HP

Business Process Insight

For the Windows® Operating System

Software Version: 2.20

System Administration Guide

Document Release Date: October 2007
Software Release Date: October 2007

Legal Notices

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notices

Trademark Notices

Java™ is a US trademark of Sun Microsystems, Inc.

Microsoft® is a US registered trademark of Microsoft Corporation.

UNIX® is a registered trademark of The Open Group.

Windows® and MS Windows® are US registered trademarks of Microsoft Corporation.

Documentation Updates

This manual's title page contains the following identifying information:

- Software version number, which indicates the software version
- Document release date, which changes each time the document is updated
- Software release date, which indicates the release date of this version of the software

To check for recent updates, or to verify that you are using the most recent edition of a document, go to:

http://ovweb.external.hp.com/lpe/doc_serv/

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

Support

You can visit the HP Software Support web site at:

www.hp.com/go/hpsoftwaresupport

HP Software online support provides an efficient way to access interactive technical support tools. As a valued support customer, you can benefit by using the support site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HP support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract.

To find more information about access levels, go to:

www.hp.com/managementsoftware/access_level

To register for an HP Passport ID, go to:

www.managementsoftware.hp.com/passport-registration.html

Contents

1 Introduction

This guide provides information about the management tasks that you need to complete for the ongoing set up and maintenance of your Business Process Insight (HPBPI) solution.

This chapter describes the tools and procedures that are available for managing HPBPI. Some of these tools are provided with HPBPI and some are tools that are available with the third-party products that HPBPI uses, for example, database tools.

HPBPI comprises a number of different components, some of which can be distributed to other systems; for example, OVIS probes are installed on the system where OVIS is installed. However, there are some HPBPI components that always need to be located together and these are collectively referred to as the HPBPI Server components.

This guide describes the administration tasks that relate to the HPBPI components, the administration tools that are available and how to use these tools to maintain your HPBPI system.

For information about the architecture of the HPBPI system and how HPBPI integrates with other products, refer to the *Business Process Insight Reference Guide*. You do not need to understand the HPBPI architecture in order to manage the HPBPI system; however, it provides useful information that helps to understand how the components relate to each other.

The *Business Process Insight Reference Guide* also provides reference information; for example, information relating to the HPBPI database tables, which you might find helpful.

Tools Available to Manage HPBPI

The following are the tools that are available to you for managing your HPBPI system:

- HPBPI Server Administration Console

  This console enables you to modify parameters relating to the HPBPI Server components. The console, and details of the components that can be managed through it, are described in Chapter 2, HPBPI Component Administration.

- Log files

  Use the log files to identify what is happening in your HPBPI system. You can set the logging to report at different levels of detail when you are trying to identify specific problems. Refer to Chapter 2, HPBPI Component Administration for details of how to set the logging levels for HPBPI Server components through the Administration Console. Refer to Chapter 3, HPBPI Modeler Administration for details of the HPBPI Modeler log files.

- Notification Server Administration Console

  Use this console to add and delete users who want to receive email alerts and HP Operations Manager SLO and SLA notifications from HPBPI. You can also use this console to configure HP Operations Manager subscriptions and to configure scripts that the Notification Server runs when it receives a notification event; see Chapter 6, Notification Server Configuration.

- Engine Instance Cleaner

  Use the Business Impact Engine Instance Cleaner to remove active or completed Flow and Data instances from the database and the HPBPI system. Typically, you do not use the Engine Instance Cleaner to remove individual Flow or Data instances. To remove individual instances, use the Intervention Client. Both the Engine Instance Cleaner and the Intervention Client are described in Chapter 7, Intervention.

- Repository Explorer

  Use the Repository Explorer to provide you with a view of the Model Repository data. It enables you to:

  - view and print details of the definitions that are stored in the Repository.
  - export current and superseded versions of your business flows.
  - delete definitions.
  - view the history of your definitions.

- Intervention Client

  The Intervention Client enables you to intervene or gain access to instances of Flow and Data definitions. You might need to do this in cases where the flow instance is blocked for some reason and cannot progress, or where you have entered inaccurate start and complete conditions in your progression rules.

  The Intervention Client also enables you to change the status of Services defined for your business flows, delete instances and undeploy definitions. Refer to Chapter 7, Intervention for details of the Intervention Client

  If you want to delete multiple Flow and Data instances that are active or complete, on a regular basis, use the Engine Instance Cleaner.

- Integrity Checker

  The Installation Integrity Checker checks the HPBPI installation status and compares the current status of your HPBPI system to its status immediately following its installation. For example, it checks file permissions and the J2SE version. The Integrity Checker is described as one of the problem solving tools in the *Business Process Insight Problem Solving Guide*.

Common Management Tasks

The following are examples of some of the management tasks that you might need to complete and details of where you can find more information about them in this guide:

- Changing logging levels for components.

  If you are trying to gather information about a particular problem that is occurring in your HPBPI system, or you are trying to debug a Flow definition that you have created, you might need to modify the level at which the HPBPI components are logging. Chapter 2, HPBPI Component Administration describes how to change the logging levels for HPBPI components.

- Configuration of the HPBPI system.

  Use the Administration Console to modify the HPBPI configuration parameters as described in Chapter 2, HPBPI Component Administration.

- Managing the HPBPI Modeler.

  There are some management tasks associated with the HPBPI Modeler, for example, importing and exporting flow definitions. These tasks are described in Chapter 3, HPBPI Modeler Administration.

- Managing the Business Process Metric Definer.

  There are some management tasks associated with the Business Process Metric Definer, which are described in this guide, for example, exporting and importing metric definitions. These tasks are described in Chapter 4, HPBPI Metric Definer Administration.

