ORACLE®

PRIMAVERA

**P6 EPPM BPM Configuration Guide
16 R2**

September 2016

Contents

Oracle BPM Setup Tasks

The Oracle Business Process Management (BPM) Suite provides an integrated environment for developing, administering, and using business applications centered around business processes. BPM supports BPM and Business Process Execution Language (BPEL) standards from modeling and implementation to run-time and monitoring.

P6 integrates with BPM which lets you initiate and manage workflows. You can use a sample project initiation workflow for P6 sample database that is available on OTN.

You can expand your investment in BPM to include workflows representing more stages of your application, program, project, or product development life cycle from design-time and implementation to run-time and application management.

The Oracle BPM Suite enables you to:

▸ Create and customize business processes, models, and standards using pre-defined components for web-based applications.
▸ Collaborate between process developers and process analysts.
▸ Expand business process management to include flexible, unstructured processes.
▸ Integrate your applications with Web Services.
▸ Add dynamic tasks and support approval routing using declarative patterns and rules-driven flow determination.

Unify different stages of your development life cycle by addressing end-to-end requirements for developing process-based applications. Oracle BPM unifies the design, implementation, run time, and monitoring stages based on a Service-Oriented Architecture (SOA) infrastructure. This allows different personas to participate through all stages of the workflow life-cycle.

In This Section

Pre-Integration Requirements for BPM: Start Here

Before continuing, ensure the following required conditions have been addressed:

> **Note**: See the *Tested Configurations* document for information on supported versions.

1) On the domain where you will install P6 EPPM, configure WebLogic Middleware with Oracle Application Development Framework (Oracle ADF). This will ensure the P6 EPPM domain will have Oracle WSM Policy Manager and Oracle JRF modules, which you must have when integrating P6 EPPM with BPM.   To download the Oracle ADF, refer to the following link: http://www.oracle.com/technetwork/developer-tools/adf/downloads/index.html

   > **Note**:   If you installed the Oracle Fusion Middleware SOA Suite to the same WebLogic Middleware home where you installed P6 EPPM, then the Oracle ADF is already available and you can skip this step.

2) Download the Oracle BPM Suite documentation from http://www.oracle.com/technetwork/middleware/soasuite/documentation/index.html. See Tested Configurations for the supported versions of BPM.

3) See http://www.oracle.com/technetwork/middleware/soasuite/downloads/index.html to accept a license agreement and download a BPM package.

4) Install Oracle BPM. See the BPM documentation at http://www.oracle.com/technetwork/middleware/soasuite/documentation/index.html to guide you. When you design and develop your own workflows in 12c, you will need JDeveloper with extensions or Oracle Business Process Composer for 12c.

   > **Note:** You can use BPM with P6 EPPM.

5) Install and configure P6 EPPM. This includes having completed any licensing requirements and downloading any documentation for P6 EPPM, including P6 and P6 EPPM Web Services.

   a. Make sure you have a working configuration of P6. The term *P6* refers to the web application user interface for the main module in the P6 EPPM suite.

   b. You must deploy the Oracle WSM Policy and JRF modules for P6 to work with BPM. If you use the P6 EPPM Configuration Wizard, you can use it to deploy the Oracle WSM Policy and JRF modules. You will have to have a WebLogic domain to deploy the policy and module. You can also deploy the Oracle WSM Policy and JRF modules manually in WebLogic if you choose not to use the configuration wizard to do so.

   > **Note**: P6 EPPM Web Services and P6 (which includes Oracle WSM Policy manager and Oracle WSM JRF modules) can be on the same domain or separate domains. However, if OWSM is enabled for P6 EPPM Web Services, then BPM will not work with P6 .

   c. Make sure you have installed P6 EPPM Web Services and ensure its authentication mode matches BPM's authentication mode. For example, *Username Token Profile* or *SAML*.

> **Note**: Cookie authentication is not currently supported at this time.

6) Configure both BPM and P6 to work together. See ***Integrating Oracle BPM 12c with P6*** in this guide for steps.

