Oracle® Fusion Middleware

Administrator's Guide for Oracle PeopleSoft Adapter for Oracle Enterprise Content Management

11*g* Release 1 (11.1.1)

E17044-01

May 2010

ORACLE

Oracle Fusion Middleware Administrator's Guide for Oracle PeopleSoft Adapter for Oracle Enterprise Content Management, 11*g* Release 1 (11.1.1)

E17044-01

Primary Author: Sarah Howland

Contributor: Tom Albrecht, Sancho Pinto, Kevin de Smidt, Kevin Cocilo

Contents

3 Configuring Imaging Solution Components

4 Imaging Solution Tables

Index

Preface

The *Administrator's Guide for Oracle PeopleSoft Adapter for Oracle Enterprise Content Management* describes Oracle PeopleSoft solution configurations for Oracle Enterprise Content Management systems.

Audience

This document is intended for administrators configuring integration solutions between Oracle PeopleSoft and Oracle Enterprise Content Management systems.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible to all users, including users that are disabled. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at http://www.oracle.com/accessibility/.

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/support/contact.html or visit http://www.oracle.com/accessibility/support.html if you are hearing impaired.

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 screen, or text that you enter.

1

Solution Overview

This chapter covers the following topics:

1.1 About This Guide

This guide contains the following chapters:

- Chapter 1, "Solution Overview," introduces AXF components, the imaging solution, security and authentication, and adapter system requirements.
- Chapter 2, "Configuring PeopleSoft Components," describes how to configure AXF components on PeopleSoft pages.
- Chapter 3, "Configuring Imaging Solution Components," describes how to configure the BPEL Connection for the imaging solution.
- Chapter 4, "Imaging Solution Tables," describes the AXF and PeopleSoft configuration tables used for the Imaging Solution, including commands and web user interface tools, and provides example implementations.

1.2 About Application Extension Framework (AXF)

Oracle's Application Extension Framework (AXF) is a command-driven, web services integration between a business application such as PeopleSoft and a content management application such as Oracle Imaging and Process Management (Oracle I/PM). The open Java-based architecture of AXF allows integrators to configure and modify multiple business process solutions separate from the systems themselves, and to upgrade systems without affecting implemented AXF solutions.

The Application Extension Framework includes the following components:

AXF Solution Templates

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

Note: To obtain a solution template, contact your systems integrator, Oracle Consulting, or Oracle Support.

AXF Commands

AXF provides reusable commands for implementing functionality. The Imaging Solution uses multiple AXF commands, as described in "About AXF Commands" on page 1-5.

AXF Web Tools

AXF provides web interface components for display to users, such as a task list and task viewer. Configured through the AXF tables, these web tools are described in "About AXF Web User Tools" on page 1-6.

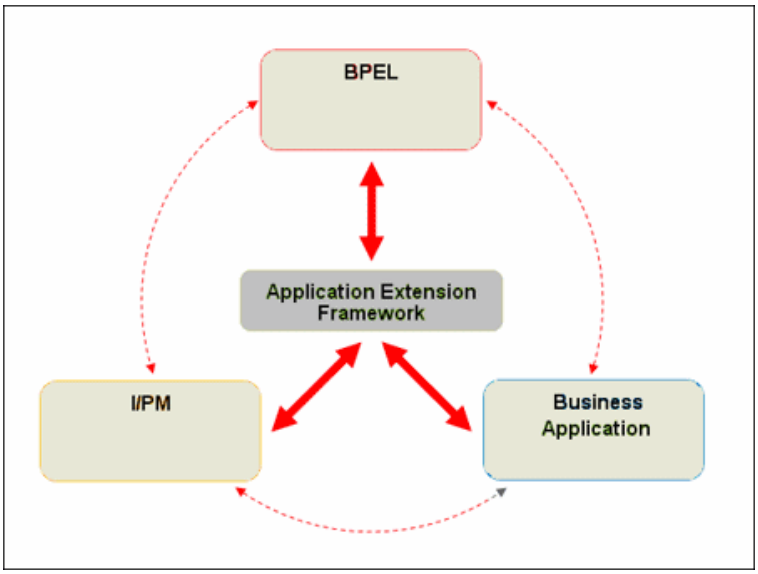
AXF Configuration Database Tables

You configure AXF solutions, commands, and web tools by configuring the AXF database tables. or information about each table and example implementations for the solution, see "Imaging Solution Tables" on page 4-1.

1.3 About the Imaging Solution

A workflow imaging solution is an integration between a business application such as PeopleSoft and BPEL (Oracle BPEL Process Manager), using Oracle I/PM as the imaging source, as illustrated in Figure 1–1. Through an AXF configuration, business users can process associated images and perform document-centric workflow tasks from their business application user interface.

Figure 1–1 Imaging Solution integrates Business Application, Oracle I/PM, and BPEL systems



This section covers the following topics:

- "Business User View for Imaging Solutions" on page 1-3
- "System Architecture" on page 1-5

1.3.1 Business User View for Imaging Solutions

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

- Launch Oracle I/PM from PeopleSoft, and select and perform workflow tasks. For example, users performing Invoice Processing tasks select a custom button, link, or menu integrated into their business application called Invoice Processing, 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 (Invoice Processing), an AXF command (Open_Tasklist), and a user name.
  - AXF returns an Open_URL command with the URL to launch, such as an AXF Task List.
  - The business application opens the Task List URL in a new browser window, enabling the user to start processing invoice images.
- View attached images and metadata values. Use Oracle I/PM's tools for viewing, annotating, and redacting images, as permissions allow.
- Key entries in PeopleSoft while viewing images and related values in the Oracle I/PM viewer.
- Perform actions related to the workflow task, such as routing, canceling, updating, and completing tasks.
- Scan or upload supporting documents for a selected PeopleSoft record.
- View supporting images for a PeopleSoft record without leaving the PeopleSoft application.

1.3.1.1 Sample Scenario 1: Processing Invoices

An Imaging Solution configured for invoice processing might work as follows:

- 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.
- From PeopleSoft, the user launches the Imaging Solution, by clicking a link, button, or menu called **Process Invoices**.
- 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.
- In the Task Viewer, users view the task's invoice image, key entries in PeopleSoft 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.
- Users complete the task and begin another, if desired.

Most often, changes users make in PeopleSoft are synchronized with Oracle I/PM, and vice versa.

Note: For details about imaging solution user tasks, see the *Oracle Fusion Middleware User's Guide for Oracle Enterprise Content Management Solutions for Oracle PeopleSoft*.

1.3.1.2 Sample Scenario 2: Capturing Supporting Employee Documents

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

- From PeopleSoft, a user retrieves a record such as an employee record.
- The user launches the document imaging solution, by clicking a link or button called **Scan Employee Document**.
- Oracle Distributed Document Capture launches and automatically initiates a scan (if a scanner is attached to the desktop) or allows the user to upload electronic images from desktop.
- The user enters index values (metadata) in Oracle Distributed Document Capture to store with the images.
- The user clicks Send, which transmits the captured document images and their metadata from Oracle Distributed Document Capture to Oracle I/PM.

1.3.1.3 Sample Scenario 3: Viewing Supporting Employee Documents

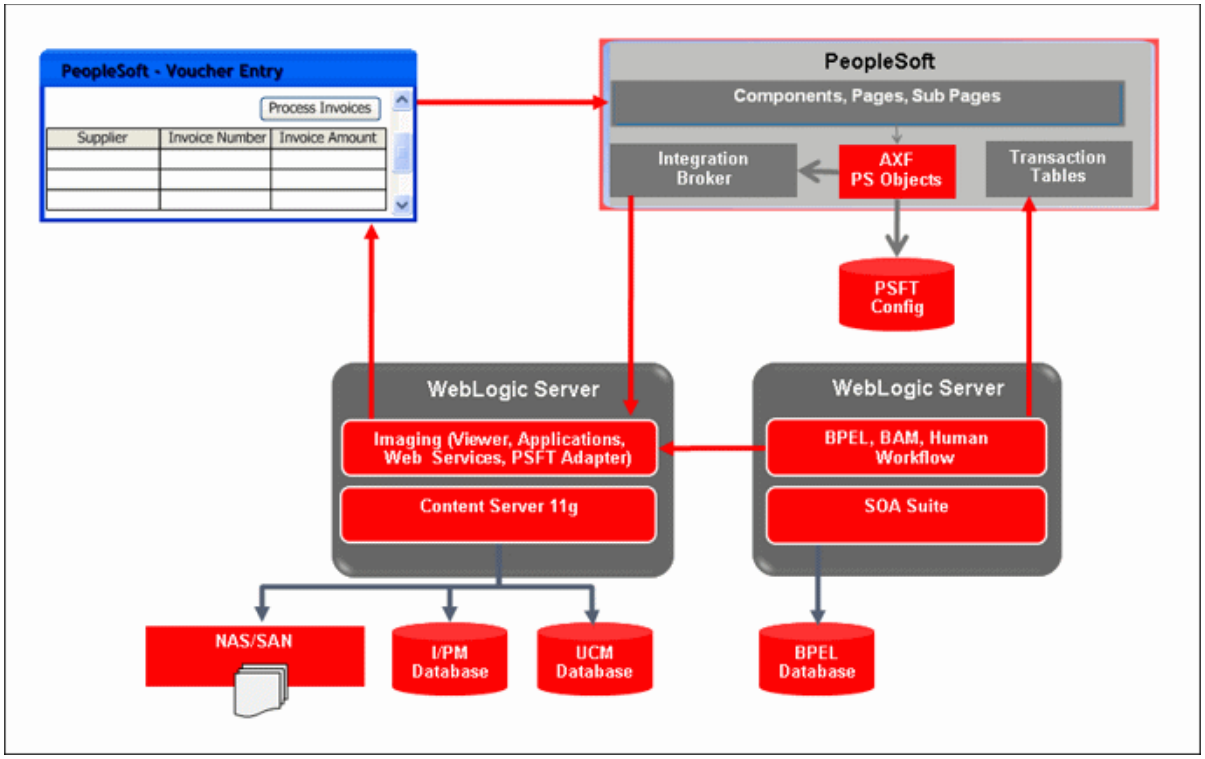
An Imaging Solution configured for viewing supporting documents might work as follows:

- From PeopleSoft, a user retrieves a record such as an employee record.
- A user launches the document imaging solution, by clicking a link or button called **View Employee Documents**.
- From the list of documents associated with the employee record and their metadata values, the user selects a document.
- The document is displayed in the Oracle I/PM viewer, where the user can view its images, and with appropriate permissions, apply annotations or redactions.

1.3.2 System Architecture

Figure 1–2 illustrates an imaging solution configuration for the PeopleSoft adapter.

Figure 1–2 System Architecture for Imaging Solution for PeopleSoft Adapter



1.3.3 About AXF Commands

The reusable AXF commands allow you to implement the functionality described below. For information about these commands, including their parameters and example implementations, see "AXF Commands" on page 4-24.

AXF Command	Description
Open Task	Initializes and displays the AXF Task Viewer web page and claims a human workflow task. See "Open Task Command" on page 4-25.
Autotask	Initializes autotask mode, in which a new human workflow task is automatically claimed in the AXF Task Viewer without displaying the Task List. See "Autotask Command" on page 4-25.
Release Task	Initializes the AXF Task List web tool for display (regardless of Autotask mode) and releases a human workflow task. See "Release Task Command" on page 4-26.
Complete Task	Completes a human workflow task and updates BPEL payload attribute values. If using the Autotask Command, claims the next task and displays it in the Task Viewer. See "Complete Task Command" on page 4-27.
Redirect	Redirects the current AXF web page to any URL specified in the configuration. See "Redirect Command" on page 4-28.
Terminate Conversation	Used by an external client to terminate a conversation with AXF. (This command does not include parameters.)