- Customizing the Business Process Dashboard.

  You can modify the code for the Business Process Dashboard in order to integrate the Business Process Dashboard into your solution, for example, you might have a Web portal that you want to use to show the impact information for your business flows. Modifying the example Business Process Dashboard is described in *HP Business Process Insight Integration Training Guide - Customizing the Business Process Dashboard.*

- Managing business flow data in the Model Repository,

  You can export definitions from the Business Impact Engine using the Repository Explorer. You can also use the Repository Explorer to view and print your flow, data and event definitions. These definitions are provided in a forms-style interface using the Explorer, which makes them easier to read and print.

  The Repository Explorer is described in Chapter 5, Repository Explorer.

- Configuring user accounts to receive alerts for SLO and SLA violations, flow event notifications and metric threshold alerts.

  You use the Notification Server to configure user accounts to subscribe to event notifications and alerts. You can also subscribe user accounts to events, such that when a specified subscription event is received by the Notification Server, a script is run.

  Using the Notification Server Web Administration Console is described in Chapter 6, Notification Server Configuration.

- Intervening in flow progression.

  The following are examples of why you might need to intervene in a flow:

  — where you have Flows that are not progressing, perhaps because someone has manually completed a step in the flow, when a service is not available

  — there is an error in the start and complete conditions.

  Chapter 7, Intervention describes the tools that are available to intervene in the progression of a flow.

- Database housekeeping.

  If you are monitoring many HPBPI flow instances, the data in the HPBPI database can build up over time. Chapter 7, Intervention describes the tools and parameters that are available to manage the data in the HPBPI database.

- Configuring the authorization mechanism used by your HPBPI clients.

  You can configure the method that you use to authorize or authenticate your HPBPI clients and components using one of:

  — HP Select Access; see Chapter 8, Select Access Authorization

  — The Servlet Engine; see Chapter 9, Servlet Engine Authentication

- Creating a high availability solution for HPBPI on Microsoft Windows:

  You can install and configure HPBPI as part of a Microsoft Server Cluster in order to provide a high availability solution for HPBPI; see Chapter 10, High Availability Using Microsoft Clusters.

- Backup and recovery.

  For production systems, you need to make sure that you have a backup policy for your HPBPI system that fits in with your overall organizational backup policy. Chapter 11, Backup and Recovery describes how to backup HPBPI and alerts you to any specific requirements that might have an effect on your organizational backups.

- Problem solving.

  If you have specific problems with your system that you need to resolve, refer to the *Business Process Insight Problem Solving Guide*, as the problem and recovery might be documented. If you are unable to solve a particular problem, report it to HP, and include the information requested in the *Business Process Insight Installation Guide*.

- Architecture and integration

  If you want to find out about the architecture of HPBPI and how it integrates with other HP BTO Software and third-party components, refer to the *Business Process Insight Reference Guide*.

2 HPBPI Component Administration

This chapter describes the HPBPI Administration Console and the parameters that can be modified through the console. Later chapters in this guide, and chapters in the *Business Process Insight Reference Guide*, provide more information about the management tasks that you can complete using the Administration Console, specifically, describing the use of the console within the context of the task that you are trying to complete.

This chapter includes the following information:

- The parameters that can be modified using the Administrative Console.
- How to make sure that any modifications made through the Administrative Console are preserved.
- Accessing the License Manager software.

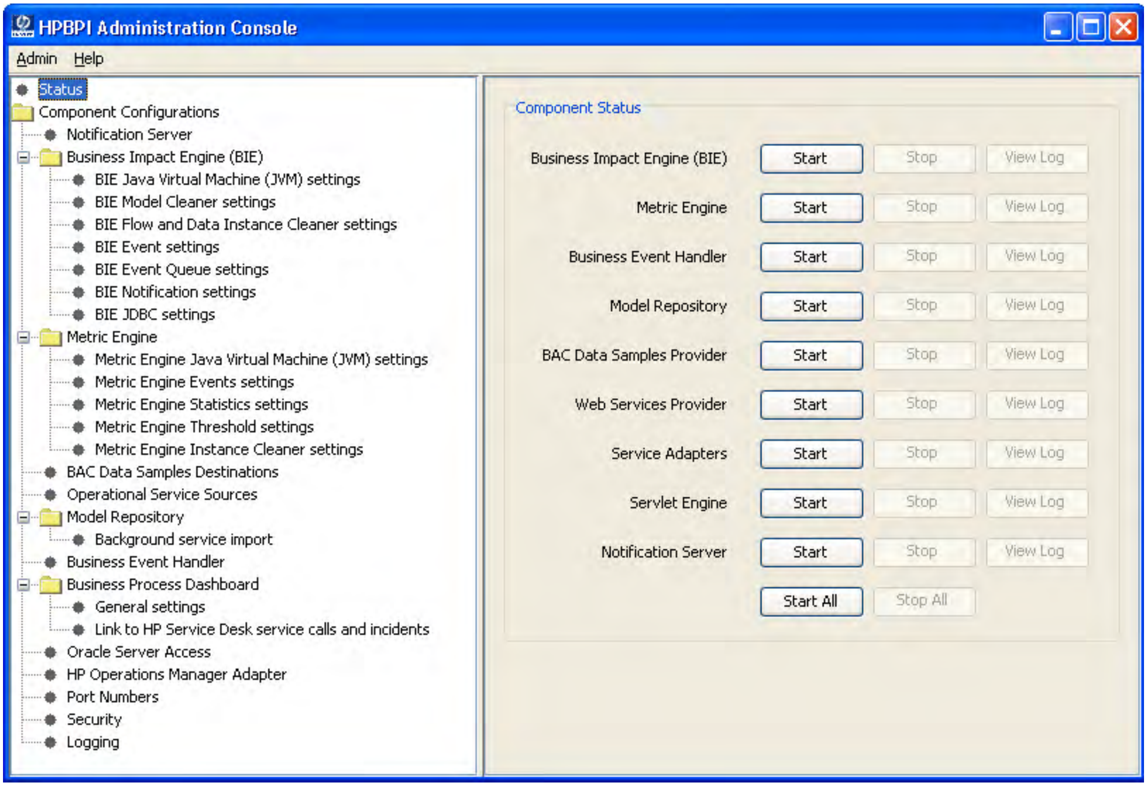
Not all the parameters described in this chapter need to be modified. What you need to change depends on your particular configuration and requirements. You are advised not to change anything unless you are recommended to in the documentation, or by your support representative.