7) To ensure the BPM functionality is available in P6, you must set up a username in BPM that matches an equivalent P6 username (only the username value must match; Passwords can be different between the environments).

   You must assign the user in BPM the appropriate role to perform the functional requirements. Refer to ***Testing the Connection for P6, BPM, and Web Services*** (on page 12).

Integrating Oracle BPM with P6

To integrate Oracle BPM with P6, you must complete the following tasks in the specified order:

1) ***Modifying the P6 Environment for BPM***
2) ***Configuring P6 to Connect to Oracle BPM***

Modifying the P6 Environment for BPM

To configure BPM with P6:

1) Create a new directory on the host where P6 is running. For example, **P6BPMConnector/P6** and **P6BPMConnector/BPM**.

In the P6 media pack `Tools\BPM Connector\12c` subfolder, locate the `prm-bpm-connector-12c-v1.0.jar`

1) Copy the connector file listed above to the **P6BPMConnector/P6** folder you created. Determine the location of the BPM installation that you are using to integrate with P6. If you have multiple SOA Suite installations, ensure that you determine the location of the specific installation that you are integrating with P6.

2) Copy all the supporting jars from the BPM installation that you are using to integrate with P6 to the **P6BPMConnector/BPM** folder you created. Their default locations are listed:

   ▸ *<Oracle_Home>*`\soa\soa\modules\oracle.soa.fabric_11.x.x\bpm-infra.jar`
   ▸ *<Oracle_Home>*`\soa\soa\modules\oracle.soa.workflow_11.x.x\bpm-services.jar`
   ▸ *<Oracle_Home>*`\soa\soa\modules\oracle.soa.fabric_11.x.x\fabric-runtime.jar`
   ▸ *<Oracle_Home>*`\oracle_common\modules\oracle.jmx_12.x.x\jmxframework.jar`
   ▸ *<Oracle_Home>*`\oracle_common\modules\oracle.jmx_12.x.x\jmxspi.jar`
   ▸ *<Oracle_Home>*`\soa\bpm\modules\oracle.bpm.mgmt_11.x.x\oracle.bpm.bpmn-em-tools.jar`

- ▸ *<Oracle_Home>*\soa\soa\modules\oracle.soa.mgmt_11.x.x\soa-infra-mgmt.jar
- ▸ *<Oracle_Home>*\wlserver\server\lib\wlclient.jar
- ▸ *<Oracle_Home>*\oracle_common\modules\oracle.xdk_12.x.x\xml.jar
- ▸ *<Oracle_Home>*\oracle_common\modules\oracle.xdk_12.x.x\xmlparserv2.jar
- ▸ *<Oracle_Home>*\soa\soa\modules\oracle.soa.fabric_11.x.x\tracking-api.jar
- ▸ *<Oracle_Home>*\soa\soa\modules\oracle.rules_11.x.x\rulesdk2.jar

> **Note:** The jar files must be from the same BPM installation that will be integrated with P6.

Configuring P6 to Connect to Oracle BPM

To connect P6 and BPM:

1) Open the Primavera P6 Administrator.
2) In the **Configurations** tab, expand your configuration.
3) Expand **Database/Instance[n]/BPM Settings**.
4) In the **Connector file location** field, enter the full path (including the file name) where you copied the connector files.

    For example:

    `c:/oracle/bpm`*<release_level>*`/eppmjars/prm-bpm-connector-12c-v1.0.jar` or `/home/oracle/bpm1213x/prm-bpm-connector-12c-v1.0.jar`

5) In the **BPM library path** field, enter the path of the directory where you copied all the   jars.

    > **Note**: If the above settings are incorrect or the BPM jar files are incomplete, you will be prompted with an error message when you try to configure the BPM. Verify that your settings and jar files are correct.

