring User Authentication"
- Section 7.7.3, "Configuring Document Security"
- Section 7.7.4, "Securing Communications Between AXF and Content Server"
- Section 7.7.5, "Managing Business Application and Content Server Users"

7.7.1 Configuring a CSF Credential for Managed Attachments

You must store the administrator user credentials in a credential store framework (CSF), a user name/password pair that is keyed by an alias and stored inside a named map in the CSF. The user name stored in the CSF is compared with the user name sent by Oracle E-Business Suite, Oracle PeopleSoft, or a custom plug-in calling the Enterprise Application Adapter Framework in the web services call. If the administrator user names do not match (see Section C.1.4), access to the Managed Attachments page is prevented.

You can create a credential through Enterprise Manger (EM) or through WebLogic Scripting Tool (WLST).

7.7.1.1 Creating a Credential Using Enterprise Manager

1. Log in to Enterprise Manager.
2. Right-click on your domain under **WebLogic Domain**.
3. Click **Security** and then **Credentials**.
4. Select the oracle.wsm.security map. If it does not exist, follow these steps:
   a. Select **Create Map**.
   b. Enter **`oracle.wsm.security`** in the map field and click **OK**.
5. Expand the oracle.wsm.security map.
6. Click **Create Key**. The key is now available for selection.
7. Enter **`MA_CSF_KEY`** as the key name.
8. Select **password** in the **Type** field.
9. Enter a user name and password for a valid Oracle WebLogic Server administrator (for example, **`weblogic`** and **`Welcome1`**).

   The user name and password are passed by the business application to AXF, and were previously configured for the Web Services call from Oracle E-Business Suite (see Section 2.4.1), Oracle PeopleSoft (see Section 3.2.2.1), and a custom plug-in calling the Enterprise Application Adapter Framework (see Section 7.4.2).
10. Click **OK**.

7.7.1.2 Creating a Credential Using WLST

1. Execute the following command, where **username** and **password** refer to a valid Oracle WebLogic Server administrator passed by the business application to AXF, and were previously configured for the Web Services call from Oracle E-Business Suite (see Section 2.4.1), Oracle PeopleSoft (see Section 3.2.2.1), and a custom plug-in calling the Enterprise Application Adapter Framework (see Section 7.4.2).

   ```
   createCred(map="oracle.wsm.security", key="MA_CSF_KEY",
   user=[username], password=[password])
   ```

   Example:

   ```
   createCred(map="oracle.wsm.security", key="MA_CSF_KEY",
   user="weblogic", password="Welcome1")
   ```

2. Execute the following command to verify that the credential was created properly:

   ```
   listCred(map="oracle.wsm.security",key="MA_CSF_KEY")
   ```

7.7.2 Configuring User Authentication

Business application users must have a Content Server account to display the Managed Attachments page within the supported business application entity. In addition, the authentication model configured for Content Server and the business application determines how users are authenticated the first time they activate managed attachments from a business application record:

- **Content Server configured for single sign-on:** If the business application is not configured for single sign-on, the Content Server single sign-on login prompt displays. (If the business application is configured for single sign-on, the user has been authenticated so no login prompt displays.)
- **Content Server *not* configured for single sign-on:** The Content Server login prompt displays, regardless of the selected business application authentication model.

> **Note:** No user authentication is needed for AXF if security checks are properly configured, as described in Section 7.7.

7.7.3 Configuring Document Security

When checking documents into Content Server through the Managed Attachments user interface, users specify their access by choosing one of two modes:

- **private** (not shared): Users can access these documents only through their associated business application entity screens through the Managed Attachments user interface. Users (including the user who checks in a document) cannot search for or access a private document using any other standard Content Server user interface. This is the default security mode when checking in a new document through the Managed Attachments user interface.

> **Note:** The **AppAdapterCheckinPrivate** global configuration variable determines if the **Share Document** field on the Content Check-In page is deselected by default (set for private check-in). For more information, see Section 7.5.1.

Note: Private (not shared) documents are automatically assigned to a special security group called *AFDocuments*, and users who have access to the business application entity are granted temporary access to the documents when they invoke the Managed Attachments user interface. The **AppAdapterPrivateSecGroup** variable enables you to override on a per entity level the security group to use for private attachments, and assign them to an alternate group. For more information, see Section 7.6.18.

In certain exceptional cases, an administrator might grant special users direct access to the AFDocuments security group by permanently assigning the AFRead, AFWrite, AFDelete, or AFAdmin roles for the AFDocuments security group, in which case the user could access a private document through any standard Content Server user interface.

- **shared**: These documents are more easily accessed than private documents, because their security is managed by Content Server. In addition to access through their associated business application entity screens through the Managed Attachments user interface, any Content Server user with a document's assigned security group access can search for and access the document using any standard Content Server user interface.

Note: Users can view shared documents only in AutoVue (if configured). They cannot view private documents. For details, see Section 7.8.

7.7.4 Securing Communications Between AXF and Content Server

Follow these steps to enable trusted communication between the host on which AXF is running and the Content Server. On the Content Server, you have two options for this configuration step: either using the System Properties application or by editing the config.cfg file.

7.7.4.1 Using the Content Server System Properties Application

1. Open the System Properties utility for the Content Server instance.
2. Select the Server tab.
3. Identify the AXF host by either entering a name in the **Hostname Filter** field or an address in the **IP Address Filter** field.
4. Click OK and restart Content Server.

7.7.4.2 By Editing the Content Server Config.cfg File

1. Open the config.cfg file.
2. Locate the SocketHostAddressSecurityFilter entry, and edit it to include the IP address of the system on which AXF is running.
3. Restart Content Server.

7.7.5 Managing Business Application and Content Server Users

Keep the following guidelines in mind when managing Oracle E-Business Suite, Oracle PeopleSoft, or other business application users for Managed Attachments access (for example, when managing users in a central repository using single sign-on):

- In order for business application users to access Managed Attachments functionality, their Oracle E-Business Suite, Oracle PeopleSoft, or other business application user names MUST match their Content Server user names.
- When creating Content Server users (global or local), do NOT assign the AFRead, AFWrite, AFDelete, or AFAdmin roles.

  The adapter dynamically assigns roles to the user: Roles are granted to the user based on dPrivilege's value when the AF_GRANT_ACCESS service is invoked (see Section C.5.1); or, if not specified, based on the configuration variable AppAdapterGrantPrivilege's value (see Section 7.5). Depending on the privilege (R, W, D, or A), a predefined role is dynamically assigned to the user.

> **Caution:** Do not delete the AFRead, AFWrite, AFDelete, AFAdmin roles from the system.

7.8 Configuring Oracle AutoVue Document Viewing

This section covers the following topics:

- Section 7.8.1, "About Configuring Oracle AutoVue"
- Section 7.8.2, "Requirements For Oracle VueLink for Content Server"
- Section 7.8.3, "Enabling AutoVue For the Adapter"

7.8.1 About Configuring Oracle AutoVue

The default document viewing option for the Managed Attachments page is the standard web rendition provided by Content Server. For enhanced viewing options, the adapter also supports Oracle AutoVue through VueLink for Content Server, where non-private attachments of virtually any document type are available for viewing, printing, collaboration, annotation, and mark up, delivered securely using Web technologies.

Important Points About the AutoVue/Adapter Integration

- If configured, users can view all non-private attachments through AutoVue. For details on shared versus private documents, see Section 7.7.2.
- The VueLink integration serves as a bridge that securely streams documents from the Content Server repository to AutoVue for viewing, digital markup and collaboration. Digital markups are saved in the Content Server repository and associated with the attached document.
- By default, AutoVue is turned off in the Managed Attachments solution. You activate it on the Configure Managed Attachments page, by displaying the AutoVue icon and identifying the URL to the AutoVue servlet to launch when the user clicks the **View in AutoVue** icon.
- As with other preferences settings, you can enable the AutoVue integration at the global, application, and business entity levels.

- AutoVue trusts the user authentication against Content Server. Once authenticated to Content Server, no further authentication to AutoVue is required for users to view attached documents using AutoVue.
- Viewing or annotating a document in AutoVue does not check it out or prevent it from being edited by others. Annotations are contained in a separate layer from their corresponding document.

7.8.2 Requirements For Oracle VueLink for Content Server

This section lists requirements for integrating AutoVue viewing with Content Server. For specific release requirements, see the configuration and certification information at the following location:

http://www.oracle.com/technetwork/middleware/webcenter/content/documentation/documentation-155348.html

Note: The requirements listed in this section apply only if providing optional document viewing with the adapter through Oracle AutoVue.

Note: The AutoVue option for Managed Attachments is not supported on WebSphere Application Server.

Requirements include:

- Oracle AutoVue (and any applicable service packs).
- Oracle VueLink for Content Server, the AutoVue VueLink integration for Content Server. (Also see the system administrator documentation for this integration.)

7.8.3 Enabling AutoVue For the Adapter

Follow these steps to enable and configure AutoVue document viewing globally, or for a specific application or business entity.

1. Configure the Content Server/AutoVue integration.

   Follow the procedures outlined in the *Oracle VueLink 20 for Oracle UCM System Administrator Manual*.

2. On Content Server, choose **Managed Attachments** from the Administration menu.
3. On the Configure Managed Attachments page, enable display globally of the AutoVue icon by setting the **AppAdapterAutoVueIconVisible** variable to True. For general information, see Section 7.5.
4. In the **AutoVueURLPath** field, specify the AutoVue servlet to run, using the following format:

   `http://ucm_host name:port/web_root/idcplg?IdcService=VIEW_IN_AUTOVUE`

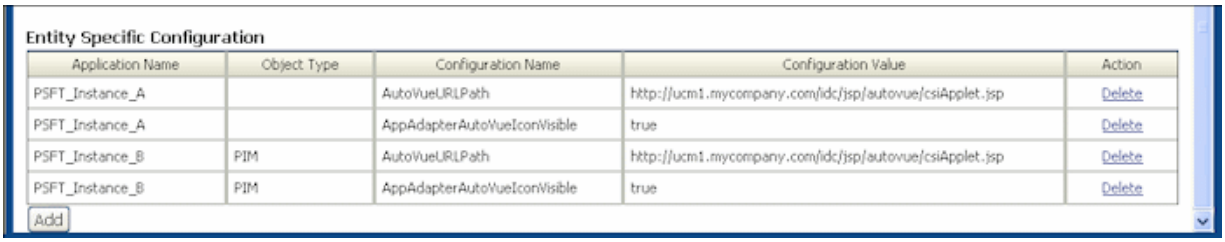
   For example:

   `http://ucm1.example.com/cs/idcplg?IdcService=VIEW_IN_AUTOVUE`

5. Optionally specify application or business entity settings by adding new rows in the AFEnvironmentConfiguration resultset for the AppAdapterAutoVueIconVisible and AutoVueURLPath variables, where:
   - dAFApplication: Application in which to display or hide button
   - dAFBusinessObjectType: Business entity within application in which to display or hide button

   Example 7–11 contains several AutoVue settings. It assumes that the **View in AutoVue** icon is set to false globally. The first two entries enable and display the icon for the PSFT_Instance_A application. The last two entries enable and display the icon for the PSFT_Instance_B application, but *only* within its PIM business entity.

Figure 7–11 Example AutoVue Document Viewing


Entity Specific Configuration

Application Name	Object Type	Configuration Name	Configuration Value	Action
PSFT_Instance_A		AutoVueURLPath	http://ucm1.mycompany.com/idc/jsp/autovue/csiApplet.jsp	Delete
PSFT_Instance_A		AppAdapterAutoVueIconVisible	true	Delete
PSFT_Instance_B	PIM	AutoVueURLPath	http://ucm1.mycompany.com/idc/jsp/autovue/csiApplet.jsp	Delete
PSFT_Instance_B	PIM	AppAdapterAutoVueIconVisible	true	Delete

Add

7.9 Configuring Oracle Distributed Document Capture

This section covers the following topics:

- Section 7.9.1, "About Document Scanning Using Oracle Distributed Document Capture"
- Section 7.9.2, "Oracle Distributed Document Capture and Related Components Requirements"
- Section 7.9.3, "Configuring Oracle Distributed Document Capture For the Adapter"
- Section 7.9.4, "Configuring Content Server for Distributed Document Capture Via Adapter"
- Section 7.9.5, "Testing the Oracle Distributed Document Capture Via Adapter Configuration"

> **Note:** For installation requirements specific to Oracle Distributed Document Capture use with this adapter, see Section 7.9.2.

7.9.1 About Document Scanning Using Oracle Distributed Document Capture

The Oracle Distributed Document Capture application enables an application such as the adapter to direct it to scan a document and pass in document index values. This allows users to scan documents or import scanned image files from the Managed Attachments page and attach them to the selected business application record.

When configured for the adapter, document scanning works as follows:

- A **Scan** button is added to the Managed Attachments page. The user clicks the button, and selects options such as a **document classification**, which is assigned to an Oracle Distributed Document Capture scan profile. For example, the user might

select a classification of *Identity Documents* to scan a photocopy of a driver's license or passport. (An Oracle Distributed Document Capture scan profile specifies scanning, importing, and indexing settings.) The user also specifies whether the document should be shared, and if so the user specifies a security group for the shared document from those to which the user has access.

- When the user clicks the **Scan Document** button, Oracle Distributed Document Capture's remote client launches in a new window, automatically authenticates and logs in the user (if configured), and passes in parameters such as the scan profile to use and business application entity values for later attachment.
- Within the Oracle Distributed Document Capture client, the user reviews the document, makes changes as needed, completes any index fields configured in the scan profile, then sends the batch. Sending the batch commits the new document to Content Server 11g using a commit profile specified for the scan profile. (A Capture/Content Server commit profile specifies connection information and field mappings between Capture and Content Server metadata fields.)
- Upon successful sending, the user returns to the Managed Attachments page and refreshes the display to view the newly scanned document or imported scanned image file.

7.9.2 Oracle Distributed Document Capture and Related Components Requirements

This section lists requirements for implementing scanning and importing functionality with Oracle Distributed Document Capture. For complete system requirements, such as browsers, see the configuration and certification information at the following location:

http://www.oracle.com/technetwork/middleware/webcenter/content/documentation/documentation-155348.html

Note: The requirements listed in this section apply only if using Oracle Distributed Document Capture for optional scanning functionality.

Requirements include:

- Oracle Distributed Document Capture Release 10.1.3.5.1 or later, including the Oracle UCM 11g Commit Driver, which is provided for installation in this release. The Oracle UCM 11g Commit Driver is required to commit documents from Oracle Distributed Document Capture to Oracle WebCenter Content Server 11g.
- Internet Explorer 6 or above for end-users.

Note: The Mozilla Firefox browser is not supported for scan functionality via Oracle Distributed Document Capture, which currently requires Internet Explorer.

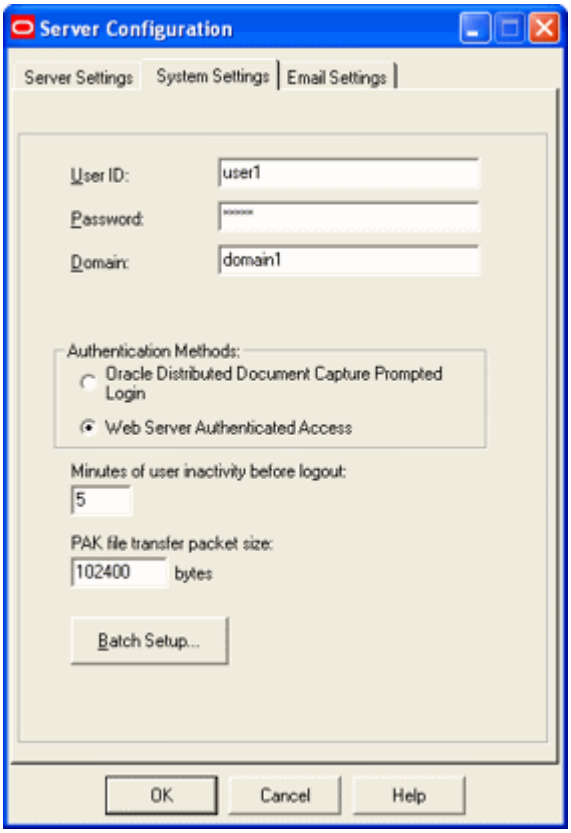
7.9.3 Configuring Oracle Distributed Document Capture For the Adapter

Follow these steps to configure attachment scanning on the Oracle Distributed Document Capture side.

1. Optionally set Oracle Distributed Document Capture for automatic login.

On the **System Settings** tab of the Distributed Document Capture Server Configuration application shown in Figure 7–12, select **Web Server Authenticated Access** from the Authentication Methods options. This setting enables the client to launch automatically without users needing to log in. For details, see the section on authentication in the *Installation Guide for Oracle Distributed Document Capture*.

Figure 7–12 Distributed Document Capture Server Configuration, System Settings Tab



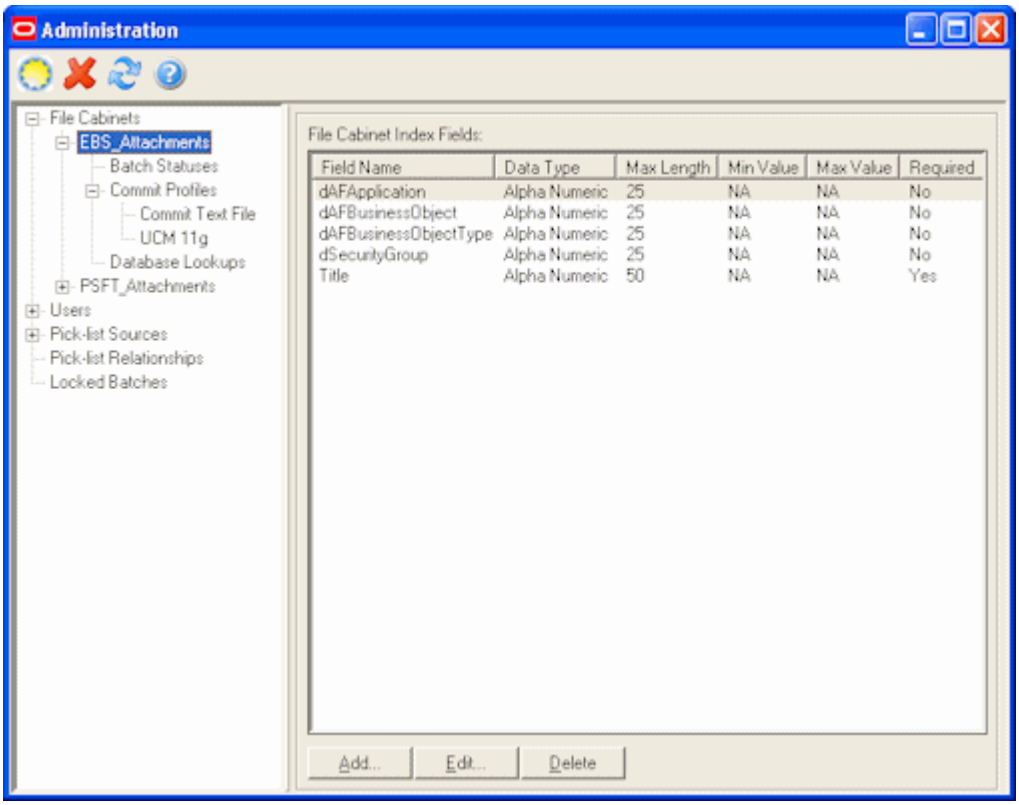
2. In Capture Administration, add index fields to a selected file cabinet for capturing values for new documents, as shown in Figure 7–13. For details, see the section on Capture Administration in the *Administrator's Guide for Oracle Distributed Document Capture*.

   a. Create index fields for values to save with attached documents on the Content Server. For example, you might configure a Doc Type pick-list index field for users to select from standard document types.

   b. Create the following alphanumeric index fields to contain the business application entity values and required fields:

      – dAFApplication
      – dAFBusinessObjectType
      – dAFBusinessObject
      – dSecurityGroup
      – Title

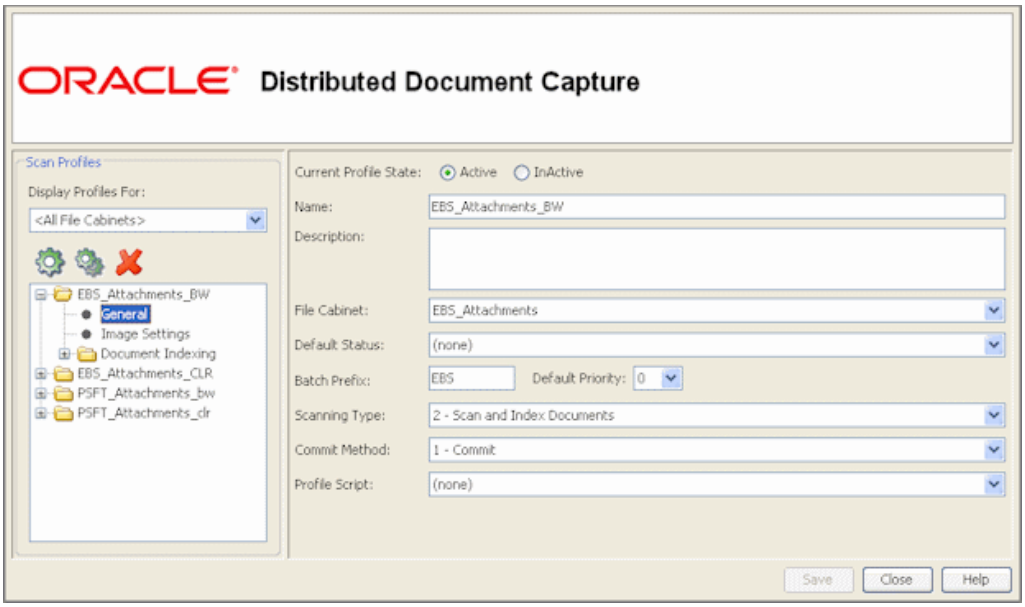
Figure 7–13 Capture Administration, Index Fields



3. In Oracle Distributed Document Capture's Profile Administration, create a scan profile, as shown in Figure 7–14. You later associate this scan profile with one or more document classifications, so that when a user selects a classification, the associated scan profile's settings are used. For details, see the section on scan profiles in the *Administrator's Guide for Oracle Distributed Document Capture*.

   a. On the General pane, select **2 - Scan and Index Documents** in the Scanning Type field. This scanning type includes indexing, and scans or imports pages into a single document in a batch. Specify a file cabinet and batch prefix.

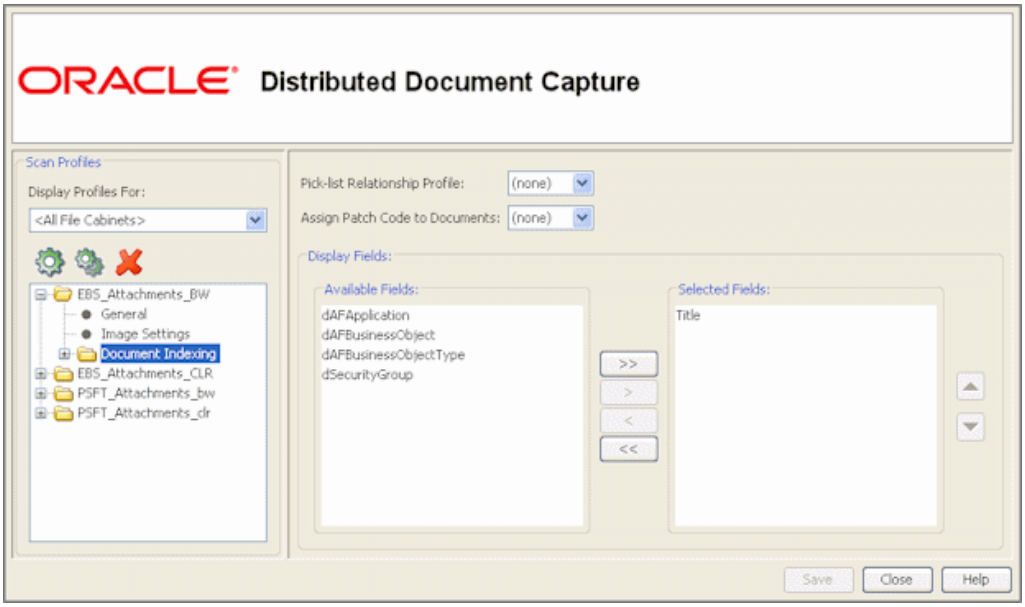
Figure 7–14 Scan Profiles, Oracle Distributed Document Capture Profile Administration



b. On the Document Indexing pane, move all fields to display to users to the Selected Fields box. It is recommended that you display the Title field for users to enter, and make the field required.

Note: Typically, you would not select the business application entity fields (dAFBusinessObject, for example) for display to users. If you choose to display them, lock them on the Field Properties pane to prevent users from changing their values.

Figure 7–15 Scan Profiles, Oracle Distributed Document Capture Profile Administration



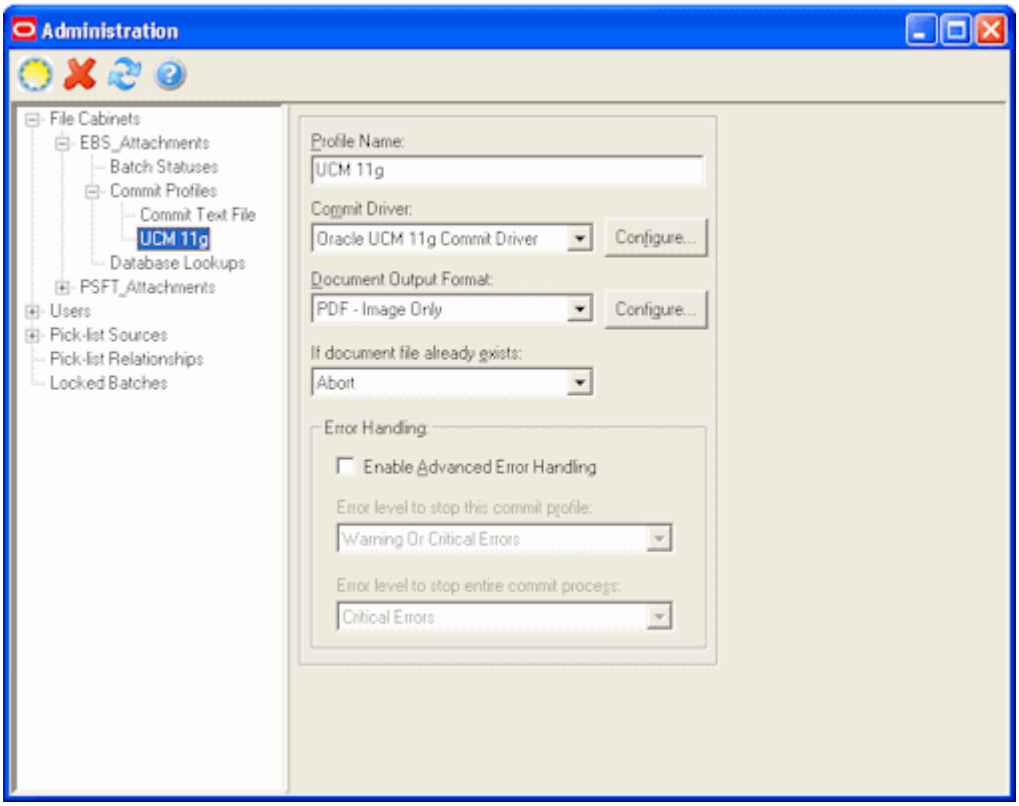
c. On the Field Properties, Auto Populate, and Database Lookup panes, configure any pick-lists, database lookups, or autopopulating needed for indexing. Save the scan profile.

4. In Capture Administration, create an Oracle UCM 11g (Content Server) commit profile to commit the scanned or imported documents to Content Server when users send a completed batch.

   This commit profile specifies how to connect to the Content Server and how the business application and Capture values are passed to the Content Server. For information about creating Content Server commit profiles, see the section on commit profiles in the *Administrator's Guide for Oracle Distributed Document Capture*.

   a. Select **Oracle UCM 11g Commit Driver** in the Commit Driver field. For this integration, it is recommended that you select **PDF - Image Only** in the Document Output Format field. Click the **Configure** button adjacent to the Commit Driver field.

Figure 7–16 Commit Profile Settings



   b. On the **Login** tab, specify settings for logging in to the Content Server instance. (The other tabs become active after you log in.) Use the following format for the Content Server URL:

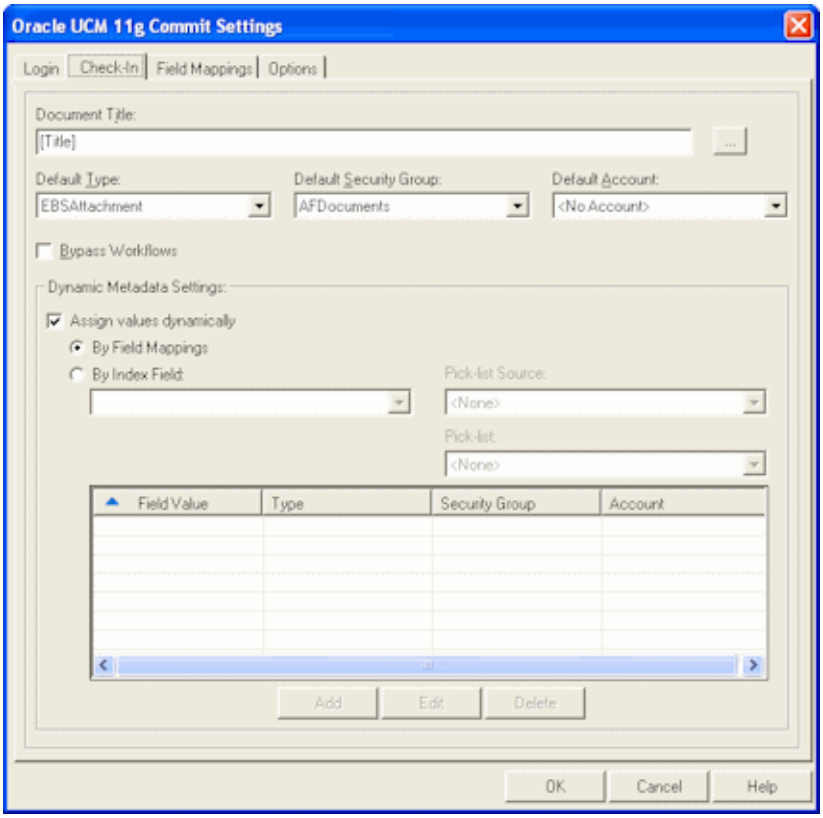
      `http://Content Server host name or IP address:WebServerPort/ucm web root/idcplg`

      The specified Content Server user must be assigned the AFWrite role for the AFDocuments security group. It is recommended that you create a special Capture user specific to the Managed Attachments solution for this purpose to ensure regular users do not gain write access to the AFDocuments security

group, which is reserved for the Managed Attachments solution. For details about the AFDocuments group, see Section 7.7.3.

c. On the **Check-In** tab, choose **Assign values dynamically** and **By Field Mappings** fields, as shown in Figure 7–17. Also specify how you want documents named in the **Document Title** field. It is recommended that you use the Title field.

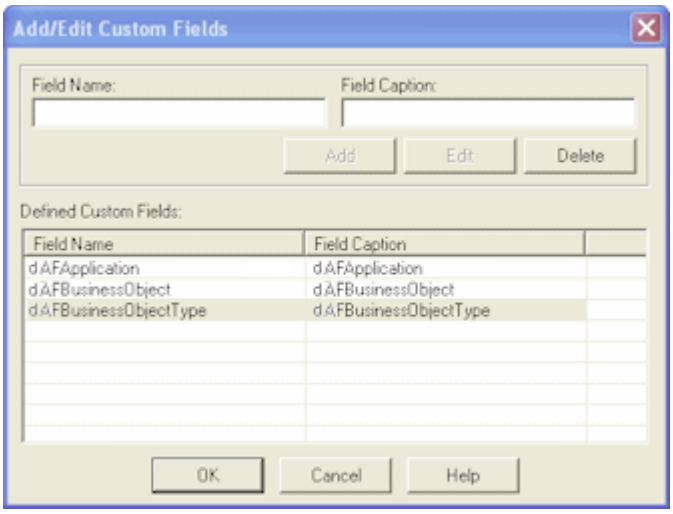
Figure 7–17 Capture Administration, Check-In for Content Server Commit Settings



d. On the **Field Mappings** tab, click the **Add/Edit Custom Fields** button, and add the following custom fields in the Add/Edit Custom Fields dialog box shown in Figure 7–18:

- dAFApplication
- dAFBusinessObjectType
- dAFBusinessObject

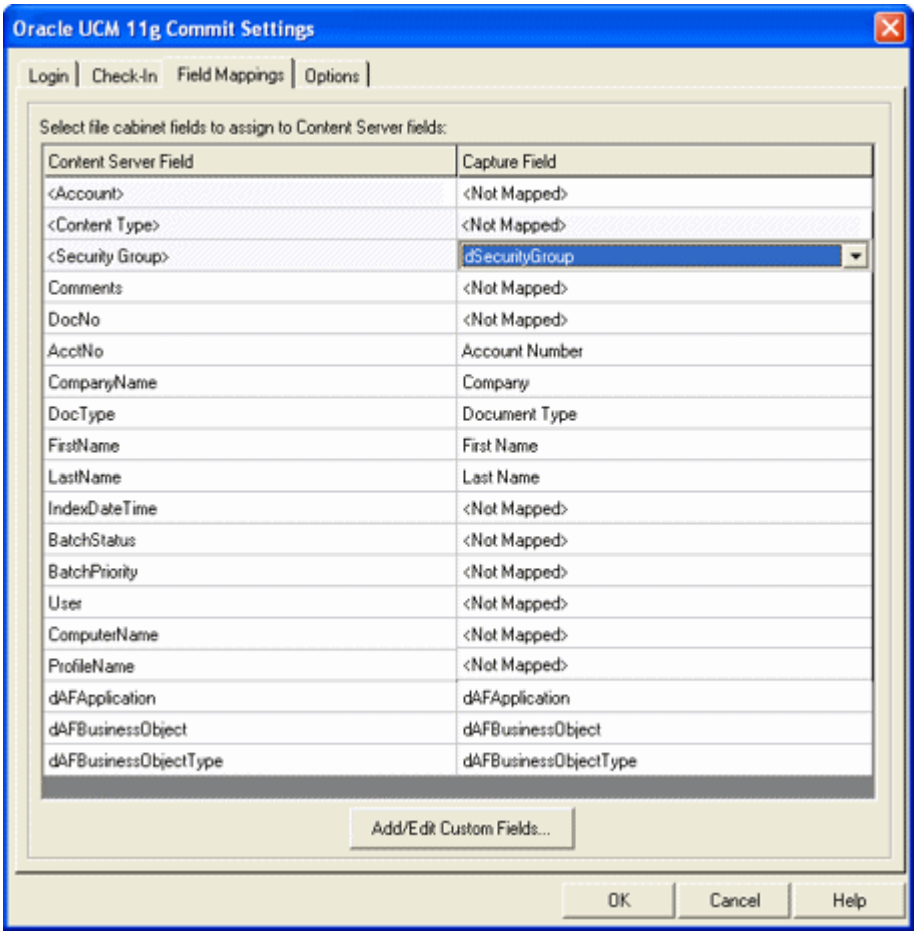
Figure 7–18 Capture Administration, Add/Edit Custom Fields



Click **OK**. The custom fields you added now display on the **Field Mappings** tab shown in Figure 7–19. Map these custom fields to the corresponding index fields you created in step 2, as they are required to attach the new Content Server document to the business application entity. Also map other Capture fields whose values you want written to Content Server fields. Click **OK**.

e. Activate the business application-Content Server commit profile by clicking **Commit Profiles** from the Administration tree and selecting the **Active** check box for the commit profile.

Figure 7–19 Capture Administration, Field Mappings for Content Server Commit Settings

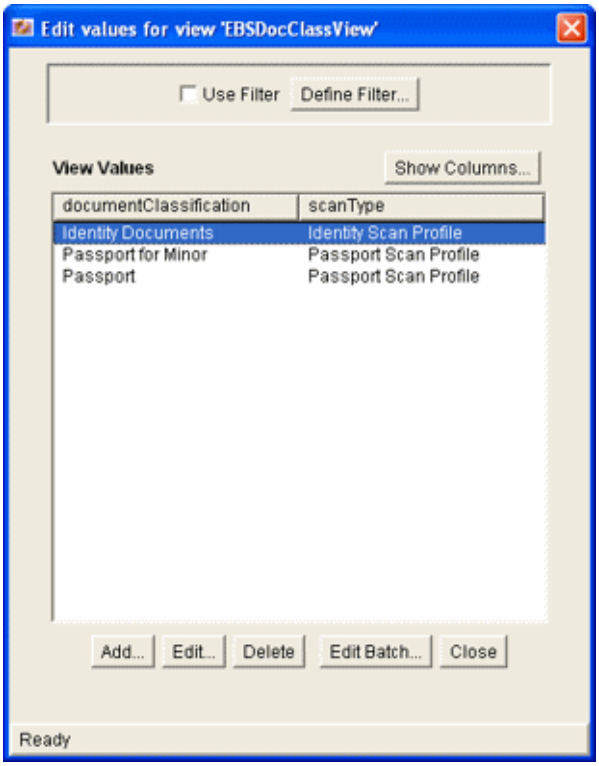


7.9.4 Configuring Content Server for Distributed Document Capture Via Adapter

Follow these steps to configure attachment scanning on the Content Server side.