Administration Console Description

You use the Administration Console to manage the HPBPI Server components. (The HPBPI components that make up the HPBPI Server are described in the *Business Process Insight Reference Guide*.) The Administration Console is a graphical management interface that displays a list of the parameters that can be modified. The Administration Console runs on both Microsoft Windows and HP-UX.

Figure 1 on page 20 shows an example of the Administration Console when it is first started on Windows following a full installation with Oracle as the configured database server.

Figure 1 Layout of Administration Console



- Status
- Component Configurations
  - Notification Server
  - Business Impact Engine (BIE)
    - BIE Java Virtual Machine (JVM) settings
    - BIE Model Cleaner settings
    - BIE Flow and Data Instance Cleaner settings
    - BIE Event settings
    - BIE Event Queue settings
    - BIE Notification settings
    - BIE JDBC settings
  - Metric Engine
    - Metric Engine Java Virtual Machine (JVM) settings
    - Metric Engine Events settings
    - Metric Engine Statistics settings
    - Metric Engine Threshold settings
    - Metric Engine Instance Cleaner settings
  - BAC Data Samples Destinations
  - Operational Service Sources
  - Model Repository
    - Background service import
  - Business Event Handler
  - Business Process Dashboard
    - General settings
    - Link to HP Service Desk service calls and incidents
  - Oracle Server Access
  - HP Operations Manager Adapter
  - Port Numbers
  - Security
  - Logging

Component Status

Business Impact Engine (BIE)	Start	Stop	View Log
Metric Engine	Start	Stop	View Log
Business Event Handler	Start	Stop	View Log
Model Repository	Start	Stop	View Log
BAC Data Samples Provider	Start	Stop	View Log
Web Services Provider	Start	Stop	View Log
Service Adapters	Start	Stop	View Log
Servlet Engine	Start	Stop	View Log
Notification Server	Start	Stop	View Log
	Start All	Stop All	

The Administration Console opens on the `Status` panel, which is where you start and stop the HPBPI Server components.

You navigate through the Administration Console configurations using the explorer-style menu in the left-hand pane. Select the option from the pane that you want to manage. For example, if you want to manage port numbers select the `Port Numbers` option.

When you select an option in the left-hand pane, the right hand pane is updated to contain the configuration details for the parameters that can be modified. The console presents only the parameters for the components that are installed on the system where it is running. For example, on HP-UX, the console displays information about the HP Operations Manager Adapter, port numbers and logging parameters as the HP Operations Manager Adapter is the only HPBPI component installed.

You do not necessarily need to complete all the tasks described in this chapter for your implementation. Many of the parameters that you can change are used to configure more than one HPBPI component, for example, changing the database password affects all components that access the database.

➤ Make sure that you follow the instructions in each section when making changes to the configuration for HPBPI Server components through the Administration Console. This is because the steps for changing the parameters can vary according to the parameter that is being modified.

Starting the Administration Console

You start the Administration Console as follows:

- On Windows, select:

  `Start|Programs|HP|HP Business Process Insight|Administration`

- On HP-UX, type the following command:

  ```
  hpbpi-install-dir/bin/biaadmin.sh
  ```

The Administration Console opens at the `Status` pane.

You can close the Administration Console at any time without applying outstanding changes and it does not impact the status of the system. You might need to click the `Reset` button to reset the page back to its state when you opened it. The next time that you start the Administration Console, it shows the current status of the HPBPI components.

Functions Provided by the Administrative Console

HPBPI comprises a number of individual components, each of which has its own configuration, or property, file that define the parameters that can be used to control how the component behaves. Examples of parameter values that can be configured are:

- retry delays
- port numbers
- logging levels

Many of the parameters in these property files do not need to be changed. Therefore, in order to simplify the management of the HPBPI system, only those parameters that do need to be modified have been made available through the HPBPI Administration Console.

If you do make changes directly to the properties in the property files, you will need to reapply the changes each time you reinstall HPBPI. This is because the files are overwritten when HPBPI is installed.

Specifically, the Administration Console provides access to the following HPBPI Server component areas:

- Server component status, which shows whether or not the HPBPI Server components are running and enables you to start and stop components; see section Status on page 25.
- Notification Server parameters; see section Component Configurations - Notification Server on page 38.
- Business Impact Engine parameters; see section Component Configurations - Business Impact Engine (BIE) on page 42.
- Metric Engine parameters; see section Component Configurations - Metric Engine on page 59.
- BAC Data Samples Destinations; see section Component Configuration - BAC Data Sample Destinations on page 73.