6) Right-click **BPM Configuration**, then select **Configure** to set options from the dialog box. If you receive an error message, check your values from the previous steps in this section.
7) To configure P6 to use BPM, add the following settings:

    > **Caution**: Do not put a forward slash / at the end of the URL. For example, it should read only `http://`*<host_name>*`:`*<port>*, and not `http://`*<host_name>*`:`*<port>*`/`.

    a. In the **bpm.user** field, enter the BPM user with administrative access to BPM.
    b. In the **bpm.password** field, enter the password for the user.
    c. In the **bpm.t3.url** field, enter the T3 URL for your Oracle SOA configuration. For example: `t3://`*<host_name>*`:`*<port>*`/soa-infra`

    d. In the **bpm.security.realm** field, enter the name of the security realm used by BPM. This is `jazn.com` by default.

    e. In the **bpm.soap.url** field, enter the URL for the SOAP services. Usually this URL takes the form of `http://<host_name>:<port>`.

    f. In the **bpm.workspace.url** field, enter an address in the form of `http://<host_name>:<port>` that indicates where the BPM Workspace application is hosted.

> **Note**: The host can be an IP address or a host name such as a machine name.

    g. In the **bpm.partition** field, enter the name of the SOA partition containing the BPM composite applications you would like to use in P6. The default partition name is `default`.

8) Click **OK** and save changes. When the configuration is complete, you will see a **BPM properties have been configured** message.

9) Restart the P6 application server.

Verifying the Sample Workflow Configuration

The sample workflow provided uses Username Token Authentication and does not implement the message protection options of Nonce, Creation Timestamp, or SSL encryption. For the sample workflow to work, you must set the following settings to false or P6 EPPM Web Services will reject the messages sent from BPM.

To verify you configured the sample workflow correctly set the following to false:

1) Open the Primavera P6 Administrator.

2) In the **Configurations** tab, expand your configuration.

3) Expand **Web Services/Security/Authentication**.

4) In the **Mode** field, select **Username Token Profile**.

5) Expand **Web Services/Security/Authentication/Username Token Profile/Nonce** and set **Require Nonce** to **false**.

6) Expand **Web Services/Security/Authentication/Username Token Profile/Created** and set **Require Created** to **false**.

7) Expand **Web Services/Security/Message Protection**:

    a. Set **Require Timestamp** to **false**.

    b. Set **Require Digital Signatures for Incoming Messages** to **false**.

    c. Set **Require Encryption for Incoming Messages** to **false**.

    d. Set **Encrypt Response** to **false**.

Creating a Cluster Environment for BPM in WebLogic

Follow the instructions in the following sections to create a cluster environment for BPM in WebLogic.

Prerequisites for Creating a Cluster for BPM in WebLogic

Before you create a cluster:

> **Note**: See the Tested Configurations document for supported versions.

1) Install the **Oracle MDS (Metadata Services) schema** using the **Repository Creation Unit (RCU)** on a separate database instance.
2) Install **Oracle ADF Runtime** on the Middleware Home for all nodes where you configured P6 EPPM.
3) Create a cluster for BPM in WebLogic. Follow WebLogic's instructions to create a cluster. The following instructions will refer to the cluster as **BPMCLUSTER_domain**.

Extending the WebLogic Domain to Create a New Cluster

To extend the WebLogic Domain:

1) Stop the P6 EPPM servers:
   ▸ If you started the servers using Node Manager (for example, start_Primavera.bat), run **stop_Primavera.bat** to stop all running servers.
   ▸ If you started the servers using startWebLogic.cmd/startManagedWebLogic.cmd in Windows or startWebLogic.sh/startManagedWebLogic.sh in Unix, use the "stop" version for those files to stop the servers.
2) Run the WebLogic **Configuration Wizard**.
3) In the **Welcome** window:
   a. Select **Extend an existing WebLogic domain**.
   b. Click **Next**.
4) In the **Select a WebLogic Domain Directory** window:
   a. Expand **domains**.
   b. Select the BPM Cluster domain (for example, BPMCLUSTER_domain), which you must extend with policy manager.
   c. Click **Next**.
5) In the **Select Extension Source** window:
   a. Select **Extend my domain automatically to support the following added products:**.
   b. Select **Oracle WSM Policy Manager - 12.1.3.0 [oracle_common]**. When you select this option, the wizard will automatically select **Oracle JRF - 12.1.3.0 [oracle_common]**.
   c. Click **Next**.
6) In the **Configure JDBC Component Schema** window:

    a. Enter your MDS schema details. The MDS details are the DB configuration details where you installed the Oracle MDS schema.

    b. Click **Next**.