1. On Content Server, edit the following configuration variables for Oracle Distributed Document Capture, if you have not done so. See Section 7.5.
   - `ODDCURLPath`
     `http://`*`ODDC_host/ODDC_webcapture_address`*
   - `ODDCScanAction`
     `ODDC Scan Action`
   - `AdapterAppScanBtnVisible`
     `true`
2. Restart Content Server.
3. On Content Server, configure document classification and scan types.
   a. From the Administration tray in Content Server, click the **Admin Applets** link.
   b. Select the Configuration Manager applet. The Configuration Manager displays.

c. Click the Views tab, select the appropriate document classification view from the alphabetical list (**EBSDocClassView** for Oracle E-Business Suite, **PSFTDocClassView** for Oracle PeopleSoft, or **UniversalDocClassView** for Enterprise Application Adapter Framework), and click **Edit Values**.



4. In the Edit Values dialog box, click **Add** and add document classifications and their corresponding scan profile. Note that documentClassification entries must be unique, but a scan profile can be used multiple times. Click **Close** when done.
5. From the Options menu, choose **Publish schema** to publish the data.

7.9.5 Testing the Oracle Distributed Document Capture Via Adapter Configuration

Follow these steps to test the configuration as an end-user. For more information, see the *Oracle WebCenter User's Guide for Application Adapters*.

1. Refresh the Managed Attachments results page. You should see a Scan button beside the New button.
2. Click the **Scan** button. In the Scan Document page, select a document classification and security group, and click **Scan Document**. The Oracle Distributed Document Capture client launches.

> **Note:** You can attach one document only via scanning or importing at a time.

Notice that the URL reflects the settings you specified to run the client. It also passes the scan profile, scan action (1 = Scan and 2 = Import), and index data containing the Oracle E-Business Suite or Oracle PeopleSoft entity values.

3. Within the Oracle Distributed Document Capture client, review, index, and send the document.

Note: If you decide to cancel a document scan or import, you must close the Oracle Distributed Document Capture window and return to the Managed Attachments page and perform a new scan or import. This ensures that index values are properly set for attachments.

4. Return to the Managed Attachments page and click **Refresh**. The newly scanned document or imported scanned image file should display in the list. (It may take a few minutes to display.)

7.10 Testing the Managed Attachments Solution

To test functionality, you can launch the Managed Attachments solution from the Command Driver page instead of a business application.

Follow these steps to use the command driver.

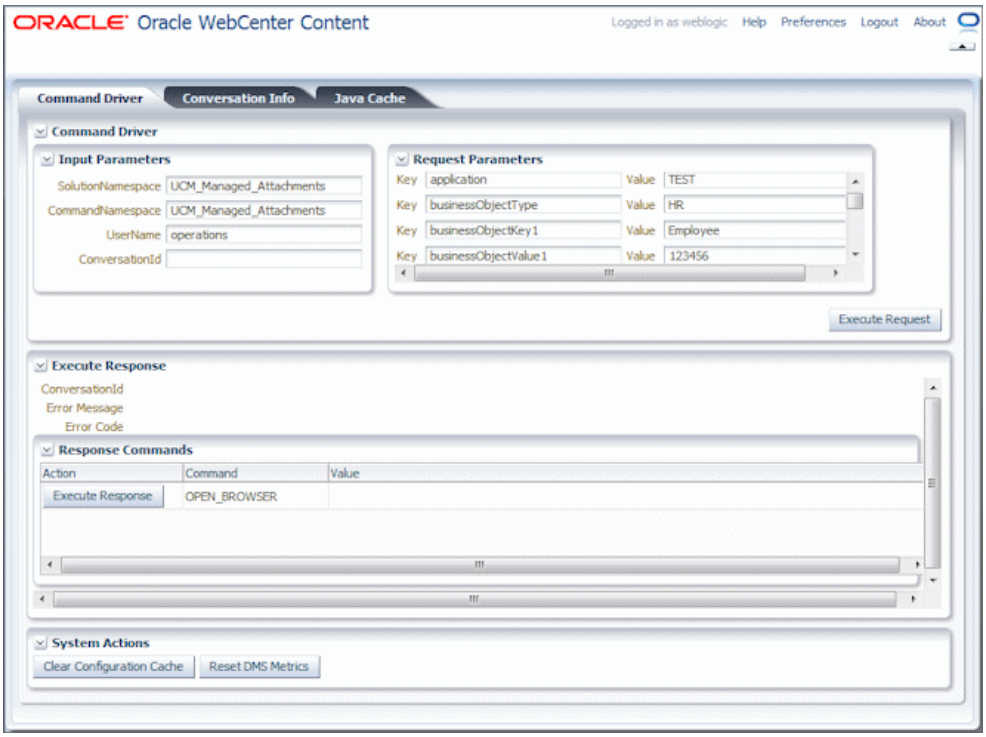
1. Display the command driver page using the following URL form:

   `http://IPM Host:IPM Port/imaging/faces/Driver.jspx`

   Note that you must log in to this page as the user set up in the CSF Credential for Managed Attachments (for example, weblogic), as described in Section 7.7.1.

2. In the command driver options that display, complete the **Input Parameters** and **Request Parameters** fields. In the **UserName** field, enter the name of a valid user already configured in Content (for example, `operations`). See examples in Figure 7–20. Leave the **ConversationId** field blank.

Figure 7–20 Example Parameters for the Managed Attachments Driver Page



3. Click the **Execute Request** button.

A conversation starts, as indicated by entries in the **conversationId** and other response fields.

4. Under **Response Commands**, click the **Execute Response** button.

   A new browser window opens and displays the Managed Attachments page.

7.11 Configuring Content Server Logging

For information about Content Server logging, see the *Oracle Fusion Middleware Administering Oracle WebCenter Content*.

7.12 Uninstalling Adapter Components on Content Server

Follow these steps to disable and then uninstall the following adapter components on the Content Server.

- AppAdapterCore
- AppAdapterEBS
- AppAdapterPSFT
- AppAdapterUniversal

For details about these components, see Section 7.3.1.

1. Log in to Content Server as an administrator.
2. Choose **Admin Server** from the Administration menu.

   The Content Admin Server page displays.
3. Select the name of the Content Server instance from which to uninstall the component.

   The Content Admin Server <*instance_name*> page displays.
4. Click **Component Manager**.

   The Component Manager page displays.
5. Select an adapter component in the Enabled Components list and click **Disable**.
6. Repeat step 5, selecting and disabling the remaining adapter components.
7. Click **Start/Stop Content Server**.

   The Content Admin Server <*instance_name*> displays.
8. Click **Restart**.
9. Click **Component Manager**.

   The Component Manager displays.
10. Select an adapter component in the Uninstall Component field and click **Uninstall**. Click **OK** to confirm uninstalling the component.
11. Repeat step 10, uninstalling the remaining adapter components.
12. Select the link to return to the Component Manager.

    The Component Manager page displays.
13. Click **Start/Stop Content Server**.

    The Content Admin Server <*instance_name*> page displays.

14. Restart Content Server to apply the changes.

A

BPM Imaging Solution Reference

This appendix defines the AXF for BPM business rule inputs, outputs, and rulesets. It also describes the AXF, Oracle E-Business Suite, and Oracle PeopleSoft configuration tables used for the BPM Imaging solution, with the HelloBPM solution shown as an example implementation.

This appendix covers the following topics:

- Section A.1, "AXF for BPM Business Rule Reference"
- Section A.2, "AXF Tables for the BPM Imaging Solution"
- Section A.3, "AXF Commands for the BPM Imaging Solution"
- Section A.4, "Oracle E-Business Suite Forms Tables for the BPM Imaging Solution"
- Section A.5, "Oracle PeopleSoft Tables for the BPM Imaging Solution"

A.1 AXF for BPM Business Rule Reference

This section provides reference information for the business rules provided with the AXF for BPM infrastructure.

- Section A.1.1, "Page Component Parameters"
- Section A.1.2, "Control Page Parameters"
- Section A.1.3, "Action Dictionary"
- Section A.1.4, "Data Dictionary"
- Section A.1.5, "DynamicTab Dictionary"
- Section A.1.6, "FunctionalBlock Dictionary"
- Section A.1.7, "LookupAndValidator Dictionary"
- Section A.1.8, "TableData Dictionary"
- Section A.1.9, "Business Rule Entity Relationships"

A.1.1 Page Component Parameters

Page components allow you to display contents, such as images, data fields, and/or tables on a dynamic tab, as described in Section 5.1.1.4.

To use a page component on a tab, specify the page component and its parameters in the DynamicTab Dictionary. As an example, the DynamicTab dictionary in the SalesQuoteEntry context uses the Tab ruleset to define page components for the

context's additional tabs and the TabParameter ruleset to define their parameters, as listed in Section 5.3.5.3. See Section A.1.5.2.

To add one of these components to the page, specify the page component in its pageComponent property in the DynamicTab business rule, Tab ruleset (the Image Data and Comments components are shown in the examples below):

pageComponent: `PageComponent."ImageData"`

pageComponent: `PageComponent."Comment"`

You specify the page component's parameters in the DynamicTab rule's TabParameter ruleset.

- Parameters for the Image Page Component
- Parameters for the Image Data Page Component
- Parameters for the Data Page Component
- Parameters for the Table Data Page Component
- Parameters for the Master Detail Page Component
- Parameters for the Identity Browser Page Component
- Parameters for the Comment Page Component

A.1.1.1 Parameters for the Image Page Component

This page component contains the Image Viewer. Table A–1 describes parameters you can set for the viewer in Imaging.

Table A–1 Image Page Component Attributes

Attribute	Description	Default Value for AXF
showTabs	Specifies if the tabs should be shown allowing the user to switch between documents in a folder.	false
showFooter	Specifies if the footer should be shown allowing the user to switch between documents in a folder.	false
forceHideProperties	Specify true to force the Properties detail panel of the viewer closed, overriding any system default and user preference settings.	true
forceHideStickyNotes	Specify true to force the StickyNotes detail panel of the viewer closed, overriding any system default and user preference settings.	true
forceHideHistory	Specify true to force the History detail panel of the viewer closed, overriding any system default and user preference settings.	true

A.1.1.2 Parameters for the Image Data Page Component

This page component is a composite page component comprised of a data page component and an image page component.

Table A–2 describes the image data page component's attributes. In addition, the Image Viewer attributes listed in Table A–1 are also configurable for this page component.

Table A–2 Image Data Page Component Attributes

Attribute	Description
DataContext	Specifies the context which determines what data is displayed in the table. This context is used to execute the appropriate rule containing the definition for this data. By default, this attribute uses the default context configured in the task flow template.
SplitterPosition	Define where to place the splitter bar between the data component and the image component. The default is 250.
HideData	Specifies whether the Data component is shown or hidden. A value of true hides the Data component and displays the Image component expanded to fill the entire tab region. A value of false shows the Data component to the left of the Image component with a splitter bar separating the Data and Image components.

A.1.1.3 Parameters for the Data Page Component

This page component is a reusable user interface component that comprises a list of input/output components grouped in different sections. Each component displayed on the control is bound to a data field defined in the BPM task data control.

This page component enables you to:

- Specify which data fields display in the task detail area
- Specify display attributes for each data field (such as label, read-only, order, and section/group name)
- Edit the value of the payload fields
- Customize how data fields are grouped

Table A–3 lists attributes for the data page component. These attributes are not required.

Table A–3 Data Page Component Attributes

Attribute	Description
Context	Specifies the display context of the control.
DisplayMode	Specifies the mode in which to display the data control, where: 0=panelAccordion 1=panelForm
NumberOfColumns	Specifies the number of columns to use to display the data. This attribute applies to the panelForm display mode only.

A.1.1.4 Parameters for the Table Data Page Component

This page component displays table data on a dynamic tab.

Table A–4 lists optional attributes for the table data page component.

Table A–4 Table Data Page Component Attributes

Attribute	Description
Context	Specifies the context which determines which data displays in the table. This context is used to execute the appropriate rule containing the definition for this table. By default, this attribute uses the default context of human task definition configured in the task flow template.

A.1.1.5 Parameters for the Master Detail Page Component

The Master Detail Page Component is used to display both a master and a detail area of data. Both a Data Component and a Data Table Component are used for this page component.

Table A–5 lists the attributes defined in the Master Detail Page Component.

Table A–5 Master Detail Page Component Attributes

Attribute	Description
DataContext	Specifies the context that determines the data displayed on the header (master) portion of the page. This context is passed to the data component on this page. By default, this attribute uses the default context configured in the task flow template.
TableDataContext	Specifies the context which determines what data is displayed on the table portion of this page. This context is passed to the table data component on this page. By default, this attribute uses the default context configured in the task flow template.
PageComponentTitle	Specifies the title displayed at the top of the page when this component is rendered.
NumberOfColumns	Specifies the number of columns to display for this table.

A.1.1.6 Parameters for the Identity Browser Page Component

This page component hosts the built-in ADF identity browser and allows users to select user and groups for assignment.

Table A–6 lists optional attributes for the identity browser page component.

Table A–6 Identity Browser Page Component Attributes

Attribute	Description

SearchFilter	Allows you to control the filtering of identities in the Identity Browser.
	To define which filters display for use, specify the SearchFilter parameter as a NamedFunctionalBlockParameter. The value for this key is a comma delimited string of 1..3 of the filters. The filter values are "User", "Group", and "ApplicationRole".
	▪ If a valid string is provided with 1..N filter values, the filters are available in the dropdown.
	Example 1, where only the Users filter is available:
	`"User"`
	Example 2, where the Groups and ApplicationRole are available:
	`"Group, ApplicationRole"`
	▪ If the SearchFilter parameter is not supplied or has an empty value, all three options are available for the filter (User, Groups, and ApplicationRole).

UserBindingName	Specifies the binding to use to store the selected user.

Table A–6 (Cont.) Identity Browser Page Component Attributes

Attribute	Description
GroupBindingName	Specifies the binding to use to store the selected group.
ApplicationRoleBindingName	Specifies the binding to use to store the selected application role.
AssignmentBindingName	Specifies the binding name for the assignment node. Rather than explicitly specifying which payload fields store each of the three possible identities, a single node name can be specified; the Identity Browser assumes a fixed User, Group, and ApplicationRole available under that node and assigns the values there.

A.1.1.6.1 Configuring the Identity Browser in Explicit or Relative Mode You can configure the Identity Browser component to use an assignment mode of explicit or relative.

- With **explicit mode**, you specify which payload fields store each of the three possible identities. In the NamedFunctionalBlockParameter properties, the UserBindingName, GroupBindingName, ApplicationRoleBindingName properties specify the payload field to which to save the values selected for users, groups, and application roles, respectively. If no values are specified, the selected user, group, or application role value or values are not saved to the payload.
- With **relative mode**, you specify a single node name instead of payload fields. The Identity Browser looks for fixed user, group, and application role values within the specified node and assigns the values there. In the NamedFunctionalBlockParameter properties, the assignment property specifies the node name in which the Identity Browser looks for fixed user, group, and application role values.

A.1.1.7 Parameters for the Comment Page Component

This page component allows users to add or view comments for the selected task. It has no parameters.

A.1.2 Control Page Parameters

Control pages allow you to quickly integrate the following predefined page components in a solution, as described in Section 5.1.4.

- Data Control Page Parameters
- Identity Browser Control Page Parameters
- Comment Control Page Parameters

A.1.2.1 Data Control Page Parameters

To use the Data control page, create a Named Functional Block to navigate to the Data control page. In the NamedFunctionalBlockParameter Properties, specify the parameters listed in Table A–7. For an example, see Table 5–23.

Table A–7 Data Control Page Parameters

Parameter Key	Description	Default Value
ControlPageName	Specifies the control page to load (required). This value maps to the structureContext in the page component.	`"DataControlPage"`
ControlPageTitle	Sets the title on the control page (optional). If not specified, no title displays.	<blank>
Context	Specifies the content context, which defines the business rule that executes and the data that displays on the page.	Uses the default context defined.
CommentRequired	Specifies if the user must enter a new comment to complete the OK action on the page ("True" or "False").	"False"

A.1.2.2 Identity Browser Control Page Parameters

To use the identity browser control page, create a Named Functional Block to navigate to the identity browser control page. In the NamedFunctionalBlockParameter Properties, specify the parameters listed in Table A–8. For an example, see Table 5–23.

Table A–8 Identity Browser Control Page Parameters

Parameter Key	Description	Default Value
ControlPageName	Specifies the control page to load (required).	`"IdentityBrowserControlPage"`
CommentRequired	Specifies if the user must enter a new comment to complete the OK action on the page ("True" or "False").	"False"
ControlPageTitle	Sets the title on the control page (optional). If not specified, no title displays.	<blank>
UserBindingName	Specifies the payload field to save the values that were selected for users. If no value is specified, the selected user value(s) are not saved to the payload.	None
GroupBindingName	Specifies the payload field to save the values that were selected for groups. If no value is specified, the selected group value(s) are not saved to the payload.	None
ApplicationRoleBindingName	Specifies the binding to use to store the selected application role.	None
AssignmentBindingName	Specifies the binding name for the assignment node. Rather than explicitly specifying which payload fields store each of the three possible identities, a single node name can be specified; the Identity Browser assumes a fixed User, Group, and ApplicationRole available under that node and assigns the values there.	None

A.1.2.3 Comment Control Page Parameters

To use the comment control page, create a Named Functional Block to navigate to the comment control page. In the NamedFunctionalBlockParameter Properties, specify the parameters listed in Table A–9. For an example, see Table 5–23.

Table A–9 Comment Control Page Parameters

Parameter Key	Description	Default Value
ControlPageName	Specifies the control page to load (required).	`"CommentControlPage"`
CommentRequired	Specifies if the user must enter a new comment to complete the OK action on the page ("True" or "False").	"False"

A.1.3 Action Dictionary

The Action dictionary contains the following rulesets:

- Action
- Section
- ActionEnableModeException
- SectionEnableModeException
- ActionParameter
- SectionParameter

A.1.3.1 Action

This ruleset defines action links or commands for the action control.

Table A–10 Action Properties

Property	Description
actionId	The unique id of this action.
displayName	Name to display for this action.
displayOrder	Order to display this action in the section.
enableMode	Enable mode for this action, where: ■ HIDE: Do not show this action ■ DISABLE: Show this action disabled ■ ENABLE: Show this action enabled
namedFunctionalBlockId	The Id of the Named Functional Block or Named Functional Block Sequence triggered by this action.
sectionId	The section in which to display this action (foreign key to Section Properties).

A.1.3.2 Section

This ruleset defines the sections in which action links display on the action control.

Table A–11 Section Properties

Property	Description
disclosed	Specifies whether the section initially displays open (true) or closed (false).
displayName	Name to display for this section.
displayOrder	Order in which to display the section.
enableMode	Enable mode for this section, where: ■ HIDE: Do not show this action ■ DISABLE: Show this action disabled ■ ENABLE: Show this action enabled
sectionId	The unique id of this section (referenced by an action).

A.1.3.3 ActionEnableModeException

This ruleset is reserved for future use.

A.1.3.4 SectionEnableModeException

This ruleset is reserved for future use.

A.1.3.5 ActionParameter

This ruleset defines parameters associated with an action, which you can use to extend an action without changing the interface.

Table A–12 ActionParameter Properties

Property	Description
actionId	The unique id of this action.
parameterKey	The key of the parameter.
parameterValue	The value of the parameter.

A.1.3.6 SectionParameter

This ruleset defines parameters associated with a section, which you can use to extend a section without changing the interface.

Table A–13 SectionParameter Properties

Property	Description
sectionId	The unique id of this section.
parameterKey	The key of the parameter
parameterValue	The value of the parameter.

A.1.4 Data Dictionary

The Data dictionary contains the following rulesets:

- Data
- Section
- DataParameter
- SectionParameter
- DataComponentAttribute
- DataValidator

A.1.4.1 Data

This ruleset specifies individual data items on the data component.

Table A–14 Data Properties

Property	Description
bindingName	Binding for this item (where the value is read/written). Required.
displayName	Name to display as the attribute's label. Required.
displayOrder	Order in which to display this item.
lookupId	Lookup to use for this field, as defined in the LookupAndValidator Dictionary. If blank, no lookup is available.
dataId	ID used to link this item with the DataParameter Properties. Required.

Table A–14 (Cont.) Data Properties

Property	Description
enableMode	Specifies the display mode of the item. Values are: ■ EnableMode.ENABLE (default) ■ EnableMode.DISABLE ■ EnableMode.HIDE
sectionId	Id of the section into which to place this item (defined by Section Properties). Each data item must be placed within a section.
componentClass	Specifies the user interface control to render for this data item. This value must identify the user interface control, generally ADF input and output components such as "ComponentClass.RichInputText". If no component class is specified, this value defaults to rich input text, unless the underlying data is a date in which case it defaults to rich input date.

A.1.4.2 Section

This ruleset specifies sections in which to display data items.

Table A–15 Section Properties

Property	Description
disclosed	Determines if section is initially displayed disclosed (Boolean)
displayName	Name of the section to display on the user interface.
displayOrder	Display order of the section in the component.
enableMode	Specifies the display mode of the item. Values are: ■ EnableMode.ENABLE (default) ■ EnableMode.DISABLE ■ EnableMode.HIDE
sectionId	Unique id of the section referenced in Data Properties.

A.1.4.3 DataParameter

This ruleset specifies parameters for individual data items on the data component. This ruleset is for future use.

Table A–16 DataParameter Properties

Property	Description
dataId	Foreign key to the Data Properties.
parameterKey	Key for this parameter.
parameterValue	Value for this parameter.

A.1.4.4 SectionParameter

This ruleset specifies parameters for data sections. This ruleset is for future use.

Table A–17 SectionParameter Properties

Property	Description
sectionId	Foreign key to the Section Properties.

Table A–17 (Cont.) SectionParameter Properties

Property	Description
parameterKey	Key for this parameter.
parameterValue	Value for this parameter.

A.1.4.5 DataComponentAttribute

This ruleset specifies user interface attributes to the user interface component specified by the data component. The example values shown set the background color of the user interface component to light blue. These attributes are standard Oracle ADF user interface attributes.

Table A–18 DataComponentAttribute Properties

Property	Description
dataId	Foreign key to the Data Properties.
parameterKey	Key for this parameter. For example, you might include: ■ "contentStyle" (example value: "background-color:lightblue") ■ ComponentAttribute."converterdatetime.pattern" (example value: "MMM dd, yyyy hh:mm:ss a ")
parameterValue	Value for this parameter.

A.1.4.6 DataValidator

This ruleset defines how the data item uses the referenced validator.

Table A–19 DataValidator Properties

Property	Description
validatorCategoryId	Use this property to categorize a group of validations related to a functional block and its mode property. For example, setting this property to "AddressInfo," then also setting the validatorCategoryId property in NamedFunctionalBlockValidator Properties to "AddressInfo" and the mode property to "PROMPT" means that any field set for validation with the specified validator in the Data dictionary will share its "PROMPT" mode.
dataId	Foreign key to the Data Properties.
validatorMode	Specifies the validator's mode, where: ■ IMMEDIATE means the field is validated as soon as possible (when the field loses focus). ■ FUNCTIONAL_BLOCK means the validation is performed when a functional block is executed.
validatorId	Validator to run on the selected data field, as defined in the Validator in the LookupAndValidator Dictionary. If blank, no validator will be applied.

A.1.5 DynamicTab Dictionary

The DynamicTab dictionary contains the following rulesets:

- Tab
- TabParameter

A.1.5.1 Tab

This ruleset defines tabs to display for the specified context, such as Image and Comments.

Table A–20 Tab Properties

Property	Description
tabId	The unique name of the tab.
displayName	Title of the tab.
displayOrder	Order of the tab.
pageComponent	Location of the page component definition file.

A.1.5.2 TabParameter

This ruleset defines parameters for tabs (page components) to be displayed. For example, it sets the data and tabledata context and the page component title.

Table A–21 TabParameter Properties

Property	Description
tabId	The unique name of the tab.
parameterKey	Parameter defined in the page component. For example, specify a parameter of SplitterPosition and value of 250 to set the default sizing for the screen, or specify the data to display by specifying context, which defines the business rule executed. For a description of parameters for each page component, see Section A.1.1.
parameterValue	Value of the parameter defined in the page component mapped to the tab. Any value that does not begin with ${ or #{ is treated as a literal. **Note:** Expression Language can be used to specify the parameter value (for example, "#{pageFlowScope.taskId}"

A.1.6 FunctionalBlock Dictionary

The FunctionalBlock dictionary contains the following rulesets:

- NamedFunctionalBlock
- NamedFunctionalBlockSequence
- NamedFunctionalBlockParameter
- NamedFunctionalBlockPrompt
- NamedFunctionalBlockValidator

A.1.6.1 NamedFunctionalBlock

This ruleset defines the functional blocks of the context.

Table A–22 NamedFunctionalBlock Properties

Property	Description
functionalBlock	Specifies what the functional block does (for example, "setOutcome," "navControlPage," or "system::SUSPEND")

Table A–22 (Cont.) NamedFunctionalBlock Properties

Property	Description
namedFunctionalBlockId	Specifies a unique name for the Named Functional Block. **Note:** This name must be unique across all instances of both NamedFunctionalBlock namedFunctionalBlockId and NamedFunctionalBlockSequence namedFunctionalBlockSequenceId.

A.1.6.2 NamedFunctionalBlockSequence

This ruleset defines the functional block sequences.

Table A–23 NamedFunctionalBlockSequence Properties

Property	Description
namedFunctionalBlockId	Specifies the name of the functional block included in the sequence, as defined in NamedFunctionalBlock Properties.
namedFunctionalBlockSequenceId	Specifies the unique Id of the NamedFunctionalBlockSequence. **Note:** This name must be unique across all instances of both NamedFunctionalBlock namedFunctionalBlockId and NamedFunctionalBlockSequence namedFunctionalBlockSequenceId.
order	The execution order of the NamedFunctionalBlock in the sequence.

A.1.6.3 NamedFunctionalBlockParameter

This ruleset defines parameters for the functional blocks.

Table A–24 NamedFunctionalBlockParameter Properties

Property	Description
namedFunctionalBlockId	The unique id of the NamedFunctionalBlock. For example values for this table, see Table 5–23.
parameterKey	The key of the parameter (for example, Outcome).
parameterValue	The value of the parameter (for example, APPROVE).

A.1.6.4 NamedFunctionalBlockPrompt

This ruleset defines prompts for the functional blocks.

Table A–25 NamedFunctionalBlockPrompt Properties

Property	Description
namedFunctionalBlockId	The unique id of the NamedFunctionalBlock. For example values for this table, see Table 5–24.
order	The display order of the prompts.
promptMessage	The message shown in the pop-up window.
promptMode	When the prompt occurs. Possible values are: ■ PromptModeType.BEFORE_VALIDATION ■ PromptModeType.AFTER_VALIDATION

Table A–25 (Cont.) NamedFunctionalBlockPrompt Properties

Property	Description
promptType	The type of the prompt. Possible values are OkCancelFunctionalBlockContinuation and RollbackSavedChanges. **OkCancelFunctionalBlockContinuation** prompts the user: ■ Yes: continues the Named Functional Block ■ No: cancels the Named Functional Block **RollbackSavedChanges** prompts the user: ■ Yes: continue the Named Functional Block after reverting task payload changes to the state when the task was opened. Note that comments added since the task has been opened will not be reverted. ■ No: continues the Named Functional Block ■ Cancel: cancels the Named Functional Block

A.1.6.5 NamedFunctionalBlockValidator

This ruleset defines settings for functional block validations, such as the associated functional block and what should occur upon error (continue, abort, or prompt).

Table A–26 NamedFunctionalBlockValidator Properties

Property	Description
namedFunctionalBlockId	The unique id of the NamedFunctionalBlock to associate with a validator.
validatorCategoryId	Use this property to categorize a group of validations related to a functional block and its mode property. For example, setting this property to "AddressInfo" and the mode property to "PROMPT," then also setting the validatorCategoryId property in DataValidator Properties and/or ColumnValidator Properties to "AddressInfo" means that any field set for validation with the specified validator in the Data and/or TableData dictionaries will share the "PROMPT" mode.
validatorMode	This property determines what occurs if the validation fails. Choices include: ■ CONTINUE ■ ABORT (default) ■ PROMPT

A.1.7 LookupAndValidator Dictionary

The LookupAndValidator dictionary contains the following rulesets:

- StaticLookup
- DynamicLookup
- DynamicLookupBindVariable
- LookupParameter
- DynamicLookupRelatedBinding
- Validator
- ValidatorParameter

A.1.7.1 StaticLookup

This ruleset defines a list of static lookups and their values and display order. You can use these fixed list lookups in Data or DataTable dictionaries to implement the lookup.

For example values for this table, see Table 5–26.

Table A–27 StaticLookup Properties

Property	Description
lookupId	Unique ID that links this item with the other lookup definition rulesets and Data Dictionary and TableData Dictionary.
displayName	Name to display in the selection list.
value	When the user selects a display name from the static dropdown, place this value into the field with which this lookup is associated.
displayOrder	Order to display this item in the list.

A.1.7.2 DynamicLookup

This ruleset defines dynamic lookups, their database connection and binding, type (dropdown or search), and SQL query to perform. You can use these dynamic lookups in Data or DataTable dictionaries to implement the lookup.

For example values for this table, see Table 5–27.

Table A–28 DynamicLookup Properties

Property	Description
lookupId	Unique ID that links this item with the other lookup definition rulesets and Data Dictionary and TableData Dictionary.
dataSource	Specify the datasource to query, such as /jdbc/IPMDS. The IPMDS portion is automatically configured during installation. For more information, see Section 5.4.10.
sqlQuery	Specify the SQL query string to execute against the database connection specified in lookupConfiguration. An alias specified in the select clause cannot contain spaces.
sqlColumn	Specifies a column from the select clause that is used to bind to any Data or TableData column that references this lookup. Note: This value must either be a column from the select clause or an alias if an alias is defined.

A.1.7.3 DynamicLookupBindVariable

This ruleset defines the bind variables to use in the specified SQL query. Typically, you use Expression Language to access a value in the bindings/payload.

Table A–29 DynamicLookupBindVariable Properties

Property	Description
lookupId	Foreign key to the DynamicLookup Properties.
variableName	Binding variable name located in the SQL query.
variableValue	Specifies a binding to retrieve a value in the payload.

A.1.7.4 LookupParameter

This ruleset specifies parameters the lookup definitions. This ruleset is provided for future use.

Table A–30 LookupParameter Properties

Property	Description
lookupId	Foreign key to the lookup rulesets.
parameterKey	Key for this parameter.
parameterValue	Value for this parameter.

A.1.7.5 DynamicLookupRelatedBinding

This ruleset defines related bindings to be updated when a user selects a value from the lookup.

Use this ruleset to bind additional data fields to a dynamic lookup. For example, the HelloBPM solution's AccountName lookup populates address fields from the AccountName lookup.

For example values for this table, see Table 5–28.

Table A–31 DynamicLookupRelatedBinding Properties

Property	Description
lookupId	Foreign key to the DynamicLookup Properties.
bindingName	Name of the binding/payload-field to be updated.
sqlColumn	Specifies a column from the Dynamic Lookup select clause that is used to update the data item specified in the bindingName. Note: This value must either be a column from the select clause or an alias if an alias is defined.

A.1.7.6 Validator

This ruleset specifies a validator to run, its validation type, and whether it should run immediately or as part of a functional block.

Table A–32 Validator Properties

Property	Description
validatorId	Specifies the validator's unique id, and acts as a primary key to ValidatorParameter Properties, where the validation's parameter are set.
validatorClass	Specifies the validator's class. See Table A–33 for a list and description of standard classes. For more information about these validators, including their required parameters, see the section on ADF Faces Validators in the Oracle ADF documentation.

Table A–33 Standard Validator Classes Properties

Validator Class	Description
ByteLengthValidator	Validates the byte length of strings when encoded. The `maximumLength` attribute of inputText is similar, but it limits the number of characters that the user can enter.

Table A–33 (Cont.) Standard Validator Classes Properties

Validator Class	Description
DateRestrictionValidator	Validates that the entered date is valid with some given restrictions.
DateTimeRangeValidator	Validates that the entered date is within a given range, which you specify as validator attributes.
DoubleRangeValidator	Validates that a component value is within a specified range. The value must be convertible to a floating-point type.
LengthValidator	Validates that the length of a component value is within a specified range. The value must be of type `java.lang.String`.
LongRangeValidator	Validates that a component value is within a specified range. The value must be any numeric type or string that can be converted to a long data type.
RegExpValidator	Validates the data using Java regular expression syntax.

A.1.7.7 ValidatorParameter

This ruleset defines parameters for validations to be run. Typically these parameters provide further information about the specified validator.

Table A–34 ValidatorParameter Properties

Property	Description
validatorId	Specifies the validator's unique id, defined in Validator Properties.
parameterKey	Key for this parameter, as required by the validatorClass specified in Validator Properties.
parameterValue	Value for this parameter.

A.1.8 TableData Dictionary

The TableData dictionary contains the following rulesets:

- Table
- Column
- TableParameter
- ColumnParameter
- ColumnComponentAttribute
- ColumnValidator

A.1.8.1 Table

This ruleset defines the table, its binding, its enabled buttons (copy, paste, delete, and insert), and stretch mode.

Table A–35 Table Properties

Property	Description
tableId	Primary key for this table, links to TableParameter Properties.
bindingName	Name of the binding (xml node) in which to load the table's rows.

Table A–35 (Cont.) Table Properties

Property	Description
tableStretchMode	Specify the stretching to apply to the table's columns. ■ **none** (default): Specify for optimal performance. ■ **last:** Specify to stretch the last column to fill up any unused space. ■ **blank:** Specify to automatically insert an empty blank column and that stretches so the row background colors span the entire width of the table. ■ **multiple:** Specify to stretch more than one column. To configure a specific leaf (non-group) column stretch to fill up any unused space, specify "column:" followed by the ID of the column to stretch, as in the following: `"column:myColId"`
enableInsert	If true, enables the Insert button.
enableDelete	If true, enables the Delete button.
enableCopyPaste	If true, enables the Copy and Paste buttons.

A.1.8.2 Column

This ruleset defines the table's columns, including their binding, display order, column width, and lookup (if any).

Table A–36 Column Properties

Property	Description
columnId	Primary Key for this column, links to ColumnParameter Properties.
displayName	The name display for the column header.
displayOrder	Order of display for this column.
columnWidth	Specifies the column's width (for example, "300").
bindingName	Name of the binding (xml node) in which to load the column's data.
enableMode	Determines the state in which the control is rendered: Possible values are: ■ EnableMode.ENABLE (default) ■ EnableMode.DISABLE ■ EnableMode.HIDE
componentClass	Specifies the user interface control to render for this data item. This must identify the user interface control, generally ADF input and output components such as "ComponentClass.RichInputText".
lookupId	Lookup to run on the selected table column, as defined in the LookupAndValidator Dictionary.

A.1.8.3 TableParameter

This ruleset defines parameters associated with a table.

Table A–37 TableParameter Properties

Property	Description
tableId	Identifies the primary key of the table with which this parameter is associated, defined in Table Properties.
parameterKey	Key for this parameter.
parameterValue	Value for this parameter.

A.1.8.4 ColumnParameter

This ruleset defines parameters associated with a table's columns.

Table A–38 ColumnParameter Properties

Property	Description
columnId	Identifies the primary key of the column with which this parameter is associated, defined in Column Properties.
parameterKey	Key for this parameter.
parameterValue	Value for this parameter.

A.1.8.5 ColumnComponentAttribute

This ruleset defines attributes associated with a table's columns.

Table A–39 ColumnComponentAttribute Properties

Property	Description
columnId	Identifies the primary key of the column with which this parameter is associated, defined in Column Properties.
attributeName	Key for this parameter.
attributeValue	Value for this parameter.

A.1.8.6 ColumnValidator

This ruleset defines how the column uses the referenced validator.

Table A–40 ColumnValidator Properties

Property	Description
validatorCategoryId	Use this property to categorize a group of validations related to a functional block and its mode property. For example, setting this property to "AddressInfo," then also setting the validatorCategoryId property in NamedFunctionalBlockValidator Properties to "AddressInfo" and the mode property to "PROMPT" means that any field set for validation with the specified validator in the Data dictionary will share its PROMPT mode.
columnId	Identifies the primary key of the column with which this parameter is associated, defined in Column Properties.
validatorMode	Specifies the validator's mode, where: IMMEDIATE: the field is validated as soon as possible (when the field loses focus). FUNCTIONAL_BLOCK: the validation is performed when a functional block is executed.

Table A–40 (Cont.) ColumnValidator Properties

Property	Description
validatorId	Validator to run on the selected table column, as defined in the Validator rule in the LookupAndValidator Dictionary.