- Operational Service Sources (section Component Configuration - Operational Service Sources on page 78), which includes:
  - — OVIS interopeability parameters; see section OVIS on page 78.
  - — HP Operations Manager interopeability parameters; see section HP Operations Manager on page 82.
  - — HP Business Availability Center interoperability parameters; see section Other Service Sources on page 87.
  - — SOA Manager interopeability parameters; see section Other Service Sources on page 87.
- Model Repository parameters; see section Component Configurations - Model Repository on page 95.
- Business Event Handler parameters; see section Component Configurations - Business Event Handler on page 98.
- Business Process Dashboard parameters; see section Component Configurations - Business Process Dashboard on page 102.
- Microsoft SQL Server access; see section Component Configurations - MS SQL Server Access on page 110.
- Oracle Server access; see section Component Configurations - Oracle Server Access on page 113.
- HP Operations Manager Adapter settings; see section Component Configurations - HP Operations Manager Adapter on page 116.
- Port number settings; see section Component Configurations - Port Numbers on page 118.
- Security options; see section Component Configurations - Security on page 123.
- Logging parameter settings; see section Component Configurations - Logging on page 128.

The configuration parameters that you can modify for these component areas are described in the remaining sections of this chapter.

The Modeler is not an HPBPI Server component; its configuration is described in Chapter 3, HPBPI Modeler Administration.

In addition to individual component configurations, the Administration console provides access to the HP License Manager software; see section License Management on page 133.

Status

Use this option to start and stop the HPBPI Server components and also to view the component log files of these components.

The Status pane lists the HPBPI Server components that you can start and stop, and includes buttons labeled, `Start`, `Stop` and `View Log`. The components included on the Status pane, following a full installation on Windows, are:

- Business Impact Engine (BIE)

  The component responsible for managing the progress of flow instances, based on the business and operational events received from external sources.

- Metric Engine

  The component that analyzes and provides the statistical results from business process metrics and performance indicators. The business process metrics are created using the Business Process Metrics definer.

- Business Event Handler

  Used to accept and manage the business events between the Business Impact Engine and other business applications, for example, databases and files.

- Model Repository

  The design-time repository where the HPBPI Modeler stores its definitions. The Repository Server uses the information in the Model Repository when deploying the business flows.

- BAC Data Samples Provider

  The HPBPI component that is responsible for sending KPI data samples to the BAC system. The data samples are configured within the Metric Definer.

- Web Services Provider

  The HPBPI interface that exposes HPBPI Web Services to an external consumer, for example HP Operations Dashboard.

- Service Adapters

  This is the SOA Manager Adapter, which is used to manage the communication between the Business Impact Engine and HP SOA Manager; this option does not affect the status of the OVIS and HPSD integrations.

- Servlet Engine

  This is the Web Server component (Tomcat) that is installed as part of HPBPI. This component manages the server-side of all the HPBPI Web-based interfaces, for example: the Business Process Dashboard, Notification Server Web Administration Console and Intervention Client.

- Notification Server

  The component used to send email notifications of business events to configured recipients.

- HP Operations Manager Adapter

  Used to manage the communications between the Business Impact Engine and HP Operations Manager (either HP-UX or Windows, according to where the adapter is installed).

  This option appears automatically if you install the HP Operations Manager Adapter for Windows or HP-UX. If you want to use the adapter on a system where the HPBPI Server is installed, you need to add the adapter to the status screen using the option under HP Operations Manager Adapter as described in section Component Configurations - HP Operations Manager Adapter on page 116.

In addition, there are the `Start All` and `Stop All` buttons, which you can use to start and stop all HPBPI components.

➤ The status pane lists only those components that you have installed; for example, the HP Operations Manager Adapter option is available only following an installation of the adapter on HP-UX, or when you explicitly add the Adapter to the status pane using the HP Operations Manager Adapter option.

The section Starting and Stopping the HPBPI Server Components on page 27 describes the procedure for starting and stopping the HPBPI server components.

You also access the log files for the HPBPI Server components from the `Status` pane. This enables you to identify status and error information for that component. The level of detail shown in the log files is controlled by the logging levels that you set. Setting log levels is described in section Component Configurations - Logging on page 128.

If the log files are not displaying correctly on your system, you can change the viewer used to display the log files as described in section Log File Locations and Changing the Log Viewer on page 37.

Starting and Stopping the HPBPI Server Components

You can use the HPBPI Administration Console to start and stop the HPBPI Server components that you have installed on the system where the Administration Console is running.

The HPBPI Server components are also defined as Windows Services and as installed are started automatically when your machine starts. This chapter describes using the Administration Console to start and stop the HPBPI Server components; refer to the *Business Process Insight Installation Guide* for details of the Windows Services for the HPBPI Server components.

Starting the HPBPI Server Components Using the Administration Console

You start the HPBPI Server components from the `Status` panel. Click the `Start All` button to start all the components installed on the system. Alternatively, click the `Start` button adjacent to the individual HPBPI components that you want to start.

➤ There are some dependencies between components, for example, some components require the database to be started before they can start. You are therefore advised to use the `Start All` option to start the components unless instructed otherwise.

Stopping the HPBPI Server Components Using the Administration Console

Before stopping the HPBPI Server components, make sure that you have stopped the other HPBPI components that might rely on them, for example the:

- HPBPI Modeler
- Metric Definer
- Repository Explorer
- Business Process Dashboard
- Service Desk Process Insight Dashboard
- Notification Server Administration Console
- Intervention Client

You stop the HPBPI Server components in the same way that you started them, from the `Status` panel of the Administration Console. Select the `Stop All` option to stop the HPBPI Server, or stop the components individually.

There are some background components, not listed, which are also automatically started and stopped through the HPBPI Administration Console. These are components that need to be running before the listed HPBPI components can be started and stopped. These components are started when the Administration Console is started and, in the case of the Windows Services, when the individual Services are started.

Note that there is also an Administration Console Server, which is always running while the Administration Consoles is open on the machine. The Administration Console Server must not be running when you are installing or removing HPBPI components, you must therefore always close the Administration Console to also shut down the Administration Console Server before completing any installation tasks.