AXF Command	Description
Update Task	Updates BPEL payload field values on a specified human task or values in the XML payload using XPATH. See "Update Task Command" on page 4-28.
Update Task From Procedure	Calls a stored procedure using a specified data source and updates values in the BPEL payload using XPATH. See "Update Task From Procedure Command" on page 4-30.
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 "Validate Task Command" on page 4-32.

Note: You can also deploy custom commands to execute using AXF. See "Custom Commands" on page 4-33.

1.3.4 About AXF Web User Tools

The imaging solution provides the following user interface components. These are web interface components displayed to users and configured through the AXF tables.

- "About the Task List" on page 1-6
- "About the Task Viewer" on page 1-7
- "About the Enumeration Picker" on page 1-8
- "About the Identity Picker" on page 1-8
- "About Comments" on page 1-9

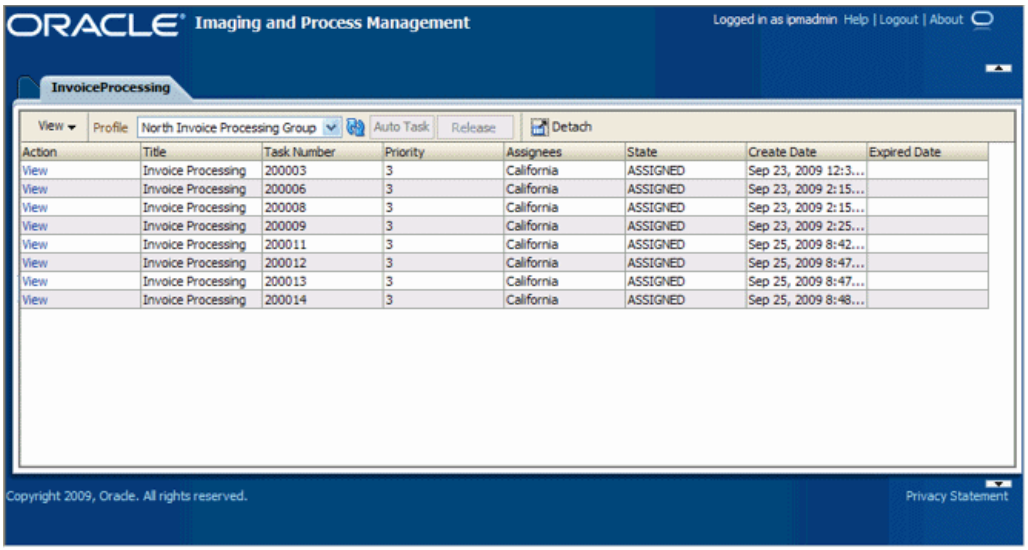
1.3.4.1 About the Task List

The Task List web page 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 "Task List Web Tool" on page 4-13.

Figure 1–3 Task List Web Tool



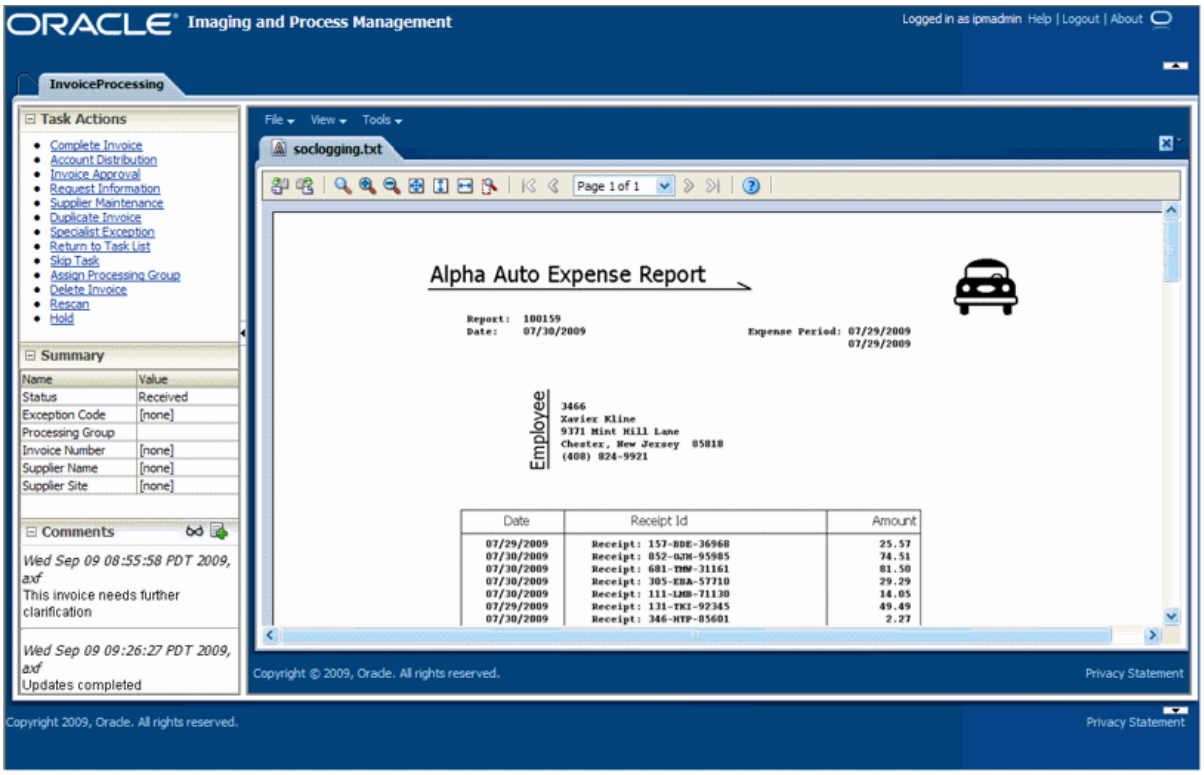
The Task List web tool can also display a list of AXF actions just like the Task Viewer, using AXF action commands. These action commands are menu components configured in the AXF_ACTIONS Table for display on a web page.

1.3.4.2 About the Task Viewer

The Task Viewer web page displays images and metadata values through interaction with the AXF infrastructure, BPEL, Oracle I/PM, and the business application. It also typically displays a side menu containing AXF action commands configured in the AXF_ACTIONS Table. It may also include a Comments side pane; see "About Comments" on page 1-9.

For configuration information, see "Task Viewer Web Tool" on page 4-15.

Figure 1–4 Task Viewer Web Tool

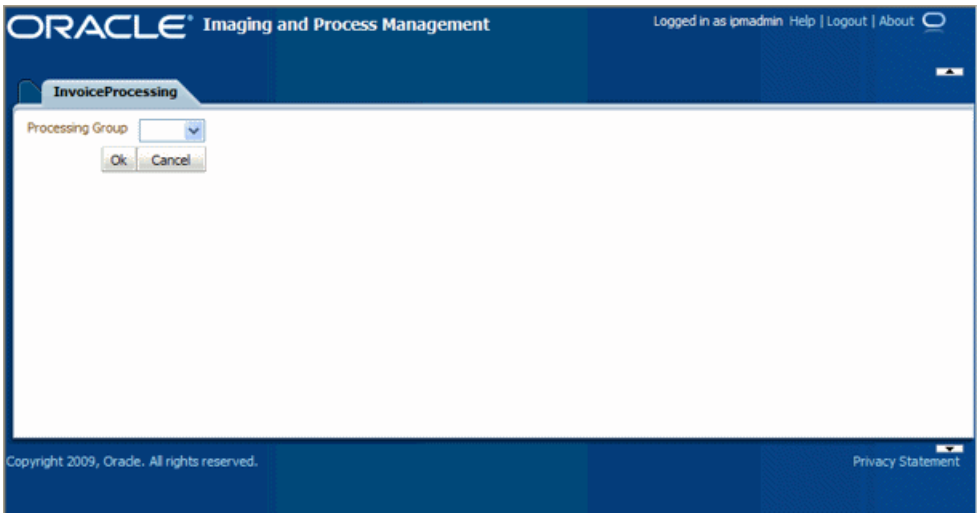


1.3.4.3 About the Enumeration Picker

The Enumeration Picker web page allows users to select from a list of enumerated values configured in the AXF database tables.

For configuration information, see "Enumeration Picker Web Tool" on page 4-19.

Figure 1–5 Enumeration Picker Web Tool



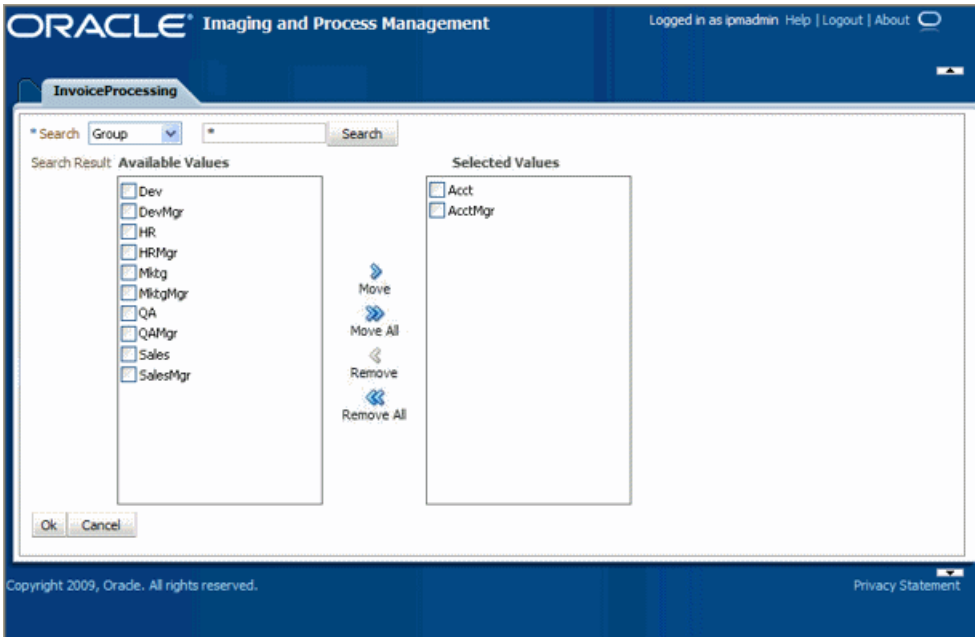
1.3.4.4 About the Identity Picker

The Identity Picker web page allows users to select one or more users or groups from an identity store configured for BPEL. After choosing an identity, a related action is

typically taken. Most likely, a task is assigned or delegated 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 "Identity Picker Web Tool" on page 4-23.

Figure 1–6 Identity Picker Web Tool

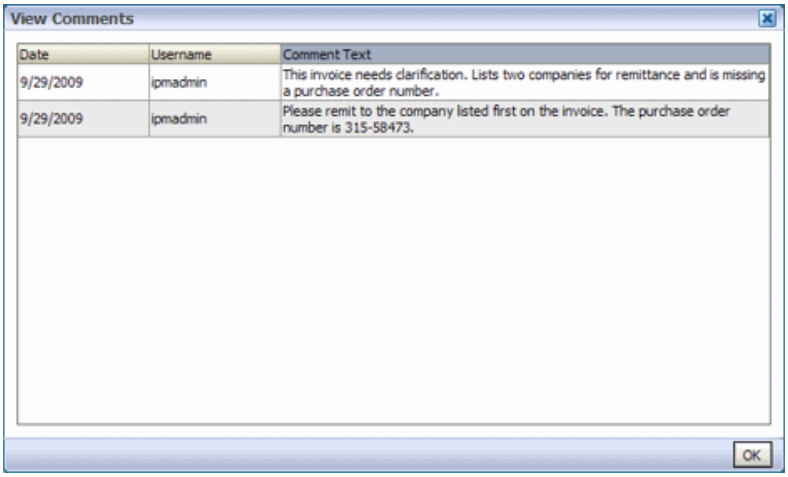


1.3.4.5 About Comments

The Comments web page allows users to enter comments related to the human task during the transaction's processing. Comments can be displayed in a side pane on the Task Viewer. Comments persist for the entire process, allowing users to view and add comments. Comments are saved using the native comments capabilities of BPEL's workflow task.