A.1.9 Business Rule Entity Relationships

The following diagrams display the relationships between the business rule dictionaries:

- Action Entity Relationships
- Functional Block Entity Relationships
- Lookup and Validator Entity Relationships
- Data Entity Relationships
- Table Data Entity Relationships
- Dynamic Tab Entity Relationships

Figure A–1 Action Entity Relationships

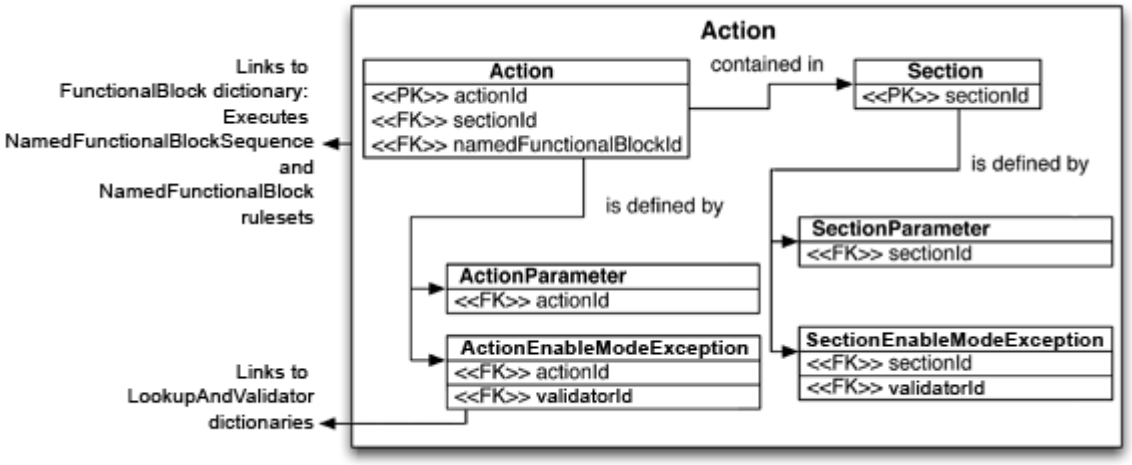


Figure A–2 Functional Block Entity Relationships

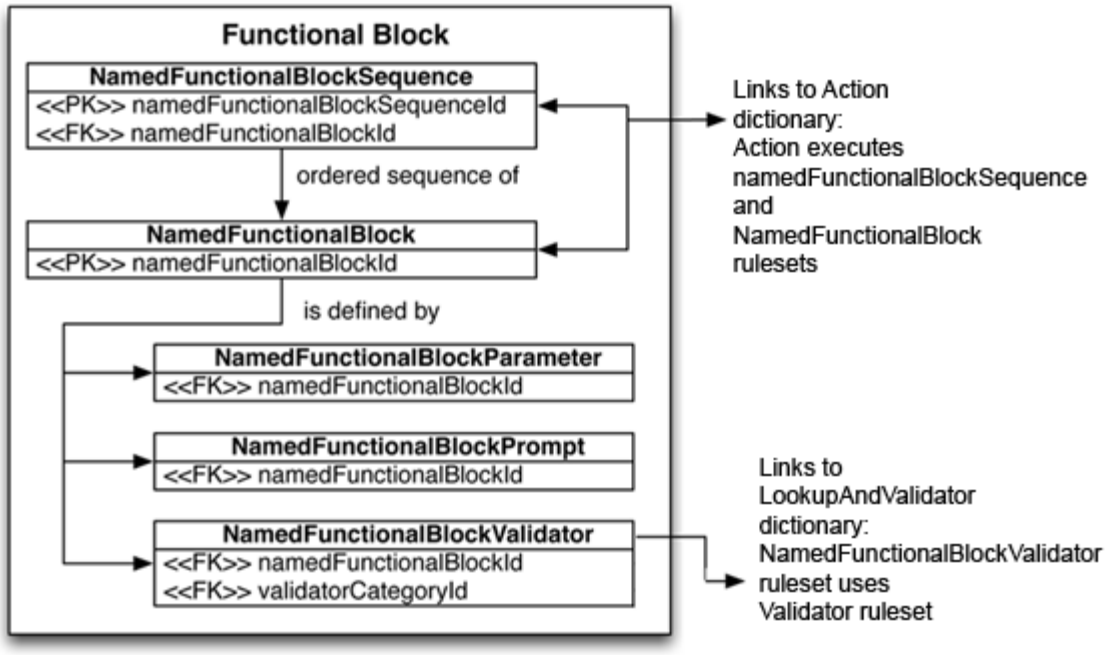


Figure A–3 Lookup and Validator Entity Relationships

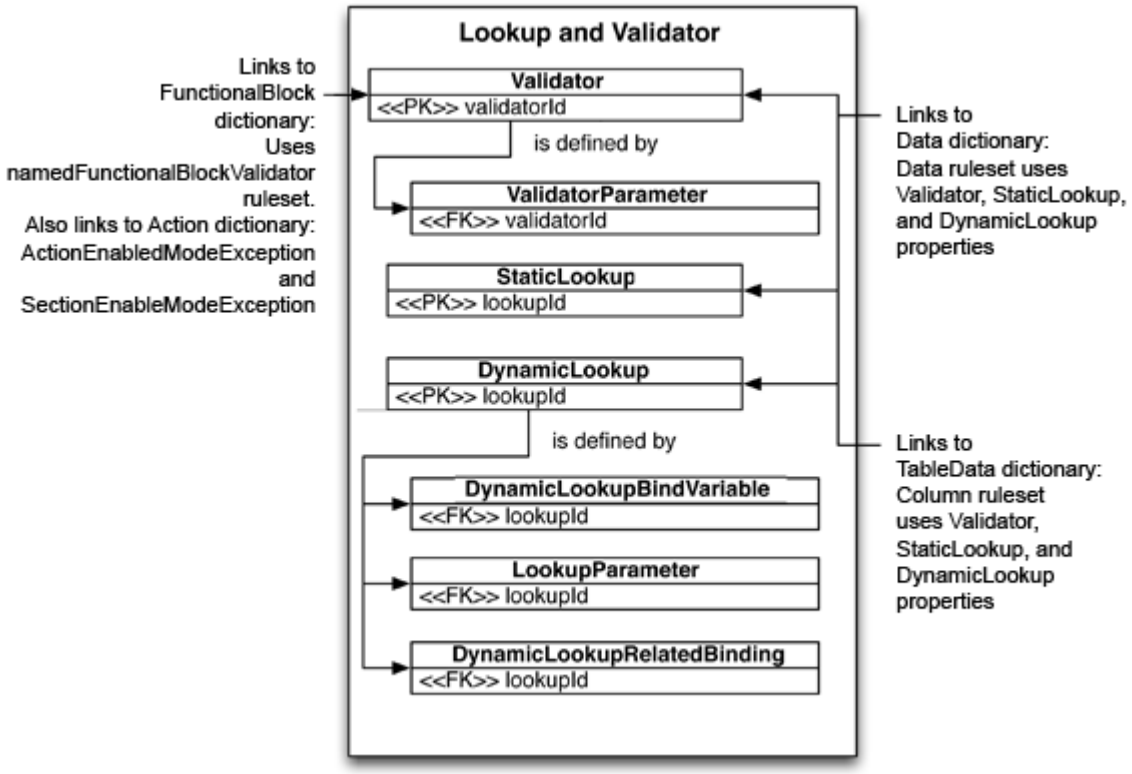


Figure A–4 Data Entity Relationships

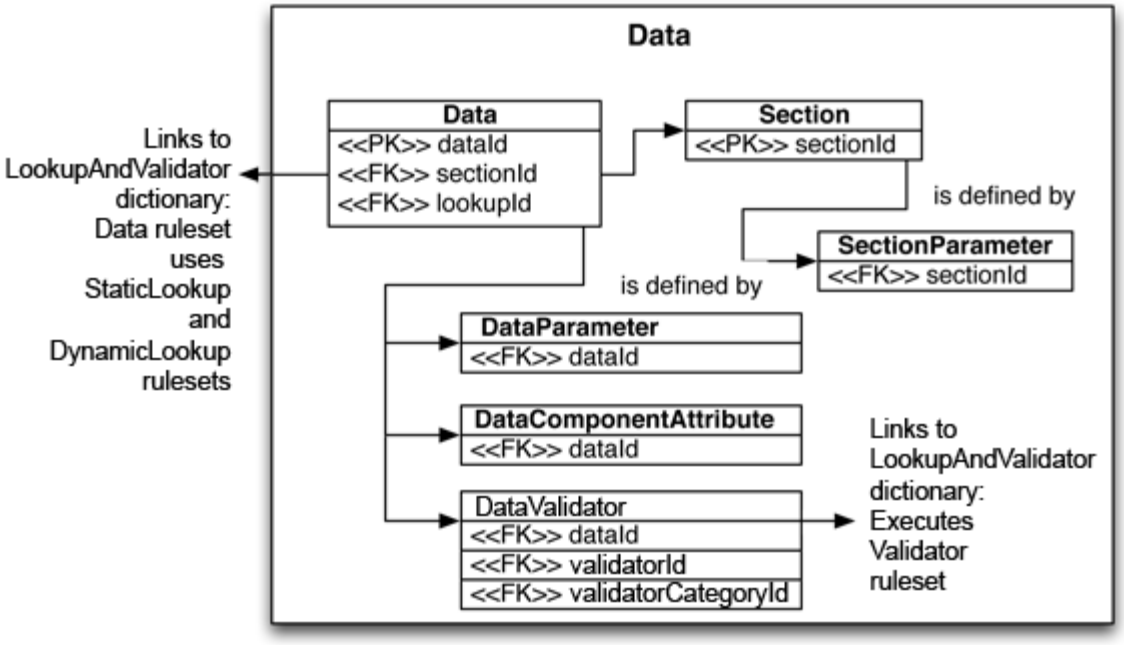


Figure A–5 Table Data Entity Relationships

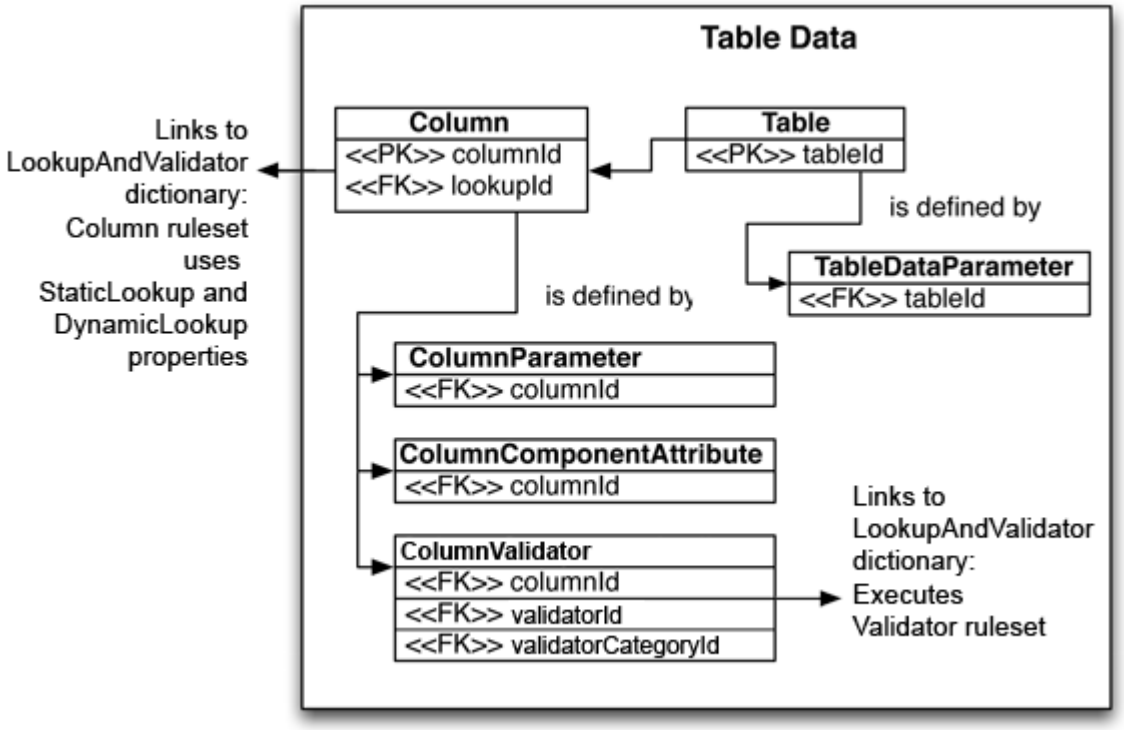
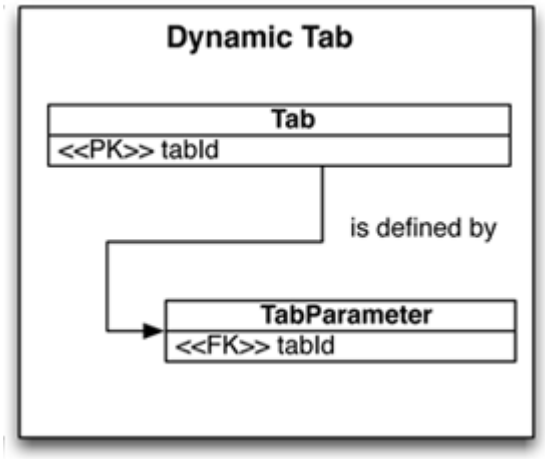


Figure A–6 Dynamic Tab Entity Relationships

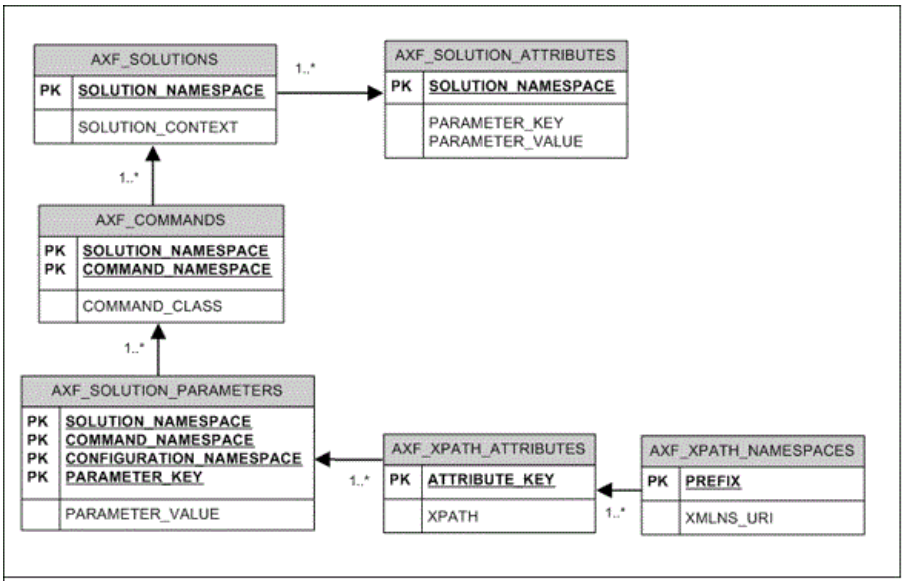


A.2 AXF Tables for the BPM Imaging Solution

This section describes the BPM AXF tables.

The diagram that follows displays the BPM AXF configuration tables and their relationships.

Figure A–7 BPM AXF Configuration Tables (BPM Imaging Solution)



AXF Table	Description
AXF_SOLUTIONS Table AXF_SOLUTION_ATTRIBUTES Table	Define AXF solutions, and general parameters for infrastructure, services, and solutions.
AXF_COMMANDS Table	Define AXF commands within solutions.
AXF_SOLUTION_PARAMETERS Table	Define parameters for AXF commands.
AXF_XPATH_ATTRIBUTES Table, AXF_XPATH_NAMESPACES Table	Define XPATH attributes for payload elements.

A.2.1 AXF_SOLUTIONS Table

The AXF_SOLUTIONS table defines the solutions used by AXF. It links to the AXF_COMMANDS Table through the SOLUTION_NAMESPACE column.

A.2.1.1 Column Description

Table A–41 Column Description for AXF_SOLUTIONS Table

Column	Description
SOLUTION_CONTEXT	Defines the JNDI name of the AXF solution implementation. (Currently, AxfCommandMediator is the only solution implementation.)
SOLUTION_NAMESPACE	Defines the AXF solution name.

A.2.1.2 Example Implementation

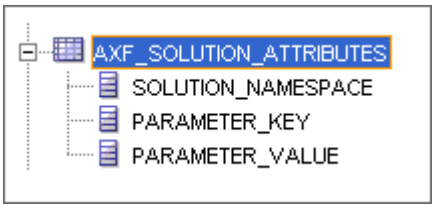
This example table shows the AXF solutions defined. Each of the solutions uses AxfCommandMediator as its solution implementation.

Table A–42 Example AXF_SOLUTIONS Table (HelloBPM)

SOLUTION_NAMESPACE	SOLUTION_CONTEXT
SalesQuoteEntry	ejb.AxfCommandMediator#oracle.imaging.axf.service.AxfCommandMediatorRemote
SalesQuoteApproval	ejb.AxfCommandMediator#oracle.imaging.axf.service.AxfCommandMediatorRemote
SalesQuoteAccountManager	ejb.AxfCommandMediator#oracle.imaging.axf.service.AxfCommandMediatorRemote

A.2.2 AXF_SOLUTION_ATTRIBUTES Table

This table defines general attributes for use by infrastructure, services, or solutions. For example, use this table to define connection settings.



A.2.2.1 Column Description

Table A–43 Column Description for AXF_SOLUTION_ATTRIBUTES Table

Column	Description
SOLUTION_NAMESPACE	Specifies the solution namespace that uses this parameter.
PARAMETER_KEY	Name of the parameter. Used when retrieving the parameter value from the database. Parameters include: ▪ Provider_URL: Identifies the location of the SOA system. Identifies where the BPM connection information is stored.
PARAMETER_VALUE	Value of the parameter.

A.2.2.2 Example Implementation

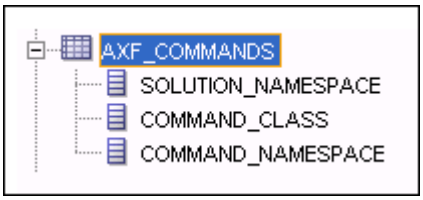
This example table sets solution attributes for the BPM HelloBPM solution.

Table A–44 Example AXF_SOLUTION_ATTRIBUTES Table

SOLUTION_ NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
SalesQuoteEntry	CONNECTION_PROVIDER	oracle.imaging.axf.servicemodules.bpel.workflow.AxfWorkflowService Module

A.2.3 AXF_COMMANDS Table

Use this table to define AXF commands and their java classes for each solution. Note that you configure each command's parameters in the AXF_SOLUTION_ PARAMETERS Table.



A.2.3.1 Column Description

Table A–45 Column Description for AXF_COMMANDS Table

Column	Description
SOLUTION_NAMESPACE	The name of the solution, as defined in the AXF_SOLUTIONS Table.
COMMAND_NAMESPACE	Defines the unique name of the command within the solution.
COMMAND_CLASS	The fully qualified class name in which the command is defined. This class is loaded and the execute() method representing the command is executed. For information about a specific task, see the specific task listed in Section A.3.

A.2.3.2 Example Implementation

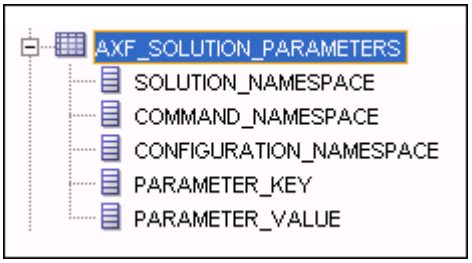
This example shows commands defined for the HelloBPM solution.

Table A–46 Example AXF_COMMANDS Table for HelloBPM

SOLUTION_NAMESPACE	COMMAND_CLASS	COMMAND_NAMESPACE
SalesQuoteApproval	oracle.ecm.axf.commands.system.OpenSolutionWorkspace	StartSalesQuoteApproval
SalesQuoteEntry	oracle.ecm.axf.commands.bpm.UpdateTaskCommand	SaveQuote
SalesQuoteEntry	oracle.ecm.axf.commands.system.OpenSolutionWorkspace	StartSalesQuoteEntry
SalesQuoteAccountManager	oracle.ecm.axf.commands.system.OpenSolutionWorkspace	StartSalesQuoteAccountManager

A.2.4 AXF_SOLUTION_PARAMETERS Table

This table defines command parameters for the solution and AXF commands.



A.2.4.1 Column Description

Table A–47 Column Description for AXF_SOLUTION_PARAMETERS Table

Column	Description
SOLUTION_NAMESPACE	Identifies the solution namespace, as defined in the AXF_SOLUTIONS Table.
COMMAND_NAMESPACE	Specifies the command name, as defined in the AXF_COMMANDS Table.
CONFIGURATION_NAMESPACE	Used to implement the command. Specify the complete package name of the implementation class. This namespace path provides the physical Java class to instantiate. The namespace also differentiates commands within the same solution namespace.
PARAMETER_KEY	Specifies the parameter key to use in the AXF command. For parameter details, see the specific command: AXF Commands: ■ Open Solution Workspace Command ■ Update Task Command ■ Document Search Command
PARAMETER_VALUE	Specifies the value of the parameter key. (For parameter details, see the specific AXF command or user component.) If the value has an XPATH: prefix, the attribute value comes from the AXF_XPATH_ATTRIBUTES Table.

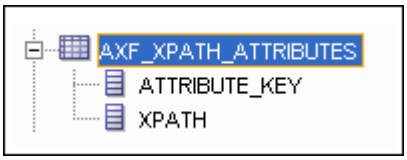
A.2.4.2 Example Implementation

Table A–48 Example AXF_SOLUTION_PARAMETERS Table for HelloBPM

SOLUTION_ NAMESPACE	COMMAND_ NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_ KEY	PARAMETER_VALUE
SalesQuoteApproval	StartSalesQuoteApproval	oracle.ecm.axf.commands.system.OpenSolutionWorkspace	BASE_URL	http://*hostname*:16000/axf-solutionWorkspace/faces/solutionworkspace
SalesQuoteEntry	SaveQuote	oracle.ecm.axf.commands.bpm.UpdateTaskCommand	OpportunityID	XPATH:Order_OpportunityID
SalesQuoteEntry	StartSalesQuoteEntry	oracle.ecm.axf.commands.system.OpenSolutionWorkspace	BASE_URL	http://*hostname*:16000/axf-solutionWorkspace/faces/solutionworkspace
SalesQuoteAccount Manager	StartSalesQuoteAccount Manager	oracle.ecm.axf.commands.system.OpenSolutionWorkspace	BASE_URL	http://*hostname*:16000/axf-solutionWorkspace/faces/solutionworkspace

A.2.5 AXF_XPATH_ATTRIBUTES Table

This table defines the XPATH attributes used in the AXF framework. This XPATH is mainly defined for payload elements.



A.2.5.1 Column Description

Table A–49 Column Description for AXF_XPATH_ATTRIBUTES Table

Column	Description
ATTRIBUTE_KEY	Attribute key referenced in the Parameter Value column in the AXF_SOLUTION_PARAMETERS Table.
XPATH	XPATH expression used to locate the value in the payload.

A.2.5.2 Example Implementation

This example follows an XPATH attribute specified for the Opportunity ID field in the AXF_SOLUTION_PARAMETERS table. The PARAMETER_VALUE column contains an XPATH: prefix, indicating that the attribute value comes from the AXF_XPATH_ATTRIBUTES table.

Fields not shown: SOLUTION_NAMESPACE=SalesQuoteEntry

Table A–50 Example AXF_SOLUTION_PARAMETERS Table

COMMAND_NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
SaveQuote	oracle.ecm.axf.commands.bpm.UpdateTaskCommand	OpportunityID	XPATH:Order_OpportunityID

In the AXF_XPATH_ATTRIBUTES table that follows, the corresponding XPATH column displays the XPATH expression used to locate the value in the payload. The last row displays an optional configuration that specifies the conversion of a value.

Table A–51 Example AXF_XPATH ATTRIBUTES Table

ATTRIBUTE_KEY	XPATH
Order_OpportunityID	//task:UserData/QuoteRequest/salesQuote:Summary/salesQuote:OpportunityID

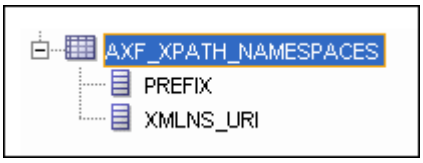
In the AXF_NAMESPACES table that follows, the XMLNS_URI column displays where within the XML file to locate the SalesQuote task information.

Table A–52 Example AXF_XPATH_NAMESPACES Table

Prefix	XMLNS_URI
task	http://xmlns.oracle.com/bpm/workflow/task

A.2.6 AXF_XPATH_NAMESPACES Table

The following table defines the namespaces used for the XPATH attributes. It links to the AXF_XPATH_ATTRIBUTES Table.



A.2.6.1 Column Description

Table A–53 Column Description for AXF_XPATH_NAMESPACES Table

Column	Description
PREFIX	The namespace prefix used in the XPATH.
XMLNS_URI	Provides a unique identifier.

A.2.6.2 Example Implementation

Table A–54 HelloBPM AXF_XPATH_NAMESPACES Table

Prefix	XMLNS_URI
task	http://xmlns.oracle.com/bpm/workflow/task
xml	http://www.w3.org/XML/1998/namespace
salesQuote	http://www.example.com/ns/salesquote

A.3 AXF Commands for the BPM Imaging Solution

BPM AXF commands include:

- Section A.3.1, "Open Solution Workspace Command"
- Section A.3.2, "Update Task Command"
- Section A.3.3, "Document Search Command"

A.3.1 Open Solution Workspace Command

This command returns an instruction to the requester to open a browser with a URL to the AXF solution workspace. Its primary purpose is to display the solution workspace and solution application.

A.3.1.1 Open Solution Workspace Command Parameters

Table A–55 OpenSolutionWorkspaceCommand Parameters

Parameter Key	Description
BASE_URL	This URL is returned in the response command upon executing this command. It is a standard URL string that directs to the specified URL.

A.3.1.2 Example Implementation

Table A–56 Example AXF_COMMANDS Table

SOLUTION_NAMESPACE	COMMAND_CLASS	COMMAND_NAMESPACE
SalesQuoteEntry	oracle.ecm.axf.commands.system.OpenSolutionWorkspace	StartSalesQuoteEntry

Fields not shown: SOLUTION_NAMESPACE=SalesQuoteEntry

Table A–57 Example Redirect Command in AXF_SOLUTION_PARAMETERS Table

COMMAND_NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
StartSalesQuoteEntry	oracle.ecm.axf.commands.system.OpenSolutionWorkspace	BASE_URL	`http://host:port/axf-solutionWorkspace/faces/solutionworkspace`

A.3.2 Update Task Command

The Update Task Command updates the BPM task referenced in the AXF conversation associated with the AXF request. The command updates BPM task payload data given the arguments passed in the request. It also signals the AXF solution application of the update to merge the updated data with changes made within the application.

The Update Task command can update values in the XML payload using XPATH or system attributes. (For an XPATH example, see Section A.2.5 and Section A.2.6.)

You can create your own parameter keys for the Update Task command and use either a system attribute or an XPATH for the parameter value. AXF searches the request parameters and finds all the values that match the parameter keys (besides Outcome), and pulls parameter keys for the list of attributes to use in that task payload.

To update a non-payload attribute in the BPM task, use a system attribute from those listed in Section A.3.2.2. For example, the UpdateTask command can take the value of `Outcome` (defined as PARAMETER_KEY) from the request parameter and update the `OUTCOME` (defined as PARAMETER_VALUE) attribute value in the task.

A.3.2.1 Update Task Command Parameters

Table A–58 Parameters for UpdateTaskCommand

Parameter Key	Description
OUTCOME	Specify the outcome defined for the human work flow system. APPROVE and REJECT are available by default in any BPM process; the process designer can create others.

A.3.2.2 System Attributes

System Attributes
ACQUIREDBY
APPROVERS
ASSIGNEDDATE
ASSIGNEDGROUP //Cannot be updated
ASSIGNEDUSER //Cannot be updated
CREATEDATE
CREATOR
DATEATTRIBUTE1-DATEATTRIBUTE5
EXPIREDDATE
ENDDATE
FORMATTRIBUTE1-FORMATTRIBUTE5
FROMUSER
NUMBERATTRIBUTE1-NUMBERATTRIBUTE5
OUTCOME
OWNERGROUP
OWNERUSER
PRIORITY
STATE
TASKID
TASKNUMBER //Cannot be updated
TITLE
TASKDEFINITIONNAME
TEXTATTRIBUTE1-TEXTATTRIBUTE10
UPDATEDBY
URLATTRIBUTE1 - URLATTRIBUTE5

A.3.2.3 Example Implementation

This example shows two attributes updated by the same comand_namespace, UpdateHelloBPM.

Fields not shown: SOLUTION_NAMESPACE=SalesQuoteEntry

Table A–59 Example UpdateTaskCommand Parameters in AXF_SOLUTION_PARAMETERS Table

COMMAND_ NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
SaveQuote	oracle.ecm.axf.commands.bpm.UpdateTaskCommand	OpportunityID	XPATH:Order_OpportunityID

A.3.3 Document Search Command

This command retrieves documents stored in the document repository system for display in the viewer, using a search URL configured in the AXF_SOLUTION_

PARAMETERS Table. You can specify parameters to pass from the business application, which are appended to the search URL.

A.3.3.1 Document Search Command Parameters

Table A–60 lists configuration parameters for this command. Use these parameters in the AXF_SOLUTION_PARAMETERS Table to configure commands.

Table A–60 Parameters for DocumentSearchCommand

Parameter Key	Description
DOCUMENT_SEARCH_URL	Specifies the URL to be returned to the caller with the passed in parameters appended to the end of the URL string.

A.3.3.2 Example Implementation

The following tables show configuration examples for the Document Search command. The first example (SearchIPM) specifies the URL with no passed request parameters. The second example expects 1..n request parameters.

Table A–61 Example AXF_COMMANDS Table

SOLUTION_NAMESPACE	COMMAND_CLASS	COMMAND_NAMESPACE
SalesQuoteEntry	oracle.ecm.axf.commands.ucm.DocumentSearchCommand	ImagingSearch

Fields not shown: SOLUTION_NAMESPACE=SalesQuoteEntry

Table A–62 Example Document Search Command in AXF_SOLUTION_PARAMETERS Table

COMMAND_ NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
ImagingSearch	oracle.ecm.axf.commands.ucm.DocumentSearchCommand	DOCUMENT_ SEARCH_URL	http://MyIPMServer:Port/imaging/faces/Pages/UrlTools.jspx?ToolName=ExecuteSearch&SearchName=HelloBPM

A.4 Oracle E-Business Suite Forms Tables for the BPM Imaging Solution

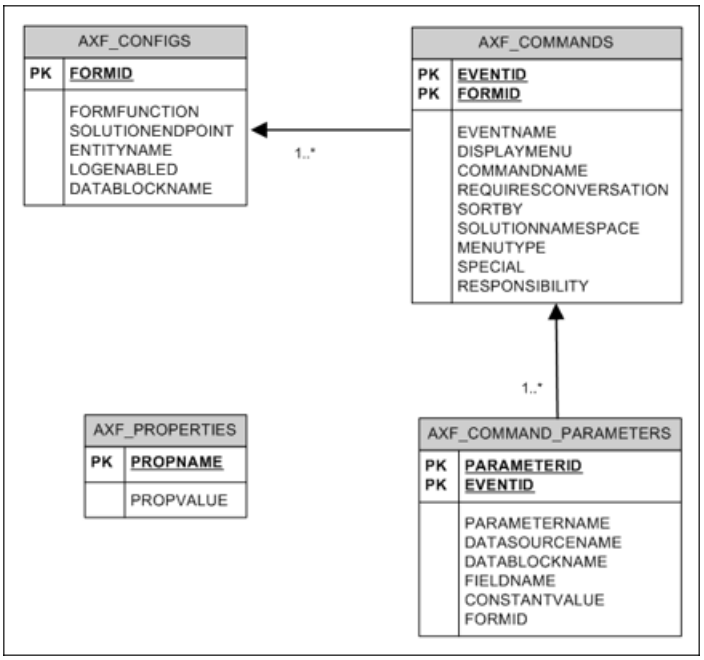
Configuring AXF for Oracle E-Business Suite requires configuring AXF-related tables in Oracle E-Business Suite. This section covers the following topics:

- Section A.4.1, "About the Oracle E-Business Suite AXF Tables For the BPM Imaging Solution"
- Section A.4.2, "AXF_CONFIGS Table (Oracle E-Business Suite)"
- Section A.4.3, "AXF_COMMANDS Table (Oracle E-Business Suite)"
- Section A.4.4, "AXF_COMMAND_PARAMETERS Table (Oracle E-Business Suite)"
- Section A.4.5, "AXF_PROPERTIES Table (Oracle E-Business Suite)"

> **Note:** The Oracle E-Business Suite BPM imaging solution is supported for Forms-based applications only, not OAF applications.

A.4.1 About the Oracle E-Business Suite AXF Tables For the BPM Imaging Solution

The following diagram shows how the tables used by the Oracle E-Business Suite system in AXF solutions are related.



A.4.2 AXF_CONFIGS Table (Oracle E-Business Suite)

Use the AXF_CONFIGS table to enable the AXF solution on various Oracle E-Business Suite Forms. This table provides a fine level of granularity when selecting which Forms to AXF-enable, up to the Data Block level.

Form events are automatically invoked when an action is performed on an Oracle E-Business Suite Form. The AXF_CUSTOM.pll makes all events available, such as POST-INSERT, for customization. You can decide which events to use, and how and when to use them.

When an action occurs, the customized code launches the specified solution and command configured for the event. In the case where the same form is being reused, FORMFUNCTION and DATABLOCKNAME uniquely identify each Form.

> **Note:** You can enable all datablocks on a form rather than a specific datablock, by specifying `AXF_DEFAULT` for the DATABLOCKNAME parameter. This allows AXF to be notified whenever a POST-INSERT event occurs for the form, regardless of its datablock. Note, however, that setting the DATABLOCKNAME parameter to AXF_DEFAULT enables specified ZOOM or SPECIAL commands on all pages related to the form. (Set ZOOM and SPECIAL commands in the AXF_COMMANDS Table (Oracle E-Business Suite).)

A.4.2.1 Column Description

Table A–63 Column Description for AXF_CONFIGS Table

Column Name	Description
FORMID	Specifies the primary key of the table.
FORMFUNCTION	Distinguishes each Oracle E-Business Suite Form based on the form's functionality.

Table A–63 (Cont.) Column Description for AXF_CONFIGS Table

Column Name	Description
SOLUTIONENDPOINT	Specifies a URL to AXF. This value should start as `https://` if SecureMode is on. See Section 2.4.2.
ENTITYNAME	Used by the attachment functionality as a unique name, which links attachments to the correct Forms.
LOGENABLED	Enables or disables the log for the specified form. Specify one of the following: ■ 1/TRUE/YES ■ 0/FALSE/NO
DATABLOCKNAME	Specify the data block on the form to enable. Note that you can also specify `AXF_DEFAULT` to enable all data blocks on the form. A Form may be reused by Oracle E-Business Suite; the FORMFUNCTION and DATABLOCKNAME together uniquely identify each form.

A.4.3 AXF_COMMANDS Table (Oracle E-Business Suite)

Use the AXF_COMMANDS table to describe the actions to take based on user activity. This table works with the AXF_CONFIGS Table (Oracle E-Business Suite).

A.4.3.1 Column Description

Table A–64 Column Description for AXF_COMMANDS Table

Column Name	Description
FORMID	Links to the AXF_CONFIGS Table (Oracle E-Business Suite).
EVENTID	Primary key of the table.
EVENTNAME	Name of the Event command to invoke (for example, ZOOM, POST-INSERT).
DISPLAYMENU	Displays the text of the menu for the command.
COMMANDNAMESPACE	Request command to pass to the back-end when the menu is selected.
REQUIRESCONVERSATION	Indicates if the command requires a valid conversation or not.
SORTBY	Order in which to display the menu.
SOLUTIONNAMESPACE	Name of the solution.

Table A–64 (Cont.) Column Description for AXF_COMMANDS Table

Column Name	Description
MENUTYPE	Specify the menu type to display to users in Oracle E-Business Suite. You can choose: ■ ZOOM: Displays a Zoom menu in the toolbar. ■ ZOOMANDSPECIAL: Displays both a Zoom menu and a Special menu. (Enter a special key in the SPECIAL column.) ■ SPECIAL: Displays a Special menu on the toolbar. (Enter a special key in the Special column.)
SPECIAL	Create new menu entries by entering a unique number for the Special type menu, where: ■ SPECIAL1-15 creates entries in the Tools menu. ■ SPECIAL16-30 creates entries in the Reports menu. ■ SPECIAL31-45 creates entries in the Actions menu. (Consult the Oracle E-Business Suite documentation for further information.)
RESPONSIBILITY	Reserved for future use.

A.4.4 AXF_COMMAND_PARAMETERS Table (Oracle E-Business Suite)

Use the AXF_COMMAND_PARAMETERS table to define the information sent for each defined command. Each command may require or omit a different set of parameters.