7) In the **Test JDBC Component Schema** window:

    a. Select your schema.

    b. Click **Test Connections**.

    c. If the test is successful, click **Next**.

       If the test fails, go back to the previous screen and ensure your credentials are correct.

8) In the **Select Optional Configuration** window:

    a. Select **Managed Servers, Clusters and Machines** and **Deployments and Services**.

    b. Click **Next**.

9) In the **Configured Managed Servers** window, do not make changes and click **Next**.

10) In the **Configure Clusters** window, do not make changes and click **Next**.

11) In the **Assign Servers to Clusters** window, do not make changes and click **Next**.

12) In the **Configure Machines** window, do not make changes and click **Next**.

13) In the **Assign Servers to Machines** window, do not make changes and click **Next**.

14) In the **Target Deployments to Clusters or Servers** window:

    a. In the left pane, select **Cluster**.

       1. In the right pane under **Applications**, select **wsm-pm**.

       2. In the right pane under **Library**, select all options.

    b. In the left pane, select another server.

       1. In the right pane under **Applications**, select the application that corresponds to that server. For example, if you selected the P6 server, select **P6** under **Applications**.

       2. Clear all other options for that server.

       3. Repeat these steps for each server.

    c. In the left pane, select **Cluster** and select all the applications that belong in that cluster from the right pane.

    d. Click **Next**.

15) In the **Target Services to Clusters or Servers** window:

    a. In the left pane, select **Cluster** and select all options in the right pane.

    b. In the left pane, select **Cluster** and clear all options in the right pane.

    c. Click **Next**.

16) On the Configuration Summary screen:

    a. Verify the configuration summary.

    b. Click Extend.

    c. When the extension is complete, click **Done**.

Finalizing the Extended Schema for BPM

To finalize the extended schema for BPM:

1) Copy this extended domain to all node server machines that are part of the created cluster.
2) Set StartScriptEnabled to true:
   a. Go to the Node Manager folder in the WebLogic home (for example, C:\Oracle\Middleware\Oracle_Home\user_projects\<P6 EPPM_Domain>\nodemanager)
   b. Edit the **nodemanager.properties** file.\
   c. Set **StartScriptEnabled = true**.
   d. Ensure that **StartScriptName=startWebLogic.cmd** or **StartScriptName=startWebLogic.sh** depending on your operating system.
3) Start the servers:
   ▸ Run **start_Primavera.bat**/ **start_Primavera.sh** to start all the servers through node manager.
   ▸ To start the servers manually, run **startWeblogic.cmd**/**startManagedWeblogic.cmd** for Windows or **startWeblogic.sh**/**startManagedWeblogic.sh** for Unix.

Testing the Connection for P6, BPM, and Web Services

This section includes steps to to deploy and configure the P6ConfigValidator workflow, which you can use to test a web services call.

Configuring WSDL Location and Service Endpoints

The configuration plan references the generic endpoint: p682ws:7011. The configuration plan also contains BPEL properties in the workflow to configure the target environment. In this workflow, these entries consist only of the P6 user name, password, and database instance to use when making the test Web Service call.