Starting and Stopping HPBPI Web Clients

HPBPI has the following Web clients:

- Metric Definer

  This is a Web application that you use to configure business process metrics and metric threshold definitions. Using the Metric Definer to configure business process metrics and thresholds is described in the *Business Process Insight Training Guide - Defining Business Metrics*. Opening the Metric Definer is described in section Accessing the HPBPI Metric Definer on page 30.

- Notification Server Administration Console

  This is a Web application that you use to add and configure users who want to receive email alerts from HPBPI. You also use it to configure HPOM to receive HP Operations Manager messages. Opening the Notification Server Web Administration Console is described in section Accessing the HPBPI Notification Server Administration Console on page 33. Using the Notification Server Web Administration Console to configure the Notification Server users is described in Chapter 6, Notification Server Configuration.

- The Business Process Dashboard

  This is a Web application where the Web pages for monitoring the business flows are displayed. The Dashboard can be on any system that has access to the HPBPI server. You can load the Business Process Dashboard pages using a Web browser client through a URL as described in section Accessing the Business Process Dashboard on page 35.

- Repository Explorer

  This is a Web application that you use to manage entries in the Model Repository, which holds the details of the business flows that you have created using the HPBPI Modeler. The Repository Explorer is described in Chapter 5, Repository Explorer. Opening the Repository Explorer is described in Accessing the Repository Explorer on page 32.

- The Intervention Client

  This is a Web application that enables you to make changes to definitions and instances that are active or completed in the Business Impact Engine. Starting and stopping the Intervention Client is described in section Accessing the Intervention Client on page 36.

There is also the HP Service Desk Process Insight Dashboard. Information about starting and stopping this Dashboard can be found in the *Business Process Insight Integration Training Guide - Monitoring Service Desk*.

The Web Server that manages the Web pages for the Web-based consoles, dashboard and clients is Tomcat. You start Tomcat using the `Servlet Engine` component on the HPBPI Administration Console. Be aware that starting and stopping the Servlet Engine has an impact on all the Web applications that use it.

You can also configure your own security options for accessing the HPBPI interfaces. You do this using either HP Select Access or the Servlet Engine (Tomcat). Refer to the following chapters for more information:

- Chapter 8, Select Access Authorization
- Chapter 9, Servlet Engine Authentication

Accessing the HPBPI Metric Definer

Before trying to start the HPBPI Metric Definer, make sure the `Servlet Engine` is started.

Opening the Metric Definer Web Interface

To start the HPBPI Metric Definer Web interface, complete the following steps:

1. Open a new Web browser window.
2. Type the following URL into the Web Browser:

   `http://`***hostname***`:44080/ovbpimetricdefiner`

   where:

   — *`hostname`* is the fully qualified domain name of the system where the HPBPI Server is installed and running. You can use `localhost` as the hostname, if you are starting the definer on the system where the HPBPI server components are installed and running.

   — `44080` is the port number for the Servlet Engine, identified by either the `ServletEngine HTTP` port number or the `ServletEngine HTTPS` port number. Use the port number configured for your system.

   You are prompted for a username and password.

3. Enter the username and password for the Metric Definer. Following a new installation, these are:

   — Username: `admin`

   — Password: `hpbpi`

   You can change the password used for the Metric Definer; see Chapter 8, Select Access Authorization or Chapter 9, Servlet Engine Authentication.

Closing the Metric Definer

You close the Metric Definer by closing the browser Window where the definer is running.

If you close the Metric Definer browser Window before completing a definition (clicking the `OK` button on the appropriate pane), the definition is lost. Before closing the Browser Window, make sure that any definition that you are creating appears in the list of definitions within the Metric Definer. If you are modifying a definition, allow the definer to respond (usually by taking you to the page where the revised definition is listed), before closing the Browser Window.

You are advised to use the Web Server authorization mechanisms to lock the Web Browser screen after a certain length of time; this increases the level of security for the Metric Definer pages. Refer to Chapter 9, Servlet Engine Authentication for details of the default authorization.

Accessing the Repository Explorer

Before trying to start the HPBPI Repository Explorer, make sure the `Servlet Engine` and the Model Repository components are started.

Opening the Repository Explorer Web Interface

To start the HPBPI Repository Explorer Web interface, complete the following steps:

1. Open a new Web browser window.
2. Type the following URL into the Web Browser:

   `http://`***hostname***`:44080/ovbpirepositoryexplorer`

   where:

   — *`hostname`* is the fully qualified domain name of the system where the HPBPI Server is installed and running. You can use `localhost` as the hostname, if you are starting the definer on the system where the HPBPI server components are installed and running.

   — `44080` is the port number for the Servlet Engine, identified by either the `ServletEngine HTTP` port number or the `ServletEngine HTTPS` port number. Use the port number configured for your system.

   You are prompted for a username and password.

3. Enter the username and password for the Repository Explorer. Following a new installation, these are:

   — Username: `admin`

   — Password: `hpbpi`

   You can change the password used for the Repository Explorer; see Chapter 8, Select Access Authorization or Chapter 9, Servlet Engine Authentication.

Closing the Repository Explorer

You close the Repository Explorer by closing the browser Window where the explorer is running. Closing the browser window has no effect on how the Repository Server operates, but if you have not completed an administration task, the task is lost when the browser window is closed.

You are advised to use the Web Server authorization mechanisms to lock the Web Browser screen after a certain length of time; this increases the level of security for the Repository Explorer pages. Refer to Chapter 9, Servlet Engine Authentication for details of the default authorization.