A.4.4.1 Column Description

Table A–65 Column Description for AXF_COMMAND_PARAMETERS Table

Column	Description
PARAMETERID	Defines a unique ID for the parameter.
EVENTID	Defines a unique ID for the event. Comes from the AXF_COMMANDS Table (Oracle E-Business Suite).
PARAMETERNAME	The name of the parameter to pass.
DATASOURCENAME	Data source for the parameter value. You can specify *Data* or *Constant*.
DATABLOCKNAME	Data block of the form from which the value is fetched.
FIELDNAME	Field name in the form from which the value is fetched.
CONSTANTVALUE	A constant value for the parameter.

A.4.5 AXF_PROPERTIES Table (Oracle E-Business Suite)

Use the AXF_PROPERTIES table to define properties for an AXF integration with Oracle E-Business Suite.

A.4.5.1 Column Description

Table A–66 Column Description for AXF_PROPERTIES Table

Column	Description
PROPNAME	Specifies properties to use. Properties include: ■ SecureMode: To enable SSL, set this property to ON, and set values for AXFWalletPath and AXFWalletKey properties. ■ AXFWalletPath: Certificate location (path). ■ AXFWalletKey: Specify the key for the AXF wallet password, to be retrieved from the Oracle E-Business Suite vault. ■ AXF_VERSION: Specify `1` for AXF 10g, `2` for AXF for 11g. ■ AXF_SOAP_POLICY: Specify the name of the SOAP policy to use. The currently supported policy is `USER_NAME_TOKEN`. ■ AXF_SOAP_SECURITY: Specify `TRUE` (required), which enables SOAP security, as described in Section 2.4.1. ■ AXF_SOAP_USER: Specify the SOAP userid used in the SOAP header for authentication, as described in Section 2.4.1. ■ AXF_PAPERCLIP: Set to `TRUE` to enable the paperclip option, or `FALSE` (default) to disable it. Also see Section 6.4.11.
PROPVALUE	Specifies the property's value.

A.5 Oracle PeopleSoft Tables for the BPM Imaging Solution

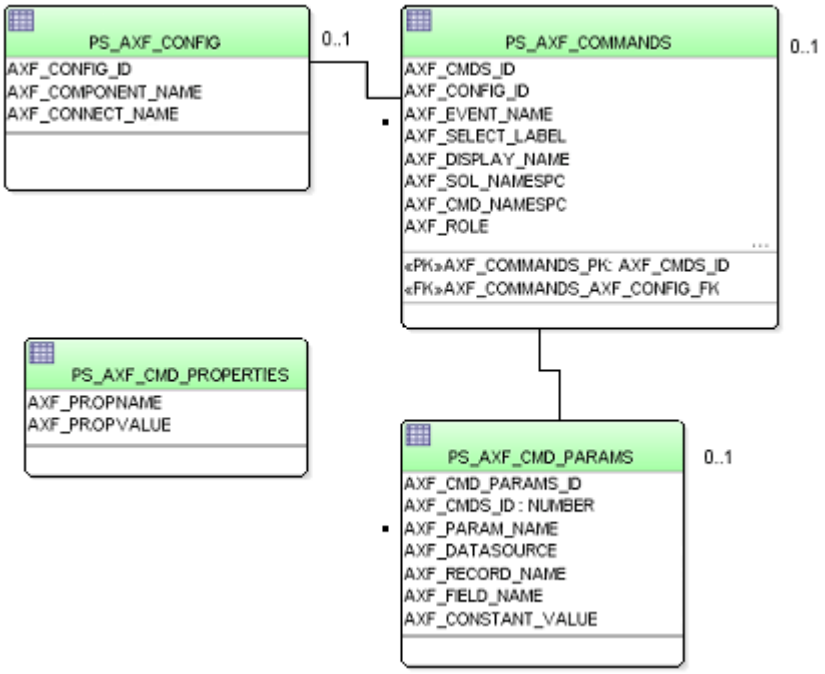
Configuring the adapter for Oracle PeopleSoft requires configuring AXF-related tables in Oracle PeopleSoft. For more information, see Section 3.5.

This section covers the following topics:

- Section A.5.1, "About the Oracle PeopleSoft AXF Tables For the BPM Imaging Solution"
- Section A.5.2, "PS_AXF_CONFIG Table"
- Section A.5.3, "PS_AXF_COMMANDS Table"
- Section A.5.4, "PS_AXF_COMMAND_PARAMS Table"
- Section A.5.5, "PS_AXF_PROPERTIES Table"

A.5.1 About the Oracle PeopleSoft AXF Tables For the BPM Imaging Solution

The following diagram shows how the tables used by the Oracle PeopleSoft system in AXF solutions are related.



Note: All AXF table fields require non-null values. You can include a space for fields that do not apply.

A.5.2 PS_AXF_CONFIG Table

Use the PS_AXF_CONFIG table to enable the AXF solution on various Oracle PeopleSoft components. This table provides a fine level of granularity when selecting which pages and components to AXF-enable.

Events are automatically invoked when an action is performed on an Oracle PeopleSoft page. The AXF_PS_Integration project component catches PRE- and POST-save events and makes them available for customization. You can decide which events to use and how and when to use them.

A.5.2.1 Column Description

Table A–67 Column Description for PS_AXF_CONFIG Table

Column Name	Description
AXF_CONFIG_ID	Specifies the primary key of the table.
AXF_COMPONENT_NAME	Specifies the name of the Oracle PeopleSoft component being enabled.
AXF_CONNECT_NAME	Specifies the Oracle PeopleSoft Integration Broker Connection name (service operation to call), as defined in the Oracle PeopleSoft Integration Broker administration interface. The default service operation is AXF_EXECUTE. You can set up and use other connections.

Table A–67 (Cont.) Column Description for PS_AXF_CONFIG Table

Column Name	Description
AXF_VERSION	Specifies the AXF version of the connection. Available values include: ■ 10g ■ 11g

A.5.3 PS_AXF_COMMANDS Table

Use the PS_AXF_COMMANDS table to describe the actions to take based on user activity. This table works with the PS_AXF_CONFIG Table.

A.5.3.1 Column Description

Table A–68 Column Description for PS_AXF_COMMANDS Table

Column Name	Description
AXF_CMDS_ID	Specifies the primary key of the table, used to identify the unique command.
AXF_CONFIG_ID	Specifies the foreign key to the PS_AXF_CONFIG Table, which associates this unique command with a particular page and component.
AXF_EVENT_NAME	Specifies the event being executed. The AXF_EVENT_NAME corresponds to the subpage that is incorporated into an existing Oracle PeopleSoft page, such as the button, link, or menu that is added to an Oracle PeopleSoft page to invoke AXF functionality. For more information, see Section 3.3. Available options include: ■ AXF_BUTTON_1, …, AXF_BUTTON_5 ■ AXF_LINK_1, ..., AXF_LINK5 ■ AXF_COMBO_1, ..., AXF_COMBO_5 ■ SAVE_PRE_CHANGE ■ SAVE_POST_CHANGE
AXF_SELECT_LABEL	Defines the name displayed on the page for the selection field. This is used only with the COMBO event.
AXF_DISPLAY_NAME	Defines the name of the button or link to display on the Oracle PeopleSoft page.
AXF_SOL_NAMESPC	Identifies the solution namespace for the command to execute.
AXF_CMD_NAMESPC	Identifies the command namespace for the command to execute.
AXF_PSFT_ROLE	Identifies the Oracle PeopleSoft roles with access to the command. It is a comma-delimited list with each role enclosed in single quotes (for example, 'Role1','Role2','Role3').
AXF_SORT_ORDER	Specifies the order of items displayed in a selection field. Sort order applies to selection fields only.
AXF_REQ_CONV	Specifies if a conversation is required to this command before execution.

A.5.4 PS_AXF_COMMAND_PARAMS Table

Use the PS_AXF_COMMAND_PARAMS table to specify the information sent for each defined command. Each command may require or omit a different set of parameters.

A.5.4.1 Column Description

Table A–69 Column Description for AXF_COMMAND_PARAMETERS Table

Column	Description
AXF_CMDS_PARAMS_ID	Specifies the primary key of the table.
AXF_CMDS_ID	Specifies the foreign key to the PS_AXF_COMMANDS Table, which associates the unique parameter with a specific command.
AXF_PARAM_NAME	Defines the parameter name.
AXF_DATASOURCE	Specifies where the parameter value is retrieved. ■ DATA: Retrieves the value in Oracle PeopleSoft defined by the AXF_RECORD_NAME and AXF_FIELD_NAME fields. ■ CONSTANT: Uses the value defined in the AXF_CONSTANT_VALUE field.
AXF_RECORD_NAME	Identifies the record of the field in the Oracle PeopleSoft page to use as the target value to retrieve when AXF_DATASOURCE is set to DATA.
AXF_FIELD_NAME	Used as the target value when AXF_DATASOURCE is set to DATA.
AXF_CONSTANT_VALUE	Used as the constant value when AXF_DATASOURCE is set to CONSTANT.

A.5.5 PS_AXF_PROPERTIES Table

Use the PS_AXF_PROPERTIES table to define properties for an AXF integration with Oracle PeopleSoft. See Section 6.4.12.

A.5.5.1 Column Description

Table A–70 Column Description for PS_AXF_PROPERTIES Table

Column	Description
AXF_PROPNAME	Specifies properties to use. Properties include: ■ WINDOW_HEIGHT: Specify the popup window's height in pixels. (This refers to any popup window that displays as part of the solution.) ■ WINDOW_WIDTH: Specify the popup window's width in pixels.
AXF_PROPVALUE	Specifies the property's value.

B

BPEL Imaging Solution Reference

This appendix describes the AXF, Oracle E-Business Suite, and Oracle PeopleSoft configuration tables used for the BPEL Imaging solution. It describes AXF commands and web user interface tools such as the Task List and Task Viewer, and provides example implementations.

This appendix covers the following topics:

- Section B.1, "AXF Tables For the BPEL Imaging Solution"
- Section B.2, "AXF User Components for the BPEL Imaging Solution"
- Section B.3, "AXF Commands for the BPEL Imaging Solution"
- Section B.4, "Oracle E-Business Suite Forms Tables For the BPEL Imaging Solution"
- Section B.5, "Oracle PeopleSoft Tables For the BPEL Imaging Solution"

B.1 AXF Tables For the BPEL Imaging Solution

This section describes the BPEL AXF tables.

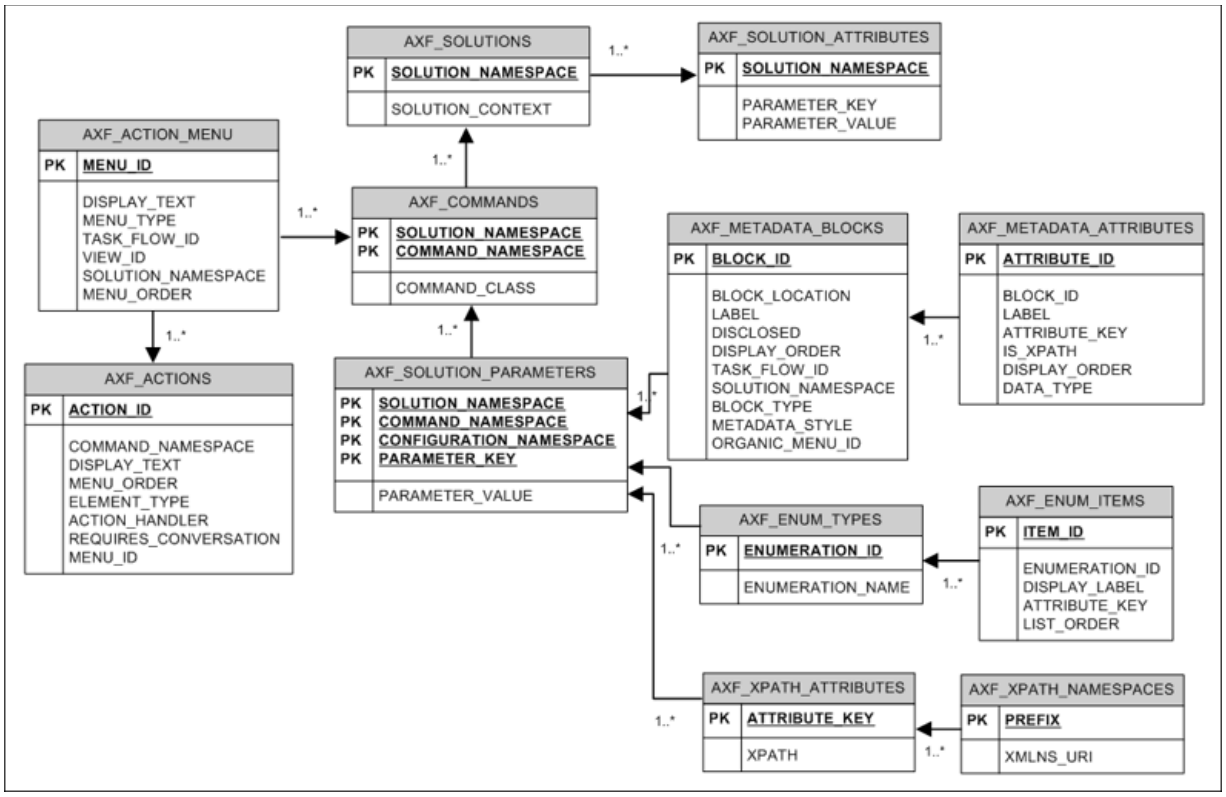
Note: Running an AXF and an Oracle BPM Worklist session at the same time can result in the session first opened ending. For example, launching an AXF session with an Oracle BPM Worklist session open ends the BPM Worklist session.

This conflict occurs because console session information is retained in browser cookies whose names are domain specific, but default to the same initial value. To prevent this conflict, set cookie names unique for each domain. To set cookie names, use the console on the advanced section of the Domain Configuration/General page.

Note: If modifying AXF table values in a running system, either execute Clear DB Cache from the Driver page or restart the AXF application within the Application Server for the changes to take effect. For information about the Driver page, see "Verifying the AXF Installation with HelloWorld" in *Oracle WebCenter Content Installation Guide*.

The diagram that follows displays the BPEL AXF configuration tables and their relationships.

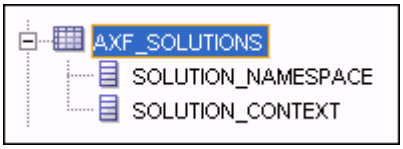
Figure B–1 BPEL AXF Configuration Tables (BPEL Imaging Solution)



AXF Table	Description
AXF_SOLUTIONS Table AXF_SOLUTION_ATTRIBUTES Table	Define AXF solutions, and general parameters for infrastructure, services, and solutions.
AXF_COMMANDS Table	Define AXF commands within solutions.
AXF_SOLUTION_PARAMETERS Table	Define parameters for AXF commands and AXF user interface components.
AXF_ACTION_MENU Table AXF_ACTIONS Table	Define task action pane itself and links in the pane.
AXF_METADATA_BLOCKS Table, AXF_METADATA_ATTRIBUTES Table	Define optional sections, such as Summary, Comments, and dynamic data tables in the Task Viewer.
AXF_ENUM_TYPES Table, AXF_ENUM_ITEMS Table	Define enumeration pickers and their values.
AXF_XPATH_ATTRIBUTES Table, AXF_XPATH_NAMESPACES Table	Define XPATH attributes for payload elements.

B.1.1 AXF_SOLUTIONS Table

The AXF_SOLUTIONS table defines the solutions used by AXF. It links to the AXF_COMMANDS Table through the SOLUTION_NAMESPACE column.



B.1.1.1 Column Description

Table B–1 Column Description for AXF_SOLUTIONS Table

Column	Description
SOLUTION_CONTEXT	Defines the JNDI name of the AXF solution implementation. (Currently, AxfCommandMediator is the only solution implementation.)
SOLUTION_NAMESPACE	Defines the AXF solution name.

B.1.1.2 Example Implementation

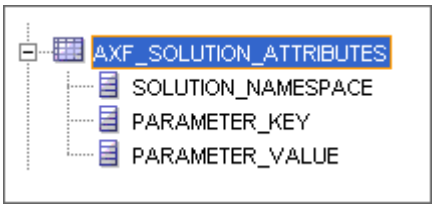
This example table shows the AXF solutions defined. Each of the solutions uses AxfCommandMediator as its solution implementation.

Table B–2 Example AXF_SOLUTIONS Table

SOLUTION_NAMESPACE	SOLUTION_CONTEXT
InvoiceProcessing	ejb.AxfCommandMediator#oracle.imaging.axf.service.AxfCommandMediatorRemote
AccountDistributionApproval	ejb.AxfCommandMediator#oracle.imaging.axf.service.AxfCommandMediatorRemote
SupplierMaintenance	ejb.AxfCommandMediator#oracle.imaging.axf.service.AxfCommandMediatorRemote
RequestInvoiceInformation	ejb.AxfCommandMediator#oracle.imaging.axf.service.AxfCommandMediatorRemote
AccountDistribution	ejb.AxfCommandMediator#oracle.imaging.axf.service.AxfCommandMediatorRemote
InvoiceApproval	ejb.AxfCommandMediator#oracle.imaging.axf.service.AxfCommandMediatorRemote
Rescan	ejb.AxfCommandMediator#oracle.imaging.axf.service.AxfCommandMediatorRemote

B.1.2 AXF_SOLUTION_ATTRIBUTES Table

This table defines general attributes for use by infrastructure, services, or solutions. For example, use this table to define error message addresses, connections, and conversation timeout settings.



B.1.2.1 Column Description

Table B–3 Column Description for AXF_SOLUTION_ATTRIBUTES Table

Column	Description
SOLUTION_NAMESPACE	Specifies the solution namespace that uses this parameter.
PARAMETER_KEY	Name of the parameter. Used when retrieving the parameter value from the database. Parameters include: ■ BPEL_CONNECTION: Identifies the BPEL connection to use. ■ CONNECTION_PROVIDER: Defines the connection (BPEL or custom). If specifying a BPEL connection, this value is *AxfWorkflowServiceModule*. ■ ConversationTimeoutSeconds: Specifies the length of time for which a ConversationID (cid) is valid. The default is 43200 seconds of inactivity. ■ USE_AUTOTASK_LOCKING: Specifies if autotask locking is enabled (TRUE) or disabled (FALSE). Enabling autotask locking can prevent collisions that may occur when multiple users acquire tasks in Autotask mode. See Section 6.4.4. ■ USERNAME_PASS_THROUGH: When FALSE, allows the authenticated Oracle WebLogic Server user to perform tasks, such as BPEL tasks, rather than the user passed from Oracle E-Business Suite or Oracle PeopleSoft (TRUE, default). See Section 6.4.2. ■ TASKLIST_RANGESIZE: Specifies the number of tasks to return in the Task List. For example, you might change the number from 25 (default) to 10 tasks.
PARAMETER_VALUE	Value of the parameter.

B.1.2.2 Example Implementation

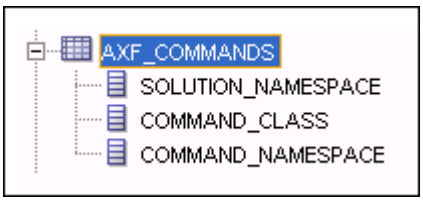
This example table sets solution attributes for the Invoice Processing solution.

Table B–4 Example AXF_SOLUTION_ATTRIBUTES Table

SOLUTION_ NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
InvoiceProcessing	BPEL_CONNECTION	axfconnection
InvoiceProcessing	CONNECTION_PROVIDER	oracle.imaging.axf.servicemodules.bpel.workflow.AxfWorkflowService Module
InvoiceProcessing	USE_AUTOTASK_LOCKING	TRUE
InvoiceProcessing	TASKLIST_RANGESIZE	10

B.1.3 AXF_COMMANDS Table

Use this table to define AXF commands and their java classes for each solution. Note that you configure each command's parameters in the AXF_SOLUTION_ PARAMETERS Table.



B.1.3.1 Column Description

Table B–5 Column Description for AXF_COMMANDS Table

Column	Description
SOLUTION_NAMESPACE	The name of the solution, as defined in the AXF_SOLUTIONS Table.
COMMAND_NAMESPACE	Defines the unique name of the command within the solution.
COMMAND_CLASS	The fully qualified class name in which the command is defined. This class is loaded and the execute() method representing the command is executed. For information about a specific task, see the specific task listed in Section B.3.

B.1.3.2 Example Implementation

This example shows commands defined for the Invoice Processing solution.

Fields not shown: SOLUTION_NAMESPACE=InvoiceProcessing

Table B–6 Example AXF_COMMANDS Table

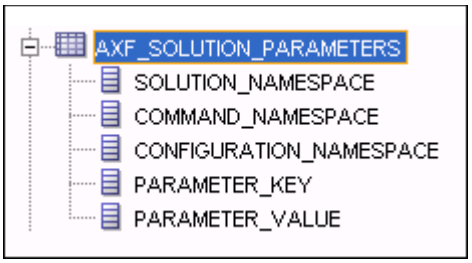
COMMAND_CLASS	COMMAND_NAMESPACE
oracle.imaging.axf.commands.bpel.AutotaskCommand	AutoOpenTask
oracle.imaging.axf.commands.bpel.ReleaseTaskCommand	ReleaseTask
oracle.imaging.axf.commands.bpel.ReleaseTaskCommand	SkipTask
oracle.imaging.axf.commands.bpel.CompleteTaskCommand	AccountDistributionComplete
oracle.imaging.axf.commands.bpel.CompleteTaskCommand	AssignProcessingGroupComplete
oracle.imaging.axf.commands.bpel.CompleteTaskCommand	CompleteInvoice
oracle.imaging.axf.commands.bpel.CompleteTaskCommand	DeleteInvoice
oracle.imaging.axf.commands.bpel.CompleteTaskCommand	DuplicateInvoice
oracle.imaging.axf.commands.bepl.CompleteTaskCommand	Hold
oracle.imaging.axf.commands.bpel.CompleteTaskCommand	InvoiceApprovalComplete
oracle.imaging.axf.commands.bpel.CompleteTaskCommand	RequestInformationComplete
oracle.imaging.axf.commands.bpel.CompleteTaskCommand	RescanComplete
oracle.imaging.axf.commands.bpel.CompleteTaskCommand	SpecialistExceptionComplete
oracle.imaging.axf.commands.bpel.CompleteTaskCommand	SupplierMaintenance
oracle.imaging.axf.commands.bpel.CompleteTaskCommand	SupplierMaintenanceComplete
oracle.imaging.axf.commands.bpel.OpenTaskCommand	OpenTask
oracle.imaging.axf.commands.bpel.UpdateTaskFromProcedureCommand	RetrieveUserList
oracle.imaging.axf.commands.bpel.UpdateTaskCommand	AttachSupplemental
oracle.imaging.axf.commands.bpel.UpdateTaskCommand	SaveInvoice
oracle.imaging.axf.commands.system.RedirectCommand	AccountDistributionEdit
oracle.imaging.axf.commands.system.RedirectCommand	AssignProcessingGroupEdit
oracle.imaging.axf.commands.system.RedirectCommand	EditComments
oracle.imaging.axf.commands.system.RedirectCommand	InvoiceApprovalEdit
oracle.imaging.axf.commands.system.RedirectCommand	RequestInformationEdit
oracle.imaging.axf.commands.system.RedirectCommand	RescanEdit
oracle.imaging.axf.commands.system.RedirectCommand	SearchIPM

Table B–6 (Cont.) Example AXF_COMMANDS Table

COMMAND_CLASS	COMMAND_NAMESPACE
oracle.imaging.axf.commands.system.RedirectCommand	SpecialistExceptionEdit
oracle.imaging.axf.commands.system.RedirectCommand	StartInvoiceProcessing
oracle.imaging.axf.commands.system.RedirectCommand	SupplierMaintenanceEdit
oracle.imaging.axf.commands.system.TerminateConversationCommand	TerminateConversation

B.1.4 AXF_SOLUTION_PARAMETERS Table

This table defines command parameters for the solution, AXF commands, and AXF user interface components.



B.1.4.1 Column Description

Table B–7 Column Description for AXF_SOLUTION_PARAMETERS Table

Column	Description
SOLUTION_NAMESPACE	Identifies the solution namespace, as defined in the AXF_SOLUTIONS Table.
COMMAND_NAMESPACE	Specifies the command name, as defined in the AXF_COMMANDS Table.

Table B–7 (Cont.) Column Description for AXF_SOLUTION_PARAMETERS Table

Column	Description
CONFIGURATION_NAMESPACE	Used to implement the command. Specify the complete package name of the implementation class. This namespace path provides the physical Java class to instantiate. The namespace also differentiates commands within the same solution namespace.
PARAMETER_KEY	Specifies the parameter key to use in the AXF command. For parameter details, see the specific command or web tool: User interface components: ■ Section B.2.2, "Task List Web Tool" ■ Section B.2.3, "Task Viewer Web Tool" ■ Section B.2.4, "Enumeration Picker Web Tool" ■ Section B.2.5, "Identity Picker Web Tool" ■ Section B.2.3.3, "Comments" AXF commands: ■ Section B.3.1, "Open Task Command" ■ Section B.3.2, "Autotask Command" ■ Section B.3.3, "Release Task Command" ■ Section B.3.4, "Complete Task Command" ■ Section B.3.5, "Redirect Command" ■ Section B.3.6, "Update Task Command (BPEL)" ■ Section B.3.7, "Update Task From Procedure Command" ■ Section B.3.8, "Terminate Conversation Command" ■ Section B.3.9, "Validate Task Command"
PARAMETER_VALUE	Specifies the value of the parameter key. (For parameter details, see the specific AXF command or web tool.) If the value has an XPATH: prefix, the attribute value comes from the AXF_XPATH_ATTRIBUTES Table.

B.1.4.2 Example Implementation

This example defines the StartInvoiceProcessing command for the Invoice Processing solution. The first row specifies that the task list be displayed, using the RedirectCommand and corresponding URL. The remaining rows call the task list (in the CONFIGURATION_NAMESPACE column) and define its behavior.

Fields not shown: SOLUTION_NAMESPACE=InvoiceProcessing

Table B–8 Example AXF_SOLUTION_PARAMETERS Table for StartInvoiceProcessing Command

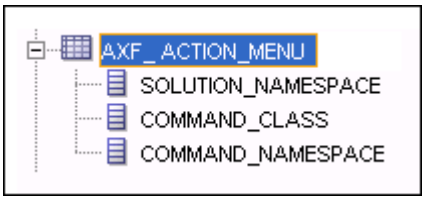
COMMAND_ NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
StartInvoiceProcessing	oracle.imaging.axf.commands.bpel.RedirectCommand	REDIRECT_URL	taskflow://WEB-INF/taskflows/axf-tasklist-tfd.xml#axf-tasklist-tfd
StartInvoiceProcessing	oracle.imaging.axf.web.backing.TaskList	CMD_OPEN_ TASK_BUTTON	OpenTask
StartInvoiceProcessing	oracle.imaging.axf.web.backing.TaskList	CMD_AUTO_ TASK_BUTTON	AutoOpenTask
StartInvoiceProcessing	oracle.imaging.axf.web.backing.TaskList	DEFAULT_VIEW	(null)

Table B–8 (Cont.) Example AXF_SOLUTION_PARAMETERS Table for StartInvoiceProcessing Command

COMMAND_ NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
StartInvoiceProcessing	oracle.imaging.axf.web.backing.TaskList	NO_OF_LINES	20
StartInvoiceProcessing	oracle.imaging.axf.web.backing.TaskList	SHOW_INBOX	FALSE
StartInvoiceProcessing	oracle.imaging.axf.web.backing.TaskList	CONNECTION_ NAME	default
StartInvoiceProcessing	oracle.imaging.axf.web.backing.TaskList	VIEW_LIST	North Invoice Processing Group, South Invoice Processing Group, East Invoice Processing Group, West Invoice Processing Group, My Holds, Exceptions

B.1.5 AXF_ACTION_MENU Table

Use this table to insert and customize an action menu on an AXF web page such as the Task Viewer or Task List page. A common use is to display a Task Actions pane in the Task Viewer for users to click action links related to the displayed task, as shown in Figure B–2. Use the AXF_ACTIONS Table to define a specified menu's actions.



B.1.5.1 Column Description

Table B–9 Column Description for AXF_ACTION_MENU Table

Column	Description
MENU_ID	Specifies a primary key to the AXF_ACTIONS Table, identifying the menu in which to place menu actions.
DISPLAY_TEXT	Specifies the pane's title (for example, *Task Actions*, *Shortcuts*, or *Re-Assignments*).
MENU_TYPE	Specifies where on the page the menu is displayed and its type. (LEFT_SIDEBAR displays a side pane leftmost on the page.)
TASK_FLOW_ID	Specifies the active task flow with which to associate the menu. For example, to add a side menu to a task list display, specify a value of `axf-tasklist-tfd`, which identifies the page that shows the menu.
VIEW_ID	(Reserved for future use.)
SOLUTION_NAMESPACE	Identifies the AXF solution, as defined in the AXF_SOLUTIONS Table.
MENU_ORDER	Defines the order in which the menu is displayed when multiple menus are set for display.

B.1.5.2 Example Implementation

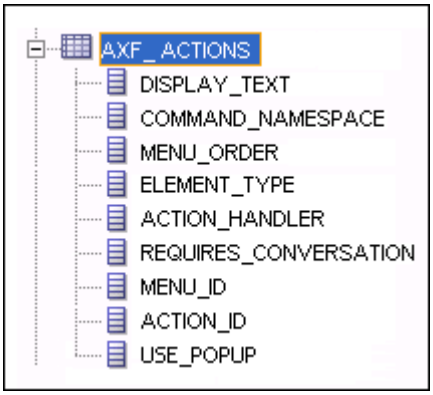
This example table implements a pane entitled *Task Actions* in the left side of the Task Viewer for the Invoice Processing solution.

Table B–10 Example AXF_ACTION_MENU

MENU_ID	DISPLAY_TEXT	MENU_TYPE	TASK_FLOW_ID	VIEW_ID	SOLUTION_NAMESPACE	MENU_ORDER
0	Task Actions	LEFT_SIDEBAR	axf-taskviewer-tfd	null	InvoiceProcessing	0

B.1.6 AXF_ACTIONS Table

This table defines the task actions used in an AXF solution. You can display action menus on AXF web pages such as the Task Viewer page (see Section B.2.3) or a Task List (see Section B.2.2). This table links to the AXF_COMMANDS Table.



B.1.6.1 Column Description

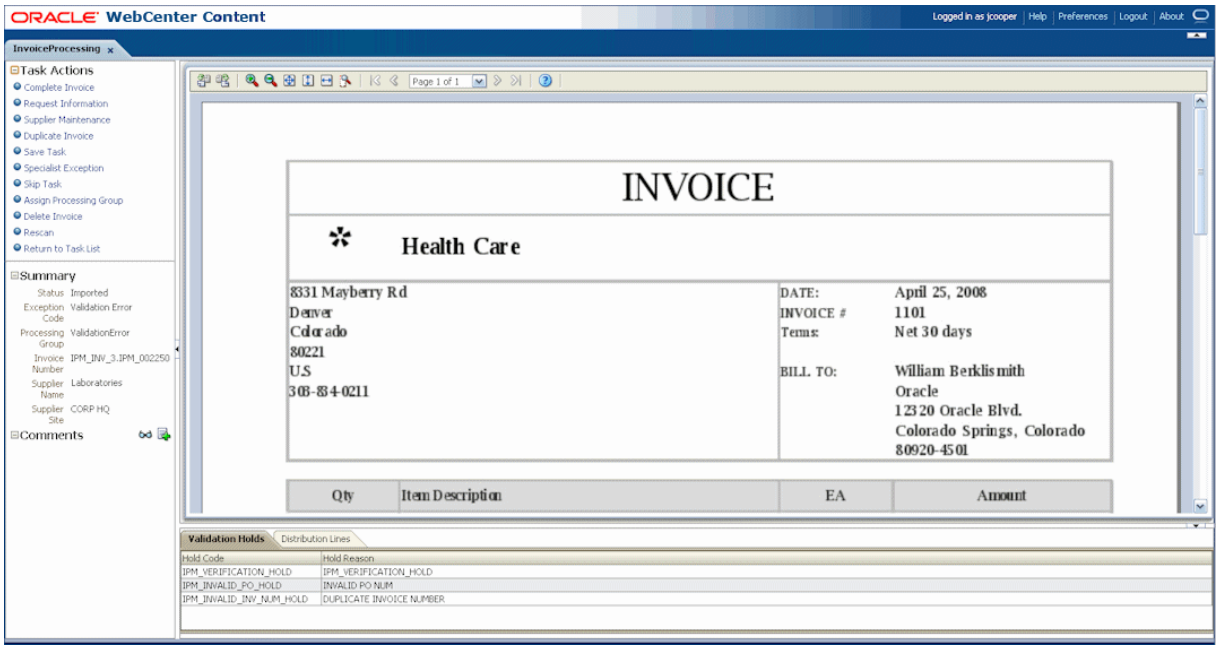
Table B–11 Column Description for AXF_ACTIONS Table

Column	Description
DISPLAY_TEXT	Specifies the name of the action (link, for example) in the pane.
COMMAND_NAMESPACE	Specifies the command called as a result of the action, as defined in the AXF_COMMANDS Table.
MENU_ORDER	Specifies the display order of the action in the pane.
ELEMENT_TYPE	Specifies how to render the action on the page, where LINK displays an HTML link.
ACTION_HANDLER	Determines who/what handles the action. COMMAND is the currently available handler. **Note:** If left (null), this value defaults to COMMAND.
REQUIRES_ CONVERSATION	Specifies whether the action requires a conversation ID to already be initialized.
MENU_ID	Specifies the ID from the AXF_ACTION_MENU Table and defines the menu in which the action is displayed.
ACTION_ID	Defines the action's unique numeric identifier.
USE_POPUP	Reserved for future use.

B.1.6.2 Example Implementation

The tables that follow provide an example AXF_ACTIONS Table.

Figure B–2 Task Viewer Page with Task Actions, Summary, and Comments Enabled



Fields not shown in Table B–12:

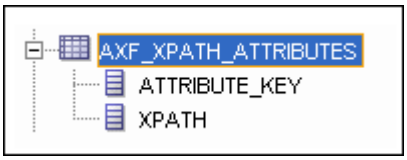
- ACTION_HANDLER=COMMAND
- REQUIRES_CONVERSATION=TRUE

Table B–12 Example AXF_ACTIONS Table

DISPLAY_TEXT	COMMAND_ NAMESPACE	MENU_ ORDER	ELEMENT_ TYPE	MENU_ID	ACTION_ID
Invoice Approval	InvoiceApprovalEdit	1	LINK	0	AXF_ACTIONS_SEQ.NEXTVAL
Return to Task List	ReleaseTask	0	LINK	0	AXF_ACTIONS_SEQ.NEXTVAL
Skip Task	SkipTask	0	LINK	0	AXF_ACTIONS_SEQ.NEXTVAL
Complete Invoice	CompleteTask	0	LINK	0	AXF_ACTIONS_SEQ.NEXTVAL

B.1.7 AXF_XPATH_ATTRIBUTES Table

This table defines the XPATH attributes used in the AXF framework. This XPATH is mainly defined for payload elements.



B.1.7.1 Column Description

Table B–13 Column Description for AXF_XPATH_ATTRIBUTES Table

Column	Description
ATTRIBUTE_KEY	Attribute key referenced in the Parameter Value column in the AXF_SOLUTION_PARAMETERS Table.

Table B–13 (Cont.) Column Description for AXF_XPATH_ATTRIBUTES Table

Column	Description
XPATH	XPATH expression used to locate the value in the payload.

B.1.7.2 Example Implementation

This example follows an XPATH attribute specified for an AssignProcessingGroupEdit command in the AXF_SOLUTION_PARAMETERS table. The PARAMETER_VALUE column contains an XPATH: prefix, indicating that the attribute value comes from the AXF_XPATH_ATTRIBUTES table.

Fields not shown: SOLUTION_NAMESPACE=InvoiceProcessing

Table B–14 Example AXF_SOLUTION_PARAMETERS Table

COMMAND_NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
AssignProcessingGroupEdit	oracle.imaging.axf.web.EnumerationPicker	ATTRIBUTE_NAME	XPATH:InvoiceProcessing_ProcessingGroup

In the AXF_XPATH_ATTRIBUTES table that follows, the corresponding XPATH column displays the XPATH expression used to locate the value in the payload.

Table B–15 Example AXF_XPATH ATTRIBUTES Table

ATTRIBUTE_KEY	XPATH
InvoiceProcessing_ProcessingGroup	//task:processingGroup

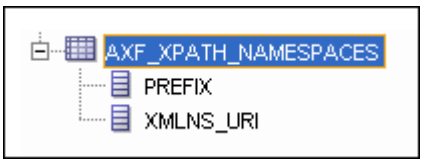
In the AXF_NAMESPACES table that follows, the XMLNS_URI column displays where within the XML file to locate the processingGroup task information.

Table B–16 Example AXF_XPATH_NAMESPACES Table

Prefix	XMLNS_URI
task	http://xmlns.oracle.com/bpel/workflow/task

B.1.8 AXF_XPATH_NAMESPACES Table

The following table defines the namespaces used for the XPATH attributes. It links to the AXF_XPATH_ATTRIBUTES Table.