To prepare the configuration plan:

1) Unzip the files in the `P6_<release_level>_Tools\BPM Connector\p6ConfigValidator 2.0 [12.1.3.0].zip` folder from the Media Pack.
2) Edit the **P6ConfigValidator_cfgplan.xml** file.
3) Find the attribute **<replace />** and change the following values:
   a. Find **<property name="bpel.preference.p6userName">** and replace the P6 username value between the "<replace> </replace>" elements with a P6 username from your environment. Oracle recommends using a user assigned to the Admin Superuser Global security profile.
   b. Find **<property name="bpel.preference.p6password">** and replace the P6 password value between the "<replace> </replace>" elements with the password of the user from above.
   c. Find **<replace>localhost:7011</replace>** and replace the value between the "<replace> </replace>" elements with the hostname:port where you deployed P6 Web Services.

4) Save the changes made to the **P6ConfigValidator_cfgplan.xml** file and close the text editor.

Deploying the SOA Composite

To deploy the JAR for the SOA composite:

1) Login to Enterprise Manager on the WebLogic Server hosting SOA.
2) Expand **SOA**, **soa-infra**, *server_name*.
3) Right-click on the target soa-infra partition and select **SOA Deployment**, **Deploy to This Partition...**.
4) Specify the location of the service archive file **sca_P6ConfigValidator_rev1.0.jar** and the edited configuration plan (**P6ConfigValidator_cfgplan.xml**) for this environment.
5) Click **Next** to confirm the remaining options for the target environment.
6) Deploy the JAR for the SOA composite.

Deploy Client Application and Task UI Project

To deploy the P6ConfigUI.ear as a standard application deployment to the same server where the SOA is running:

1) Login to Enterprise Manager.
2) Expand **WebLogic Domain**, *domain_name*.
3) Right-click on the server name and select **Application Deployment**, **Deploy**.
4) Specify the location of **P6ConfigUI.ear**.
5) (optional) Specify the location of the deployment plan if needed.
6) Click **Next** to confirm the remaining options for the target environment.
7) Deploy **P6ConfigUI.ear**.

Assigning the TestConfig Role to Users

To use this workflow, you need to assign a BPM role for the user to initiate the test and receive the confirmation the test was successful.

To assign the role:

1) Login to the BPM Workspace as a user with administrative rights.
2) Click **Administration** on the top right toolbar.
3) On the **Organization Roles** list, select **P6ConfigValidator.TestConfig**.
4) On the **Details** pane, make one or more BPM users or user groups a member of the **TestConfig** role. The BPM users or user groups assigned to the TestConfig role must match a P6 username for the workflow to be visible from P6.
5) Click **Apply** to save these changes.

Troubleshooting and Known Issues for BPM

Use the BPM and P6 EPPM Web Services log viewers to troubleshoot problems if they arise and check for know issues with BPM.

> **Note**: As a general rule when troubleshooting workflow failures, first check the BPM diagnostics to determine at what point in the process the workflow failed. If the failure is related to retrieving data from P6, then you should check the P6 EPPM Web Services logs. Also be sure to check the P6 Help and other documentation for both P6 and BPM.

Known Issues

When integrating P6 and BPM, you can use only English as the language. While both P6 and BPM both support localization, the API for integrating these two applications supports only English.

Checking the P6 EPPM Web Services Logs

The P6 Integration API uses the Java Logging API to handle log messages. Message levels that the API logs range from FINEST to SEVERE, in which FINEST logs the most messages and SEVERE logs the least messages. Additionally, there is a level ALL, which logs all messages; however, this setting could potentially impact performance.

Configuring the Logging Level

You configure the logging level by specifying and then editing your own declared logging configuration file (see http://download.oracle.com/javase/) by adding or modifying the following lines:

```
com.primavera.integration.level = <level>
com.primavera.ws.level = <level>
```

Where <level> is one of the following values: FINEST, FINER, FINE, CONFIG, INFO, WARNING, SEVERE, ALL, OFF. For example, to set the logging level to ALL, use the following:

```
com.primavera.integration.level = ALL
com.primavera.ws.level = ALL
```

Setting Logging On and Off

By default the API logging is turned off. You can turn logging on by uncommenting the following line in the cxf.xml that is supplied in the default P6 Integration API server deployment:

```
<!-- <cxf:logging /> -->
```