Accessing the HPBPI Notification Server Administration Console

Before trying to start the HPBPI Notification Server Administration Console, make sure the `Servlet Engine` is started.

Opening the Web Administration Console

To start the HPBPI Notification Server Web Administration complete the following steps:

1. Open a new Web browser window.
2. Type the following URL into the Web Browser:

   `http://`*`hostname`*`:44080/ovbpinotifyadmin`

   where:

   — *`hostname`* is the fully qualified domain name of the system where the HPBPI Server is installed and running. You can use `localhost` as the hostname, if you are starting the definer on the system where the HPBPI server components are installed and running.

   — `44080` is the port number for the Servlet Engine, identified by either the `ServletEngine HTTP` port number or the `ServletEngine HTTPS` port number. Use the port number configured for your system.

   You are prompted for a username and password.

3. Enter the username and password for the Web Administration Console, following a new installation, these are:

   — Username: `admin`

   — Password: `hpbpi`

   You can change the password used for the Notification Server Administration console; see Chapter 8, Select Access Authorization or Chapter 9, Servlet Engine Authentication.

Closing the Web Administration Console

You close the Notification Server Administration Console by closing the browser Window where the Console is running. Closing the browser window has no effect on how the Notification Server operates, but if you have not completed an administration task, the task is lost when the browser window is closed.

You are advised to use the Web Server authorization mechanisms to lock the Web Browser screen after a certain length of time; this increases the level of security for the Notification Server Administration pages. Refer to Chapter 9, Servlet Engine Authentication for details of the default authorization that is provided for the Notification Server.

Accessing the Business Process Dashboard

Before trying to start the HPBPI Business Process Dashboard, make sure that the `Servlet Engine` component is started from the HPBPI Administration Console.

Opening the Business Process Dashboard

To access the Business Process Dashboard, complete the following steps:

1. Open a new Web browser window.
2. Type the following URL into the Web Browser:

   `http://`*`hostname`*`:44080/hpbpi-bpd`

   where:

   — *`hostname`* is the fully qualified domain name of the system where the HPBPI Server is installed and running. You can use `localhost` as the hostname, if you are starting the definer on the system where the HPBPI server components are installed and running.

   — `44080` is the port number for the Servlet Engine, identified by either the `ServletEngine HTTP` port number or the `ServletEngine HTTPS` port number. Use the port number configured for your system.

   On a new installation you do not need to log in: a browser window opens on the `Home` page for the Business Process Dashboard. You can select to view Flow summaries or Service summaries from this Home page.

   You can add and change the password used for the Business Process Dashboard; see Chapter 8, Select Access Authorization or Chapter 9, Servlet Engine Authentication.

Closing the Business Process Dashboard

To close the Business Process Dashboard, close the browser Window where the Dashboard is running.

Accessing the Intervention Client

Before trying to start the HPBPI Intervention Client, make sure that the `Servlet Engine` component is started from the HPBPI Administration Console.

Opening the Intervention Client Web Application

You access the Intervention Client as follows:

1. Open a new Web browser window
2. Type the following URL:

   `http://`*`hostname`*`:44080/ovbpiintclient`

   where:

   — *`hostname`* is the fully qualified domain name of the system where the HPBPI Server is installed and running. You can use `localhost` as the hostname, if you are starting the definer on the system where the HPBPI server components are installed and running.

   — `44080` is the port number for the Servlet Engine, identified by either the `ServletEngine HTTP` port number or the `ServletEngine HTTPS` port number. Use the port number configured for your system.

   You are prompted for a username and password.

3. You are presented with a dialog to enter the login credentials for the Intervention Client. Enter a User Name and Password for the Client. On a new installation the User Name is `admin` and the Password is `hpbpi`.

   Section Security and the Intervention Client on page 223 describes how to change the username and password credentials for the Intervention Client.

Closing the Intervention Client Web Application

To close the Intervention Client Web application, close the browser Window where it is running.

If you want to increase the level of security for the Intervention Client, you can use the Web Server authorization mechanisms to lock the Web Browser screen after a certain length of time. This is advisable as someone could inadvertently use the Intervention Client to delete flows and flow instances within your HPBPI system. Section Security and the Intervention Client on page 223 provides details of the authorization that is provided for the Intervention Client by default.

Log File Locations and Changing the Log Viewer

All the HPBPI components have log files; however the log files presented through the `Status` pane relate only to the HPBPI Server components, which are managed using the Administration Console.

The Modeler is not an HPBPI server component and its log file is described in Chapter 3, HPBPI Modeler Administration. The structure of all HPBPI log files is described in the *Business Process Insight Problem Solving Guide*.

All the HPBPI log files, including the Modeler log files, are located in the following directory:

`HPBPI-install-dir\data\log`

Use the `View Log` button on the `Status` pane (adjacent to the Start and Stop buttons) to view the log files for the server components. By default the logs are displayed using the following commands:

- `HPBPI-install-dir/lbin/bia/logviewer.sh` (HP-UX)
- `notepad` (Windows)

Section Component Configurations - Logging on page 128 describes the parameters for modifying the viewer application that is used to display the log files for HPBPI Server components.

Component Configurations - Notification Server

Table 1 lists the parameters that you can modify to change the configuration of the Notification Server using the Administration Console.

Table 1 Notification Server Parameters

Descriptive Parameter Name	Description
SMTP hostname	The fully qualified domain name of the system where the SMTP email server is running. The HPBPI system needs access to a running SMTP email server in order to send impact alerts through email to registered users. Refer to Chapter 6, Notification Server Configuration for information about registering users to receive email alerts.
SMTP port number	The port number that the SMTP server is communicating on. The HPBPI system communicates with the SMTP email server through a specific port number, which is usually port 25.
Sender email address	The email address that you want to appear in the `Sender:` field on email notifications sent from the HPBPI system. This is the return email address that is sent out in response to impact alerts for all email messages. This is required for email messages sent out in response to impact alerts.
Retry interval to SMTP server (minutes)	The time (in minutes) that the Notification Server waits before attempting to send a notification to the SMTP server, if it fails on the first attempt. If the Notification Server is unable to send the message, it logs a warning message and temporarily stores the message in the database. The message is removed from the database when the notification is successfully delivered.