For configuration information, see "Comments" on page 4-19.

Figure 1–7 Comments Web Tool



1.4 About Adapter Authentication and Security

The PeopleSoft adapter provides the following authentication points:

- *Browser level authentication*, where end-users invoke an AXF web tool or the Oracle I/PM viewer from a PeopleSoft page configured for access. Authentication at this level is handled by Oracle WebLogic Server.
- *Service call authentication*, where web service calls are made to the Application Extension Framework. The PeopleSoft adapter supports user authentication against the AXF solution mediator web services using a username token security installed on the application server on which AXF resides. This is handled through Integration Broker configuration, as described in "Configuring Integration Broker to Communicate with AXF" on page 2-3.

1.5 Adapter System Requirements

The PeopleSoft Adapter consists of optional solutions installed over a base configuration of AXF and Oracle I/PM files standard to adapters that use AXF functionality. Requirements for the PeopleSoft Adapter for Enterprise Content Management are listed below.

Note: Supported configuration and certification information is available at:

`http://www.oracle.com/technology/software/products/ias/files/fusion_certification.html`

PeopleSoft Enterprise

A fully functioning Oracle PeopleSoft Enterprise system.

- PeopleTools 8.49.x
- To avoid duplicate logins, Oracle Single Sign-On or Oracle Access Manager is required.

Oracle SOA Suite 11*g*

SOA 11*g*R1 (with patchset 1) is required. In addition, you need a BPEL server instance. BPEL is part of SOA 11*g*R1.

2

Configuring PeopleSoft Components

This chapter covers the following topics:

2.1 About Configuring AXF Components on PeopleSoft Pages

You configure AXF components on PeopleSoft pages using the PeopleSoft Application Designer and the AXF tables in PeopleSoft. For procedures, see "Enabling AXF Components on PeopleSoft Pages" on page 2-6. For a description of tables, see "About the AXF Tables in PeopleSoft" on page 4-34.

There are two categories of AXF components:

2.1.1 User Interface Items

User interface items refer to AXF buttons, links, and menus you place on a PeopleSoft page. Five of each are provided (for example, AXF_BUTTON_1 through _5). For more information and configuration examples, see "PS_AXF_COMMANDS Table" on page 4-35.

Important Points About User Interface Items

- You must specify a PeopleSoft role (in the PS_AXF_COMMANDS Table) to grant users permission to use a specified user interface item. If a user lacks the permissions, the item is not displayed.
- A user interface item becomes active when it is positioned on a page. If AXF_BUTTON_1 is placed on multiple pages in the same PeopleSoft component, it shares the same configuration across the pages. To configure different actions for each page, use a different user interface item (for example, AXF_BUTTON_2). Note that you can reuse user interface items on different PeopleSoft components without conflicts.

2.1.2 System Events

System event components refer to pre- and post-save items you place on a PeopleSoft page that trigger a pre- or post-save system event.

- The AXF_POSTSAVE_SBP item captures SAVE_POST_CHANGE system events, which execute a configured command *after* a PeopleSoft component is saved.

  For example, you might configure a SaveInvoice command to be invoked during the SAVE_POST_CHANGE event, so that whenever an action inserts a new 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* a PeopleSoft component is saved.

Important Points About System Events

- You must specify a PeopleSoft role (in the PS_AXF_COMMANDS Table) to allow a specified system event to be triggered for a user.
- System events, like user interface items, are configured at the PeopleSoft component level. However, because they are not visible to the user, they are triggered as 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 a PeopleSoft component that is being saved, the configured command executes.

2.2 Configuring PeopleSoft Solution Components and Communications

This section describes how to configure PeopleSoft components and communications for the solution. It covers the following topics:

2.2.1 Importing the PeopleSoft Project

Follow these steps to import the PeopleSoft project:

1. Start the PeopleSoft Application Designer in two-tier mode.
2. From the menu, select **Tools**, then **Copy Project**, then **From File**. The Copy From File screen is displayed.
3. Locate the following directory:

   MW_HOME/ECM_HOME/axf/adapters/psft/

4. Select **psft**, the parent directory of the **AXF_PS_INTEGRATION** directory. The AXF_PS_INTEGRATION directory contains the project XML files, but you must select its parent directory to access them from the PeopleSoft Application Designer. The project is displayed in the lower pane of the screen.
5. Click **Select**, select all definition types, and click **Copy**. The project files begin copying.

6. After all files have copied, select **Build**, then **Project** from the menu. The Build screen is displayed.
7. Select the **Create Tables** and **Execute SQL Now** options, and click **Build**. You can monitor the build status as the files are imported into PeopleSoft.
8. Once done, check the PSBUILD.LOG file to verify that the project imported successfully.
9. If the log file shows *tablespace 'AXF' does not exist* errors, manually create a tablespace called `AXF` in the same database you logged into when starting the PeopleSoft Application Designer, then return to step 6 to rebuild the project.

2.2.2 Configuring Integration Broker to Communicate with AXF

Importing the AXF_PS_INTEGRATION project to PeopleSoft also imports the Integration Broker connection information. These components contain information needed to connect from PeopleSoft to the AXF Server. The following section describes how to access this information in the 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:

- "Configuring Communication With a 10g AXF Server" on page 2-3
- "Configuring Communication With an 11g AXF Server" on page 2-4

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

- "Setting Up the Service Operation Routings" on page 2-4
- "Validating Domain Status" on page 2-5
- "Securing Web Services" on page 2-5

2.2.2.1 Configuring Communication With a 10g AXF Server

1. In 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**, 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. Set the Host property name to the AXF Server name or IP address.
7. Set the URL property name 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* is displayed in the message text area of the Ping NodeResults page. Proceed to "Setting Up the Service Operation Routings" on page 2-4 and "Validating Domain Status" on page 2-5.

- If *Success* is not displayed, return to the Node Configuration page to reenter values until you can ping the node successfully.

2.2.2.2 Configuring Communication With an 11g AXF Server

1. In 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**, enter `AXF` in the Node Name field, and click **Search**.
4. From the search results, click the **AXF_SOLUTION_MEDIATOR_11G** link.
5. On the Node Definitions tab, enter a valid username and password in the **External User ID** and **External Password** fields. 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. Set the Host property name to the AXF Server name or IP address.
8. Set the URL property name to the following URL:

   `http://`*`AXF_Server name or IP address:port_number`*`/axf-ws/AxfSolutionMediatorService`

   For example:

   ```
   http://myserver.us.oracle.com:16000/axf-ws/AxfSolutionMediatorService
   ```

9. Click **Save**.
10. Click **Ping Node** to verify that the node is configured properly.
    - If configured properly, the word *Success* is displayed in the message text area of the Ping NodeResults page. Proceed to "Setting Up the Service Operation Routings" on page 2-4 and "Validating Domain Status" on page 2-5.
    - If *Success* is not displayed, return to the Node Configuration page to reenter values until you can ping the node successfully.

2.2.2.3 Setting Up the Service Operation Routings

1. From the Integration Setup Menu, 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 a **10g AXF Server**, verify that the AXF_SM_ROUTING_10G routing definition's status is Active. If it is not, choose the Selected field, click the Activate Selected Routings button, and ensure that all other routing definition are set to Inactive.
   - 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

the Selected field, click the Activate Selected Routings button, and ensure that all other routing definition are set to Inactive.

6. Click the Save button.

2.2.2.4 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 is displayed.
2. Click **Domain Status** under the **Administration** section. The Domain Status page is displayed.
3. Verify that the domain status is listed as **Active** next to the PeopleSoft Server. If not, select **Active** from the Domain Status menu and click **Update**.

2.2.2.5 Securing Web Services

Follow the steps in this section to apply the wss_username_token_service_policy to all services.

1. Log in to the WebLogic Server Administration Console.
2. From the Domain Structure options, select **Deployments**. The Summary of Deployments page is displayed.
3. From the Deployments table, select **imaging**.
4. Click the **Update** button. The Update Application Assistant page is displayed.
5. Select the lower **Redeploy this application using** option to redeploy the ear file with a deployment plan.
6. Click the **Change Path** button for the Deployment plan path option, browse to the following location, and select the Plan.xml file:

   MW_HOME/user_projects/applications/*domain_name*/server/ipm
7. Continue the wizard and complete the deployment.

2.2.3 Adding the AXF_ROLE to PeopleSoft

To ensure that appropriate PeopleSoft users have access to AXF functionality, follow these steps to add the AXF_ROLE and permissions to PeopleSoft.

1. In PeopleSoft Server, open **PeopleTools**, then **Security**, then **Permissions & Roles**. The Permissions & Roles menu expands to show available options.
2. Click **Roles**. The Roles page is displayed.
3. Click the **Add New Value** tab.
4. Enter `AXF_ROLE` in the Role Name field and click **Add**. The Role Name field is cleared and the AXF_ROLE is added.
5. Add a description in the Description field.
6. Click the **Permission Lists** tab, enter `AXF_PERMS` in the Permission List field, and click **Save**.

2.2.4 Assigning Users the AXF_ROLE

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

> **Note:** For AXF calls to be processed correctly from PeopleSoft, all users needing access to AXF functionality must be assigned the AXF_ROLE in PeopleSoft.

1. In 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 is displayed.
3. In the search field, specify criteria to identify the user to find, and click **Search**. A listing of users is displayed.
4. Click the user to which to assign the role. A detailed page of user information is displayed.
5. Click the **Roles** tab. A listing of roles assigned to the user is displayed.
6. Click the + (plus) sign to add a blank field, and enter `AXF_ROLE` in the blank field. To search for the role, click the magnifying glass icon.
7. Click **Save**. The user can now access AXF functionality on PeopleSoft pages.

2.2.5 Setting User Locales

To prevent issues with different locales when invoking AXF, PeopleSoft users should set the same values for their user locale preference and their browser locale.

2.3 Enabling AXF Components on PeopleSoft Pages

Follow these steps to enable an AXF component on a PeopleSoft page. For more information, see "About Configuring AXF Components on PeopleSoft Pages" on page 2-1.

1. Open the AXF_PS_Integration Project in PeopleSoft Application Designer.
2. Open a PeopleSoft page to enable. For example, open VCHR_HDR_QV2.
3. In the Project Workspace, select one of the AXF pages from the Pages node (for example, AXF_BTN1_SBP) and drag it onto the PeopleSoft page or subpage.
4. Save the page.
5. Configure the PeopleSoft AXF tables to associate a command with the newly added user interface item or system event as described in the following sections:
   - "PS_AXF_CONFIG Table" on page 4-35
   - "PS_AXF_COMMANDS Table" on page 4-35
   - "PS_AXF_COMMAND_PARAMS Table" on page 4-37

2.4 Configuring and Viewing Log Files

You may want to examine the following AXF-related logs:

- AXF logs

- PeopleSoft logs

2.4.1 Configuring AXF Logging

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'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'/>
   <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 2–1. 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 console.

Table 2–1 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. Restart Administration Server if it is running. The logger is displayed in Enterprise Manager. You can change the logging level at run time.