After removing the comment markers, the line would appear as follows:

```
<cxf:logging />
```

> **Note**: You can find the cfx.xml here:
> ```
> p6ws.war\WEB-INF\classes\cxf.xml
> ```

Oracle BPM Logging

Refer to the following file to access BPM messages:

```
<weblogic_domain>/servers/AdminServer/logs/AdminServer.log
```

In a typical BPM installation on Linux, the WebLogic domain for Fusion Middleware is:

```
<WebLogic_domain>/user_projects/domains/domain1
```

Troubleshooting Scenarios

Outside of the scope of general issues with BPM or P6 EPPM, the P6 BPM Integration can potentially yield two types of issues:

1) connector configuration issues
2) data implementation issues rooted in either P6 or the BPM server

Troubleshooting Connector Failures to Load Due to Configuration Issues

1) In the Primavera P6 Administrator, set the **Log/Console Logger/Severity Level** to *debug* or *info*.
2) Check the P6 log for BPM related messages. The log file is `P6WebAccess.html` and its location is specified in `BREBootStrap.xml` located in your P6 EPPM home folder. These messages should indicate the cause of the connector failing to load.

Troubleshooting P6 Workflows Portlet Failures To Load Data, Show Forms, Status Images, or Initiate a Process

These type of errors could have their root cause in either the P6 application or the BPM server.

1) In Primavera P6 Administrator, set the **Log/Console Logger/Severity Level** to *debug* or *info*.
2) Check the P6 log for BPM related messages. The log file is `P6WebAccess.html` and its location is specified in `BREBootStrap.xml` located in your P6 EPPM home folder. These messages should indicate why the BPM code failed to process normally.

3) Check the BPM logs in these cases to make sure that the cause of the failure is not due to the BPM server.

Modifying the P6 Environment for BPM

To configure BPM with P6:

1) Create a new directory on the host where P6 is running. For example, **P6BPMConnector/P6** and **P6BPMConnector/BPM**.

In the P6 media pack `Tools\BPM Connector\12c` subfolder, locate the `prm-bpm-connector-12c-v1.0.jar`

1) Copy the connector file listed above to the **P6BPMConnector/P6** folder you created. Determine the location of the BPM installation that you are using to integrate with P6. If you have multiple SOA Suite installations, ensure that you determine the location of the specific installation that you are integrating with P6.

2) Copy all the supporting jars from the BPM installation that you are using to integrate with P6 to the **P6BPMConnector/BPM** folder you created. Their default locations are listed:

  ‣ *<Oracle_Home>*`\soa\soa\modules\oracle.soa.fabric_11.x.x\bpm-infra.jar`

  ‣ *<Oracle_Home>*`\soa\soa\modules\oracle.soa.workflow_11.x.x\bpm-services.jar`

  ‣ *<Oracle_Home>*`\soa\soa\modules\oracle.soa.fabric_11.x.x\fabric-runtime.jar`

  ‣ *<Oracle_Home>*`\oracle_common\modules\oracle.jmx_12.x.x\jmxframework.jar`

  ‣ *<Oracle_Home>*`\oracle_common\modules\oracle.jmx_12.x.x\jmxspi.jar`

  ‣ *<Oracle_Home>*`\soa\bpm\modules\oracle.bpm.mgmt_11.x.x\oracle.bpm.bpmn-em-tools.jar`

  ‣ *<Oracle_Home>*`\soa\soa\modules\oracle.soa.mgmt_11.x.x\soa-infra-mgmt.jar`

  ‣ *<Oracle_Home>*`\wlserver\server\lib\wlclient.jar`

  ‣ *<Oracle_Home>*`\oracle_common\modules\oracle.xdk_12.x.x\xml.jar`

  ‣ *<Oracle_Home>*`\oracle_common\modules\oracle.xdk_12.x.x\xmlparserv2.jar`

  ‣ *<Oracle_Home>*`\soa\soa\modules\oracle.soa.fabric_11.x.x\tracking-api.jar`

  ‣ *<Oracle_Home>*`\soa\soa\modules\oracle.rules_11.x.x\rulesdk2.jar`

  > **Note:** The jar files must be from the same BPM installation that will be integrated with P6.