Table 1 Notification Server Parameters

Descriptive Parameter Name	Description
Maximum retries to SMTP server	Maximum number of retry attempts made by the Notification Server to send a notification message to the SMTP server. When the maximum number of retries is reached, the Notification Server deletes the message from the database, issues an error message in its log file and makes no further attempts to deliver the message to the email server.
Path to HP Operations Manager Opcmsg utility	The full path name for the `opcmsg` utility within HP Operations Manager. Used by the Notification Server to send notifications to HP Operations Manager.
Retry interval to HP Operations Manager (minutes)	The time (in minutes) that the Notification Server waits before attempting to send an HP Operations Manager message to HP Operations Manager, if it fails on the first attempt. If the Notification Server is unable to send the message, it logs a warning message and temporarily stores the message in the database. The warning message is removed from the database when the notification is successfully delivered.
Maximum retries to HP Operations Manager	Maximum number of retry attempts made by the Notification Server to send an HP Operations Manager message to HP Operations Manager. When the maximum number of retries is reached the Notification Server deletes the message from the database, issues an error message in its log file and makes no further attempts to deliver it.

Table 1 Notification Server Parameters

Descriptive Parameter Name	Description
Retry interval for script execution (minutes)	The time (in minutes) that the Notification Server waits before attempting to execute a script that you have created, if the script fails to execute on the first attempt. If the Notification Server is unable to execute a script, it logs a warning message in the Notification Server log file.
Maximum retries for script execution	Maximum number of retry attempts made by the Notification Server to execute a script that you have created. When the maximum number of retries is reached the Notification Server, issues an error message in its log file and makes no further attempts to execute the script.
Timeout for script execution (seconds)	The time (in seconds) that the Notification Server waits before aborting the execution of the script, after the script has started executing. Once aborted, the Notification Server waits for the retry interval before attempting to execute the script again.

Changing the Notification Server Parameters

To change the Notification Server parameters, complete the following steps on the Windows system, where the HPBPI Server is installed:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Select the `Component Configuration > Notification Server` option from the Navigator pane on the HPBPI Administration Console.
3. Enter the changes that you want to make to the Notification Server parameters in the right-hand pane as appropriate.
4. Click `Apply` to apply the changes to the HPBPI configuration.

5. Select the `Status` option to move to the panel where the HPBPI component status are shown.

6. Stop and restart all the HPBPI Server components using the `Stop All` and `Start All` buttons. You could stop and restart only the Notification Server and the Servlet Engine; however, you are recommended to use `Stop All` and `Start All` to make sure that components are started and stopped in the correct order.

The new Notification Server parameter values are now applied to your HPBPI system.

If you want to change the login password for the Notification Server Administration Console, refer to Chapter 9, Servlet Engine Authentication.

Component Configurations - Business Impact Engine (BIE)

The Business Impact Engine configuration parameter settings are divided into a number of logical sections. These sections appear hierarchically in the Administration Console under the `Engine` option as follows:

- BIE Java Virtual Machine (JVM) settings
- BIE Model Cleaner settings
- BIE Flow and Data Instance Cleaner settings
- BIE Event settings
- BIE Event Queue settings
- BIE Notification settings
- BIE JDBC settings

The parameters relating to these settings appear on the right-hand pane of the Administration Console when you select one of these options in the console's navigation tree. If the options are not visible, you need to expand the entries under `Engine` using the usual Explorer style navigation techniques.

BIE Java Virtual Machine (JVM) Settings

This section describes the parameters that you can modify through the `BIE Java Virtual Machine (JVM) settings` option.

Table 2 shows the Business Impact Engine parameters that enable you to modify the amount of memory heap available to the JVM for the system.

A heap is a storage management structure for tracking and allocating memory. In this case, the Java heap is used for allocating the Java objects used by the Engine.

Table 2 BIE Java Virtual Machine (JVM) Settings

Descriptive Parameter Name	Description
Initial size of the JVM heap (MB)	The initial size of the storage allocated by the JVM for Java objects.
Maximum size of the JVM heap (MB)	The maximum size of the storage allocated by the JVM for Java objects.

Modifying the BIE Java Virtual Machine (JVM) Settings

To change the `BIE Java Virtual Machine (JVM) Settings`, complete the following steps on the Windows system, where the HPBPI Server is installed:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Select the `Component Configuration > Business Impact Engine (BIE) > BIE Java Virtual Machine (JVM) settings` option from the Navigator pane on the HPBPI Administration Console.
3. Enter the changes that you want to make to the settings as appropriate in the right-hand pane.
4. Click `Apply` to apply the changes to the property files.
5. Select the `Status` option to move to the panel where the HPBPI component status are shown.
6. Stop and restart all the HPBPI Server components using the `Stop All` and `Start All` options.

The new Business Impact Engine settings are now applied to your HPBPI system.

BIE Model Cleaner Settings

This section describes the parameters that you can modify through the `BIE Model Cleaner Settings` option. The Model Cleaner is a thread that runs at intervals and deletes Flow and Data definitions that have been undeployed or superseded, and that no longer have any associated instances running.

Table 3 shows the Business Impact Engine parameters that enable you to remove undeployed Flow and Data definitions, and unused Service definitions, from the Business Impact Engine database.