2.4.2 Configuring PeopleSoft Logging

The AXF PeopleSoft adapter uses standard apache log4j logging. Logging can be activated and managed 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=%d{DATE} [%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.

2.5 Uninstalling AXF from 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.

3

Configuring Imaging Solution Components

This chapter describes how to configure the following imaging solution components:

3.1 Configuring the BPEL Connection

Configuring the BPEL connection for use by an AXF solution involves the following tasks:

3.1.1 Creating a CSF Credential Alias

The Credential Store Framework (CSF) enables you to create a username/password alias for use in an Oracle I/PM connection configuration. With a CSF alias, you supply a key instead of a username and password, and use this key in creating an Oracle I/PM 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*.

3.1.2 Creating a Connection in Oracle I/PM 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 Oracle I/PM 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 the system name, port, and credential alias (previously created as described in "Creating a CSF Credential Alias" on page 3-1).

   For example:

- System: *system*
- Port: *port number*
- Credential Alias: axfconnection

5. Click **Next**, then **Submit**.

3.1.3 Referencing the Connection in the AXF_SOLUTION_ATTRIBUTES Table

Follow these steps to identify the Oracle I/PM imaging connection to the AXF solution. Run the commands from SQL Developer (or other suitable tool that can connect to the imaging database schema).

1. Run the two configuration rows specified below, where:
   - *CONNECTION_NAME* identifies the connection name configured in Oracle I/PM Imaging Connections, as described in "Creating a Connection in Oracle I/PM Imaging Connections" on page 3-1.
   - *SOLUTION_NAMESPACE* identifies the solution. *InvoiceProcessing* is used in the example below. Modify this value if needed.

```
Insert into AXF_SOLUTION_ATTRIBUTES (SOLUTION_NAMESPACE,PARAMETER_KEY,PARAMETER_VALUE) values \
('InvoiceProcessing','BPEL_CONNECTION','CONNECTION_NAME');
Insert into AXF_SOLUTION_ATTRIBUTES (SOLUTION_NAMESPACE,PARAMETER_KEY,PARAMETER_VALUE) values \
('InvoiceProcessing','CONNECTION_PROVIDER','oracle.imaging.axf.servicemodules.bpel.workflow.
AxfWorkflowServiceModule');
```

3.1.4 Configuring the URI to Display Images in the Task Viewer

Follow the steps below to configure the URI for displaying images in the Task Viewer. You configure the DocURL payload element from the BPEL Injector as the URI and hide the banner.

1. Under Manage Applications, display the BPEL Payload Properties page for the Oracle I/PM application.

   For more information, see the *Oracle Fusion Middleware Administrator's Guide for Oracle Imaging and Process Management*.

2. For the URI payload element, choose **Document URL** in the Mapped Value field.

4

Imaging Solution Tables

This chapter describes the AXF and PeopleSoft configuration tables used for the Imaging Solution, including commands, web user interface tools, and example implementations. This chapter covers the following topics:

4.1 Overview of AXF Configuration Tables

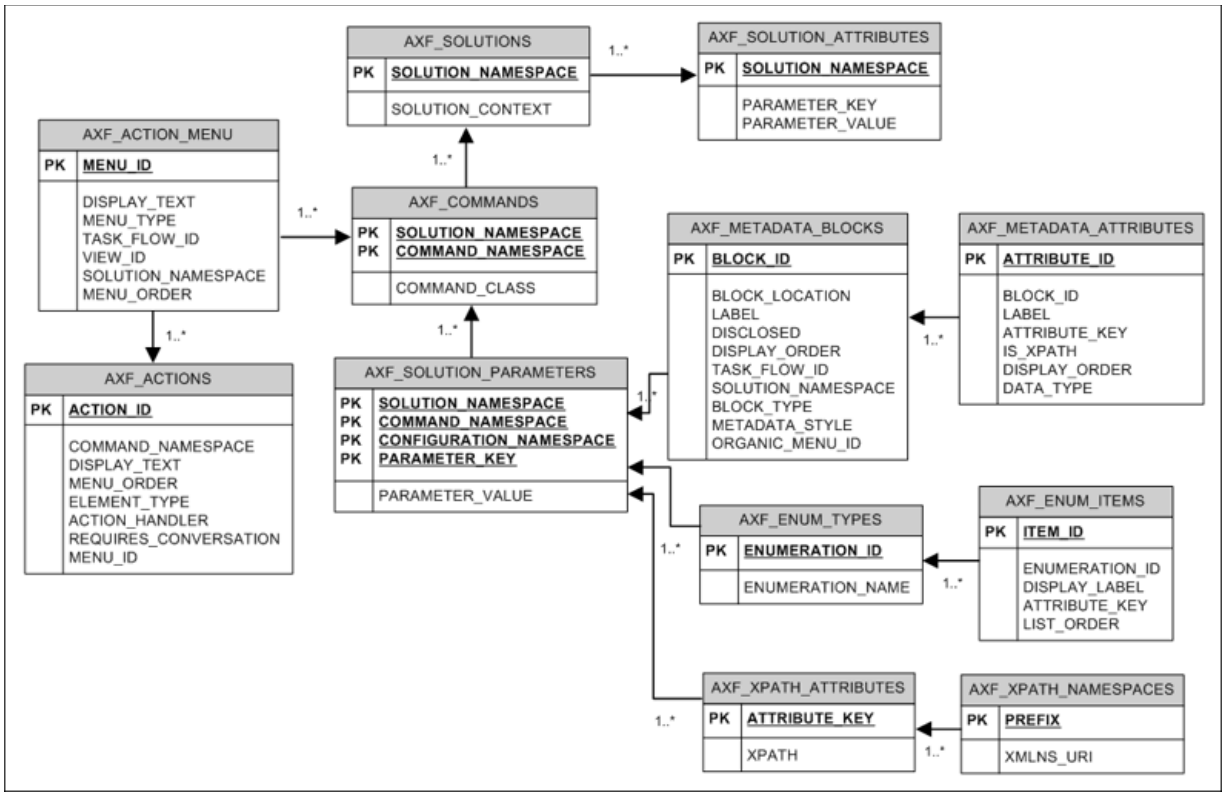
Note: Running an AXF and an Oracle BPM Worklist session at the same time can result in the session first opened ending. For example, launching an AXF session with an Oracle BPM Worklist session open ends the BPM Worklist session.

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 Fusion Middleware Installation Guide for Oracle Enterprise Content Management Suite*.

The diagram that follows displays the AXF configuration tables and their relationships.

Figure 4–1 AXF Configuration Tables



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 web tools.
AXF_ACTION_MENU Table AXF_ACTIONS Table	Define task action pane itself, links in the pane, and their parameters.
AXF_METADATA_BLOCKS Table, AXF_METADATA_ATTRIBUTES Table	Define optional panes, such as Summary and Comments 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.

4.2 AXF Tables

This section describes the following AXF tables. See "AXF Web User Tools" on page 4-13 for web tool-related tables.

4.2.1 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.

4.2.1.1 Column Description

Table 4–1 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.

4.2.1.2 Example Implementation

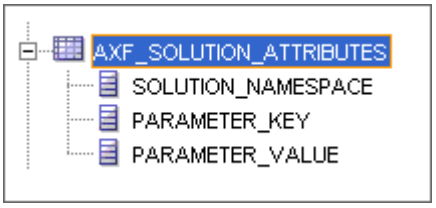
This example table shows the AXF solutions defined. Each of the solutions uses AxfCommandMediator as its solution implementation.

Table 4–2 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

4.2.2 AXF_SOLUTION_ATTRIBUTES Table

This table defines general attributes for use by infrastructure, services, or solutions. For example, use this table to define error message addresses, connections, and conversation timeout settings.



4.2.2.1 Column Description

Table 4–3 Column Description for AXF_SOLUTION_ATTRIBUTES Table

Column	Description
SOLUTION_NAMESPACE	Specifies the functional area that utilizes the parameter. Must correspond to a valid BPEL value. ■ *AXF* namespace is used by AXF. ■ *AccountsPayable* is used by the AccountsPayable template. ■ *BPEL.default* specifies the name of the BPEL connection, where BPEL is a constant and default is the name of connection.
PARAMETER_KEY	Name of the parameter. Used when retrieving the parameter value from the database. Parameters include: ■ BPEL_CONNECTION: Identifies the BPEL connection to be used. ■ 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. ■ IDENTITY_SERVICE_ENDPOINT: Specifies the URL point to BPEL identity web services to query the defined users in BPEL. ■ 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 are acquiring tasks in Autotask mode. See "Configuring Autotask Locking" on page 4-26.
PARAMETER_VALUE	Value of the parameter.

4.2.2.2 Example Implementation

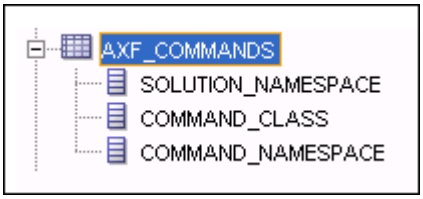
This example table sets solution attributes for the Invoice Processing solution.

Table 4–4 Example AXF_SOLUTION_ATTRIBUTES Table

SOLUTION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
InvoiceProcessing	BPEL_CONNECTION	axfconnection
InvoiceProcessing	CONNECTION_PROVIDER	oracle.imaging.axf.servicemodules.bpel.workflow.AxfWorkflowServiceModule
InvoiceProcessing	USE_AUTOTASK_LOCKING	TRUE

4.2.3 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.



4.2.3.1 Column Description

Table 4–5 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 under "Imaging Solution Tables" on page 4-1.

4.2.3.2 Example Implementation

This example shows commands defined for the Invoice Processing solution.

Fields not shown: SOLUTION_NAMESPACE=InvoiceProcessing

Table 4–6 Example AXF_COMMANDS Table

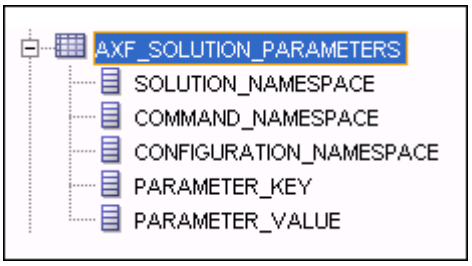
COMMAND_CLASS	COMMAND_NAMESPACE
oracle.imaging.axf.commands.bpel.AutotaskCommand	AutoOpenTask
oracle.imaging.axf.commands.bpel.ReleaseTaskCommand	ReleaseTask
oracle.imaging.axf.commands.bpel.ReleaseTaskCommand	SkipTask
oracle.imaging.axf.commands.bpel.CompleteTaskCommand	AccountDistributionComplete
oracle.imaging.axf.commands.bpel.CompleteTaskCommand	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

Table 4–6 (Cont.) Example AXF_COMMANDS Table

COMMAND_CLASS	COMMAND_NAMESPACE
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
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

4.2.4 AXF_SOLUTION_PARAMETERS Table

This table defines command parameters for the solution, AXF commands, and AXF web tools.



4.2.4.1 Column Description

Table 4–7 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 4–7 (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 be instantiated. The namespace also differentiates commands within the same solution namespace.
PARAMETER_KEY	Specifies the parameter key to be used in the AXF command. For parameter details, see the specific command or web tool: Web Tools: ■ "Task List Web Tool" on page 4-13 ■ "Task Viewer Web Tool" on page 4-15 ■ "Enumeration Picker Web Tool" on page 4-19 ■ "Identity Picker Web Tool" on page 4-23 ■ "Comments" on page 4-19 AXF Commands: ■ "Open Task Command" on page 4-25 ■ "Autotask Command" on page 4-25 ■ "Release Task Command" on page 4-26 ■ "Complete Task Command" on page 4-27 ■ "Redirect Command" on page 4-28 ■ "Update Task Command" on page 4-28 ■ "Update Task From Procedure Command" on page 4-30 ■ "Validate Task Command" on page 4-32
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.