B.1.8.1 Column Description

Table B–17 Column Description for AXF_XPATH_NAMESPACES Table

Column	Description
PREFIX	The namespace prefix used in the XPATH.
XMLNS_URI	Provides a unique identifier.

B.1.8.2 Example Implementation

Table B–18 Example AXF_XPATH_NAMESPACES Table

Prefix	XMLNS_URI
task	http://xmlns.oracle.com/bpel/workflow/task
documentContent	http://xmlns.oracle.com/imaging/axf/documentContentTypes
solution	http://xmlns.oracle.com/imaging/axf/solutionTypes
invoiceProcessing	http://xmlns.oracle.com/imaging/axf/InvoiceProcessing
xml	http://www.w3.org/XML/1998/namespace

B.2 AXF User Components for the BPEL Imaging Solution

This section covers the following topics:

- Section B.2.1, "Solution Editor"
- Section B.2.2, "Task List Web Tool"
- Section B.2.3, "Task Viewer Web Tool"
- Section B.2.4, "Enumeration Picker Web Tool"
- Section B.2.5, "Identity Picker Web Tool"
- Section B.2.6, "Account Distribution Coding Form (Oracle E-Business Suite Only)"

B.2.1 Solution Editor

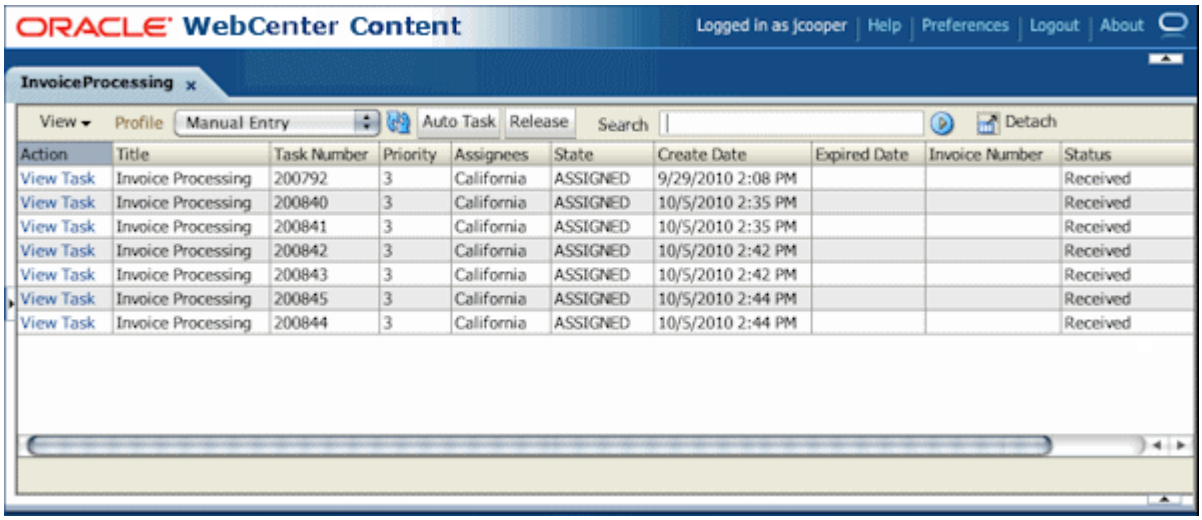
After initial configuration of a BPEL imaging solution (using a solution accelerator or the tables described in this appendix), you can use the Solution Editor in Imaging to quickly make changes to the user interface components. For information about the Solution Editor, see "Solutions Editor Page" in *Oracle Fusion Middleware Administrator's Guide for Oracle Imaging and Process Management*.

For example, use the Solution Editor for such tasks as:

- Changing the profiles (BPEL views) available to users in the Task List
- Changing the task actions available to users in the Task Viewer
- Changing the metadata items and bottom panel tabs displayed in the Task Viewer
- Specifying the enumeration values available for specified enumerations on the Enumeration Picker page

B.2.2 Task List Web Tool

The AXF Task List web tool is a reusable web interface for displaying human workflow tasks controlled by an AXF solution.



Task List Features

- The **Profile** list uses standard BPEL views to restrict the task list view based on user/group, BPEL Process versions, and BPEL payload attribute values. Use the BPM Worklist application to create views and share them with other users or groups. Use the BPEL Imaging Solution Editor's **General** tab to edit the profile list, assign a default profile, and identify the uses of comment blocks within the solution. For more information, see "Solutions Editor Page: General Tab" in *Oracle Fusion Middleware Administrator's Guide for Oracle Imaging and Process Management*.
- Users can select a task from the table by clicking its **View Task** link, which retrieves the workflow task from a specified BPEL server and process, claims it and displays it in the Task Viewer. After users complete the selected task, they return to the Task List. You can configure the number of tasks to return in the Task List at a time.
- When autotask mode is selected, the AXF solution automatically claims and opens tasks as users complete them, until all of a user's tasks have been processed or the user chooses to stop processing tasks by returning to the Task List. Users activate autotask mode by clicking **Auto Task**.
- Users can release an assigned task by clicking the **Release** button. The task is then released back into the pool of available tasks. If the user clicks Release but did not previously acquire the selected task, a message indicates that the task cannot be released.
- Users can sort tasks by clicking a column heading. They can also search for specific tasks using the **Search** field. For details, see the *Oracle WebCenter User's Guide for Application Adapters*.

B.2.2.1 Task List Parameters

Table B–19 Task List Parameters in AXF_SOLUTION_PARAMETERS Table

Parameter	Description
CMD_OPEN_TASK_BUTTON	Specify a COMMAND_NAMESPACE to execute when a user clicks the View Task link on the Task List web page.
CMD_AUTO_TASK_BUTTON	Specify a COMMAND_NAMESPACE to execute when a user clicks the Auto Task button on the Task List web page.

Table B–19 (Cont.) Task List Parameters in AXF_SOLUTION_PARAMETERS Table

Parameter	Description
NO_OF_LINES	Specify the maximum number of tasks to display before displaying multiple pages. To configure the number of tasks to return in the Task List, use the TASKLIST_RANGESIZE parameter in the AXF_SOLUTION_ATTRIBUTES Table.
VIEW_LIST	Specify the list of views (defined in the Human workflow system) displayed to users in the Profile field.
DEFAULT_VIEW	Specify the default profile. If used with the VIEW_LIST parameter, this value must also be listed in the VIEW_LIST value.

B.2.2.2 Example Implementation

This example defines the StartInvoiceProcessing command for the Invoice Processing solution. The first row uses the Redirect Command to display the task list. The remaining rows define the task list's behavior.

Fields not shown: SOLUTION_NAMESPACE=InvoiceProcessing

Table B–20 Example Task List Parameters in AXF_SOLUTION_PARAMETERS Table

COMMAND_NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
StartInvoiceProcessing	oracle.imaging.axf.commands.bpel.RedirectCommand	REDIRECT_URL	taskflow://WEB-INF/taskflows/axf-tasklist-tfd.xml#axf-tasklist-tfd
StartInvoiceProcessing	oracle.imaging.axf.web.backing.TaskList	CMD_OPEN_TASK_BUTTON	OpenTask
StartInvoiceProcessing	oracle.imaging.axf.web.backing.TaskList	CMD_AUTO_TASK_BUTTON	AutoOpenTask
StartInvoiceProcessing	oracle.imaging.axf.web.backing.TaskList	DEFAULT_VIEW	(null)
StartInvoiceProcessing	oracle.imaging.axf.web.backing.TaskList	SHOW_INBOX	FALSE
StartInvoiceProcessing	oracle.imaging.axf.web.backing.TaskList	VIEW_LIST	North Invoice Processing Group, South Invoice Processing Group, East Invoice Processing Group, West Invoice Processing Group, My Holds, Exceptions

B.2.3 Task Viewer Web Tool

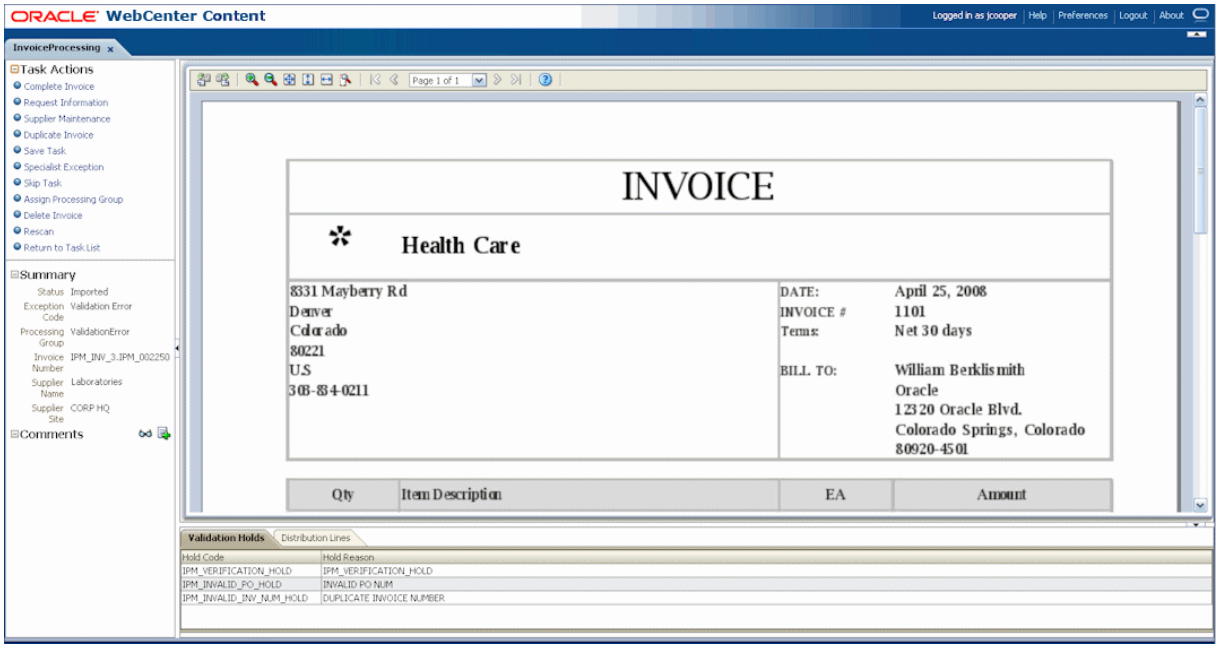
The AXF Task Viewer web tool is a reusable web interface that displays the content associated with a Human Workflow Task. In a typical configuration, a business user displays the Task Viewer on one monitor, and keys values shown in the image into a business application on another monitor.

Once configured, use the BPEL Imaging Solution Editor to make the following changes to the Task Viewer:

- Edit the display label of the Action menu, define the menu item order, hide or display menu items and edit menu item labels, as described in "Solutions Editor Page: Actions Tab" in *Oracle Fusion Middleware Administrator's Guide for Oracle Imaging and Process Management*.
- Edit the metadata block label, change the metadata item order, hide or display metadata items, edit metadata item labels, and change the dynamic data table table (bottom panel) display, as described in "Solutions Editor Page: Task Viewer

Tab" in *Oracle Fusion Middleware Administrator's Guide for Oracle Imaging and Process Management*.

You can also customize the web page through database configuration.



Task Viewer Features

- Users view image documents in the Image Viewer pane, using either the basic or advanced Imaging viewer mode. The Task Viewer uses the Imaging viewer tool to render image documents.
- If configured, users can select actions in the side Task Actions pane, which invoke a solution's AXF commands. You enable the Task Actions pane in the AXF_ACTION_MENU Table, configure the action links to invoke AXF commands in the AXF_ACTIONS Table, and the commands themselves in the AXF_COMMANDS Table. If autotask mode is selected, users disable it by returning to the Task List, typically by clicking a **Return to Task List** link in the Task Actions pane.
- If configured, users can view a Summary section that displays metadata values about the task. You configure these items for display in the AXF_METADATA_BLOCKS Table and AXF_METADATA_ATTRIBUTES Table. You can also configure the section's title and the task payload values displayed.
- If configured, users can view a Comments section that displays comment fragments and provides icons for displaying full comments or adding them for the task. You configure comments for display in the AXF_METADATA_BLOCKS Table.
- If configured, users can view a dynamic data table (bottom tabs) from the BPEL payload XML, such as General Ledger lines for an invoice processing solution. You configure the table in the AXF_METADATA_BLOCKS Table and its data lines in the AXF_METADATA_ATTRIBUTES Table. For more information, see Section 6.4.10.

Configuring the Task Viewer

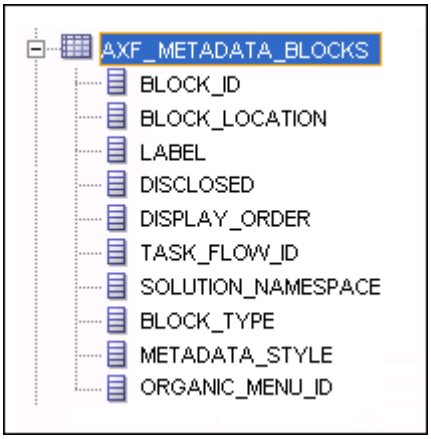
You configure the Task Viewer in the following tables:

- Section B.1.4, "AXF_SOLUTION_PARAMETERS Table"

- Section B.1.5, "AXF_ACTION_MENU Table"
- Section B.1.6, "AXF_ACTIONS Table"
- Section B.2.3.1, "AXF_METADATA_BLOCKS Table"
- Section B.2.3.2, "AXF_METADATA_ATTRIBUTES Table"

B.2.3.1 AXF_METADATA_BLOCKS Table

This table defines the task viewer itself and its sections for display on the Task Viewer page.



B.2.3.1.1 Column Description

Table B–21 Column Description for AXF_METADATA_BLOCKS Table

Column	Description
BLOCK_ID	Identifies the row in the database. Links to the AXF_METADATA_ATTRIBUTES Table.
BLOCK_LOCATION	Specifies where the block displays on the Task Viewer page. *LEFT_SIDEBAR* displays a left sidebar pane.
LABEL	Defines the pane's label (for example, *Summary* or *Comments*).
DISCLOSED	Specify TRUE to display the block in expanded form. If you specify FALSE, the item is displayed but collapsed.
DISPLAY_ORDER	Specifies the order in which the block is displayed.
TASK_FLOW_ID	Specifies the task flow on which to display the metadata block (for example, *axf-taskviewer-tfd* or *axf-tasklist-tfd*).
SOLUTION_NAMESPACE	Specifies the AXF solution name.
BLOCK_TYPE	Specifies the type of values contained in the block (for example, METADATA or COMMENT).

Table B–21 (Cont.) Column Description for AXF_METADATA_BLOCKS Table

Column	Description
METADATA_STYLE	Specifies how metadata is used. You can specify `null` (not used) or `table`. Specify `table` to display a table of dynamic data from the BPEL payload XML in the bottom tabs. See Section 6.4.10.1. Note that this block displays only if there is data available. To hide the bottom panel, see Section 6.4.10.2. To configure a dynamic table, the following values must be specified: BLOCK_LOCATION = `BOTTOM_PANEL` BLOCK_TYPE = `METADATA` METADATA_STYLE = `table`
ORGANIC_MENU_ID	Reserved for future use.

B.2.3.1.2 Example Implementation

This table displays the Summary and Comments section on the Task Viewer page.

Columns not shown: DISCLOSED=TRUE

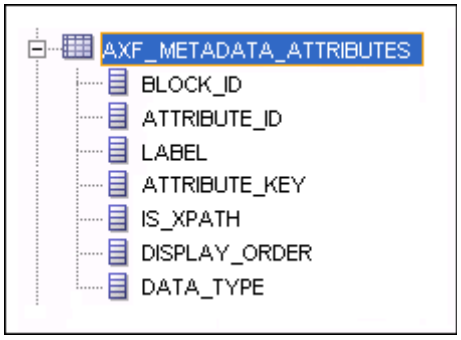
ORGANIC_MENU_ID=null

Table B–22 Example AXF_METADATA_BLOCKS Table

BLOCK_ ID	BLOCK_LOCATION	LABEL	DISPLAY_ ORDER	TASK_FLOW_ID	SOLUTION_ NAMESPACE	BLOCK_ TYPE	METADATA _STYLE
1	LEFT_SIDEBAR	Summary	1	axf-taskviewer-tfd	InvoiceProcessing	METADATA	null
2	LEFT_SIDEBAR	Comments	2	axf-taskviewer-tfd	InvoiceProcessing	COMMENT	null
3	BOTTOM_PANEL	GL Lines	3	axf-taskviewer-tfd	InvoiceProcessing	METADATA	TABLE

B.2.3.2 AXF_METADATA_ATTRIBUTES Table

This table defines the labels and values to show in metadata sections specified in the AXF_METADATA_BLOCKS Table. It also defines how attribute values are retrieved for display using XPath attributes.



B.2.3.2.1 Column Description

Table B–23 Column Description for AXF_METADATA_ATTRIBUTES Table

Column	Description
BLOCK_ID	References the AXF_METADATA_BLOCKS Table in which to display metadata labels and values.
ATTRIBUTE_ID	Primary key for the metadata attribute.
LABEL	Specifies the metadata label displayed to users in the metadata section (for example, *Status* in a Summary section). When a dynamic data table is specified, this label becomes a column header.
ATTRIBUTE_KEY	This attribute is updated in the BPEL task when a user clicks the OK button on the Task Viewer page. The attribute value is either a system attribute name (see Section B.3.6.2) or prefixed with XPATH, which uses the XPath table to find the value. Only XPATH is supported for a dynamic data table. If the IS_XPATH value is TRUE, the attribute_key value comes from the AXF_XPATH_ATTRIBUTES Table and is the XPATH to update the value in the task payload.
IS_XPATH	If TRUE, the attribute value comes from the XPATH in the BPEL payload. If FALSE, the value comes from system attributes or flex field attributes.
DISPLAY_ORDER	Specifies the order in which the metadata label/value display in the metadata section. For a dynamic data table, specifies the column order.
DATA_TYPE	Specifies the metadata item's data type (for example, String). This value must be STRING for a dynamic data table.

B.2.3.2.2 Example Implementation

Table B–24 defines metadata labels and values displayed in a Task Viewer's Summary section, as shown in Figure B–2. For another example, see Section 6.4.10.

Columns not shown:

DATA_TYPE=String

Table B–24 Example AXF_METADATA_ATTRIBUTES Table for Summary Section

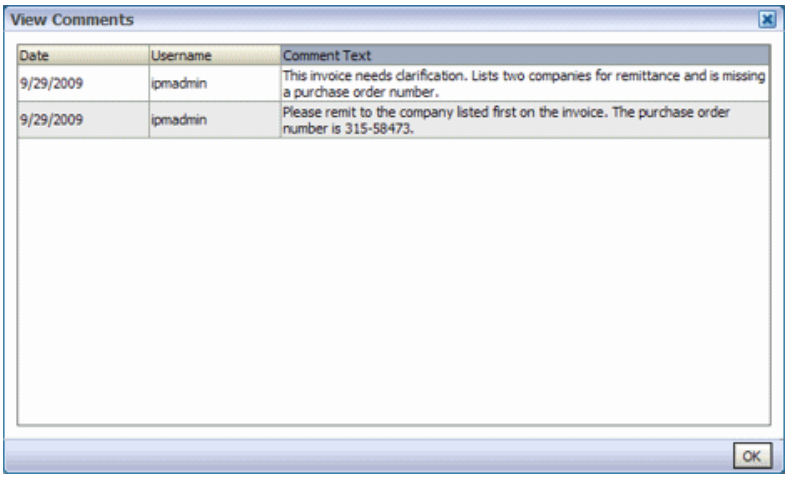
BLOCK_ID	ATTRIBUTE_ID	LABEL	ATTRIBUTE_KEY	IS_XPATH	DISPLAY_ORDER
1	1	Status	InvoiceProcessing_Status	TRUE	0
1	2	Exception Code	InvoiceProcessing_ExceptionCode	TRUE	1
1	3	Processing Group	InvoiceProcessing_ProcessingGroup	TRUE	2
1	4	Invoice Number	InvoiceProcessing_InvoiceNumber	TRUE	3
1	5	Supplier Name	InvoiceProcessing_SupplierName	TRUE	4
1	6	Supplier Site	InvoiceProcessing_SupplierSiteName	TRUE	5

B.2.3.3 Comments

The Comments pane enables users to view and enter comments related to the human task during the transaction's processing. You configure comments in the AXF_METADATA_BLOCKS Table. Also see Section 1.3.3.5.

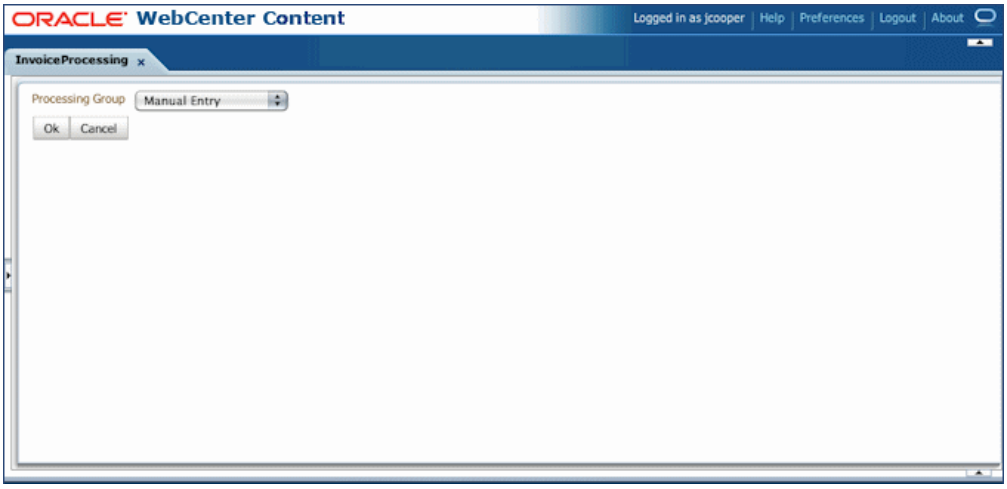
You can specify whether the Comments field is displayed on the **General** tab of Imaging's Solution Editor, as described in "Solutions Editor Page: General Tab" in *Oracle Fusion Middleware Administrator's Guide for Oracle Imaging and Process*

Management. You can display comments on the Task Viewer, Identity Picker, and Enumeration Picker pages.



B.2.4 Enumeration Picker Web Tool

The Enumeration Picker web tool enables users to select a choice from a list of values configured in AXF tables. For example, the Enumeration Picker shown in the graphic that follows displays a Processing Group dropdown field containing North, South, East, and West values.



After the user selects a value, the value is updated into the BPEL payload before the configured command executes, typically a command to return to the Task Viewer or to complete the command.

You configure the Enumeration Picker in the following tables:

- Define configuration parameters for the picker in the AXF_SOLUTION_PARAMETERS table (see Table B–25).
- Define the picker in the AXF_ENUM_TYPES Table.
- Define the picker's values in the AXF_ENUM_ITEMS Table.

After configuring enumerations, you can edit them in Imaging's Solution Editor. On the Enumeration tab, add, delete, and reorder enumerations, or specify their available

values, as described in "Solutions Editor Page: Enumerations Tab" in *Oracle Fusion Middleware Administrator's Guide for Oracle Imaging and Process Management*.

B.2.4.1 Enumeration Picker Parameters

Table B–25 Enumeration Picker Parameters in AXF_SOLUTION_PARAMETERS Table

Parameter	Description
LOV_REFERENCE	This list of values reference links to the AXF_ENUM_TYPES Table, whose ID value links to the AXF_ENUM_ITEMS Table, where all picker values are stored.
ATTRIBUTE_NAME	This attribute is updated in the BPEL task when a user clicks the OK button on the Enumeration Picker web page. The attribute value is a constant; see Section B.3.6.2. If the value has an XPATH: prefix, the value comes from the AXF_XPATH_ATTRIBUTES Table and it is the XPATH to update the value in the task payload.
CMD_ON_CANCEL	Specify the command (COMMAND_NAMESPACE) to execute when a user clicks the Cancel button on the Enumeration Picker page.
CMD_ON_OK	Specify the command (COMMAND_NAMESPACE) to execute when a user clicks the OK button on the Enumeration Picker page.
ATTRIBUTE_LABEL	Specify the label name to display on the web page for attributes to be updated in the BPEL task.
DEFAULT_VALUE	Specify a default value for the picker. If no default is specified, a blank value displays.
DEFAULT_ALWAYS	Specify TRUE to always show the value specified in the DEFAULT_VALUE parameter when displaying the Enumeration Picker, even if another value was previously selected. Otherwise, specify FALSE.

Example Implementation

This example shows an enumeration picker referenced for selecting the processing group.

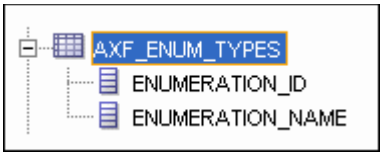
Fields not shown: SOLUTION_NAMESPACE=InvoiceProcessing

Table B–26 Example Enumeration Picker Parameters in AXF_SOLUTIONS Table

COMMAND_NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
AssignProcessingGroupEdit	oracle.imaging.axf.web.backing.EnumerationPicker	LOV_REFERENCE	ProcessingGroups
AssignProcessingGroupEdit	oracle.imaging.axf.web.backing.EnumerationPicker	ATTRIBUTE_LABEL	Processing Group
AssignProcessingGroupEdit	oracle.imaging.axf.web.backing.EnumerationPicker	CMD_ON_OK	AssignProcessingGroupComplete
AssignProcessingGroupEdit	oracle.imaging.axf.web.backing.EnumerationPicker	ATTRIBUTE_NAME	XPATH:InvoiceProcessing_ProcessingGroup
AssignProcessingGroupEdit	oracle.imaging.axf.web.backing.EnumerationPicker	CMD_ON_CANCEL	OpenTask
AssignProcessingGroupEdit	oracle.imaging.axf.web.backing.EnumerationPicker	DEFAULT_VALUE	North
AssignProcessingGroupEdit	oracle.imaging.axf.web.backing.EnumerationPicker	DEFAULT_ALWAYS	FALSE

B.2.4.2 AXF_ENUM_TYPES Table

This table defines Enumeration Pickers.



Column Description

Table B–27 Column Description for AXF_ENUM_TYPES Table

Column	Description
ENUMERATION_ID	Specify an ID for the enumeration picker.
ENUMERATION_NAME	Specify a name for the enumeration picker configuration.

Example Implementation

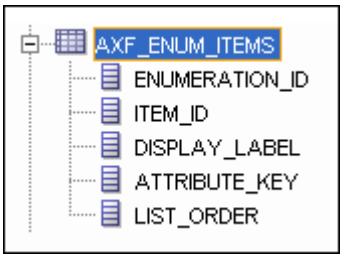
This example defines enumeration pickers for the Invoice Processing solution. You define each picker's items in the AXF_ENUM_ITEMS Table.

Table B–28 Example AXF_ENUM_TYPES Table

ENUMERATION_ID	ENUMERATION_NAME
1	ProcessingGroups
2	SupplierMaintenanceCodes
3	SpecialistExceptionCodes
4	RescanCodes

B.2.4.3 AXF_ENUM_ITEMS Table

This table defines a specified Enumeration Picker's values.



Column Description

Table B–29 Column Description for AXF_ENUM_ITEMS Table

Columns	Description
ENUMERATION_ID	Specify the picker's ID, as defined in the AXF_ENUM_TYPES Table.
ITEM_ID	Specify an ID for the picker item.
DISPLAY_LABEL	Specify the item name to display in the picker field.

Table B–29 (Cont.) Column Description for AXF_ENUM_ITEMS Table

Columns	Description
ATTRIBUTE_KEY	Specify the literal value to store in the payload. This value is often the same as the DISPLAY_LABEL's value, but can differ.
LIST_ORDER	Specify the order in which to list the value in the picker field.

Example Implementation

This example defines the items for the ProcessingGroups, SupplierMaintenanceCodes, SpecialistExceptionCodes, and RescanCodes enumeration pickers defined in the AXF_ENUM_TYPES Table.

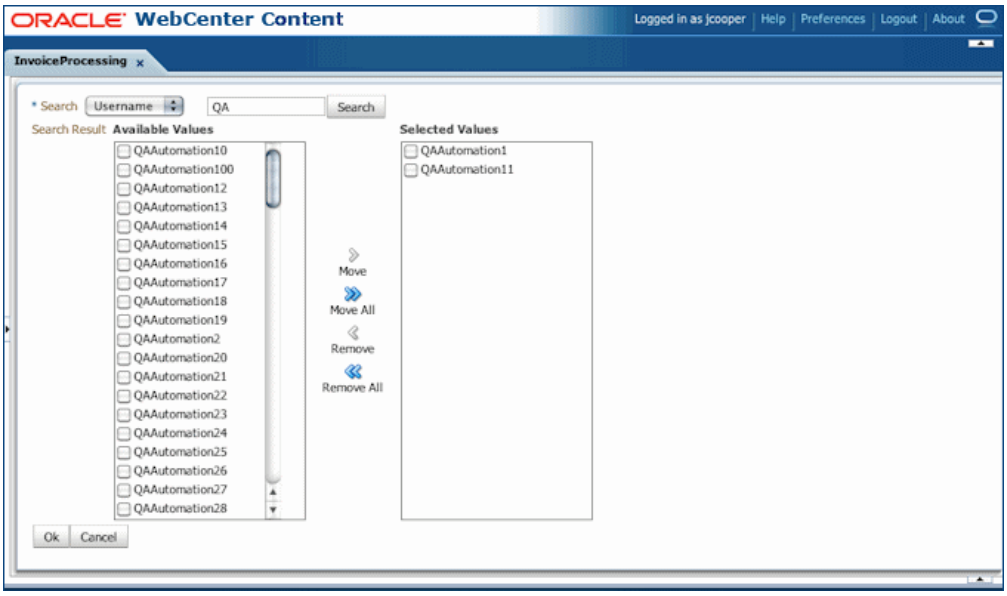
Table B–30 Example AXF_ENUM_ITEMS Table

ENUMERATION_ID	ITEM_ID	DISPLAY_LABEL	ATTRIBUTE_KEY	LIST_ORDER
1	1	North	North	1
1	2	South	South	2
1	3	East	East	3
1	4	West	West	4
2	5	No Supplier	No Supplier	1
2	6	No Supplier Site	No Supplier Site	2
3	7	Duplicate Invoice	Duplicate Invoice	1
3	8	Invalid Invoice Number	Invalid Invoice Number	2
3	9	No PO	No PO	3
3	10	Invalid PO	Invalid PO	4
3	11	PO Overbill	PO Overbill	5
4	12	Poor Image Quality	Poor Image Quality	1
4	13	Pages Out of Order	Pages Out of Order	2
4	14	Pages Missing	Pages Missing	3
4	15	Other	Other	4

B.2.5 Identity Picker Web Tool

The Identity Picker web tool enables users to select one or more users or groups from an identity store configured for BPEL. Typically, a related action is taken after choosing an identity; for example, a task is assigned or delegated. The action to take after selecting an identity is configured in the AXF_SOLUTION_PARAMETERS Table.

Note: The command updates the task payload when the user clicks OK. The BPEL process is responsible for using this information to delegate the task.



B.2.5.1 Identity Picker Parameters

Table B–31 Identity Picker Parameters in AXF_SOLUTION_PARAMETERS Table

Parameter Key	Description
CMD_ON_CANCEL	Specify the command (COMMAND_NAMESPACE) to execute when a user clicks the Cancel button on the Identity Picker page.
CMD_ON_OK	Specify the command (COMMAND_NAMESPACE) to execute when a user clicks the OK button on the Identity Picker page.
IDENTITY_FILTER	Define how the identity picker searches, where: ▪ USER: The picker searches for user information defined in BPEL. ▪ GROUP: The picker searches for group information defined in BPEL. **Note:** Specify USER or GROUP for a command. To allow both search types, create an additional command that uses the other type to open the identity picker. For example, you might create AssignByGroup and AssignByUser commands.
IDENTITY_FILTER_ATTRIBUTE	Define a wildcard search for the identity picker, where: ▪ TRAILING: Adds an asterisk (*) to the end of the search value, as in **`jc*`** ▪ LEADING: Adds an asterisk (*) to the start of the search value, as in **`*cooper`** ▪ BOTH (default): Adds an asterisk (*) to the start and end of the search value, as in **`*jc*`** ▪ NONE: No asterisks are added to the search value. This setting is useful for implementations where wildcard characters may cause searches to take a long time. **Note:** A null value for this parameter results in an asterisk added to the start and end of search values (the default setting).

Table B–31 (Cont.) Identity Picker Parameters in AXF_SOLUTION_PARAMETERS Table

Parameter Key	Description
IDENTITY_ATTRIBUTE	This attribute is updated in the BPEL task when a user clicks OK on the Identity Picker page. The Attribute value is a constant as defined under System Attributes; see Section B.3.6.2. If the value has an XPATH: prefix, then the value comes from the AXF_XPATH_ATTRIBUTES Table and it is the XPATH to update the value in the task payload.

B.2.5.2 Example Implementation

This example shows an InvoiceApprovalEdit command that searches for user information stored in BPEL, and updates the BPEL task using an XPATH variable.

Fields not shown: SOLUTION_NAMESPACE=InvoiceProcessing

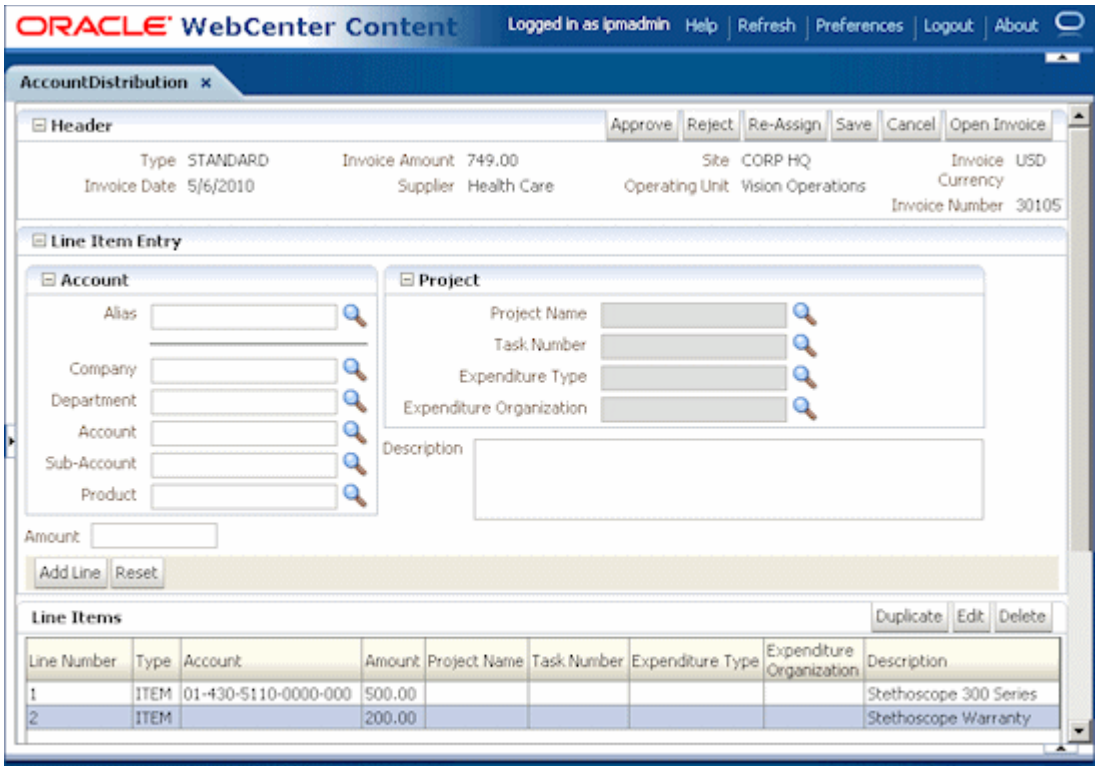
Table B–32 Example Identity Picker Parameters in AXF_SOLUTION_PARAMETERS table

COMMAND_ NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
InvoiceApprovalEdit	oracle.imaging.axf.web.backing.IdentityPicker	IDENTITY_FILTER	USER
InvoiceApprovalEdit	oracle.imaging.axf.web.backing.IdentityPicker	IDENTITY_FILTER_ ATTRIBUTE	NONE
InvoiceApprovalEdit	oracle.imaging.axf.web.backing.IdentityPicker	CMD_ON_CANCEL	OpenTask
InvoiceApprovalEdit	oracle.imaging.axf.web.backing.IdentityPicker	IDENTITY_ ATTRIBUTE	XPATH:InvoiceProcessing_ InvoiceApprovalAssignment
InvoiceApprovalEdit	oracle.imaging.axf.web.backing.IdentityPicker	CMD_ON_OK	InvoiceApprovalComplete

B.2.6 Account Distribution Coding Form (Oracle E-Business Suite Only)

The AXF Account Distribution Coding Form is a reusable web interface for entering and selecting detailed account distribution information for an invoice. Figure B–3 shows the coding form used for non-PO invoices needing line item information. In this case, authorized users access this page from an Account Distribution task list, then key in account distribution codes.