BPM Workflows in P6

The following sections detail information about workflows and how to work with them.

In This Section

About Workflows

A workflow is an automated business process that routes information and tasks between participants according to a defined set of procedures or rules designed to coordinate a specific business goal. Workflows are primarily characterized by their level of procedural automation involving one or more dynamic related series of processes, and their combination of human and machine-based tasks involving interaction with software and systems.

The following industry segments, marked by relatively high office labor costs and transaction volume, have demonstrated successful workflow implementations:

▶ Insurance
▶ Banking
▶ Legal
▶ General & Administrative
▶ Design
▶ Engineering
▶ Manufacturing

Business process modeling and workflow automation allow transactions to be conducted electronically without the need for manual intervention such as conducting certain validations or re-keying data. When workflow IT systems are processing repetitive, mundane, and often error-prone work, talented staff resources become available to handle activities that add real value to the enterprise.

Working with Workflows in P6

You can use workflows to route business processes such as project initiation requests through your organization to gather information and visibility before a go/no go decision is made. Template data, routing designators, and approval rules can be set for each stage of a workflow. To illustrate these options, pretend we have a workflow involving five key approval managers. You can define the workflow such that all five must approve and even specify a particular sequence, if any. A much more relaxed approval rule would require only one out of the five to approve. The following are just some example of how you can use workflows.

Workflows are defined, deployed, and configured in BPM where your workflow designer defines the workflow tasks involved and assigns them to specific users, roles, or groups. Then, in P6, a business need kicks off an instance of the workflow and its required tasks are automatically routed to their users, roles, or groups.

When a specific user or any user assigned to a role or group logs into P6, the Workflows portlet on their dashboard will display their relevant tasks at this stage of the workflow, as authenticated by BPM. As a workflow participant, you can select a task in the workflow instance and claim ownership for it. This means you will be responsible for performing the task. The application refreshes itself to show only the actions permitted for this stage of the workflow for you (the currently logged in user).

After your administrator sets up BPM for P6, they can configure a dashboard to display the Workflows portlet. The following list represents a list of the key Workflow elements that you can observe in the portlet depending on your configuration.

▸ **Action Required Tab**: This tab shows the tasks that are important to you (the currently logged in user).
▸ **My Workflows Tab**: This tab enables you to view all workflows according to role and status filters you can set.
▸ **Initiate a Workflow**: Click Initiate a Workflow to start a new instance of a workflow based on a predesigned template.
▸ **BPM Workspace**: Use the BPM Workspace to update the progress of tasks, initiate a change, request a project, and retrieve project information. You are also able to apply a bulk action to multiple work items.

> **Note**: If SSO authentication is not configured with BPM, you must log into BPM in the resulting window, close that window, and then return to P6 and click _ View Form again. This procedure is required whenever your BPM session expires.

▸ **Sample Workflow**: A basic workflow image with tasks for a business user, two project offices, and a project manager.
▸ **Workflow History**: View a chronological sequence of all the previous actions, users, and stages in the current workflow.

Appendix

In This Section

Configuring SAML Web Service Clients for Identity Switching without Message Protection

Oracle Web Services Manager (WSM) includes
wss11_saml_token_identity_switch_with_message_protection_client_policy, which
enables identity switching. Identity switching means that the policy propagates a different identity
than the one based on the authenticated Subject.

The Service-Oriented Architecture (SOA) application requires you to specify which user identity to
use in client-side Web service policies, and then dynamically switches the user associated with
the SAML token in the outbound Web service request. Instead of using the user name from the
Subject, this policy allows you to set a new user name when sending the SAML Web service
request.

The **wss11_saml_token_identity_switch_with_message_protection_client_policy** creates
the SAML token based on the user ID set via the property
javax.xml.ws.security.auth.username.