4.2.4.2 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 4–8 Example AXF_SOLUTION_PARAMETERS Table for StartInvoiceProcessing Command

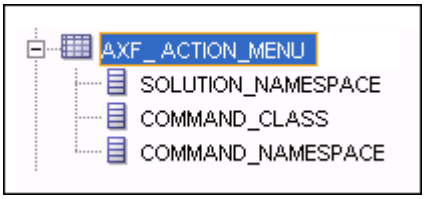
COMMAND_ NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
StartInvoiceProcessing	oracle.imaging.axf.commands.bpel.RedirectCommand	REDIRECT_URL	taskflow://WEB-INF/taskflows/axf-tasklist-tfd.xml#axf-tasklist-tfd
StartInvoiceProcessing	oracle.imaging.axf.web.backing.TaskList	CMD_OPEN_TASK_BUTTON	OpenTask
StartInvoiceProcessing	oracle.imaging.axf.web.backing.TaskList	CMD_AUTO_TASK_BUTTON	AutoOpenTask
StartInvoiceProcessing	oracle.imaging.axf.web.backing.TaskList	DEFAULT_VIEW	(null)
StartInvoiceProcessing	oracle.imaging.axf.web.backing.TaskList	NO_OF_LINES	20

Table 4–8 (Cont.) Example AXF_SOLUTION_PARAMETERS Table for StartInvoiceProcessing Command

COMMAND_ NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
StartInvoiceProcessing	oracle.imaging.axf.web.backing.TaskList	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

4.2.5 AXF_ACTION_MENU Table

Use this table to insert and customize an action menu on the Task Viewer or Task List screen. 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 4–2. Use the AXF_ ACTIONS Table to define a specified menu's actions.



4.2.5.1 Column Description

Table 4–9 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 screen the menu is displayed and its type. (LEFT_SIDEBAR displays a side pane leftmost on the screen.)
TASK_FLOW_ID	Specifies a task flow String that corresponds to a known task flow ID which loads a page or pages on the task flow.
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 if multiple menus are set for display.

4.2.5.2 Example Implementation

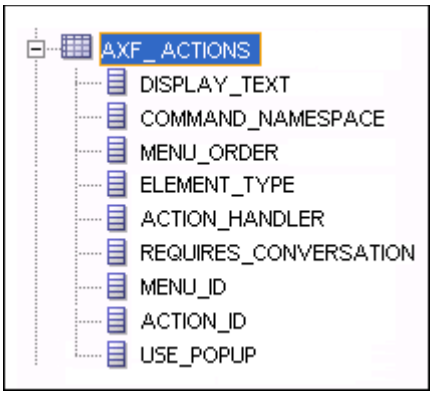
This example table implements a pane entitled *Task Actions* in the left side of the Task Viewer for the Invoice Processing solution.

Table 4–10 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

4.2.6 AXF_ACTIONS Table

This table defines the task actions used in an AXF solution. You can display action menus in a Task Viewer page (see "Task Viewer Web Tool" on page 4-15) or a Task List (see "Task List Web Tool" on page 4-13). This table links to the AXF_COMMANDS Table.



4.2.6.1 Column Description

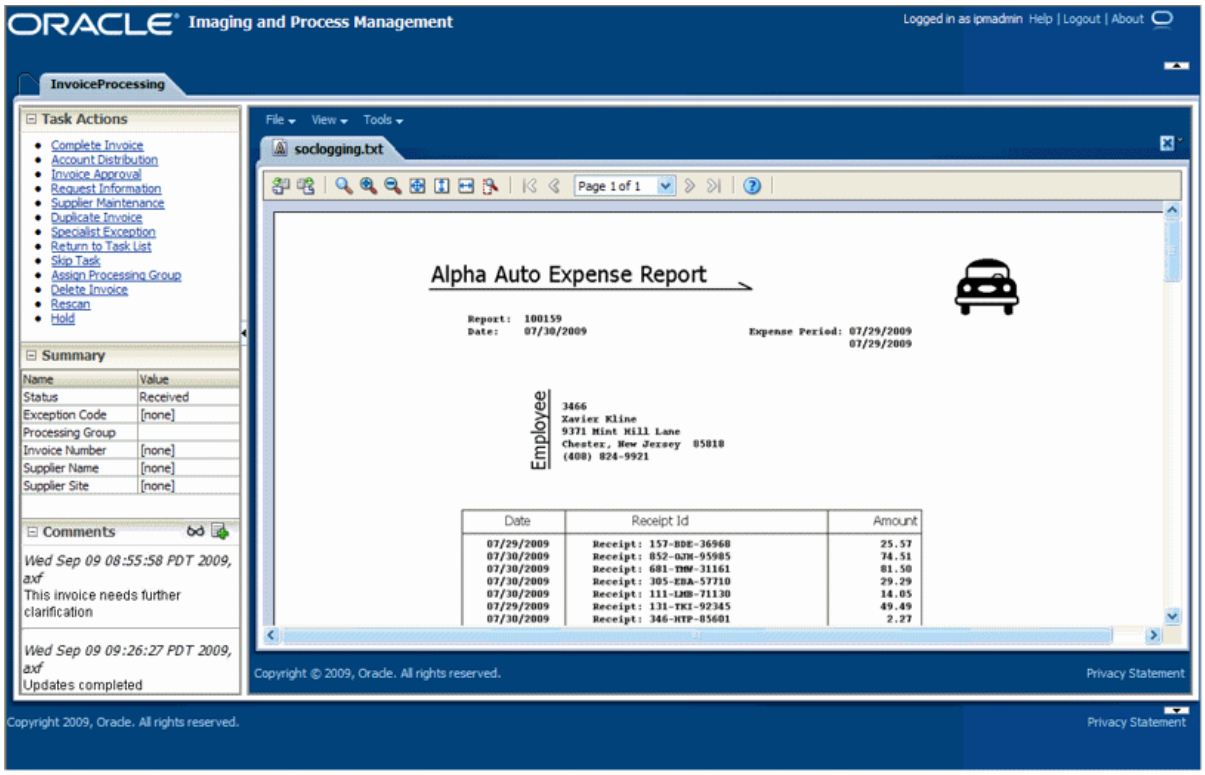
Table 4–11 Column Description for AXF_ACTIONS Table

Column	Description
DISPLAY_TEXT	Specifies the name of the action (link, for example) in the pane.
COMMAND_NAMESPACE	Specifies the command that is 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 ■ BUTTON: Displays a button
ACTION_HANDLER	Determines how the command is handled, where COMMAND sends the command specified in the COMMAND_NAMESPACE column to the Solution Mediator. **Note:** If left (null), this value defaults to COMMAND.
REQUIRES_CONVERSATION	Specifies whether the action requires a conversation ID.
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.

4.2.6.2 Example Implementation

The tables that follow provide an example AXF_ACTIONS Table.

Figure 4–2 Task Viewer Page with Task Actions, Summary, and Comments Enabled



Fields not shown in Table 4–12:

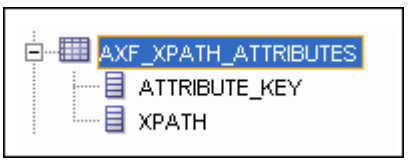
- ACTION_HANDLER=COMMAND
- REQUIRES_CONVERSATION=TRUE

Table 4–12 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

4.2.7 AXF_XPATH_ATTRIBUTES Table

This table defines the XPATH attributes used in the AXF framework. This XPATH is mainly defined for payload elements.



4.2.7.1 Column Description

Table 4–13 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.
XPATH	XPATH expression used to locate the value in the payload.

4.2.7.2 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 4–14 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 4–15 Example AXF_XPATH ATTRIBUTES Table

ATTRIBUTE_KEY	XPATH
InvoiceProcessing_ProcessingGroup	//task:processingGroup

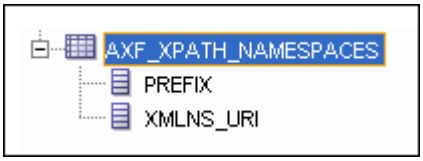
In the AXF_NAMESPACES table that follows, the XMLNS_URI column displays where within the XML file to locate the processingGroup task information.

Table 4–16 Example AXF_XPATH_NAMESPACES Table

Prefix	XMLNS_URI
task	http://xmlns.oracle.com/bpel/workflow/task

4.2.8 AXF_XPATH_NAMESPACES Table

The following table defines the namespaces used for the XPATH attributes. It links to the AXF_XPATH_ATTRIBUTES Table.



4.2.8.1 Column Description

Table 4–17 Column Description for AXF_XPATH_NAMESPACES Table

Column	Description
PREFIX	The namespace prefix used in the XPATH.

Table 4–17 (Cont.) Column Description for AXF_XPATH_NAMESPACES Table

Column	Description
XMLNS_URI	Provides a unique identifier.

4.2.8.2 Example Implementation

Table 4–18 Example AXF_XPATH_NAMESPACES Table

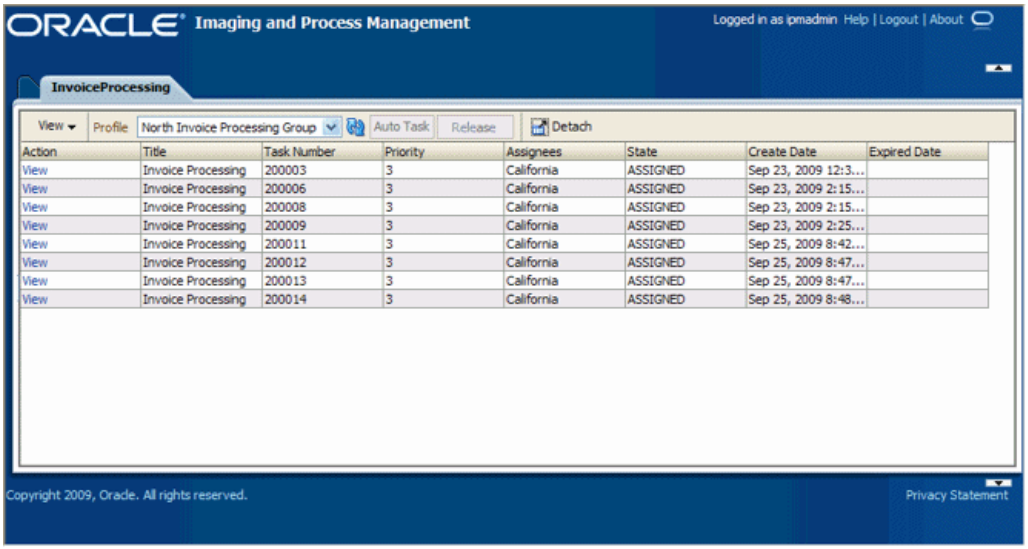
Prefix	XMLNS_URI
task	http://xmlns.oracle.com/bpel/workflow/task
documentContent	http://xmlns.oracle.com/imaging/axf/documentContentTypes
solution	http://xmlns.oracle.com/imaging/axf/solutionTypes
invoiceProcessing	http://xmlns.oracle.com/imaging/axf/InvoiceProcessing
xml	http://www.w3.org/XML/1998/namespace

4.3 AXF Web User Tools

This section covers the following topics:

4.3.1 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** field 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.)

- 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.
- 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 skip (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.
- You can configure the Task List to include a side pane action list with links.