Figure B–3 Example Account Distribution Coding Form



Important Points About the Account Distribution Coding Form

- The coding form is available for Oracle E-Business Suite only.
- The coding form performs lookups against the Oracle E-Business Suite system. A JDBC Data Source must be set up on the Oracle WebCenter Imaging WebLogic Managed Server to enable the lookups. Refer to the solution accelerator documentation for information on setting up this JDBC Data Source.
- The coding form is invoked via a solution accelerator, as described in Section 6.2. To use the coding form, you must deploy and configure a solution, along with supporting AXF commands and solution parameters.
- The buttons in the upper right portion of the form (Approve, Reject, Re-Assign, Save, Cancel, and Open Invoice) are configurable via the AXF tables.

The sections listed below provide general configuration information for the coding form.

- Section B.2.6.1, "AXF_LOOKUPS Script for Coding Form Lookup Elements"
- Section B.2.6.2, "Example Account Distribution Coding Form Solution"
- Section B.2.6.3, "Example Supporting Commands for the Account Distribution Coding Form"
- Section B.2.6.4, "Example OpenTask and AutoOpenTask Commands"

B.2.6.1 AXF_LOOKUPS Script for Coding Form Lookup Elements

The **AXF_LOOKUPS**.sql script creates lookup elements for the coding form. You must execute it as the APPS user in the Oracle E-Business Suite database under the APPS schema.

1. Connect to the APPS schema in the Oracle E-Business Suite database using the appropriate APPS user.
2. If needed, uninstall the existing coding form lookups package.

   You only need to perform this step if you are running an update/patch to an existing system that already has an AXF_LOOKUPS coding form lookups package installed.

   ```
   DROP TYPE AXF_LOOKUPS_SEGMENT_VALUE_COL;
   show errors
   DROP TYPE AXF_LOOKUPS_SEGMENT_OBJ;
   show errors
   DROP PACKAGE AXF_LOOKUPS;
   show errors
   ```

3. Execute the AXF_LOOKUPS script.

   The script is delivered as part of the Oracle E-Business Suite adapter. It is located in one of the following locations, depending on Oracle E-Business Suite release:

   - *MW_HOME/ECM_HOME*/axf/adapters/ebs/R11/
   - *MW_HOME/ECM_HOME*/axf/adapters/ebs/R12/

   Execute the script by entering:

   ```
   @AXF_LOOKUPS.sql
   ```

B.2.6.2 Example Account Distribution Coding Form Solution

Table B–33 illustrates an example solution for Account Distribution.

Table B–33 Example AXF_SOLUTIONS Table for Account Distribution Coding Form

SOLUTION_NAMESPACE	SOLUTION_CONTEXT
AccountDistribution	ejb.AxfCommandMediator#oracle.imaging.axf.service.AxfCommandMediatorRemote

B.2.6.3 Example Supporting Commands for the Account Distribution Coding Form

Table B–34 lists example AXF commands that support the Account Distribution coding form solution.

Table B–34 Example AXF_COMMANDS Table for Account Distribution Coding Form

SOLUTION_ NAMESPACE	COMMAND_CLASS	COMMAND_NAMESPACE	Description
AccountDistribution	oracle.imaging.axf.commands.bpel.CompleteTaskCommand	AccountDistributionApprove	Creates a command for the coding form's Approve button.
AccountDistribution	oracle.imaging.axf.commands.bpel.ReleaseTaskCommand	AccountDistributionCancel	Creates a command for the coding form's Cancel button.
AccountDistribution	oracle.imaging.axf.commands.bpel.CompleteTaskCommand	AccountDistributionReject	Creates a command for the coding form's Reject button.
AccountDistribution	oracle.imaging.axf.commands.bpel.CompleteTaskCommand	AccountDistributionRequestInfo	Creates a command for the coding form's Request Information button.
AccountDistribution	oracle.imaging.axf.commands.system.RedirectCommand	AssignUser	Creates a command for the coding form's Re-Assign button.
AccountDistribution	oracle.imaging.axf.commands.bpel.CompleteTaskCommand	AssignUserComplete	Creates a command to redirect users to the Identity Picker.

Table B–34 (Cont.) Example AXF_COMMANDS Table for Account Distribution Coding Form

SOLUTION_ NAMESPACE	COMMAND_CLASS	COMMAND_NAMESPACE	Description
AccountDistribution	oracle.imaging.axf.commands.bpel.AutotaskCommand	AutoOpenTask	Creates a command for the autotask operation on the Task List.
AccountDistribution	oracle.imaging.axf.commands.bpel.OpenTaskCommand	OpenTask	Creates an OpenTask command link on the Task List.
AccountDistribution	oracle.imaging.axf.commands.system.RedirectCommand	StartAccountDistribution	Creates a command to redirect users from the Task List to the coding form.

B.2.6.4 Example OpenTask and AutoOpenTask Commands

Options for the OpenTask and AutoOpenTask commands are configured for the coding form's solution. Table B–35 shows an example of OpenTask options.

Fields not shown: SOLUTION_NAMESPACE and COMMAND_NAMESPACE

Table B–35 Example AXF_SOLUTION_PARAMETERS Table for Account Distribution Coding Form

CONFIGURATION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE	Description
oracle.imaging.axf.web.backing.Coding	CMD_APPROVE	AccountDistributionApprove	COMMAND_ NAMESPACE to execute when a user clicks the coding form's Approve button.
oracle.imaging.axf.web.backing.Coding	CMD_CANCEL	AccountDistributionCancel	COMMAND_ NAMESPACE to execute when a user clicks the coding form's Cancel button.
oracle.imaging.axf.web.backing.Coding	CMD_REASSIGN	AssignUser	COMMAND_ NAMESPACE to execute when a user clicks the coding form's Re-Assign button.
oracle.imaging.axf.web.backing.Coding	CMD_REJECT	AccountDistributionReject	COMMAND_ NAMESPACE to execute when a user clicks the coding form's Reject button.
oracle.imaging.axf.web.backing.Coding	CMD_REQUEST_INFO	AccountDistributionRequestInfo	COMMAND_ NAMESPACE to execute when a user clicks the coding form's Request Information button.
oracle.imaging.axf.web.backing.Coding	DISABLE_PROJECT	FALSE	A flag to disable the Project lookups box on the coding form.
oracle.imaging.axf.web.backing.Coding	READ_ONLY	TRUE	Specify if users can enter and edit values on the coding form (FALSE, the default) or view them as read-only values (TRUE).

B.3 AXF Commands for the BPEL Imaging Solution

BPEL AXF commands include:

- Section B.3.1, "Open Task Command"

- Section B.3.2, "Autotask Command"
- Section B.3.3, "Release Task Command"
- Section B.3.4, "Complete Task Command"
- Section B.3.5, "Redirect Command"
- Section B.3.6, "Update Task Command (BPEL)"
- Section B.3.7, "Update Task From Procedure Command"
- Section B.3.8, "Terminate Conversation Command"
- Section B.3.9, "Validate Task Command"

Note: You can also deploy custom commands and chained commands to execute through AXF. See Section 6.4.8 and Section 6.4.9.

B.3.1 Open Task Command

This command acquires a task from BPEL (human work flow) for a given task ID; the specific task is likely selected from the task list. If the task can be acquired by the user, the command obtains the details of the task and displays the specified web page.

B.3.1.1 Open Task Command Parameters

Table B–36 lists configuration parameters for this command. Use these parameters in the AXF_SOLUTION_PARAMETERS Table to configure commands.

Table B–36 Parameters for OpenTask Command

Parameter Key	Description
TASK_VIEW_URL	This task flow is returned in the response command upon executing this command. The value for the TASK_VIEW_URL parameter uses one of the following strings to represent task flows. Each string can be thought of as a special URL where *taskflow:* is the protocol instead of *http*. ■ taskflow://WEB-INF/taskflows/axf-tasklist-tfd.xml#axf-tasklist-tfd (displays the Task List) ■ taskflow://WEB-INF/taskflows/axf-taskviewer-tfd.xml#axf-taskviewer-tfd (displays the Task Viewer) ■ taskflow://WEB-INF/taskflows/axf-identity-picker-tfd.xml#axf-identity-picker-tfd (displays the Identity Picker) ■ taskflow://WEB-INF/taskflows/axf-enumeration-picker-tfd.xml#axf-enumeration-picker-tfd (displays the Enumeration Picker) ■ taskflow://WEB-INF/taskflows/axf-comments-tfd.xml#axf-comments-tfd (displays Comments)

B.3.1.2 Example Implementation

This example uses the OpenTask command to display the Task Viewer for the Invoice Processing solution.

Fields not shown: SOLUTION_NAMESPACE=InvoiceProcessing

Table B–37 Example Open Task Command in AXF_SOLUTION_PARAMETERS Table

COMMAND_ NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
OpenTask	oracle.imaging.axf.commands.bpel.OpenTaskCommand	TASK_VIEW_URL	taskflow://WEB-INF/taskflows/axf-taskviewer-tfd.xml#axf-taskviewer-tfd

B.3.2 Autotask Command

This command displays autotask mode, in which a new human workflow task is automatically claimed for the user. Also see Section 6.4.4.

B.3.2.1 Autotask Command Parameters

Use the parameters in the AXF_SOLUTION_PARAMETERS Table to configure Autotask commands.

Table B–38 Autotask Command Parameters in AXF_SOLUTION_PARAMETERS Table

Parameter Key	Description
TASK_VIEW_URL	Task flow returned in the response command upon executing this command.
CMD_ON_NO_TASKS	COMMAND_NAMESPACE to execute when there are no tasks.
BPEL_TRY_AUTO	Time in milliseconds between attempts to obtain the next task from the Human workflow system.

B.3.2.2 Example Implementation

This example uses the Autotask command to automatically claim tasks and display them in the Task Viewer for the Invoice Processing solution.

Fields not shown: SOLUTION_NAMESPACE=InvoiceProcessing

Table B–39 Autotask Command Parameters in AXF_SOLUTION_PARAMETERS Table

COMMAND_ NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
AutoOpenTask	oracle.imaging.axf.commands.bpel.AutotaskCommand	TASK_VIEW_URL	taskflow://WEB-INF/taskflows/axf-taskviewer-tfd.xml#axf-taskviewer-tfd
AutoOpenTask	oracle.imaging.axf.commands.bpel.AutotaskCommand	CMD_ON_NO_TASKS	StartInvoiceProcessing
AutoOpenTask	oracle.imaging.axf.commands.bpel.AutotaskCommand	BPEL_TRY_AUTO	3000

B.3.3 Release Task Command

The Release Task command releases a human workflow task. You can configure this command with actions to execute after it is complete, using CMD_AUTOTASK_ON and CMD_AUTOTASK_OFF parameters.

B.3.3.1 Release Task Command Parameters

Table B–40 lists configuration parameters for this command. Use these parameters in the AXF_SOLUTION_PARAMETERS Table to configure commands.

Table B–40 Release Task Command Parameters

Parameter Key	Description
CMD_AUTOTASK_OFF	Specify the command (COMMAND_NAMESPACE) to execute when AUTOTASK mode is off.
CMD_AUTOTASK_ON	Specify the command (COMMAND_NAMESPACE) to execute when AUTOTASK mode is on.

B.3.3.2 Example Implementation

Fields not shown: SOLUTION_NAMESPACE=InvoiceProcessing

Table B–41 Example Release Task Commands in AXF_SOLUTION_PARAMETERS Table

COMMAND_ NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
SkipTask	oracle.imaging.axf.commands.bpel.ReleaseTaskCommand	CMD_AUTOTASK_OFF	StartInvoiceProcessing
SkipTask	oracle.imaging.axf.commands.bpel.ReleaseTaskCommand	CMD_AUTOTASK_ON	AutoTaskOpen
ReleaseTask	oracle.imaging.axf.commands.bpel.ReleaseTaskCommand	CMD_AUTOTASK_OFF	StartInvoiceProcessing
ReleaseTask	oracle.imaging.axf.commands.bpel.ReleaseTaskCommand	CMD_AUTOTASK_ON	StartInvoiceProcessing

B.3.4 Complete Task Command

The Complete Task command updates the list of attributes and outcome for a specified task in the human task workflow. This command also takes the parameters defined for the Update Task Command (BPEL).

In addition, the Complete Task command can update BPEL payload attribute values using request parameters to the command. You can also configure this command with actions to execute after it is complete, using CMD_AUTOTASK_ON and CMD_AUTOTASK_OFF parameters.

B.3.4.1 Complete Task Command Parameters

Table B–42 lists configuration parameters for this command. Use these parameters in the AXF_SOLUTION_PARAMETERS Table to configure commands.

Table B–42 CompleteTask Command Parameters

Parameter Key	Description
OUTCOME	Specify the outcome defined for the human work flow system. APPROVE and REJECT are available by default in any BPEL process; the process designer can create others.
CMD_AUTOTASK_ON	Specify the command (COMMAND_NAMESPACE) to execute when AUTOTASK mode is on.
CMD_AUTOTASK_OFF	Specify the command (COMMAND_NAMESPACE) to execute when AUTOTASK mode is off.

B.3.4.2 Example Implementation

Fields not shown: SOLUTION_NAMESPACE=InvoiceProcessing

Table B–43 Example Complete Task Command in AXF_SOLUTION_PARAMETERS Table

COMMAND_ NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
DeleteInvoice	oracle.imaging.axf.commands.bpel.CompleteTaskCommand	CMD_AUTOTASK_OFF	StartInvoiceProcessing
DeleteInvoice	oracle.imaging.axf.commands.bpel.CompleteTaskCommand	CMD_AUTOTASK_ON	AutoOpenTask
DeleteInvoice	oracle.imaging.axf.commands.bpel.CompleteTaskCommand	OUTCOME	DELETE_INVOICE

B.3.5 Redirect Command

The Redirect command redirects the browser to an AXF web tool or other external URL. The request parameters valid for internal URLs only include:

- CID (Conversation ID)
- PID (ParameterSet ID)

Store any user defined request parameters as part of the PID.

The command returns the URL in the response command.

Table B–44 lists configuration parameters for this command. Use these parameters in the AXF_SOLUTION_PARAMETERS Table to configure commands.

B.3.5.1 Redirect Command Parameters

Table B–44 RedirectCommand Parameters

Parameter Key	Description
REDIRECT_URL	This URL is returned in the response command upon executing this command. It is either: ■ a task flow String corresponding to a task flow ID which loads one or more pages on the task flow. ■ a standard URL string that redirects to the specified URL.
EXTERNAL	If this has a value of TRUE, then the redirect page does not have a CID and PID appended to it. The re-direct URL is an external website and all request parameters are appended in the URL.

B.3.5.2 Example Implementation

Fields not shown: SOLUTION_NAMESPACE=InvoiceProcessing

Table B–45 Example Redirect Command in AXF_SOLUTION_PARAMETERS Table

COMMAND_NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
StartInvoiceProcessing	oracle.imaging.axf.commands.system. RedirectCommand	EXTERNAL	FALSE
StartInvoiceProcessing	oracle.imaging.axf.commands.system. RedirectCommand	REDIRECT_URL	taskflow://WEB-INF/taskflows/axf-tasklist-tfd.xml#axf-tasklist-tfd

B.3.6 Update Task Command (BPEL)

The Update Task command can update one of the following: the list of attributes in the BPEL task, values in the XML payload using XPATH, or system attributes. (For an XPATH example, see Section B.3.6.3.)

You can create your own parameter keys for the Update Task command and use either a system attribute or an XPATH for the parameter value. AXF searches the request

parameters and finds all the values that match the parameter keys (besides outcome), and pulls parameter keys for the list of attributes to use in that task payload.

To update a non-payload attribute in the BPEL task, use a system attribute from those listed in Section B.3.6.2. For example, the UpdateTask command can take the value of `outcome` (defined as PARAMETER_KEY) from the request parameter and update the `OUTCOME` (defined as PARAMETER_VALUE) attribute value in the task.

B.3.6.1 Update Task Command Parameters

Table B–46 Parameters for UpdateTaskCommand

Parameter Key	Description
OUTCOME	Specify the outcome defined for the human work flow system. APPROVE and REJECT are available by default in any BPEL process; the process designer can create others.

B.3.6.2 System Attributes

System Attributes
ACQUIREDBY
APPROVERS
ASSIGNEDDATE
ASSIGNEDGROUP //Cannot be updated
ASSIGNEDUSER //Cannot be updated
CREATEDATE
CREATOR
DATEATTRIBUTE1-DATEATTRIBUTE5
EXPIREDDATE
ENDDATE
FORMATTRIBUTE1-FORMATTRIBUTE5
FROMUSER
NUMBERATTRIBUTE1-NUMBERATTRIBUTE5
OUTCOME
OWNERGROUP
OWNERUSER
PRIORITY
STATE
TASKID
TASKNUMBER //Cannot be updated
TITLE
TASKDEFINITIONNAME
TEXTATTRIBUTE1-TEXTATTRIBUTE10
UPDATEDBY

System Attributes
URLATTRIBUTE1 - URLATTRIBUTE5

B.3.6.3 Example Implementation

This example shows two attributes updated by the same comand_namespace, UpdateHelloBPEL. For more information about HelloBPEL, see "Verifying the AXF Installation with HelloBpel" in *Oracle WebCenter Content Installation Guide*.

Fields not shown: SOLUTION_NAMESPACE=HelloBPEL

Table B–47 Example UpdateTaskCommand Parameters in AXF_SOLUTION_PARAMETERS Table

COMMAND_ NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
UpdateHelloBPEL	oracle.imaging.axf.commands.bpel. UpdateTaskCommand	String1	XPATH:HelloBPEL_String1
UpdateHelloBPEL	oracle.imaging.axf.commands.bpel. UpdateTaskCommand	String2	XPATH:HelloBPEL_String2

B.3.7 Update Task From Procedure Command

The Update Task From Procedure command calls a stored pl/sql procedure using a specified data source and updates the task payload using XPATH. See Section B.3.7.2 for an example implementation and Section B.3.7.3 for an example stored pl/sql procedure.

B.3.7.1 Update Task From Procedure Command Parameters

Table B–48 Parameters for UpdateTaskFromProcedure Command

Parameter Key	Description
XPATH_DATASTRING	Specifies an XPATH variable contained in the AXF_XPATH_ ATTRIBUTES Table that refers to the XPATH where the list of returned data is stored.
XPATH_MESSAGESTRING	Specify an XPATH variable to allow the UpdateFromStoredProcedure function to write data to the BPEL payload for communicating basic status or error codes encountered during execution of the function (axfretrieveuserlist).
CMD_NO_DATA	Specifies the command to execute if no results return from the pl/sql function.
CMD_DATA	Specifies the command to execute if results return from the pl/sql function.
CMD_ERROR	Specifies the command to execute if an error is returned from the pl/sql function.
JNDI_DS	Specifies the name of the JNDI data source, configured on the Application Server, to use for execution of the pl/sql function.
PLSQL_PROC	Specifies the name of the pl/sql function to call.

B.3.7.2 Example Implementation

This example uses the Update Task From Procedure command to call a stored pl/sql procedure called AXFRETRIEVEUSERLIST using a data source (EBSDS).

AXFRETRIEVEUSERLIST retrieves the invoice ID, queries for the invoice amount for that transaction, and based on that amount, returns a set of users for invoice approval.

Table B–49 AXF_SOLUTIONS Table for UpdateTaskFromProcedureCommand Example

SOLUTION_NAMESPACE	SOLUTION_CONTEXT
InvoiceProcessing	ejb.AxfCommandMediator#oracle.imaging.axf.service.AxfCommandMediatorRemote

Table B–50 AXF_COMMANDS Table for UpdateTaskFromProcedureCommand Example (Fields not shown: SOLUTION_NAMESPACE=InvoiceProcessing)

COMMAND_CLASS	COMMAND_NAMESPACE
oracle.imaging.axf.commands.bpel.UpdateTaskFromProcedureCommand	RetrieveUserList

Table B–51 AXF_SOLUTION_PARAMETERS Table for UpdateTaskFromProcedureCommand Example (Fields not shown: SOLUTION_NAMESPACE=InvoiceProcessing)

COMMAND_NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
RetrieveUserList	oracle.imaging.axf.commands.bpel.UpdateTaskFromProcedureCommand	XPATH_DATASTRING	XPATH:InvoiceProcessing_InvoiceApprovalAssignment
RetrieveUserList	oracle.imaging.axf.commands.bpel.UpdateTaskFromProcedureCommand	XPATH_MESSAGESTRING	*<INSERT: an XPATH variable to use for communicating status through the payload>*
RetrieveUserList	oracle.imaging.axf.commands.bpel.UpdateTaskFromProcedureCommand	CMD_DATA	InvoiceApprovalEdit
RetrieveUserList	oracle.imaging.axf.commands.bpel.UpdateTaskFromProcedureCommand	CMD_NO_DATA	CompleteInvoice
RetrieveUserList	oracle.imaging.axf.commands.bpel.UpdateTaskFromProcedureCommand	CMD_ERROR	StartInvoiceProcessing
RetrieveUserList	oracle.imaging.axf.commands.bpel.UpdateTaskFromProcedureCommand	JNDI_DS	jdbc/EBSDS
RetrieveUserList	oracle.imaging.axf.commands.bpel.UpdateTaskFromProcedureCommand	PLSQL_PROC	AXFRETRIEVEUSERLIST

In the AXF_XPATH_ATTRIBUTES table that follows, the corresponding XPATH column displays the XPATH expression used to locate the value in the payload.

Table B–52 AXF_XPATH ATTRIBUTES Table for UpdateTaskFromProcedureCommand Example

ATTRIBUTE_KEY	XPATH
InvoiceProcessing_InvoiceApprovalAssignment	//task:InvoiceApprovalAssignment

In the AXF_NAMESPACES table that follows, the XMLNS_URI column displays where within the XML file to locate the InvoiceApprovalAssignment task information.

Table B–53 AXF_XPATH_NAMESPACES Table for UpdateTaskFromProcedureCommand Example

Prefix	XMLNS_URI
task	http://xmlns.oracle.com/bpel/workflow/task

B.3.7.3 Example PL/SQL Procedure For Updating the Task Payload

The pl/sql procedure that follows loads the xml into the DOM, retrieves the invoice ID, queries for the invoice amount for that transaction, and based on that amount, returns a set of users.

```
create or replace FUNCTION axfretrieveuserlist(xmlPayload IN VARCHAR2, pDataString OUT VARCHAR2,
pMsgString OUT VARCHAR2) RETURN INTEGER IS

    v_node    xmldom.DOMNode;
    v_node2   xmldom.DOMNode;
    v_nl      xmldom.DOMNodeList;
    v_doc     xmldom.DOMDocument;
    v_elem    xmldom.DOMElement;
    v_parser  xmlparser.Parser;
    invoiceID     VARCHAR2(256);
    invoiceAmount NUMBER(8,2);
    userList      VARCHAR2(256);
    rtnCode      INTEGER;

BEGIN

    rtnCode := 0;  -- execute CMD_NO_DATA
    -- rtnCode := -1; -- execute CMD_ERROR
    pmsgstring := ''; -- empty msg string
    v_parser := xmlparser.newParser;
    xmlparser.parseBuffer(v_parser, xmlPayload);
    v_doc := xmlparser.getDocument(v_parser);
    xmlparser.freeParser(v_parser);

    -- Retrieve the invoice ID
    v_nl := xmldom.getElementsByTagName(v_doc, 'invoiceID');
    v_node := xmldom.item(v_nl, 0);
    v_node2 := xmldom.getFirstChild(v_node);
    invoiceID := xmldom.getNodeValue(v_node2);

    -- Retrieve Invoice Amount for given invoice id
    select INVOICE_AMOUNT into invoiceAmount from ap_invoices_all where INVOICE_ID = invoiceid;

    if invoiceamount > 10000 then
      pdatastring := 'jlondon';
      rtnCode := 1;  -- execute CMD_DATA
      pmsgstring := 'Additional approval required due to invoice amount.';
    else
      pdatastring := 'jcooper,mtwain';
      rtnCode := 1;  -- execute CMD_DATA
      pmsgstring := 'No further approval is required.';
    end if;

    RETURN rtnCode;

END;
```

Follow these steps to use this example.

1. Modify this procedure to retrieve the specific pieces of data from the payload you would like. The only requirement is that the pl/sql function you create must take a VARCHAR2, set up two VARCHAR2 out parameters, and return an INTEGER. The name of the function is in the AXF configuration.
2. Install the stored procedure on the Oracle E-Business Suite database instance. The procedure must be run as the Apps user.

3. On the AXF instance and as the AXF user, run the following command to update the AXF_ACTIONS table to insert the new RetrieveUserList Command into the execution path:

```
UPDATE AXF_ACTION SET COMMAND_NAMESPACE = 'RetrieveUserList' where DISPLAY_TEXT
= 'Invoice Approval' and COMMAND_NAMESPACE = 'ValidateInvoiceApprovalEdit'
```

B.3.8 Terminate Conversation Command

The Terminate Conversation Command is used by an external client to terminate a conversation with AXF.

B.3.9 Validate Task Command

The Validate Task command validates BPEL system attribute data or BPEL payload data, and based on validation results, executes a subsequent command.

Table B–54 lists configuration parameters for this command. Use these parameters in the AXF_SOLUTION_PARAMETERS Table to configure commands.

B.3.9.1 Validate Task Command Parameters

Table B–54 ValidateTaskCommand Parameters

Parameter Key	Description
ATTRIBUTE_TO_VALIDATE	Specifies the attribute in the BPEL task to validate. This can be either a system attribute or a payload attribute. If specifying a payload attribute, use a prefix value of *XPATH:* and reference a value from the AXF_XPATH_ATTRIBUTES Table.
REGULAR_EXPRESSION	Defines a standard Regular Expression for validating the specified attribute.
CMD_ON_PASS	Specifies the command to execute after this command, if the validation is successful.
CMD_ON_FAIL	Specifies the command to execute after this command if the validation fails.
FAIL_MESSAGE	Specifies the message to display if the validation fails.

B.3.9.2 Example Implementation

The following configuration validates that the invoice has been saved (Invoice Transaction ID is not 0). If it is 0, the command reports the error message specified in the FAIL_MESSAGE parameter. For steps to add a validation, see Section 6.4.7.

Fields not shown: SOLUTION_NAMESPACE=InvoiceProcessing

Table B–55 Example ValidateTask Command in AXF_SOLUTION_PARAMETERS Table

COMMAND_ NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_ KEY	PARAMETER_VALUE
ValidateTransactionID	oracle.imaging.axf.commands.bpel. ValidateTaskCommand	ATTRIBUTE_TO_ VALIDATE	XPATH:InvoiceProcessing_TransactionID
ValidateTransactionID	oracle.imaging.axf.commands.bpel. ValidateTaskCommand	CMD_ON_PASS	CompleteInvoice
ValidateTransactionID	oracle.imaging.axf.commands.bpel. ValidateTaskCommand	REGULAR_ EXPRESSION	[^0]
ValidateTransactionID	oracle.imaging.axf.commands.bpel. ValidateTaskCommand	FAIL_MESSAGE	Please save the transaction before completing the task.

B.4 Oracle E-Business Suite Forms Tables For the BPEL Imaging Solution

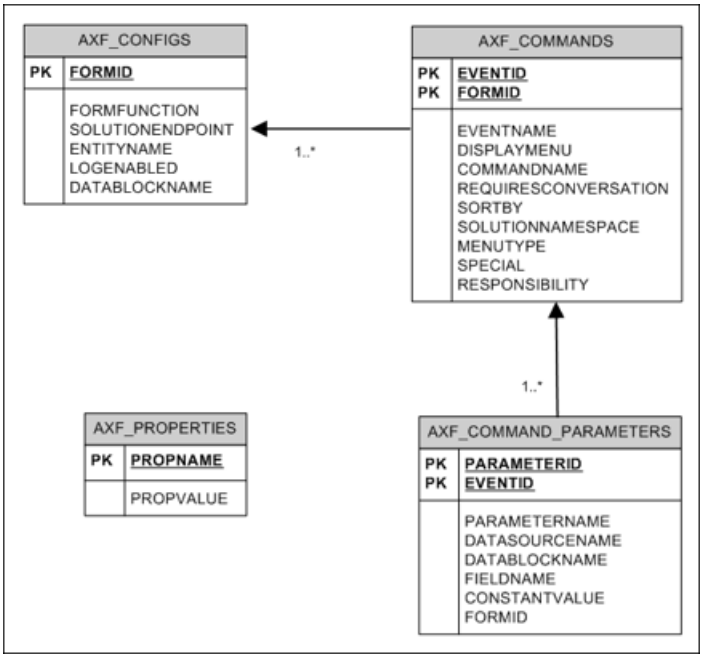
Configuring AXF for Oracle E-Business Suite requires configuring AXF-related tables in Oracle E-Business Suite. This section covers the following topics:

- Section B.4.1, "About the Oracle E-Business Suite AXF Tables For the BPEL Imaging Solution"
- Section B.4.2, "AXF_CONFIGS Table (Oracle E-Business Suite)"
- Section B.4.3, "AXF_COMMANDS Table (Oracle E-Business Suite)"
- Section B.4.4, "AXF_COMMAND_PARAMETERS Table (Oracle E-Business Suite)"
- Section B.4.5, "AXF_PROPERTIES Table (Oracle E-Business Suite)"

Note: The Oracle E-Business Suite BPEL imaging solution is supported for Forms-based applications only, not OAF applications.

B.4.1 About the Oracle E-Business Suite AXF Tables For the BPEL Imaging Solution

The following diagram shows how the tables used by the Oracle E-Business Suite system in AXF solutions are related.



B.4.2 AXF_CONFIGS Table (Oracle E-Business Suite)

Use the AXF_CONFIGS table to enable the AXF solution on various Oracle E-Business Suite Forms. This table provides a fine level of granularity when selecting which Forms to AXF-enable, up to the Data Block level.

Form events are automatically invoked when an action is performed on an Oracle E-Business Suite Form. The AXF_CUSTOM.pll makes all events available, such as POST-INSERT, for customization. You can decide which events to use, and how and when to use them.

When an action occurs, the customized code launches the specified solution and command configured for the event. In the case where the same form is being reused, such as Invoice Entry and Invoice Query, FORMFUNCTION and DATABLOCKNAME uniquely identify each Form.

Note: You can enable all datablocks on a form rather than a specific datablock, by specifying `AXF_DEFAULT` for the DATABLOCKNAME parameter. This allows AXF to be notified whenever a POST-INSERT event occurs for the form, regardless of its datablock. Note, however, that setting the DATABLOCKNAME parameter to AXF_DEFAULT enables specified ZOOM or SPECIAL commands on all pages related to the form. (Set ZOOM and SPECIAL commands in the AXF_COMMANDS Table (Oracle E-Business Suite).)

B.4.2.1 Column Description

Table B–56 Column Description for AXF_CONFIGS Table

Column Name	Description
FORMID	Specifies the primary key of the table.
FORMFUNCTION	Distinguishes each Oracle E-Business Suite Form based on the form's functionality.
SOLUTIONENDPOINT	Specifies a URL to AXF. This value should start as `https://` if SecureMode is on. See Section 2.4.2.
ENTITYNAME	Used by the attachment functionality as a unique name, which links attachments to the correct Forms.
LOGENABLED	Enables or disables the log for the specified form. Specify one of the following: ■ 1/TRUE/YES ■ 0/FALSE/NO
DATABLOCKNAME	Specify the data block on the form to enable. Note that you can also specify `AXF_DEFAULT` to enable all data blocks on the form. A Form may be reused by Oracle E-Business Suite (for example, Invoice Entry and Invoice Query); the FORMFUNCTION and DATABLOCKNAME together uniquely identify each form.

B.4.2.2 Example Implementation

This example defines that the entire Invoices Form is AXF-enabled. (Without the first row, the INV_SUM_FOLDER Data Block of the Invoices Form would be enabled.)

The last row in this example table shows the paperclip attachment feature enabled, by disabling the Managed Attachments solution. For more information, see Section 6.4.11.

Fields not shown: LOGENABLED=YES

Table B–57 Example AXF_CONFIGS Table

FORMID	FORMFUNCTION	SOLUTIONENDPOINT	ENTITYNAME	DATABLOCKNAME
1	AP_APXINWKB	https://*ApplicationServerName:Port*/axf-ws/AxfSolutionMediatorService	AP_INVOICES	AXF_DEFAULT
2	AP_APXINWKB_SUMMARY_VIEW	https://*ApplicationServerName:Port*/axf-ws/AxfSolutionMediatorService	AP_INVOICES	INV_SUM_FOLDER
6	AP_APXINWKB_BATCHES	https://*ApplicationServerName:Port*/axf-ws/AxfSolutionMediatorService	AP_INVOICES	INV_SUM_FOLDER
7	AXF_MANAGED_ATTACHMENTS-DISABLED	https://*ApplicationServerName:Port*/axf-ws/AxfSolutionMediatorService	(null)	(null)

B.4.3 AXF_COMMANDS Table (Oracle E-Business Suite)

Use the AXF_COMMANDS table to describe the actions to take based on user activity. This table works with the AXF_CONFIGS Table (Oracle E-Business Suite).

B.4.3.1 Column Description

Table B–58 Column Description for AXF_COMMANDS Table

Column Name	Description
FORMID	Links to the AXF_CONFIGS Table (Oracle E-Business Suite).
EVENTID	Primary key of the table.
EVENTNAME	Name of the Event command to invoke (for example, ZOOM, POST-INSERT).
DISPLAYMENU	Displays text of the menu for the command.
COMMANDNAMESPACE	Request command to pass to the back-end when the menu is selected.
REQUIRESCONVERSATION	Indicates if the command requires a valid conversation or not.
SORTBY	Order in which to display the menu.
SOLUTIONNAMESPACE	Name of the solution.
MENUTYPE	Specify the menu type to display to users in Oracle E-Business Suite. You can choose: ■ ZOOM: Displays a Zoom menu in the toolbar. ■ ZOOMANDSPECIAL: Displays both a Zoom menu and a Special menu. (Enter a special key in the SPECIAL column.) ■ SPECIAL: Displays a Special menu on the toolbar. (Enter a special key in the Special column.)
SPECIAL	Create new menu entries by entering a unique number for the Special type menu, where: ■ SPECIAL1-15 creates entries in the Tools menu. ■ SPECIAL16-30 creates entries in the Reports menu. ■ SPECIAL31-45 creates entries in the Actions menu. (Consult the Oracle E-Business Suite Documentation for further information.)
RESPONSIBILITY	Reserved for future use.

B.4.3.2 Example Implementation

This example shows two commands invoked from the Zoom menu (Attach Supplemental and Process Invoices). Each command is listed twice because the commands enable the same functionality, but on two different pages.

In addition, the solution has been configured to invoke the SaveInvoice command during the POST-INSERT event, which specifies that whenever an action inserts a new Oracle E-Business Suite transaction record, the integration automatically invokes the SaveInvoice command on the back-end, performing the actions associated with the command. Note that POST-INSERT is not called by a subsequent save of the same transaction record in Oracle E-Business Suite.

Fields not shown: SOLUTION_NAMESPACE=InvoiceProcessing, SPECIAL=(null), RESPONSIBILITY=(null)

Table B–59 Example AXF_COMMANDS Table

EVENT ID	FORM ID	EVENTNAME	DISPLAYMENU	COMMANDNAMESPACE	REQUIRESCONVERSATION	SORT BY	MENU TYPE
16	1	ZOOM	Attach Supplemental	AttachSupplemental	YES	2	ZOOM
14	6	ZOOM	Attach Supplemental	AttachSupplemental	YES	2	ZOOM
13	6	ZOOM	Process Batch Invoices	StartInvoiceProcessing	NO	1	ZOOM
10	1	ZOOM	Process Invoices	StartInvoiceProcessing	NO	1	ZOOM
11	1	POST-INSERT	(null)	SaveInvoice	YES	0	(null)
15	6	POST-INSERT	(null)	SaveInvoice	YES	0	(null)

B.4.4 AXF_COMMAND_PARAMETERS Table (Oracle E-Business Suite)

Use the AXF_COMMAND_PARAMETERS table to define the information sent for each defined command. Each command may require or omit a different set of parameters.

B.4.4.1 Column Description

Table B–60 Column Description for AXF_COMMAND_PARAMETERS Table

Column	Description
PARAMETERID	Defines a unique ID for the parameter.
EVENTID	Defines a unique ID for the event. Comes from the AXF_COMMANDS Table (Oracle E-Business Suite).
PARAMETERNAME	The name of the parameter to pass.
DATASOURCENAME	Data Source for the parameter value. You can specify *Data* or *Constant*.
DATABLOCKNAME	Data Block of the Form from which the value is fetched.
FIELDNAME	Field Name in the form from which the value is fetched.
CONSTANTVALUE	A constant value for the parameter.

B.4.4.2 Example Implementation

The example that follows contains two parameters sent for EventID 2: a constant value (InvoicesByVendor) and a data value (VENDOR_NAME) in the *INVOICES_QF* Data Block.

The COMPLETE command requires that a conversation be established between Oracle E-Business Suite and AXF. A Conversation is a session unique ID that allows communication between Oracle E-Business Suite and AXF Framework.

If a command requires a separate window to open, then Oracle E-Business Suite opens another instance of the browser. Users may then take additional steps in the newly created window.