The initial identity switching policy requires message encryption, which requires the server-side
policy to be the same. You will not want this policy when working in P6. To change the policy, you
need to create a new client-side policy based on the existing identity switching policy (this is done
through Enterprise Manager (EM), using the "create like" option). Within the new policy definition,
you can remove the existing assertion (SAML 1.1 SAML with Certificates) and replace it with a
new assertion based on an appropriate template, which in this case is WS-Security SAML Token
Client.

Message Protection Policy

You can configure policies in the **Web Services Policies, Edit Policy** section of EM. You can also
copy the custom SAML Identity Policy. To do this, copy the
oracle_wss11_saml_token_identity_switch_with_message_protection_client_policy.txt file
located here:
*http://download.oracle.com/docs/cd/E20686_01/English/Technical_Documentation/Oracle
_BPM/oracle_wss11_saml_token_identity_switch_with_message_protection_client_polic
y.txt*
*http://docs.oracle.com/cd/E20686_01/English/Technical_Documentation/Oracle_BPM/orac
le_wss11_saml_token_identity_switch_with_message_protection_client_policy.txt*.

Setting the WSIdentityPermission

The Web service client (for example, the SOA reference binding component) where you attached the **wss11_saml_token_identity_switch_with_message_protection_client_policy** must have the **oracle.wsm.security.WSIdentityPermission**.

To use Fusion Middleware Control and add the **oracle.wsm.security.WSIdentityPermission** to the SOA reference binding component as a System Grant, perform the following steps:

1) In the **Navigator** pane, expand **WebLogic Domain** to show the domain where you need to configure the application. Select the domain.

2) Using **Fusion Middleware Control**, click **WebLogic Domain**, **Security**, **System Policies**. System policies are the system-wide policies applied to all applications deployed to the current WebLogic Domain.

3) From the **System Policies** page, select the arrow icon in the **Permission** field to search the system security grants.

4) Select one of the codebase permissions to use as a starting point and click **Create Like**.

5) In the **Grant Details** section of the page, enter **file:${common.components.home}/modules/oracle.wsm.agent.common_11.1.1/wsm-agent-core.jar** in the **Codebase** field.

6) In the **Permissions** section of the page, select the starting point permission class and click **Edit**.

7) In the **Permission Class** field, enter **oracle.wsm.security.WSIdentityPermission**. The resource name is the composite name for SOA and the application name for a J2EE client. The action is always **assert**.

Creating the basic.credentials Key

You also need to add the basic.credentials key to the csf store via EM. You might need to create a default keystore if you have not done that already.

1) Right-click **Domain** then select **Security**, **Credentials**.

2) Create a **basic.credentials** key.

Applying the New Policy

1) Before applying the new policy, you need to import into JDeveloper. Copy the new custom policy to your JDev store directory (either use the attached policy from this document or export your custom policy from EM). The location of the store could appear as follows:

   USER_HOME\AppData\Roaming\JDeveloper\system11.1.1.4.37.59.23\DefaultDomain\oracle\store\gmds\owsm\policies

2) Apply this new client policy to your service reference in your composite app via EM.

With this policy in place you can leverage the javax.xml.ws.security.auth.username inbound service property. If you are hardcoding, set the value without quotes. The value is set to jcooper; however, you can also extract the username from the payload of execData variable.

You do not have to import the policy to JDev, you can deploy the composite without a client-side policy, and then set the client policy through EM.

EM has a feature for setting the client-side policies that shows you compatible client-side policies based on the service you are calling.

References

http://download.oracle.com/docs/cd/E17904_01/web.1111/b32511/setup_config.htm#WSSEC3585

Configuring a Keystore if One Is Not Configured

1) Right-click your WebLogic domain and select **Security Provider Configuration**.
2) In the **Keystore Section**, select **Configure**.
3) Provide credentials.

Legal Notices

Oracle Primavera P6 EPPM BPM Configuration Guide