4.3.1.1 Task List Parameters

Table 4–19 Task List Parameters in AXF_SOLUTION_PARAMETERS Table

Parameter	Description
CMD_OPEN_TASK_BUTTON	Specify a COMMAND_NAMESPACE to be executed when a user clicks the View Task link on the Task List web page.
CMD_AUTO_TASK_BUTTON	Specify a COMMAND_NAMESPACE to be executed when a user clicks the Auto Task button on the Task List web page.
CONNECTION_NAME	Specify the BPEL connection, as defined in the AXF_SOLUTION_ATTRIBUTES Table.
NO_OF_LINES	Specify the maximum number of tasks to be displayed before multiple pages are used.
BPEL_CONNECTION	Specify the BPEL connection that obtains the task list from BPEL, defined 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.
SHOW_INBOX	Specify whether the Inbox is listed in the view list. If TRUE, the Inbox is listed; if FALSE, the Inbox is not listed.
TASK_DEF	Specify the BPEL human workflow tasks to which the user has access. (For example, a value of *Rescan* means that Rescan tasks are displayed in the Inbox.) This parameter applies only when the SHOW_INBOX parameter is set to TRUE and the Inbox profile is selected.
REDIRECT_URL	Specify 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.

4.3.1.2 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

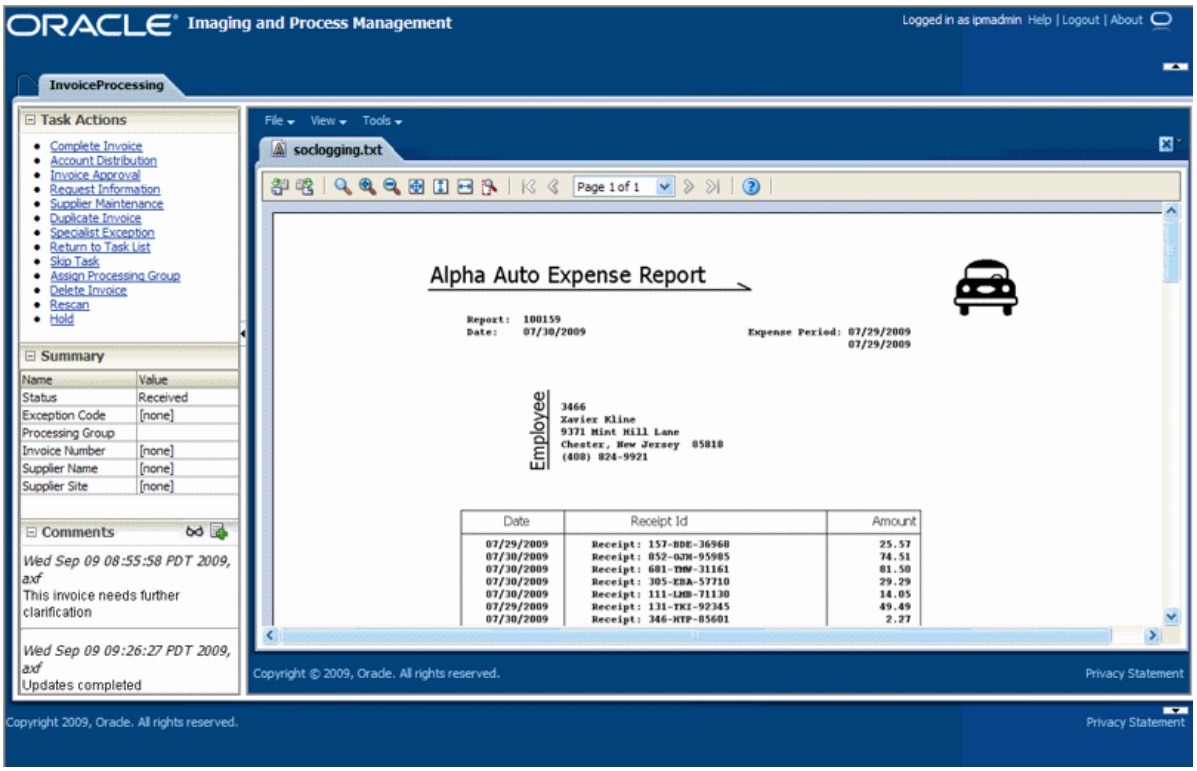
Table 4–20 Example Task List Parameters in AXF_SOLUTION_PARAMETERS Table

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

4.3.2 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.

You can customize the web page through database configuration using Java commands or AXF action commands.



Task Viewer Features

- Users view Oracle I/PM image documents in the **Image Viewer pane**, using either the basic or advanced Oracle I/PM viewer modes. Typically, the Task Viewer uses the Oracle I/PM viewer tool to render image documents. However, another tool can be used; the Task Viewer uses whichever URL has been passed into the BPEL process by Oracle I/PM's BPEL Injector.
- 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.
- 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 also 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 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.

Configuring the Task Viewer

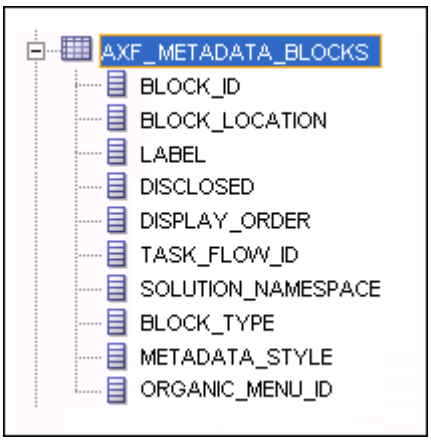
You configure the Task Viewer in the following tables:

- AXF_SOLUTION_PARAMETERS Table
- AXF_ACTION_MENU Table
- AXF_ACTIONS Table

- AXF_METADATA_BLOCKS Table
- AXF_METADATA_ATTRIBUTES Table

4.3.2.1 AXF_METADATA_BLOCKS Table

This table defines the task viewer itself and its sections to be displayed on the Task Viewer page.



4.3.2.1.1 Column Description

Table 4–21 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 is displayed on the Task Viewer page. *LEFT_SIDEBAR* displays a left sidebar pane. (Currently, this is the only value supported.)
LABEL	Defines the pane's label (for example, *Summary* or *Comments*).
DISCLOSED	TRUE if the block is displayed; FALSE if it is not displayed.
DISPLAY_ORDER	Specifies the order in which the block is displayed. The default value is 1.
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.
METADATA_STYLE	Reserved for future use.
ORGANIC_MENU_ID	Reserved for future use.

4.3.2.1.2 Example Implementation

This table displays the Summary and Comments section on the Task Viewer page.

Columns not shown: DISCLOSED=TRUE

METADATA_STYLE=null

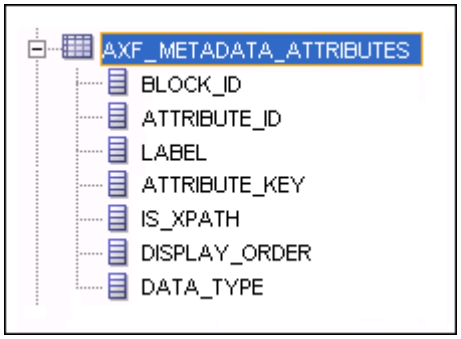
ORGANIC_MENU_ID=null

Table 4–22 Example AXF_METADATA_BLOCKS Table

BLOCK_ID	BLOCK_LOCATION	LABEL	DISPLAY_ ORDER	TASK_FLOW_ID	SOLUTION_ NAMESPACE	BLOCK_ TYPE
1	LEFT_SIDEBAR	Summary	1	axf-taskviewer-tfd	InvoiceProcessing	METADATA
AXF_METADATA_ BLOCKS_ SEQ.NEXTVAL	LEFT_SIDEBAR	Comments	1	axf-taskviewer-tfd	InvoiceProcessing	COMMENT

4.3.2.2 AXF_METADATA_ATTRIBUTES Table

This table defines the labels and values to be shown in metadata sections specified in the "AXF_METADATA_BLOCKS Table" on page 4-17. It also defines how attribute values are retrieved for display using Xpath attributes.



4.3.2.2.1 Column Description

Table 4–23 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).
ATTRIBUTE_KEY	Specifies an attribute key that matches the Xpath attribute key in the AXF_XPATH_ATTRIBUTES Table, where it is used to look up and display the metadata value.
IS_XPATH	If TRUE, the attribute value comes from the xpath in the BPEL payload. If FALSE, the value comes from BPEL system attributes.
DISPLAY_ORDER	Specifies the order in which the metadata label/value are displayed in the metadata section.
DATA_TYPE	Specifies the metadata item's data type (for example, String).

4.3.2.2.2 Example Implementation

This table defines metadata labels and values displayed in a Task Viewer's Summary section, as shown in Figure 4–2.

Columns not shown:

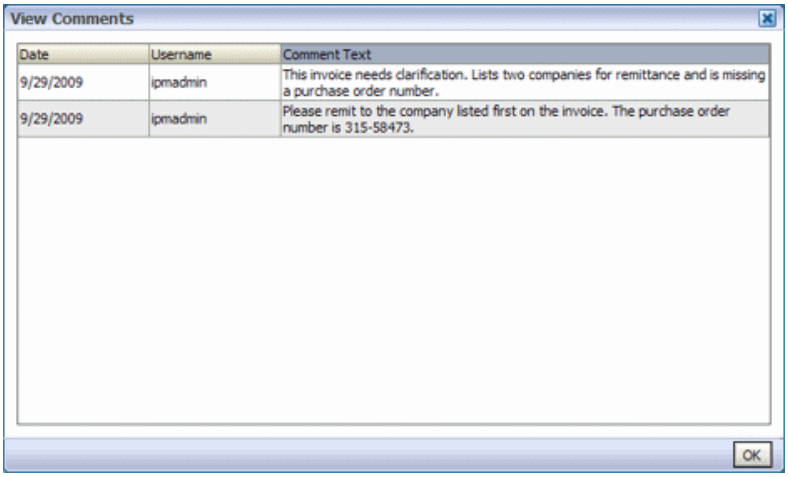
DATA_TYPE=String

Table 4–24 Example AXF_METADATA_ATTRIBUTES Table

BLOCK_ID	ATTRIBUTE_ID	LABEL	ATTRIBUTE_KEY	IS_XPATH	DISPLAY_ORDER
1	AXF_METADATA_ATTRIBUTES_SEQ.NEXTVAL	Status	InvoiceProcessing_Status	TRUE	1
1	AXF_METADATA_ATTRIBUTES_SEQ.NEXTVAL	Exception Code	InvoiceProcessing_ExceptionCode	TRUE	2
1	AXF_METADATA_ATTRIBUTES_SEQ.NEXTVAL	Processing Group	InvoiceProcessing_ProcessingGroup	TRUE	3
1	AXF_METADATA_ATTRIBUTES_SEQ.NEXTVAL	Invoice Number	InvoiceProcessing_InvoiceNumber	TRUE	4
1	AXF_METADATA_ATTRIBUTES_SEQ.NEXTVAL	Supplier Name	InvoiceProcessing_SupplierName	TRUE	5
1	AXF_METADATA_ATTRIBUTES_SEQ.NEXTVAL	Supplier Site	InvoiceProcessing_SupplierSiteName	TRUE	6

4.3.2.3 Comments

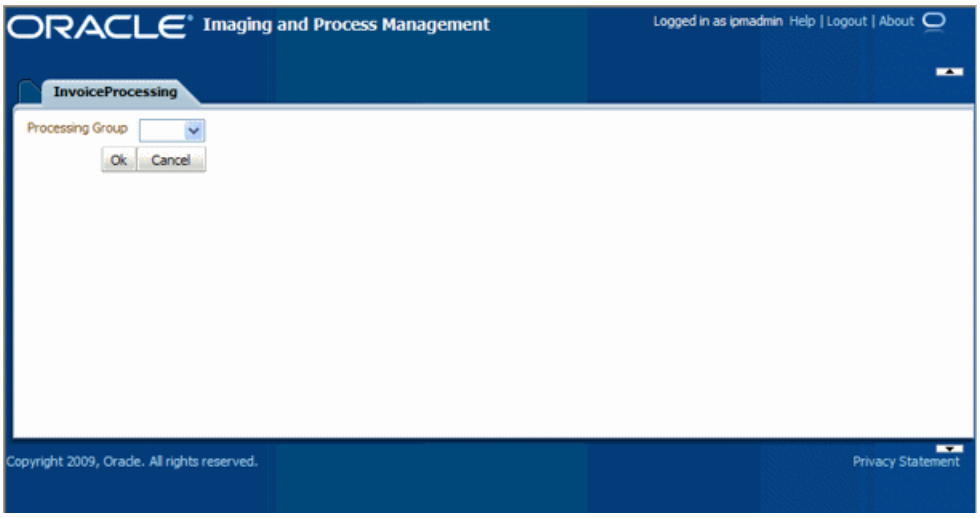
The Comments pane allows users to view and enter comments related to the human task during the transaction's processing. You configure comments in the AXF_METADATA_BLOCKS Table.