Table B–61 Example AXF_COMMAND_PARAMETERS Table

PARAMETERID	EVENTID	PARAMETERNAME	DATASOURCE NAME	DATABLOCKNAME	FIELDNAME	CONSTANT VALUE
1	2	SearchName	CONSTANT	(null)	(null)	InvoicesByVendor
2	2	VendorName	DATA	INVOICES_QF	VENDOR_NAME	(null)
18	11	InvoiceTransactionID	DATA	INV_SUM_FOLDER	INVOICE_ID	(null)
21	14	InvoiceTransactionID	DATA	INV_SUM_FOLDER	INVOICE_ID	(null)
20	15	InvoiceTransactionID	DATA	INV_SUM_FOLDER	INVOICE_ID	(null)
22	16	InvoiceTransactionID	DATA	INV_SUM_FOLDER	INVOICE_ID	(null)

B.4.5 AXF_PROPERTIES Table (Oracle E-Business Suite)

Use the AXF_PROPERTIES table to define properties for an AXF integration with Oracle E-Business Suite.

B.4.5.1 Column Description

Table B–62 Column Description for AXF_PROPERTIES Table

Column	Description
PROPNAME	Specifies properties to use. Properties include: ■ SecureMode: To enable SSL, set this property to ON, and set values for AXFWalletPath and AXFWalletKey properties. ■ AXFWalletPath: Certificate location (path). ■ AXFWalletKey: Specify the key for the AXF wallet password, to be retrieved from the Oracle E-Business Suite vault. ■ AXF_VERSION: Specify `1` for AXF 10g, or `2` for AXF 11g. ■ AXF_SOAP_POLICY: Specify the name of the SOAP policy to use. The currently supported policy is `USER_NAME_TOKEN`. ■ AXF_SOAP_SECURITY: Specify `TRUE` (required), which enables SOAP security, as described in Section 2.4.1. ■ AXF_SOAP_USER: Specify the SOAP userid used in the SOAP header for authentication, as described in Section 2.4.1. ■ AXF_PAPERCLIP: Set to `TRUE` to enable the paperclip option, or `FALSE` (default) to disable it. Also see Section 6.4.11. ■ AXF_MULTINSTANCE: Allows you to target the same instance/conversation when multiple instances of a particular Oracle E-Business Suite page are displayed. See Section 6.4.6. Specify TRUE to treat each Oracle E-Business Suite window as a separate conversation and target a separate Imaging or AXF window. Specify FALSE (default) to treat each Oracle E-Business Suite instance window as the same conversation and use the same Imaging or AXF target window.

Table B–62 (Cont.) Column Description for AXF_PROPERTIES Table

Column	Description
PROPVALUE	Specifies the property's value.

B.4.5.2 Example Implementation

This example table shows the default properties values.

Table B–63 Example AXF_PROPERTIES Table

PROPNAME	PROPVALUE
SecureMode	OFF
AXFWalletPath	file:*walletpath*
AXFWalletKey	*AXFWalletKey*
AXF_VERSION	2
AXF_SOAP_POLICY	USER_NAME_TOKEN
AXF_SOAP_SECURITY	TRUE
AXF_SOAP_USER	weblogic
AXF_PAPERCLIP	TRUE

B.5 Oracle PeopleSoft Tables For the BPEL Imaging Solution

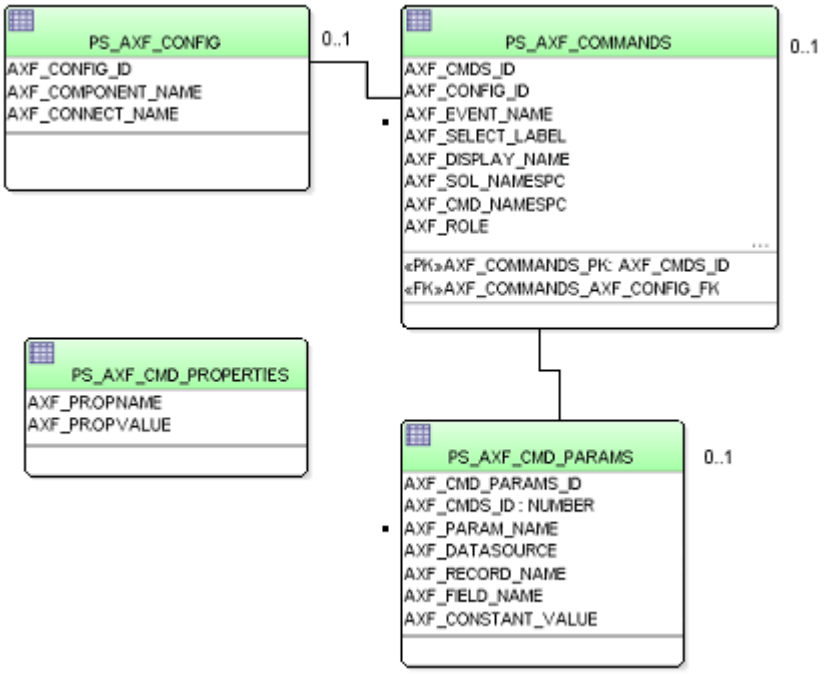
Configuring the adapter for Oracle PeopleSoft requires configuring AXF-related tables in Oracle PeopleSoft. For more information, see Section 3.5.

This section covers the following topics:

- Section B.5.1, "About the Oracle PeopleSoft AXF Tables For the BPEL Imaging Solution"
- Section B.5.2, "PS_AXF_CONFIG Table"
- Section B.5.3, "PS_AXF_COMMANDS Table"
- Section B.5.4, "PS_AXF_COMMAND_PARAMS Table"
- Section B.5.5, "PS_AXF_PROPERTIES Table"

B.5.1 About the Oracle PeopleSoft AXF Tables For the BPEL Imaging Solution

The following diagram shows how the tables used by the Oracle PeopleSoft system in AXF solutions are related.



Note: All AXF table fields require non-null values. You can include a space for fields that do not apply.

B.5.2 PS_AXF_CONFIG Table

Use the PS_AXF_CONFIG table to enable the AXF solution on various Oracle PeopleSoft components. This table provides a fine level of granularity when selecting which pages and components to AXF-enable.

Events are automatically invoked when an action is performed on an Oracle PeopleSoft page. The AXF_PS_Integration project component catches PRE- and POST-save events and makes them available for customization. You can decide which events to use and how and when to use them.

B.5.2.1 Column Description

Table B–64 Column Description for PS_AXF_CONFIG Table

Column Name	Description
AXF_CONFIG_ID	Specifies the primary key of the table.
AXF_COMPONENT_NAME	Specifies the name of the Oracle PeopleSoft component being enabled.
AXF_CONNECT_NAME	Specifies the Oracle PeopleSoft Integration Broker Connection name (service operation to call), as defined in the Oracle PeopleSoft Integration Broker administration interface. The default service operation is AXF_EXECUTE. You can set up and use other connections.

Table B–64 (Cont.) Column Description for PS_AXF_CONFIG Table

Column Name	Description
AXF_VERSION	Specifies the AXF version of the connection. Available values include: ■ 10g ■ 11g

B.5.2.2 Example Implementation

This example defines that the VCHR_QUICK_PNL component is AXF-enabled.

Table B–65 Example PS_AXF_CONFIG Table

AXF_CONFIG_ID	AXF_COMPONENT_NAME	AXF_CONNECT_NAME	AXF_VERSION
1	VCHR_QUICK_PNL	AXF_EXECUTE	11g

B.5.3 PS_AXF_COMMANDS Table

Use the PS_AXF_COMMANDS table to describe the actions to take based on user activity. This table works with the PS_AXF_CONFIG Table.

B.5.3.1 Column Description

Table B–66 Column Description for PS_AXF_COMMANDS Table

Column Name	Description
AXF_CMDS_ID	Specifies the primary key of the table, used to identify the unique command.
AXF_CONFIG_ID	Specifies the foreign key to the PS_AXF_CONFIG Table, which associates this unique command with a particular page and component.
AXF_EVENT_NAME	Specifies the event being executed. The AXF_EVENT_NAME corresponds to the subpage that is incorporated into an existing Oracle PeopleSoft page, such as the button, link, or menu that is added to an Oracle PeopleSoft page to invoke AXF functionality such as the Image Viewer or Task List. For more information, see Section 3.3. Available options include: ■ AXF_BUTTON_1, …, AXF_BUTTON_5 ■ AXF_LINK_1, ..., AXF_LINK5 ■ AXF_COMBO_1, ..., AXF_COMBO_5 ■ SAVE_PRE_CHANGE ■ SAVE_POST_CHANGE
AXF_SELECT_LABEL	Defines the name displayed on the page for the selection field. This is used only with the COMBO event.
AXF_DISPLAY_NAME	Defines the name of the button or link to display on the Oracle PeopleSoft page.
AXF_SOL_NAMESPC	Identifies the solution namespace for the command to execute.
AXF_CMD_NAMESPC	Identifies the command namespace for the command to execute.
AXF_PSFT_ROLE	Identifies the Oracle PeopleSoft roles with access to the command. It is a comma-delimited list with each role enclosed in single quotes (for example, 'Role1','Role2','Role3').
AXF_SORT_ORDER	Specifies the order of items displayed in a selection field. Sort order applies to selection fields only.
AXF_REQ_CONV	Specifies if a conversation is required to this command before execution. For example, execution of the UpdateTask command requires a conversation be running in order for the user to select a current task.

B.5.3.2 Example Implementation

This example shows two commands added to an invoice processing page. One inserts a button that when clicked initiates invoice processing. The other inserts a link that when clicked initiates a search of Imaging for duplicate invoices.

In addition, the solution has been configured to invoke the SaveInvoice command during the SAVE_POST_CHANGE event, which specifies that whenever an action inserts a new Oracle PeopleSoft transaction record, the integration automatically invokes the SaveInvoice command on the back-end, performing the actions associated with the command. Note that SAVE_POST_CHANGE is not called by a subsequent save of the same transaction record in Oracle PeopleSoft.

You must specify an Oracle PeopleSoft Role in the AXF_PSFT_ROLE field to give permissions to use the commands. If a person does not have proper permissions to use the commands, the commands do not display. If the commands display but do not function, this indicates that the commands are not properly configured.

Fields not shown: AXF_SELECT_LABEL=(null), AXF_SORT_ORDER=1, AXF_REQ_CONV=N

Table B–67 Example PS_AXF_COMMANDS Table

AXF_CMDS_ID	AXF_CONFIG_ID	AXF_EVENT_NAME	AXF_DISPLAY_NAME	AXF_SOL_NAMESPC	AXF_CMD_NAMESPC	AXF_PSFT_ROLE
1	1	AXF_BUTTON_1	Start Invoice Processing	InvoiceProcessing	StartInvoiceProcessing	'Employee'
2	1	AXF_LINK_1	Search For Duplicates	InvoiceInquiry	SearchIPM	'Employee'
3	1	SAVE_POST_CHANGE	(null)	InvoiceProcessing	SaveInvoice	'Employee'

B.5.4 PS_AXF_COMMAND_PARAMS Table

Use the PS_AXF_COMMAND_PARAMS table to specify the information sent for each defined command. Each command may require or omit a different set of parameters.

B.5.4.1 Column Description

Table B–68 Column Description for AXF_COMMAND_PARAMETERS Table

Column	Description
AXF_CMDS_PARAMS_ID	Specifies the primary key of the table.
AXF_CMDS_ID	Specifies the foreign key to the PS_AXF_COMMANDS Table, which associates the unique parameter with a specific command.
AXF_PARAM_NAME	Defines the parameter name.
AXF_DATASOURCE	Specifies where the parameter value is retrieved. ■ DATA: Retrieves the value in Oracle PeopleSoft defined by the AXF_RECORD_NAME and AXF_FIELD_NAME fields. ■ CONSTANT: Uses the value defined in the AXF_CONSTANT_VALUE field.
AXF_RECORD_NAME	Identifies the record of the field in the Oracle PeopleSoft page to use as the target value to retrieve when AXF_DATASOURCE is set to DATA.

Table B–68 (Cont.) Column Description for AXF_COMMAND_PARAMETERS Table

Column	Description
AXF_FIELD_NAME	Used as the target value when AXF_DATASOURCE is set to DATA.
AXF_CONSTANT_VALUE	Used as the constant value when AXF_DATASOURCE is set to CONSTANT.

B.5.4.2 Example Implementation

The example that follows contains three parameters sent for AXF_CMDS_ID 2: a constant value (SearchName) and two data values (INVOICENUMBER and InvoiceTransactionID).

Table B–69 Example PS_AXF_COMMAND_PARAMS Table

AXF_CMDS_ PARAMS_ID	AXF_ CMDS_ID	AXF_PARAM_NAME	AXF_ DATASOURCE	AXF_RECORD_ NAME	AXF_FIELD_ NAME	AXF_ CONSTANT_ VALUE
1	2	SearchName	CONSTANT	(null)	(null)	SearchByInvoice ID
2	2	INVOICENUMBER	DATA	VCHR_HDR_QV	INVOICE_ID	(null)
3	3	InvoiceTransactionID	DATA	VCHR_HDR_QV	VOUCHER_ID	(null)

B.5.5 PS_AXF_PROPERTIES Table

Use the PS_AXF_PROPERTIES table to define properties for an AXF integration with Oracle PeopleSoft. See Section 6.4.12.

B.5.5.1 Column Description

Table B–70 Column Description for PS_AXF_PROPERTIES Table

Column	Description
AXF_PROPNAME	Specifies properties to use. Properties include: ▪ WINDOW_HEIGHT: Specify the popup window's height in pixels. (This refers to any popup window that displays as part of the solution.) ▪ WINDOW_WIDTH: Specify the popup window's width in pixels.
AXF_PROPVALUE	Specifies the property's value.

B.5.5.2 Example Implementation

This example table shows the default properties values.

Table B–71 Example PS_AXF_PROPERTIES Table

AXF_PROPNAME	AXF_PROPVALUE
WINDOW_HEIGHT	600
WINDOW_WIDTH	800

C

Managed Attachments Solution Reference

This appendix describes the AXF, Oracle E-Business Suite, Oracle PeopleSoft, and Content Server configuration tables used for the Managed Attachments solution. It describes the GrantAccess command and the Managed Attachments URL, and provides example implementations.

This appendix covers the following topics:

- Section C.1, "AXF Tables For the Managed Attachments Solution"
- Section C.2, "Oracle E-Business Suite Forms Tables For the Managed Attachments Solution"
- Section C.3, "Oracle E-Business Suite OAF Tables For the Managed Attachments Solution"
- Section C.4, "Oracle PeopleSoft Tables For the Managed Attachments Solution"
- Section C.5, "Content Server Tables For the Managed Attachments Solution"

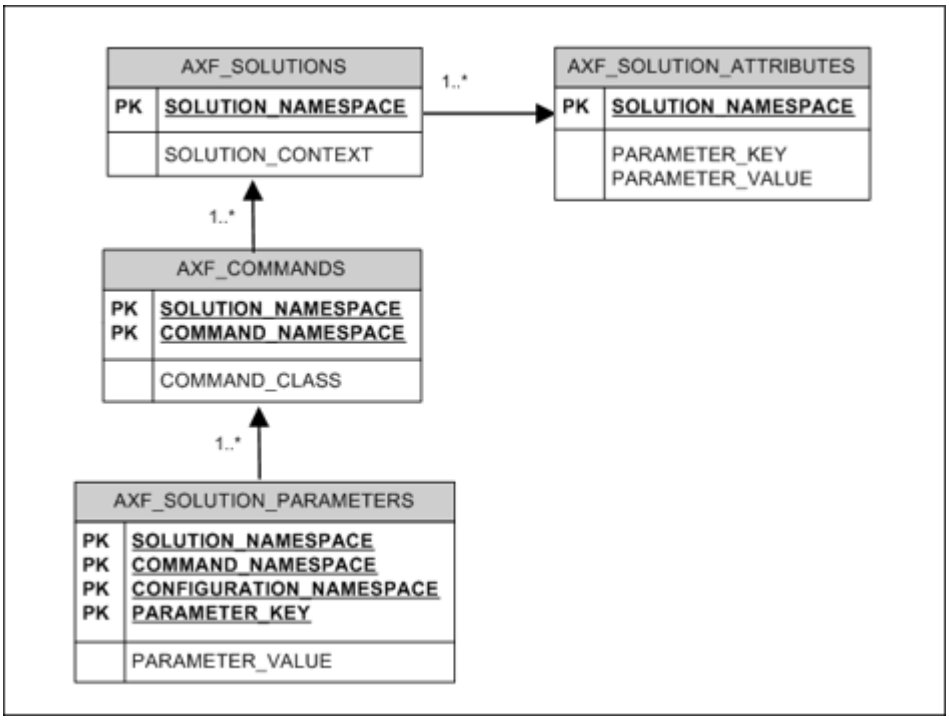
Many of the Managed Attachments tables are automatically populated during the business application plug-in configuration (see Chapter 2 or Chapter 3) and Managed Attachments configuration (see Chapter 7). This appendix describes how to configure additional pages for Managed Attachments or to make changes to the tables if needed.

C.1 AXF Tables For the Managed Attachments Solution

Application Extension Framework (AXF) tables define the solution, its system parameters, and the GrantAccess command used.

> **Note:** If modifying AXF table values in a running system, either execute Clear DB Cache from the Driver page or restart the AXF application within the Application Server for the changes to take effect.

The diagram that follows displays the AXF configuration tables used for the Managed Attachments solution and their relationships.



AXF Table	Description
AXF_SOLUTIONS Table AXF_SOLUTION_ATTRIBUTES Table	Defines AXF solutions and general parameters for infrastructure, services, and solutions.
AXF_COMMANDS Table	Defines AXF commands within solutions.
AXF_SOLUTION_PARAMETERS Table	Defines parameters for AXF commands.

C.1.1 AXF_SOLUTIONS Table

The AXF_SOLUTIONS table defines the solutions used by AXF. It links to the AXF_COMMANDS Table through the SOLUTION_NAMESPACE column.

C.1.1.1 Column Description

Table C–1 Column Description for AXF_SOLUTIONS Table

Column	Description
SOLUTION_CONTEXT	Defines the JNDI name of the AXF solution implementation. (Currently, AxfCommandMediator is the only solution implementation.)
SOLUTION_NAMESPACE	Defines the AXF solution name.

C.1.1.2 Example Implementation

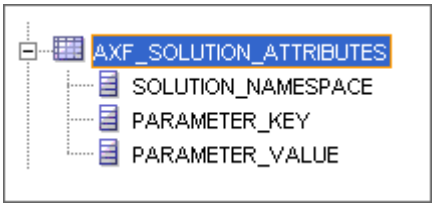
This example shows the Managed Attachments solution defined, using AxfCommandMediator as its solution implementation.

Table C–2 Example AXF_SOLUTIONS Table

SOLUTION_NAMESPACE	SOLUTION_CONTEXT
UCM_Managed_Attachments	ejb.AxfCommandMediator#oracle.imaging.axf.service.AxfCommandMediatorRemote

C.1.2 AXF_SOLUTION_ATTRIBUTES Table

This table defines general attributes for use by infrastructure, services, or solutions.



C.1.2.1 Column Description

Table C–3 Column Description for AXF_SOLUTION_ATTRIBUTES Table

Column	Description
SOLUTION_NAMESPACE	Specifies the solution namespace that uses this parameter.
PARAMETER_KEY	Name of the parameter. Used when retrieving the parameter value from the database. Applicable parameters include: ▪ USERNAME_PASS_THROUGH: When set to FALSE for an imaging solution, this parameter allows an authenticated Oracle WebLogic Server user to perform tasks, such as BPEL tasks, rather than the user passed from Oracle E-Business Suite or Oracle PeopleSoft. Note that this parameter must be set to TRUE for Managed Attachments configurations. See Section 7.6.19.
PARAMETER_VALUE	Value of the parameter.

C.1.2.2 Example Implementation

This example table sets solution attributes for the Managed Attachments solution.

Table C–4 Example AXF_SOLUTION_ATTRIBUTES Table

SOLUTION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
UCM_Managed_Attachments	USERNAME_PASS_THROUGH	TRUE

C.1.3 AXF_COMMANDS Table

This table defines AXF commands and their java classes for the solution. Each command's parameters are configured in the AXF_SOLUTION_PARAMETERS Table.

C.1.3.1 Column Description

Table C–5 Column Description for AXF_COMMANDS Table

Column	Description
SOLUTION_NAMESPACE	The name of the solution, as defined in the AXF_SOLUTIONS Table.

Table C–5 (Cont.) Column Description for AXF_COMMANDS Table

Column	Description
COMMAND_NAMESPACE	Defines the unique name of the command within the solution.
COMMAND_CLASS	The fully qualified class name in which the command is defined. This class is loaded and the execute() method representing the command is executed.

C.1.3.2 Example Implementation

This example shows the Managed Attachments command defined for the Managed Attachments solution.

Table C–6 Example AXF_COMMANDS Table

SOLUTION_NAMESPACE	COMMAND_CLASS	COMMAND_NAMESPACE
UCM_Managed_Attachments	oracle.imaging.axf.commands.ucm.AfGrantAccessCommand	UCM_Managed_Attachments

C.1.4 AXF_SOLUTION_PARAMETERS Table

This table defines command parameters for the solution and AXF command.

C.1.4.1 Column Description

Table C–7 Column Description for AXF_SOLUTION_PARAMETERS Table

Column	Description
SOLUTION_NAMESPACE	Identifies the solution namespace, as defined in the AXF_SOLUTIONS Table.
COMMAND_NAMESPACE	Specifies the command name, as defined in the AXF_COMMANDS Table.
CONFIGURATION_NAMESPACE	Used to implement the command. Specify the complete package name of the implementation class. This namespace path provides the physical Java class to instantiate. The namespace also differentiates commands within the same solution namespace.

Table C–7 (Cont.) Column Description for AXF_SOLUTION_PARAMETERS Table

Column	Description
PARAMETER_KEY	Specifies the parameter key to use in the AXF command. Parameters include: ▪ RIDC_CONNECTION_STR: Specifies the RIDC connection string used to execute the AF_GRANT_ACCESS Content Server service. Includes the host name or IP address of the system on which Content Server is running, and the Content Server port that receives RIDC calls. (To find the value for the Content Server port, locate the IntradocServerPort config value in config.cfg.) ▪ UCM_CONNECTION_STR: Specifies the base URL that executes the Content Server attachments framework search, which lists associated documents. This parameter also sets the following values: Host name or IP address of the system on which Content Server is running Port on which the web server is listening. Content Server profile (default profile is **EBSProfile** or **PSFTProfile**. If using the universal adapter described in Section 7.4, specify **UniversalProfile**.) GET_SEARCH_RESULTS_FORCELOGIN service ResultTemplate (default template is **EBS_LIST** or **PSFT_LIST**. If using the universal adapter described in Section 7.4, use **UNIVERSAL_LIST**). ▪ UCM_ADMIN_USER: Specifies the administrative Content Server user that executes the AF_GRANT_ACCESS service for the user logged into the business application. ▪ MA_CSF_KEY: Stores the admin user credentials in the credential store framework (CSF) using a configurable key name. You can change the key name if needed. The same key name is used to create the credential in CSF, as described in Section 7.7.1.
PARAMETER_VALUE	Specifies the value of the parameter key.

C.1.4.2 Example Implementations

These examples define the UCM_Managed_Attachments command for the UCM_Managed_Attachments solution. Table C–8 provides an Oracle E-Business Suite example and Table C–9 provides an Oracle PeopleSoft example. Notice that the CONFIGURATION_NAMESPACE value for the AXF for BPM example shown in Table C–10 and Table C–11 replaces `imaging` with `ecm`, which is required for the AXF for BPM infrastructure. These values were set in Section 7.2.1.

- The first row specifies the RIDC connection string that executes the AF_GRANT_ACCESS Content Server service.
- The second row specifies the Managed Attachments URL that invokes the Content Server attachments framework search.
- The third row specifies the Content Server administrative user who runs the AF_GRANT_ACCESS service; this user name is dynamically retrieved.
- The fourth row specifies the CSF key. The username stored in the CSF is compared with the username value in the incoming request from the business application.

Fields not shown: SOLUTION_NAMESPACE=UCM_Managed_Attachments

Table C–8 Example AXF_SOLUTION_PARAMETERS Table for Oracle E-Business Suite Managed Attachments Solution (AXF for BPEL)

COMMAND_ NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
UCM_Managed_ Attachments	oracle.imaging.axf.commands.ucm. AfGrantAccessCommand	RIDC_CONNECTION_STR	idc://*Content Server host name or IP address:Content Server RIDC port*
UCM_Managed_ Attachments	oracle.imaging.axf.commands.ucm. AfGrantAccessCommand	UCM_CONNECTION_STR	http://*Content Server Host name or IP address:* *Content Server Port*/cs/idcplg/_ p/min/af/trigger-EBSProfile?IdcServi ce=GET_SEARCH_RESULTS_ FORCELOGIN&ResultTemplate=EBS_LIST
UCM_Managed_ Attachments	oracle.imaging.axf.commands.ucm. AfGrantAccessCommand	UCM_ADMIN_USER	*Content Server admin user*
UCM_Managed_ Attachments	oracle.imaging.axf.commands.ucm. AfGrantAccessCommand	MA_CSF_KEY	MA_CSF_KEY

Table C–9 Example AXF_SOLUTION_PARAMETERS Table for Oracle PeopleSoft Managed Attachments Solution (AXF for BPEL)

COMMAND_ NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
UCM_Managed_ Attachments	oracle.imaging.axf.commands.ucm. AfGrantAccessCommand	RIDC_CONNECTION_STR	idc://*Content Server host name or IP address:Content Server RIDC port*
UCM_Managed_ Attachments	oracle.imaging.axf.commands.ucm. AfGrantAccessCommand	UCM_CONNECTION_STR	http://*Content Server Host name or IP address:* *Content Server Port*/cs/idcplg/_ p/min/af/trigger-PSFTProfile?IdcServ ice=GET_SEARCH_RESULTS_ FORCELOGIN&ResultTemplate=PSFT_LIST
UCM_Managed_ Attachments	oracle.imaging.axf.commands.ucm. AfGrantAccessCommand	UCM_ADMIN_USER	*Content Server admin user*
UCM_Managed_ Attachments	oracle.imaging.axf.commands.ucm. AfGrantAccessCommand	MA_CSF_KEY	MA_CSF_KEY

Table C–10 Example AXF_SOLUTION_PARAMETERS Table for Oracle E-Business Suite Managed Attachments Solution (AXF for BPM)

COMMAND_ NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
UCM_Managed_ Attachments	oracle.ecm.axf.commands.ucm.AfG rantAccessCommand	RIDC_CONNECTION_STR	idc://*Content Server host name or IP address:Content Server RIDC port*
UCM_Managed_ Attachments	oracle.ecm.axf.commands.ucm.AfG rantAccessCommand	UCM_CONNECTION_STR	http://*Content Server Host name or IP address:* *Content Server Port*/cs/idcplg/_ p/min/af/trigger-EBSProfile?IdcServi ce=GET_SEARCH_RESULTS_ FORCELOGIN&ResultTemplate=EBS_LIST
UCM_Managed_ Attachments	oracle.ecm.axf.commands.ucm.AfG rantAccessCommand	UCM_ADMIN_USER	*Content Server admin user*
UCM_Managed_ Attachments	oracle.ecm.axf.commands.ucm.AfG rantAccessCommand	MA_CSF_KEY	MA_CSF_KEY

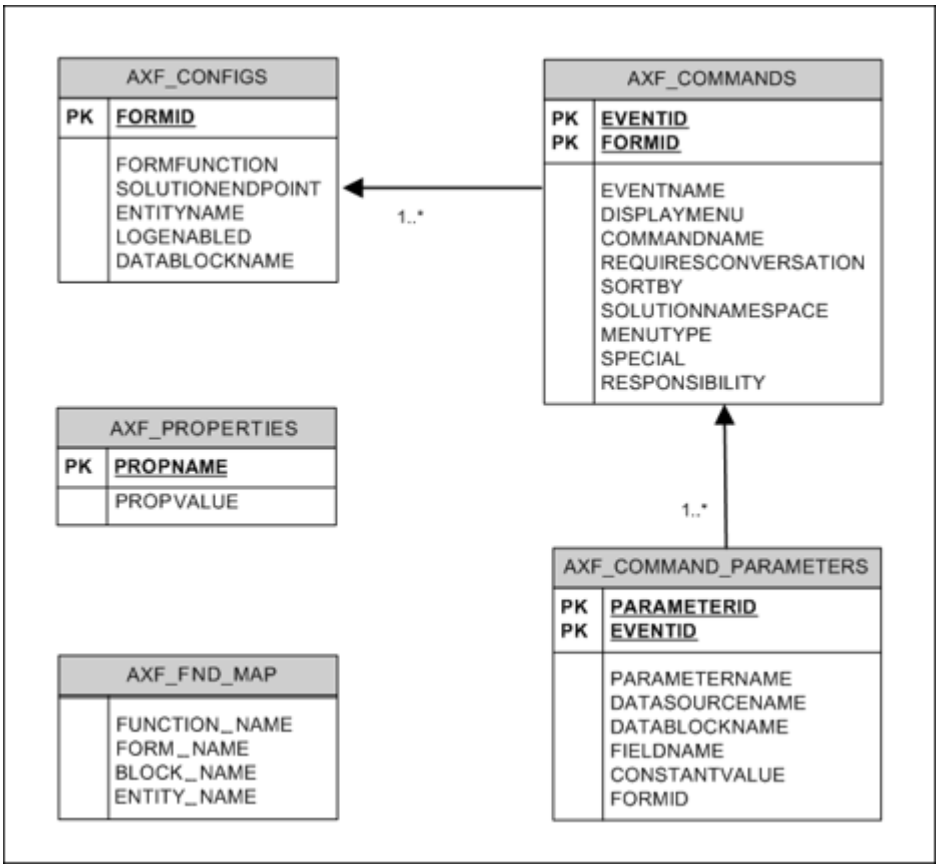
Table C–11 Example AXF_SOLUTION_PARAMETERS Table for Oracle PeopleSoft Managed Attachments Solution (AXF for BPM)

COMMAND_ NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
UCM_Managed_ Attachments	oracle.ecm.axf.commands.ucm.AfGrantAccessCommand	RIDC_CONNECTION_STR	idc://*Content Server host name or IP address:Content Server RIDC port*
UCM_Managed_ Attachments	oracle.ecm.axf.commands.ucm.AfGrantAccessCommand	UCM_CONNECTION_STR	http://*Content Server Host name or IP address:* *Content Server Port*/cs/idcplg/_p/min/af/trigger-PSFTProfile?IdcService=GET_SEARCH_RESULTS_FORCELOGIN&ResultTemplate=PSFT_LIST
UCM_Managed_ Attachments	oracle.ecm.axf.commands.ucm.AfGrantAccessCommand	UCM_ADMIN_USER	*Content Server admin user*
UCM_Managed_ Attachments	oracle.ecm.axf.commands.ucm.AfGrantAccessCommand	MA_CSF_KEY	MA_CSF_KEY

C.2 Oracle E-Business Suite Forms Tables For the Managed Attachments Solution

Oracle E-Business Suite Forms tables define how the Managed Attachments page is activated through the Zoom menu on selected Oracle E-Business Suite forms. Each Oracle E-Business Suite form enabled for the AF integration requires an AXF Oracle E-Business Suite configuration that defines a Zoom Menu item with the label Managed Attachments and a set of parameters that include the Oracle E-Business Suite instance name, business object type, business object key(s), and user friendly description of the business object instance.

The diagram that follows displays the Oracle E-Business Suite Forms configuration tables used for the Managed Attachments solution and their relationships.



Oracle E-Business Suite Forms Tables	Description
AXF_CONFIGS Table (Oracle E-Business Suite Forms)	Enables the AXF solution on various Oracle E-Business Suite Forms.
AXF_PROPERTIES Table (Oracle E-Business Suite Forms)	Defines properties for AXF integration for Oracle E-Business Suite.
AXF_FND_MAP Table (Oracle E-Business Suite Forms)	Defines Oracle E-Business Suite form values to pass to the AfGrantAccessCommand when a user activates the Managed Attachments functionality from an Oracle E-Business Suite form.
AXF_COMMANDS Table (Oracle E-Business Suite Forms)	Describes the actions to take based on user activity.
AXF_COMMAND_PARAMETERS Table (Oracle E-Business Suite Forms)	Defines the information sent for the AfGrantAccess command.

C.2.1 AXF_CONFIGS Table (Oracle E-Business Suite Forms)

Use the AXF_CONFIGS table to enable the AXF solution on various Oracle E-Business Suite Forms. This table provides a fine level of granularity when selecting which forms to AXF-enable.

When an action occurs, the customized code launches the specified solution and command configured for the event. When configured for the Oracle E-Business Suite adapter for Content Server, this table invokes the AfGrantAccess command.

C.2.1.1 Column Description

Table C–12 Column Description for AXF_CONFIGS Table

Column Name	Description
FORMID	Specifies the primary key of the table.
FORMFUNCTION	Distinguishes each Oracle E-Business Suite Form based on the form's functionality.
SOLUTIONENDPOINT	Specifies a URL to AXF, and defines if using the AXF for BPM or AXF for BPEL infrastructure. If specifying for AXF for BPM, enter a 2 at the end of this column, as shown in Table C–14 This value should start as `https://` if SecureMode is on. See Section 2.4.2.
ENTITYNAME	Used by the attachment functionality as a unique name, which links attachments to the correct Forms.
LOGENABLED	Enables or disables the log for the specified form. See Section 2.5. Specify one of the following: ■ 1/TRUE/YES ■ 0/FALSE/NO
DATABLOCKNAME	Specify the data block on the form to enable. Note that you can also specify `AXF_DEFAULT` to enable all data blocks on the form. A Form may be reused by Oracle E-Business Suite (for example, Invoice Entry and Invoice Query); the FORMFUNCTION and DATABLOCKNAME together uniquely identify each form.

C.2.1.2 Example Implementation

These examples define the AfGrantAccess command in the AXF_CONFIGS table for a form. Notice that the SolutionEndpoint value for the AXF for BPM example shown in Table C–14 contains a `2` after `axf-ws`, which is required for the AXF for BPM infrastructure.

Fields not shown: ENTITYNAME=(null), LOGENABLED=YES, and DATABLOCKNAME=(null)

Table C–13 Example AXF_CONFIGS Table (AXF for BPEL)

FORMID	FORMFUNCTION	SOLUTIONENDPOINT
1	AXF_MANAGED_ ATTACHMENTS	`https://`*`ApplicationServerName:Port`*`/axf-ws/AxfSolutionMediatorService`

Table C–14 Example AXF_CONFIGS Table (AXF for BPM)

FORMID	FORMFUNCTION	SOLUTIONENDPOINT
1	AXF_MANAGED_ ATTACHMENTS	`https://`*`ApplicationServerName:Port`*`/axf-ws2/AxfSolutionMediatorService`

C.2.2 AXF_PROPERTIES Table (Oracle E-Business Suite Forms)

Use the AXF_PROPERTIES table to define properties for AXF integration for Oracle E-Business Suite. You can also use its AXF_PAPERCLIP property to enable or disable the paperclip (display attached document) feature, as described in Section 7.6.15.

C.2.2.1 Column Description

Table C–15 Column Description for AXF_PROPERTIES Table (Oracle E-Business Suite)

Column	Description
PROPNAME	Specifies properties to use. Properties include: ■ SecureMode: To enable SSL, set this property to ON, and set values for AXFWalletPath and AXFWalletKey properties. ■ AXFWalletPath: Certificate location (path). ■ AXFWalletKey: Specify the key for the AXF wallet password, to be retrieved from the Oracle E-Business Suite vault. ■ AXF_VERSION: Specify `1` for AXF 10g, or `2` for AXF 11g. ■ AXF_SOAP_POLICY: Specify the name of the SOAP policy to use. The currently supported policy is `USER_NAME_TOKEN`. ■ AXF_SOAP_SECURITY: Specify `TRUE` to enable SOAP security, as described in Section 2.4.1. This setting is required. ■ AXF_SOAP_USER: Specify the SOAP userid used in the SOAP header for authentication, as described in Section 2.4.1. ■ AXF_PAPERCLIP: Set to `TRUE` to enable the Oracle E-Business Suite native attachments paperclip option, or `FALSE` (default) to disable it. See Section 7.6.15.
PROPVALUE	Specifies the property's value.

C.2.2.2 Example Implementation

This example table shows the default properties values.

Table C–16 Example AXF_PROPERTIES Table

PROPNAME	PROPVALUE
SecureMode	OFF
AXFWalletPath	file:*walletpath*
AXFWalletKey	*AXFWalletKey*
AXF_VERSION	2
AXF_SOAP_POLICY	USER_NAME_TOKEN
AXF_SOAP_SECURITY	TRUE
AXF_SOAP_USER	weblogic
AXF_PAPERCLIP	FALSE

C.2.3 AXF_FND_MAP Table (Oracle E-Business Suite Forms)

This table relates to an Oracle E-Business Suite form's values passed to the AfGrantAccessCommand when a user activates the Managed Attachments functionality from an Oracle E-Business Suite form. The adapter looks up values for the Oracle E-Business Suite form in this table and passes them to the AfGrantAccessCommand for executing the Content Server AF_GRANT_ACCESS service and attachments framework search.