4.3.3 Enumeration Picker Web Tool

The Enumeration Picker web tool allows 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 is executed, 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 4–25).
- Define the picker in the AXF_ENUM_TYPES Table.
- Define the picker's values in the AXF_ENUM_ITEMS Table.

4.3.3.1 Enumeration Picker Parameters

Table 4–25 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 "System Attributes" on page 4-29. 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 be executed when a user clicks the Cancel button on the Enumeration Picker page.
CMD_ON_OK	Specify the command (COMMAND_NAMESPACE) to be executed 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 None value is displayed.
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.

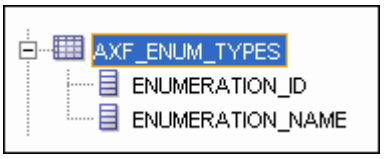
Fields not shown: SOLUTION_NAMESPACE=InvoiceProcessing

Table 4–26 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

4.3.3.2 AXF_ENUM_TYPES Table

This table defines Enumeration Pickers.



Column Description

Table 4–27 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

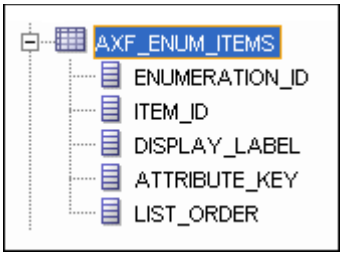
This example defines enumeration pickers for the Invoice Processing solution. Each picker's items are defined in the AXF_ENUM_ITEMS Table.

Table 4–28 Example AXF_ENUM_TYPES Table

ENUMERATION_ID	ENUMERATION_NAME
1	ProcessingGroups
2	SupplierMaintenanceCodes
3	SpecialistExceptionCodes
4	RescanCodes

4.3.3.3 AXF_ENUM_ITEMS Table

This table defines a specified Enumeration Picker's values.



Column Description

Table 4–29 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 be displayed in the picker field.
ATTRIBUTE_KEY	Specify the value to be stored 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 the value is to be listed 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 4–30 Example AXF_ENUM_ITEMS Table

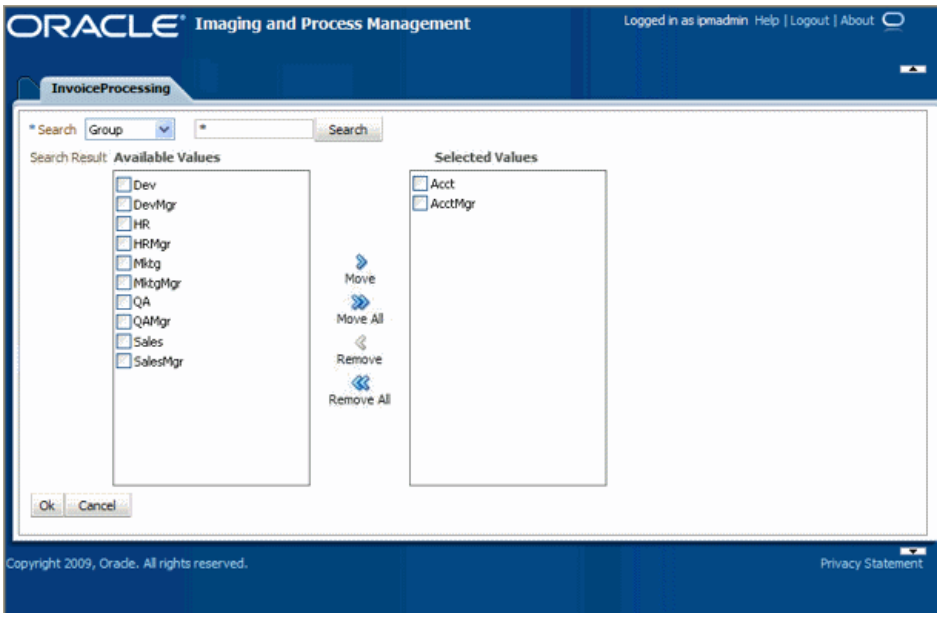
ENUMERATION_ID	ITEM_ID	DISPLAY_LABEL	ATTRIBUTE_KEY	LIST_ORDER
1	1	North	North	1
1	2	South	South	2
1	3	East	East	3
1	4	West	West	4
2	5	No Supplier	No Supplier	1
2	6	No Supplier Site	No Supplier Site	2
3	7	Duplicate Invoice	Duplicate Invoice	1
3	8	Invalid Invoice Number	Invalid Invoice Number	2
3	9	No 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

4.3.4 Identity Picker Web Tool

The Identity Picker web tool allows 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 be taken 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.

Note: The Filter and Search Attribute settings use the BPEL Worklist views configuration. Use the BPEL Workflow application to change these settings.



4.3.4.1 Identity Picker Parameters

Table 4–31 Identity Picker Parameters in AXF_SOLUTION_PARAMETERS Table

Parameter Key	Description
CMD_ON_CANCEL	Specify the command (COMMAND_NAMESPACE) to be executed when a user clicks the Cancel button on the Identity Picker page.
CMD_ON_OK	Specify the command (COMMAND_NAMESPACE) to be executed when a user clicks the OK button on the Identity Picker page.

Table 4–31 (Cont.) Identity Picker Parameters in AXF_SOLUTION_PARAMETERS Table

Parameter Key	Description
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_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" on page 4-29. 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.

4.3.4.2 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 4–32 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	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

4.4 AXF Commands

AXF commands include:

- "Open Task Command" on page 4-25
- "Autotask Command" on page 4-25
- "Release Task Command" on page 4-26
- "Complete Task Command" on page 4-27
- "Redirect Command" on page 4-28
- "Update Task Command" on page 4-28
- "Update Task From Procedure Command" on page 4-30
- "Terminate Conversation Command" on page 4-32
- "Validate Task Command" on page 4-32

AXF command-related topics include:

- "Custom Commands" on page 4-33

- "Configuring Chained Commands and Web Tools" on page 4-33

4.4.1 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.

4.4.1.1 Open Task Command Parameters

Table 4–33 lists configuration parameters for this command. These parameters are used in the AXF_SOLUTION_PARAMETERS Table to configure commands.

Table 4–33 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)

4.4.1.2 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 4–34 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

4.4.2 Autotask Command

This command initializes autotask mode, in which a new human workflow task is automatically claimed for the user.

4.4.2.1 Autotask Command Parameters

These parameters are used in the AXF_SOLUTION_PARAMETERS Table to configure Autotask commands.

Table 4–35 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 executed when there are no tasks.
BPEL_TRY_AUTO	Time in milliseconds between attempts to get the next task from the Human workflow system.

4.4.2.2 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 4–36 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

4.4.2.3 Configuring Autotask Locking

In AXF configurations with multiple simultaneous users, collisions may occur when attempting to acquire tasks in Autotask mode. You can enable or disable 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:

NAMESPACE	PARAMETER_ TYPE	PARAMETER_NAME	PARAMETER_VALUE
BPEL.default	connection	USE_AUTOTASK_LOCKING	true

4.4.3 Release Task Command

The Release Task command releases a human workflow task and displays the AXF Task List web tool, regardless of autotask mode.

4.4.3.1 Release Task Command Parameters

Table 4–37 lists configuration parameters for this command. These parameters are used in the AXF_SOLUTION_PARAMETERS Table to configure commands.

Table 4–37 Release Task Command Parameters

Parameter Key	Description
CMD_AUTOTASK_OFF	Specify the command (COMMAND_NAMESPACE) to be executed when AUTOTASK mode is off.
CMD_AUTOTASK_ON	Specify the command (COMMAND_NAMESPACE) to be executed when AUTOTASK mode is on.

4.4.3.2 Example Implementation

Fields not shown: SOLUTION_NAMESPACE=InvoiceProcessing

Table 4–38 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

4.4.4 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.

In addition, the Complete Task command can also update BPEL payload attribute values using request parameters to the command. If auto-task mode is active, the command claims the next available task and displays in the Task Viewer. If auto-task mode is not active, the command displays the Task List.

4.4.4.1 Complete Task Command Parameters

Table 4–39 lists configuration parameters for this command. These parameters are used in the AXF_SOLUTION_PARAMETERS Table to configure commands.

Table 4–39 CompleteTask Command Parameters

Parameter Key	Description
OUTCOME	Specify the outcome defined for the human work flow system. Default bpel outcomes are singular, APPROVE, or REJECT.
CMD_AUTOTASK_ON	Specify the command (COMMAND_NAMESPACE) to be executed when AUTOTASK mode is on.
CMD_AUTOTASK_OFF	Specify the command (COMMAND_NAMESPACE) to be executed when AUTOTASK mode is off.

4.4.4.2 Example Implementation

Fields not shown: SOLUTION_NAMESPACE=InvoiceProcessing

Table 4–40 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

4.4.5 Redirect Command

The Redirect command redirects the browser to an AXF web tool or other URL. The request parameters included in this URL are:

- CID (Conversation ID)
- PID (ParameterSet ID)

Any user defined request parameters should be stored as part of the PID.

The base URL comes from the database configuration. This command returns the URL in the response command.

Table 4–41 lists configuration parameters for this command. These parameters are used in the AXF_SOLUTION_PARAMETERS Table to configure commands.

4.4.5.1 Redirect Command Parameters

Table 4–41 RedirectCommand Parameters

Parameter Key	Description
REDIRECT_URL	This URL is returned in the response command upon executing this command.
EXTERNAL	If this has a value of TRUE, then the redirect page does not have a CID and PID appended to it. The redirect URL is an external Web site and all request parameters are appended in the URL.

4.4.5.2 Example Implementation

Fields not shown: SOLUTION_NAMESPACE=InvoiceProcessing

Table 4–42 Example Redirect Command in AXF_SOLUTION_PARAMETERS Table

COMMAND_NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
SearchIPM	oracle.imaging.axf.commands.system. RedirectCommand	EXTERNAL	TRUE
StartInvoiceProcessing	oracle.imaging.axf.commands.system. RedirectCommand	REDIRECT_URL	taskflow://WEB-INF/taskflows/axf-tasklist-tfd.xml#axf-tasklist-tfd

4.4.6 Update Task Command

The Update Task command updates the list of attributes in the BPEL task or updates values in the XML payload using XPATH. (For an XPATH example, see "Example Implementation" on page 4-30.)

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 "System Attributes" on page 4-29. 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.

4.4.6.1 Update Task Parameters

Table 4–43 Parameters for UpdateTaskCommand

Parameter Key	Description
outcome	Specify the outcome defined for the human work flow system. Default bpel outcomes are singular, APPROVE, or REJECT.

4.4.6.2 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
URLATTRIBUTE1 - URLATTRIBUTE5

4.4.6.3 Example Implementation

This XPATH example updates `transactionID` in the payload: the parameter key `InvoiceTransactionID` is the key defined in the request parameter. The value is `XPATH:TransactionID` where XPATH defines that the attribute `TransactionID` is defined in the AXF_XPATH_ATTRIBUTES Table.

Fields not shown: SOLUTION_NAMESPACE=InvoiceProcessing

Table 4–44 Example UpdateTaskCommand Parameters in AXF_SOLUTION_PARAMETERS Table

COMMAND_ NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
AttachSupplemental	oracle.imaging.axf.commands.bpel. UpdateTaskCommand	InvoiceTransactionID	XPATH:InvoiceProcessing_TransactionID
SaveInvoice	oracle.imaging.axf.commands.bpel. UpdateTaskCommand	InvoiceTransactionID	XPATH:InvoiceProcessing_TransactionID

4.4.7 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.

4.4.7.1 Update Task From Procedure Parameters

Table 4–45 Parameters for UpdateTaskFromProcedure Command

Parameter Key	Description
XPATH_USERS	Specifies an XPATH variable contained in the AXF_XPATH_ ATTRIBUTES Table that refers to the XPATH where the list of returned data is to be stored.
CMD_EMPTY_LIST	Specifies the command to be executed if no results are returned from the pl/sql function.
CMD_NON_EMPTY_LIST	Specifies the command to be executed if results are 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.

4.4.7.2 Example Implementation

Fields not shown: SOLUTION_NAMESPACE=InvoiceProcessing

Table 4–46 Example UpdateTaskFromProcedureCommand Parameters in AXF_SOLUTION_PARAMETERS Table

COMMAND_ NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
RetrieveUserList	oracle.imaging.axf.commands.bpel. UpdateTaskFromProcedureCommand	XPATH_USERS	XPATH:InvoiceProcessing_ InvoiceApprovalAssignment
RetrieveUserList	oracle.imaging.axf.commands.bpel. UpdateTaskFromProcedureCommand	CMD_NON_EMPTY_LIST	InvoiceApprovalEdit

Table 4–46 (Cont.) Example UpdateTaskFromProcedureCommand Parameters in AXF_SOLUTION_PARAMETERS Table

COMMAND_NAMESPACE	CONFIGURATION_NAMESPACE	PARAMETER_KEY	PARAMETER_VALUE
RetrieveUserList	oracle.imaging.axf.commands.bpel.UpdateTaskFromProcedureCommand	CMD_EMPTY_LIST	CompleteInvoice
RetrieveUserList	oracle.imaging.axf.commands.bpel.UpdateTaskFromProcedureCommand	JNDI_DS	jdbc/AXFPSF11DataSource
RetrieveUserList	oracle.imaging.axf.commands.bpel.UpdateTaskFromProcedureCommand	PLSQL_PROC	AXFRETRIEVEUSERLIST

4.4.7.3 Example PL/SQL Procedure

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 and return a VARCHAR2. The name of the function is in the AXF configuration.

```
create or replace FUNCTION axfretrieveuserlist(  xmlPayload IN VARCHAR2 ) RETURN VARCHAR2 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);

BEGIN

    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
      userList := 'jlondon';
    else
      userList := 'jcooper,mtwain';
    end if;

    RETURN userList;

END;
```

4.4.8 Terminate Conversation Command

The Terminate Conversation Command is used by an external client to terminate a conversation with AXF.

4.4.9 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 4–47 lists configuration parameters for this command. These parameters are used in the AXF_SOLUTION_PARAMETERS Table to configure commands.

4.4.9.1 Validate Task Command Parameters

Table 4–47 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.

4.4.9.2 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.

Fields not shown: SOLUTION_NAMESPACE=InvoiceProcessing

Table 4–48 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.

4.4.9.3 Example Implementation Instructions

Follow these steps to add a validation that verifies that a Transaction ID is present before allowing a task to be completed.

Note: This example assumes that you have installed the Invoice Processing template data.

Note: This configuration change should be applied only in use cases where users must create the business application invoice before the task can be completed. 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 4–49 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 4–48 to the AXF_SOLUTION_PARAMETERS table.
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 4–50 AXF_ACTIONS Table

ACTION_ID	VIEW_ID	DISPLAY_NAME	COMMAND_NAMESPACE	MENU_ORDER
CompleteInvoice	/TaskViewer.jspx	Complete Invoice	ValidateTransactionID	3

4.4.10 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.

4.4.11 Configuring Chained Commands and Web Tools

Some AXF commands have parameter keys that specify what occurs after the command is completed, allowing you to chain them. For example, Table 4–51 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 4–51 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

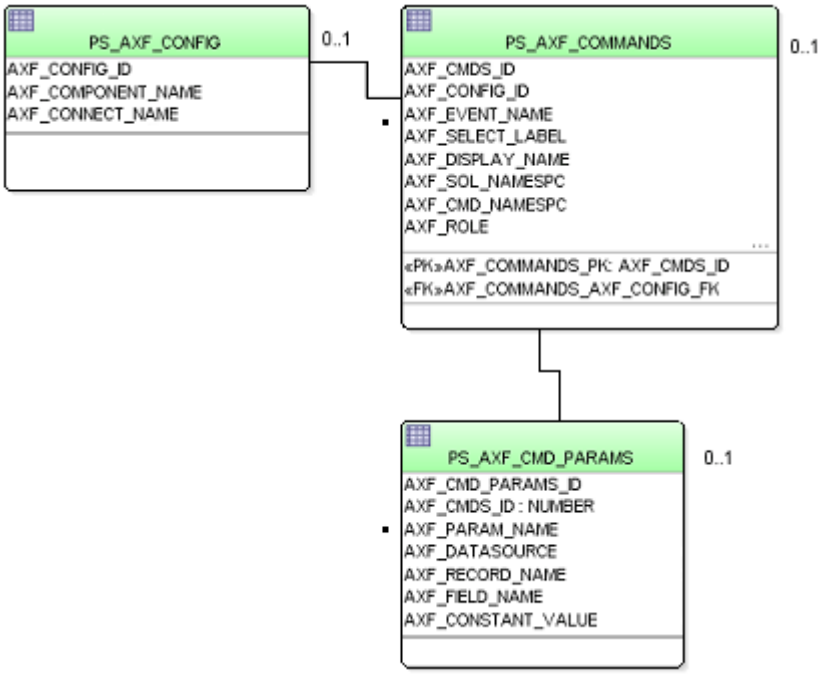
4.5 PeopleSoft Tables

Configuring AXF for PeopleSoft requires configuring AXF-related tables in PeopleSoft. For more information, see "About Configuring AXF Components on PeopleSoft Pages" on page 2-1 and "Enabling AXF Components on PeopleSoft Pages" on page 2-6.

This section covers the following topics:

4.5.1 About the AXF Tables in PeopleSoft

The following diagram shows how the tables used by the PeopleSoft system in AXF solutions are related.



Note: All AXF table fields require non-null values. You can include a space for fields that are not applicable.

4.5.2 PS_AXF_CONFIG Table

Use the PS_AXF_CONFIG table to enable the AXF solution on various PeopleSoft components. This table allows a fine level of granularity when selecting which pages and components are AXF-enabled.

Events are invoked automatically when an action is performed on a 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.

4.5.2.1 Column Description

Table 4–52 Column Description for PS_AXF_CONFIG Table

Column Name	Description	Data Type
AXF_CONFIG_ID	Specifies the primary key of the table.	Number
AXF_COMPONENT_ NAME	Specifies the name of the PeopleSoft component being enabled.	Char18
AXF_CONNECT_NAME	Specifies the PeopleSoft Integration Broker Connection name (service operation to call), as defined in the PeopleSoft Integration Broker administration interface. The default service operation is AXF_ EXECUTE. You can set up and use other connections.	Char64
AXF_VERSION	Specifies the AXF version of the connection. Available values include: ▪ 10g ▪ 11g	Char20

4.5.2.2 Example Implementation

This example defines that the VCHR_QUICK_PNL component is AXF-enabled.

Table 4–53 Example PS_AXF_CONFIG Table

AXF_CONFIG_ID	AXF_COMPONENT_NAME	AXF_CONNECT_NAME	AXF_VERSION
1	VCHR_QUICK_PNL	AXF_EXECUTE	11g

4.5.3 PS_AXF_COMMANDS Table

Use the PS_AXF_COMMANDS table to describe the actions to be taken based on user activity. This table works with the PS_AXF_CONFIG Table.

4.5.3.1 Column Description

Table 4–54 Column Description for PS_AXF_COMMANDS Table

Column Name	Description	Data Type
AXF_CMDS_ID	Specifies the primary key of the table, used to identify the unique command.	Number
AXF_CONFIG_ID	Specifies the foreign key to the PS_AXF_CONFIG Table, which associates this unique command with a particular page and component.	Number

Table 4–54 (Cont.) Column Description for PS_AXF_COMMANDS Table

Column Name	Description	Data Type
AXF_EVENT_NAME	Specifies the event being executed. The AXF_ EVENT_NAME corresponds to the subpage that is incorporated into an existing PeopleSoft page, such as the button, link, or menu that is added to a PeopleSoft page to invoke AXF functionality such as the Image Viewer or Task List. Available options are: ■ 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	char(80)
AXF_SELECT_LABEL	Defines the name displayed on the page for the selection field. This is used only with the COMBO event.	char(64)
AXF_DISPLAY_NAME	Defines the name of the button or link to display on the PeopleSoft screen.	char(20)
AXF_SOL_NAMESPC	Identifies the solution namespace for the command to execute.	char(254)
AXF_CMD_NAMESPC	Identifies the command namespace for the command to execute.	char(254)
AXF_PSFT_ROLE	Identifies the 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').	char(254)
AXF_SORT_ORDER	Specifies the order of items displayed in a selection field. Sort order applies to selection fields only.	Number
AXF_REQ_CONV	Specifies if a conversation is required to this command before execution. For example, execution of the UpdateTask command requires a conversation to be running in order for the user to select a current task.	char(1)

4.5.3.2 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 Oracle I/PM 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 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 PeopleSoft.

You must specify a 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 are not displayed. If the commands are displayed but not functioning, this indicates that the commands are not configured properly.

Table 4–55 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
1	1	AXF_BUTTON_1	(null)	Start Invoice Processing	InvoiceProcessing	StartInvoiceProcessing	'Employee'	1	N
2	1	AXF_LINK_1	(null)	Search For Duplicates	InvoiceInquiry	SearchIPM	'Employee'	1	N
3	1	SAVE_POST_CHANGE	(null)	(null)	InvoiceProcessing	SaveInvoice	'Employee'	1	N

4.5.4 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.

4.5.4.1 Column Description

Table 4–56 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 from where the parameter value is retrieved. ▪ DATA: Retrieves the value in 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 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.

4.5.4.2 Example Implementation

The example that follows contains two parameters sent for AXF_CMDS_ID 2: a constant value (SearchName) and a data value (INVOICENUMBER).

Table 4–57 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)	SearchByInvoiceID
2	2	INVOICENUMBER	DATA	VCHR_HDR_QV	INVOICE_ID	(null)
3	3	InvoiceTransactionID	DATA	VCHR_HDR_QV	VOUCHER_ID	(null)

Index

A

B

C

D

E

G

I

L

M

N

O

P

R

S

T

U

V

W

X