C.2.3.1 Column Description

Table C–17 Column Description for AXF_FND_MAP Table

Column	Description
FUNCTION_NAME	Defines the Oracle E-Business Suite Form based on its functionality.
FORM_NAME	Defines the name of the Oracle E-Business Suite form to enable.
BLOCK_NAME	Defines the data block on the form to enable.
ENTITY_NAME	Used by the attachment functionality as a unique name, which links attachments to the correct forms.

C.2.3.2 Example Implementation

The AXF_FND_MAP example that follows displays fields configured for the AfGrantAccess command for the Invoice Entry form.

Table C–18 Example Implementation for AXF_FND_MAP Table

FUNCTION_NAME	FORM_NAME	BLOCK_NAME	ENTITY_NAME
AP_APXINWKB	APXINWKB	INV_SUM_FOLDER	AP_INVOICES

C.2.4 AXF_COMMANDS Table (Oracle E-Business Suite Forms)

Use the AXF_COMMANDS table to describe the actions to take based on user activity. This table works with the AXF_CONFIGS Table (Oracle E-Business Suite Forms).

C.2.4.1 Column Description

Table C–19 Column Description for AXF_COMMANDS Table

Column Name	Description	Nullable
FORMID	Links to the AXF_CONFIGS Table (Oracle E-Business Suite Forms).	No
EVENTID	Primary key of the table.	Yes
EVENTNAME	Name of the Event command to invoke (ZOOM for this adapter).	Yes
DISPLAYMENU	Displays text of the menu for the command.	Yes
COMMANDNAMESPACE	Request command to pass to the back-end when a user selects the menu.	Yes
REQUIRESCONVERSATION	Indicates if the command requires a valid conversation or not. For this adapter, this value must be NO.	Yes
SORTBY	Order in which the menu is displayed.	Yes
SOLUTIONNAMESPACE	Name of the solution.	Yes
MENUTYPE	Specify the menu type to display to users in Oracle E-Business Suite. ZOOM displays a Zoom menu in the toolbar.	Yes
SPECIAL	Create new menu entries by entering a unique number for the Special type menu. (Not applicable for this adapter.)	Yes
RESPONSIBILITY	Use this column to filter the menu options based on user responsibility. Enter a value to display the menu only to end users with responsibilities associated with that value. (Not applicable for this adapter.)	Yes

C.2.4.2 Example Implementation

The example AXF_COMMANDS table that follows displays fields for an AfGrantAccess command configuration for the Invoice Entry form.

Fields not shown: SPECIAL=(null), RESPONSIBILITY=(null)

Table C–20 Example AXF_COMMANDS Table for Invoice Entry Form

EVENT ID	FORM ID	EVENT NAME	DISPLAYMENU	COMMAND NAMESPACE	REQUIRESCONVERSATION	SORT BY	SOLUTION NAMESPACE	MENU TYPE
8	1	ZOOM	Managed Attachments	UCM_Managed_Attachment	NO	3	UCM_Managed_Attachments	ZOOM

C.2.5 AXF_COMMAND_PARAMETERS Table (Oracle E-Business Suite Forms)

Use the AXF_COMMAND_PARAMETERS table to define the information sent for each defined command. Each command may require or omit a different set of parameters.

C.2.5.1 Column Description

Table C–21 Column Description for AXF_COMMAND_PARAMETERS Table

Column	Description
PARAMETERID	Defines a unique ID for the parameter.
EVENTID	Defines a unique ID for the event. Comes from the AXF_COMMANDS Table (Oracle E-Business Suite Forms).
PARAMETERNAME	The name of the parameter to pass, where: ■ application: Specifies the business application, as defined in the AFObjects Table. This name/value pair is passed as a configuration parameter to the Content Server AF_GRANT_ACCESS service and attachments framework search. ■ extra*ParamName*: Extra parameter to be passed as metadata from the business application to Content Server in check-in or scan forms. See Section 7.6.16.
DATASOURCENAME	Data Source for the parameter value. You can specify *Data* or *Constant*.
DATABLOCKNAME	Data Block of the Form from which the value is fetched
FIELDNAME	Field Name in the form from which the value is fetched.
CONSTANTVALUE	Must be set to a value that uniquely identifies the Oracle E-Business Suite instance.

C.2.5.2 Example Implementation

The AXF_COMMAND_PARAMETERS example that follows displays fields configured for the AfGrantAccess command for the Invoice Entry form.

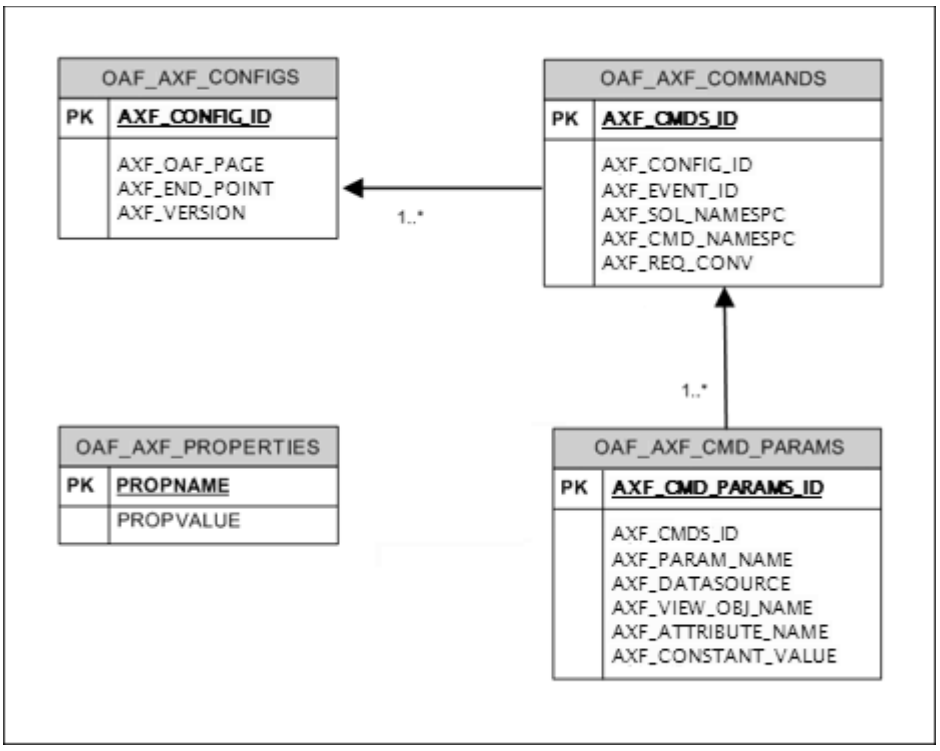
Table C–22 Example AXF_COMMAND_PARAMETERS Table

PARAMETER ID	EVENT ID	PARAMETERNAME	DATASOURCE NAME	DATABLOCKNAME	FIELDNAME	CONSTANTVALUE
1	8	application	CONSTANT	(null)	(null)	EBS_instanceA

C.3 Oracle E-Business Suite OAF Tables For the Managed Attachments Solution

Oracle E-Business Suite OAF tables define how the Managed Attachments page is activated through a button added via personalization to selected OAF pages, which redirects to the Managed Attachments URL in Content Server.

The diagram that follows displays the Oracle E-Business Suite OAF configuration tables used for the Managed Attachments solution and their relationships.



Oracle E-Business Suite OAF Table	Description
OAF_AXF_CONFIGS Table (Oracle E-Business Suite OAF)	Enables the AXF solution on various Oracle E-Business Suite OAF pages.
OAF_AXF_PROPERTIES Table (Oracle E-Business Suite OAF)	Defines properties for AXF integration for Oracle E-Business Suite.
OAF_AXF_COMMANDS Table (Oracle E-Business Suite OAF)	Describes the actions to take based on user activity.
OAF_AXF_CMD_PARAMS Table (Oracle E-Business Suite OAF)	Defines the information sent for the AfGrantAccess command.

C.3.1 OAF_AXF_CONFIGS Table (Oracle E-Business Suite OAF)

Use the OAF_AXF_CONFIGS table to store the AXF endpoints for a particular OAF page. You can define multiple endpoints for the same page.

When an action occurs, the customized code launches the specified solution and command configured for the event. When configured for the Oracle E-Business Suite adapter for Content Server, this table invokes the AfGrantAccess command.

C.3.1.1 Column Description

Table C–23 Column Description for OAF_AXF_CONFIGS Table

Column Name	Description
AXF_CONFIG_ID	Specifies the primary key of the table.
AXF_OAF_PAGE	Specifies the full path name of the Oracle E-Business Suite OAF page. See Section 7.2.3.2.
AXF_END_POINT	Specifies a URL to the AXF endpoint to which to connect. This value should start as `https://` if SecureMode is on. See Section 2.4.2.
AXF_VERSION	Specifies the AXF version of the connection, where `1` indicates AXF 10g, and `2` indicates AXF 11g.

C.3.1.2 Example Implementation

This example enables the AXF solution on the specified Oracle E-Business Suite OAF pages. Notice that the first two rows show two commands for the same page, each pointing to a different endpoint. These commands should be configured to perform different operations, such as a Managed Attachments button and an Invoice Processing button. (Note that it is recommended to place only one Managed Attachments button on an OAF page.)

Field not shown: AXF_VERSION=2

Table C–24 Example OAF_AXF_CONFIGS Table

AXF_CONFIG_ID	AXF_OAF_PAGE	AXF_END_POINT
1	/oracle/apps/icx/por/reqmgmt/webui/ReqDetailsPG	http://*ImagingServerHostname1*:*Port1*/axf-ws/AxfSolutionMediatorService
2	/oracle/apps/icx/por/reqmgmt/webui/ReqDetailsPG	http://*ImagingServerHostname2*:*Port2*/axf-ws/AxfSolutionMediatorService
3	/oracle/apps/po/document/order/webui/OrderPG	http://*ImagingServerHostname1*:*Port1*/axf-ws/AxfSolutionMediatorService
4	/oracle/apps/pos/changeorder/webui/PosViewPOPG	http://*ImagingServerHostname1*:*Port1*/axf-ws/AxfSolutionMediatorService
5	/oracle/apps/pos/supplier/webui/QuickUpdatePG	http://*ImagingServerHostname1*:*Port1*/axf-ws/AxfSolutionMediatorService
6	/oracle/apps/pon/negotiation/creation/webui/ponDefineHdrPG	http://*ImagingServerHostname1*:*Port1*/axf-ws/AxfSolutionMediatorService
7	/oracle/apps/pon/negotiation/creation/webui/ponItemHeaderPG	http://*ImagingServerHostname1*:*Port1*/axf-ws/AxfSolutionMediatorService
8	/oracle/apps/pa/project/webui/ProjAttachmentPG	http://*ImagingServerHostname1*:*Port1*/axf-ws/AxfSolutionMediatorService

C.3.2 OAF_AXF_PROPERTIES Table (Oracle E-Business Suite OAF)

Use the OAF_AXF_PROPERTIES table to define properties for AXF integration for Oracle E-Business Suite OAF applications.

C.3.2.1 Column Description

Table C–25 Column Description for OAF_AXF_PROPERTIES Table (Oracle E-Business Suite)

Column	Description
PROPNAME	Specifies properties to use. Properties include: ■ SecureMode: To enable SSL, set this property to ON, and set values for AXFWalletPath and AXFWalletKey properties. ■ AXFWalletPath: Certificate location (path). ■ AXFWalletKey: Specify the key for the AXF wallet password, to be retrieved from the Oracle E-Business Suite vault. ■ AXF_SOAP_POLICY: Specify the name of the SOAP policy to use. The currently supported policy is `USER_NAME_TOKEN`. ■ AXF_SOAP_SECURITY: Specify `TRUE` to enable SOAP security, as described in Section 2.4.1. This setting is required. ■ AXF_SOAP_USER: Specify the SOAP userid used in the SOAP header for authentication, as described in Section 2.4.1. ■ AXF_OAF_REDIRECT_PAGE_URL: Specifies the URL to the custom intermediate OAF page. This URL is configured in Section 2.3.1. It begins with `https://` if SSL is configured on the Oracle E-Business Suite instance.
PROPVALUE	Specifies the property's value.

C.3.2.2 Example Implementation

This example table shows the default properties values.

Table C–26 Example OAF_AXF_PROPERTIES Table

PROPNAME	PROPVALUE
SecureMode	OFF
AXFWalletPath	file:*walletpath*
AXFWalletKey	*AXFWalletKey*
AXF_SOAP_POLICY	USER_NAME_TOKEN
AXF_SOAP_SECURITY	TRUE
AXF_SOAP_USER	weblogic
AXF_OAF_REDIRECT_PAGE_URL	http://*EBS Host : EBS Port*/OA_HTML/OA.jsp?page=/oracle/apps/ak/ucm/axf/webui/RedirectToAxfPG

C.3.3 OAF_AXF_COMMANDS Table (Oracle E-Business Suite OAF)

This table stores the different commands configured for a particular OAF page. Each command is associated with an EVENT_ID, which refers to the ID of the button/link embedded on an OAF page via personalization (configured in Section 2.3).

C.3.3.1 Column Description

Table C–27 Column Description for OAF_AXF_COMMANDS Table

Column Name	Description
AXF_CMDS_ID	Specifies the primary key of the table, used to identify the unique command.
AXF_CONFIG_ID	Specifies the foreign key to the OAF_AXF_CONFIGS Table (Oracle E-Business Suite OAF), which associates this unique command with a particular page and component on the page.
AXF_EVENT_ID	Specifies the event being executed in the user interface. The AXF_EVENT_ID corresponds to the ID for the button or link added via personalization to invoke AXF functionality. Available options include: ■ MAButton (for Managed Attachments) ■ IPM_BUTTON (for imaging)
AXF_SOL_NAMESPC	Identifies the solution namespace for the command to execute.
AXF_CMD_NAMESPC	Identifies the command namespace for the command to execute.
AXF_REQ_CONV	Specifies if a conversation is required for this command before execution.

C.3.3.2 Example Implementation

This example shows the UCM_Managed_Attachments command defined for multiple OAF pages. Each command is associated with an EVENT_ID, which refers to the ID of the button/link embedded on an OAF page via personalization. Notice that the second row configures an Imaging (IPM) button that links to the SupplierMaintenance command of the Imaging solution; this type of configuring is described in Section 2.3.3.

Field not shown: AXF_REQ_CONV=N

Table C–28 Example OAF_AXF_COMMANDS Table

AXF_CMDS_ID	AXF_CONFIG_ID	AXF_EVENT_ID	AXF_SOL_NAMESP	AXF_CMD_NAMESPC
1	1	MAButton	UCM_Managed_Attachments	UCM_Managed_Attachments
2	2	IPM_BUTTON	SupplierMaintenance	StartSupplierMaintenance
3	3	MAButton	UCM_Managed_Attachments	UCM_Managed_Attachments
4	4	MAButton	UCM_Managed_Attachments	UCM_Managed_Attachments
5	5	MAButton	UCM_Managed_Attachments	UCM_Managed_Attachments
6	6	MAButton	UCM_Managed_Attachments	UCM_Managed_Attachments
7	7	MAButton	UCM_Managed_Attachments	UCM_Managed_Attachments
8	8	MAButton	UCM_Managed_Attachments	UCM_Managed_Attachments

C.3.4 OAF_AXF_CMD_PARAMS Table (Oracle E-Business Suite OAF)

Each command is associated with different parameters that are required for the SOAP call. The parameters can be constant values or they can refer to the view attribute name. The attributes values can be obtained from the view object from the page context.

C.3.4.1 Column Description

Table C–29 Column Description for OAF_AXF_CMD_PARAMS Table

Column	Description
AXF_CMD_PARAMS_ID	Specifies the primary key of the table.
AXF_CMDS_ID	Specifies the foreign key to the OAF_AXF_COMMANDS Table (Oracle E-Business Suite OAF), which associates the unique parameter with a specific command.
AXF_PARAM_NAME	Defines the parameter name, where: ■ application: Specifies the business application, as defined in the AFObjects Table. ■ businessObjectType: Specifies the business application's object type, as defined in the AFObjects Table. ■ businessObjectKey: Specifies the business application's object key, as defined in the AFObjects Table. ■ businessObjectValue1...5: Represents the primary key value for the document. ■ labelValue1...5: Labels displayed on Managed Attachments user interface for displaying key values for the attachments integration. ■ extra*ParamName*: Extra parameter to be passed as metadata from the business application to Content Server in check-in or scan forms. See Section 7.6.16.
AXF_DATASOURCE	Specifies where the parameter value is retrieved. ■ DATA: Retrieves the value in Oracle PeopleSoft defined by the AXF_RECORD_NAME and AXF_FIELD_NAME fields. ■ CONSTANT: Uses the value defined in the AXF_CONSTANT_VALUE.
AXF_VIEW_OBJ_NAME	Specifies the OAF ViewObject name, which is used to obtain the command parameter values at runtime. See Section 7.2.3.2.
AXF_ATTRIBUTE_NAME	Used when the AXF_DATASOURCE is set to DATA. This attribute name retrieves the actual runtime value from the view object in the controller class.
AXF_CONSTANT_VALUE	Used as the constant value when AXF_DATASOURCE is set to CONSTANT.

C.3.4.2 Example Implementation

The OAF_AXF_CMD_PARAMS example that follows displays fields configured for the AfGrantAccess command for multiple OAF pages whose key values you identified in Section 7.2.3.2.

Table C–30 Example OAF_AXF_CMD_PARAMS Table

AXF_CMD_PARAMS_ID	AXF_CMDS_ID	AXF_PARAM_NAME	AXF_DATASOURCE	AXF_VIEW_OBJ_NAME	AXF_ATTRIBUTE_NAME	AXF_CONSTANT_VALUE
1	1	application	CONSTANT			EBS_instanceA
2	1	businessObjectType	CONSTANT			REQ_HEADERS
3	1	businessObjectKey1	CONSTANT			REQUISITION_HEADER_ID

Table C–30 (Cont.) Example OAF_AXF_CMD_PARAMS Table

AXF_CMD_PARAMS_ID	AXF_CMDS_ID	AXF_PARAM_NAME	AXF_DATASOURCE	AXF_VIEW_OBJ_NAME	AXF_ATTRIBUTE_NAME	AXF_CONSTANT_VALUE
4	1	businessObjectValue1	DATA	ReqHeaderVO	RequisitionHeaderId	
5	1	labelValue1	DATA	ReqHeaderVO	PreparerName	
6	1	labelValue2	DATA	ReqHeaderVO	ReqNum	
7	1	businessObjectContext	CONSTANT			ScanContext
8	3	application	CONSTANT			EBS_instanceA
9	3	businessObjectType	CONSTANT			PO_HEADERS_MERGE_V
10	3	businessObjectKey1	CONSTANT			PO_HEADER_ID
11	3	businessObjectValue1	DATA	POHeaderMergeVO	POHeaderId	
12	4	application	CONSTANT			EBS_instanceA
13	4	businessObjectType	CONSTANT			PO_HEADERS
14	4	businessObjectKey1	CONSTANT			PO_HEADER_ID
15	4	businessObjectValue1	DATA	PosViewHeadersVO	PO_HEADER_ID	
16	5	application	CONSTANT			EBS_instanceA
17	5	businessObjectType	CONSTANT			PO_VENDORS
18	5	businessObjectKey1	CONSTANT			VENDOR_ID
19	5	businessObjectValue1	DATA	SupplierVO	VendorID	
20	6	application	CONSTANT			EBS_instanceA
21	6	businessObjectType	CONSTANT			PON_AUCTION_HEADERS_ALL
22	6	businessObjectKey1	CONSTANT			AUCTION_HEADER_ID
23	6	businessObjectValue1	DATA	AuctionHeadersAllVO	AuctionHeaderId	
24	7	application	CONSTANT			EBS_instanceA
25	7	businessObjectType	CONSTANT			PON_AUCTION_ITEM_PRICES_ALL
26	7	businessObjectKey1	CONSTANT			AUCTION_HEADER_ID
27	7	businessObjectValue1	DATA	ActionItemPricesAllVO	AuctionHeaderId	
28	7	businessObjectKey2	CONSTANT			LINE_NUMBER
29	7	businessObjectValue2	DATA	AuctionItemPricesAllVO	LineNumbers	
30	8	application	CONSTANT			EBS_instanceA
31	8	businessObjectType	CONSTANT			PA_PROJECTS
32	8	businessObjectKey1	CONSTANT			PROJECT_ID
33	8	businessObjectValue1	DATA	ProjAttachmentsVO	PROJECT_ID	

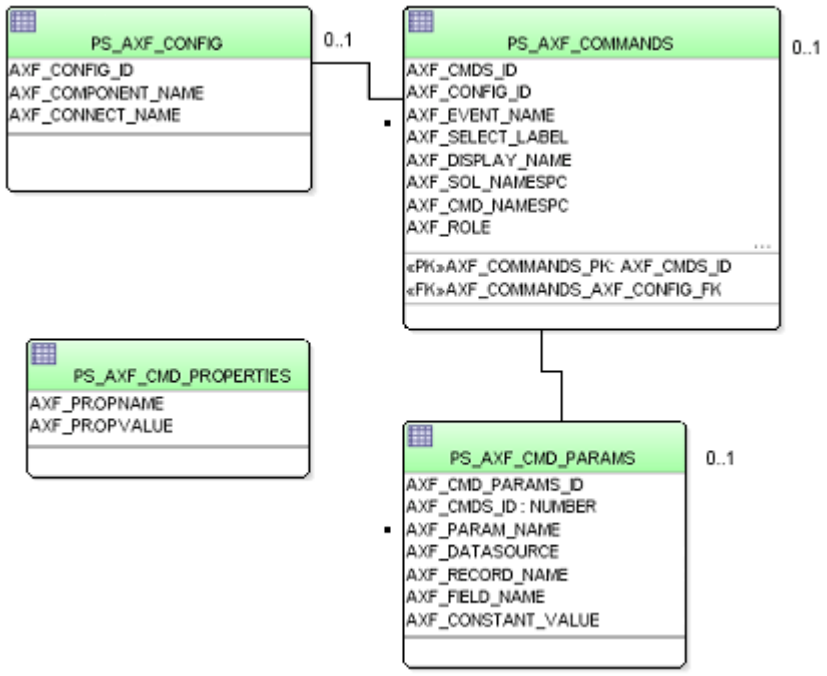
C.4 Oracle PeopleSoft Tables For the Managed Attachments Solution

Oracle PeopleSoft tables define how the Managed Attachments page is activated through a button, link, or menu on selected Oracle PeopleSoft pages.

Configuring AXF for Oracle PeopleSoft requires configuring these AXF-related tables in Oracle PeopleSoft:

- Section C.4.1, "PS_AXF_CONFIG Table"
- Section C.4.2, "PS_AXF_COMMANDS Table"
- Section C.4.3, "PS_AXF_CMD_PARAMS Table"
- Section C.4.4, "PS_AXF_PROPERTIES Table"

The following diagram shows how the tables used by the Oracle PeopleSoft system in AXF solutions are related.



Note: All AXF table fields require non-null values. You can include a space for fields that do not apply.

C.4.1 PS_AXF_CONFIG Table

Use the PS_AXF_CONFIG table to enable the AXF solution on various Oracle PeopleSoft components. This table provides a fine level of granularity when selecting which pages and components to AXF-enable.

Events are automatically invoked when an action is performed on an Oracle PeopleSoft page. The AXF_PS_Integration project component catches PRE- and POST-save events and makes them available for customization. You can decide which events to use and how and when to use them.

C.4.1.1 Column Description

Table C–31 Column Description for PS_AXF_CONFIG Table

Column	Description
AXF_CONFIG_ID	Specifies the primary key of the table.

Table C–31 (Cont.) Column Description for PS_AXF_CONFIG Table

Column	Description
AXF_COMPONENT_NAME	Specifies the name of the Oracle PeopleSoft component being enabled.
AXF_CONNECT_NAME	Specifies the Oracle PeopleSoft Integration Broker Connection name (service operation to call), as defined in the Oracle PeopleSoft Integration Broker administration interface. The default service operation is AXF_EXECUTE. You can set up and use other connections.
AXF_VERSION	Specifies the AXF version of the connection. This value should be set to `11g`.

C.4.1.2 Example Implementation

This example defines that the WM_WO (work order) component is AXF-enabled.

Table C–32 Example PS_AXF_CONFIG Table

AXF_CONFIG_ID	AXF_COMPONENT_NAME	AXF_CONNECT_NAME	AXF_VERSION
101	WM_WO	AXF_EXECUTE	11g
102	CA_HDR_PNG	AXF_EXECUTE	11g
103	REQUISITIONS	AXF_EXECUTE	11g

C.4.2 PS_AXF_COMMANDS Table

Use the PS_AXF_COMMANDS table to describe the actions to take based on user activity. This table works with the PS_AXF_CONFIG Table.

C.4.2.1 Column Description

Table C–33 Column Description for PS_AXF_COMMANDS Table

Column	Description
AXF_CMDS_ID	Specifies the primary key of the table, used to identify the unique command.
AXF_CONFIG_ID	Specifies the foreign key to the PS_AXF_CONFIG Table, which associates this unique command with a particular page and component.
AXF_EVENT_NAME	Specifies the event being executed. The AXF_EVENT_NAME corresponds to the subpage incorporated into an existing Oracle PeopleSoft page, such as a button, link, or menu added to an Oracle PeopleSoft page to invoke AXF functionality such as the Image Viewer or Task List. Available options include: ▪ AXF_BUTTON_1, …, AXF_BUTTON_5 ▪ AXF_LINK_1, ..., AXF_LINK5 ▪ AXF_COMBO_1, ..., AXF_COMBO_5 ▪ AXF_PRE_SAVE_SBP ▪ AXF_POST_SAVE_SBP
AXF_SELECT_LABEL	Represents the label that for display if using a drop-down component (for example, AXF_COMBO_1).

Table C–33 (Cont.) Column Description for PS_AXF_COMMANDS Table

Column	Description
AXF_DISPLAY_NAME	Defines the name of the button or link to display on the Oracle PeopleSoft page.
AXF_SOL_NAMESPC	Identifies the solution namespace for the command to execute.
AXF_CMD_NAMESPC	Identifies the command namespace for the command to execute.
AXF_PSFT_ROLE	Identifies the Oracle PeopleSoft roles with access to the command. Include roles in a comma-delimited list with each role enclosed in single quotes (for example, 'Role1','Role2','Role3')
AXF_SORT_ORDER	Specifies the order of items displayed in a selection field. Sort order applies to selection fields only.
AXF_REQ_CONV	Specifies if a conversation is required for this command before execution. For example, execution of the UpdateTask command requires a conversation be running in order for the user to select a current task.

C.4.2.2 Example Implementation

This example shows the UCM_Managed_Attachments command added as a link to an Oracle PeopleSoft page.

You must specify an Oracle PeopleSoft Role in the AXF_PSFT_ROLE field to give permissions to use the commands. If a person does not have proper permissions to use the commands, the commands do not display. If the commands display but do not function, this indicates that the commands are not configured properly.

Table C–34 Example PS_AXF_COMMANDS Table

AXF_CMDS_ID	AXF_CONFIG_ID	AXF_EVENT_NAME	AXF_SELECT_LABEL	AXF_DISPLAY_NAME	AXF_SOL_NAMESPC	AXF_CMD_NAMESPC	AXF_PSFT_ROLE	AXF_SORT_ORDER	AXF_REQ_CONV
101	101	AXF_LINK_1	(null)	Managed Attachments	UCM_Managed_Attachments	UCM_Managed_Attachments	'Employee'	1	N
102	102	AXF_LINK_1	(null)	Managed Attachments	UCM_Managed_Attachments	UCM_Managed_Attachments	'Employee'	1	N
103	103	AXF_LINK_1	(null)	Managed Attachments	UCM_Managed_Attachments	UCM_Managed_Attachments	'Employee'	1	N

C.4.3 PS_AXF_CMD_PARAMS Table

Use the PS_AXF_COMMAND_PARAMS table to specify the information sent for each defined command. Each command may require or omit a different set of parameters.

Use the PS_AXF_CMD_PARAMS Table to configure a label for the list of attachments. You can use a combination of constant and data values to display key information. Similar to configuring labels, you can also add rows to pass in extra metadata values, as described in Section 7.6.16.

C.4.3.1 Column Description

Table C–35 Column Description for PS_AXF_CMD_PARAMS Table

Column	Description
AXF_CMD_PARAM_ID	Specifies the primary key of the table.
AXF_CMD_ID	Specifies the foreign key to the PS_AXF_COMMANDS Table, which associates the unique parameter with a specific command.
AXF_PARAM_NAME	Defines the parameter name, where: ■ application: Specifies the business application, as defined in AFObjects Table. ■ businessObjectType: Specifies the business application's object type, as defined in Section 3.3. ■ businessObjectKey1...5: Represents the primary key name for the document. ■ businessObjectValue1...5: Represents the primary key value for the document. ■ labelValue1...5: Labels displayed on Managed Attachments user interface for displaying key values for the attachments integration. ■ extra*ParamName*: Extra parameter to be passed as metadata from the business application to Content Server in check-in or scan forms. See Section 7.6.16.
AXF_DATASOURCE	Specifies where the parameter value is retrieved. ■ DATA: Retrieves the value in Oracle PeopleSoft defined by the AXF_RECORD_NAME and AXF_FIELD_NAME fields. ■ CONSTANT: Uses the value defined in the AXF_CONSTANT_VALUE field.
AXF_RECORD_NAME	Identifies the record of the field in the Oracle PeopleSoft page to use as the target value to retrieve when AXF_DATASOURCE is set to DATA.
AXF_FIELD_NAME	Used as the constant value when AXF_DATASOURCE is set to DATA.
AXF_CONSTANT_VALUE	Used as the constant value when AXF_DATASOURCE is set to CONSTANT.

C.4.3.2 Example Implementation

This example shows the information sent for the UCM_Managed_Attachments command, including its application and business object type.

- The labelValue entries are configured to display key labels/values for the attachment. For example, the labelValues configuration in this table might display the following for work orders on the Managed Attachments page:

  [Business Unit], [US001], [Work Order], [123456]

Table C–36 Example PS_AXF_CMD_PARAMS Table

AXF_CMD_PARAM_ID	AXF_CMD_ID	AXF_PARAM_NAME	AXF_DATASOURCE	AXF_RECORD_NAME	AXF_FIELD_NAME	AXF_CONSTANT_VALUE
101	101	application	CONSTANT			PSFT_INSTANCE_A
102	101	businessObjectType	CONSTANT			WM_WO_HDR
103	101	businessObjectKey1	CONSTANT			BUSINESS_UNIT
104	101	businessObjectValue1	DATA	WM_WO_HDR	BUSINESS_UNIT	
105	101	businessObjectKey2	CONSTANT			WO_ID
106	101	businessObjectValue2	DATA	WM_WO_HDR	WO_ID	
107	101	labelValue1	CONSTANT			Business Unit
108	101	labelValue2	DATA	WM_WO_HDR	BUSINESS_UNIT	
109	101	labelValue3	CONSTANT			Work Order
110	101	labelValue4	DATA	WM_WO_HDR	WO_ID	
111	101	businessObjectContext	CONSTANT			ScanContext
112	102	application	CONSTANT			PSFT_INSTANCE_A
113	102	businessObjectType	CONSTANT			CA_CONTR_HDR
114	102	businessObjectKey1	CONSTANT			BUSINESS_UNIT
115	102	businessObjectValue1	DATA	CA_CONTR_HDR	BUSINESS_UNIT	
116	102	businessObjectKey2	CONSTANT			CONTRACT_NUM
117	102	businessObjectValue2	DATA	CA_CONTR_HDR	CONTRACT_NUM	
118	102	labelValue1	CONSTANT			Business Unit
119	102	labelValue2	DATA	CA_CONTR_HDR	BUSINESS_UNIT	
120	102	labelValue3	CONSTANT			Contract Number
121	102	labelValue4	DATA	CA_CONTR_HDR	CONTRACT_NUM	
122	103	application	CONSTANT			PSFT_INSTANCE_A
123	103	businessObjectType	CONSTANT			REQ_HDR
124	103	businessObjectKey1	CONSTANT			BUSINESS_UNIT
125	103	businessObjectValue1	DATA	REQ_HDR	BUSINESS_UNIT	
126	103	businessObjectKey2	CONSTANT			REQ_ID
127	103	businessObjectValue2	DATA	REQ_HDR	REQ_ID	
128	103	labelValue1	CONSTANT			Business Unit
129	103	labelValue2	DATA	REQ_HDR	BUSINESS_UNIT	
130	103	labelValue3	CONSTANT			Requisition
131	103	labelValue4	DATA	REQ_HDR	REQ_ID	

C.4.4 PS_AXF_PROPERTIES Table

Use the PS_AXF_PROPERTIES table to define properties for AXF integration with Oracle PeopleSoft. See Section 7.6.17.

C.4.4.1 Column Description

Table C–37 Column Description for PS_AXF_PROPERTIES Table

Column	Description
AXF_PROPNAME	Specifies properties to use. Properties include: ■ WINDOW_HEIGHT: Specify the popup window's height in pixels. (This refers to any popup window that displays as part of the solution.) ■ WINDOW_WIDTH: Specify the popup window's width in pixels.
AXF_PROPVALUE	Specifies the property's value.

C.4.4.2 Example Implementation

This example table shows the default properties values.

Table C–38 Example PS_AXF_PROPERTIES Table

AXF_PROPNAME	AXF_PROPVALUE
WINDOW_HEIGHT	600
WINDOW_WIDTH	800

C.5 Content Server Tables For the Managed Attachments Solution

Content Server tables define how Content Server documents are mapped to Oracle E-Business Suite or Oracle PeopleSoft business objects and how users are temporarily granted access to documents associated with a particular business object.

The adapter utilizes the following additional database tables:

- Section C.5.1, "AFGrants Table"
- Section C.5.2, "AFObjects Table"
- Section C.5.3, "AFRelationshipAttributes Table"
- Section C.5.4, "AFKeys Table"

These tables are automatically populated at run-time. AFGrants is initially populated when the user accesses Content Server from the business application and the AF_GRANT_ACCESS service is run. AFObjects is populated when the user checks in a document to Content Server from the Managed Attachments page.

C.5.1 AFGrants Table

This table stores the grants given to users, allowing them to temporarily access documents associated with a particular business object.

C.5.1.1 Column Description

Table C–39 Column Description for AFGrants Table

Columns	Description
dUserName	Stores the name of the user.
dAFApplication	Stores the business application's instance name (for example, PSFT_Instance1).

Table C–39 (Cont.) Column Description for AFGrants Table

Columns	Description
dAFBusinessObject	Stores the business application's object
dAFBusinessObjectType	Stores the business application's object type
dPrivilege	Stores the privilege to grant to the user: R (read), W (write), D (delete), or A (admin). This parameter is optional. If not specified, the access level specified for the AppAdapterGrantPrivilege configuration variable is used, as described in Section 7.5.
dExpirationDate	Stores the date and time at which to expire this grant.
dMaxExpiration	Stores the time at which the maximum access period (in hours) expires.

C.5.2 AFObjects Table

This table maps Content Server documents to AF business objects, in an N-to-N relationship, which enables multiple content items to be associated with a single business object and multiple business objects to contain the same content item.

Each time a user attaches or detaches a document, a record is added or updated in this table.

C.5.2.1 Column Description

Table C–40 Column Description for AFObjects Table

Column	Description
dAFID	Stores the unique ID of each attachment.
dAFApplication	Stores the business application's instance name (for example, PSFT_Instance or EBS_Instance).
dAFBusinessObjectType	Stores the business application's object type (for example, CallRecords or HRMS).
dAFBusinessObject	Stores the business object's ID in the business application instance.
dDocName	Stores a content item's ID associated with the business object.
DID	Stores the ID of the document associated with the business entity. In revision-specific attachments mode, this ID is used in returning a specific version of the document. For more information, see Section 7.6.11.

C.5.3 AFRelationshipAttributes Table

This table stores relationship attributes associated with an attachment. For more information, see Section 7.6.10. Depending on the configuration of relationship attributes, a document can have zero or more relationship attributes.

C.5.3.1 Column Description

Table C–41 Column Description for AFRelationshipAttributes Table

Columns	Description
dAFID	Stores the unique ID of each attachment.

Table C–41 (Cont.) Column Description for AFRelationshipAttributes Table

Columns	Description
attribute	Stores the relationship attribute name.
value	Stores the value of the relationship attribute the user selected.

C.5.4 AFKeys Table

This table stores individual key values for later reference.

The business applications support up to five primary key/value combinations, which can be used instead of dAFBusinessObject to represent a business entity. In a Managed Attachments solution, these keys are mapped to dAFBusinessObject by concatenating the PK_Value fields separated by a | (pipe) character. The original PK_Key and PK_Values are logged to the AFKeys table before performing the mapping operation.

C.5.4.1 Column Description

Table C–42 Column Description for AFKeys Table

Columns	Description
dAFApplication	Stores the business application's instance name.
dAFBusinessObjectType	Stores the business application's object type.
dAFBusinessObject	Stores the business object's ID in the business application instance.
PK1_Key ... PK5_Key	Stores up to five key/value combinations.
PK1_Value ... PK5_Value	Stores up to five key/value combinations.