Table 3 BIE Model Cleaner Settings

Descriptive Parameter Name	Description
Delay between checking for all Models (minutes)	The time interval (in minutes) that the Business Impact Engine waits before reading the latest list of definitions. This list is used as input for the `Delay between checking each model` parameter.
Delay between checking each Model (seconds)	The time (in seconds) that the Business Impact Engine waits between checking each entry in the list of definitions created as a result of the `Delay between checking for all models` parameter. It is used to establish those definitions that have been undeployed and which are therefore candidates to be deleted. Definitions are not deleted if: • the definition is deployed. • instances of the definition exist within the Business Impact Engine database. • the definition is referenced from another definition that is currently defined in the Business Impact Engine database; for example, a Service Definition is referenced from a Flow definition.
Disable cleanup of files for Models not in database?	A check box indicating whether or not to remove the Java source, Java Class and script files for flow and data models that have been cleaned from the Business Impact Engine. When checked, the source, classes and scripts are not removed.

Modifying the BIE Model Cleaner Settings

To change the BIE Model Cleaner Settings, complete the following steps on the Windows system, where the HPBPI Server is installed:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Select the Component Configuration > Business Impact Engine (BIE) > BIE Model Cleaner settings option from the Navigator pane on the HPBPI Administration Console.
3. Enter the changes that you want to make to the Engine settings as appropriate in the right-hand pane.
4. Click Apply to apply the changes to the property files.
5. Select the Status option to move to the panel where the HPBPI component status are shown.
6. Stop and restart all the HPBPI Server components using the Stop All and Start All options.

The new Business Impact Engine settings are now applied to your HPBPI system.

BIE Flow and Data Instance Cleaner Settings

This section describes the parameters for the BIE Flow and Data Instance Cleaner settings. For details of the Metric Engine Instance Cleaner settings, refer to section Metric Engine Instance Cleaner Settings on page 68.

The BIE Flow and Data Instance Cleaner settings section describes how often the instance cleaner thread is run and therefore how often the Completed and Active instances are deleted (or not deleted) from the database. Chapter 7, Intervention provides more information on the business impact Engine instance cleaner thread and the parameters that control it.

Instances are initially marked as being candidates to be deleted and the business impact Engine instance cleaner then runs at intervals that you specify to delete the instances that have been marked as candidates. More details of this process are provided in Engine Flow and Data Instance Cleaner Parameters on page 224.

You also have the option to delete complete instances as soon as they are complete using this setting. There is no requirement for the Engine Instance Cleaner to run in this case, as the instances are deleted as soon as they are complete.

For a new installation, the Engine Instance Cleaner thread does not run and instances are not deleted when they complete. Completed and Active instances can therefore accumulate in the database and this might have an impact on the performance of your HPBPI system.

If performance is a consideration, you need to modify this default to remove these instances and also consider archiving data in order that you can monitor historical data. Refer to Chapter 7, Intervention for details of how you can specify your own SQL to archive flow data before it is removed from the HPBPI database.

There are settings for how often the Engine Instance Cleaner marks instances as being candidates to be deleted, plus individual settings for controlling when the Completed and Active Instances are deleted. These are described in Table 4, Table 5 and Table 6.

➤ If you delete all Completed instances as soon as they are completed, data is no longer available to be viewed through the Business Process Dashboard for monitoring or archiving purposes.

Table 4 lists the Engine Flow and Data Instance Cleaner thread settings, which enable you to control how often the instance cleaner thread is executed.

Table 4 BIE Flow and Data Instance Cleaner Settings

Descriptive Parameter Name	Description
Mark instances for deletion:	The option for marking flow and data instances as candidates to be deleted by the Instance Cleaner thread at a specified interval (`Delete interval`). You have the following options: • `Never` - instances are never marked to be deleted. • `Periodically` - instances are marked to be deleted at the interval that you specify. • `Once a day` - instances are marked to be deleted once a day, at the time that you specify. • `Twice a day` - instances are marked to be deleted twice a day, at the times that you specify.
Delete interval for marked instances	The time interval used to control when the instance cleaner thread runs to delete any instances that have been marked for deletion.
Delete batch size	The maximum number of active and complete flow and data instances that the instance cleaner thread deletes in one `Delete interval`. There are four batches, one each for active and complete flow instances, and active and complete data instances.

Table 5 lists the settings that control when Completed flow and data instances are deleted from the Business Impact Engine database.

Table 5 Completed Flow and Data Instance Cleaner Settings

Descriptive Parameter Name	Description
Delete completed instances from the database	The option for deleting Completed instances from the database. You have the following options for deleting Completed instances: • `Never` - completed instances are never deleted and remain in the database • `As scheduled above` - completed instances are deleted from the database at an interval that you specify in the `Engine Flow and Data Instance Cleaner settings` options. • `Immediately on completion` - completed instances are deleted as soon as they reach the state `COMPLETED` and are not under the control of the settings for the Engine Instance Cleaner thread.
Age of the completed instances to be removed (minutes)	The age (in minutes) that the Completed instances must be before they can be deleted. This is the age of the instance at the time when the Engine Instance Cleaner runs. The following are quick references to common time durations expressed as minutes: 1 day = 1440 minutes 1 week (7 days) = 10080 minutes

Table 6 lists the settings that control how often Active flow and data instances are deleted from the Business Impact Engine database.

Table 6 Active Flow and Data Instances Cleaner Settings

Descriptive Parameter Name	Description
Delete active instances from the database?	This is an indication of whether or not you want to delete active instances from the database. • `Never` - active instances are never deleted and remain in the database • `As scheduled above` - active instances are deleted from the database at an interval that you specify in the `Engine Flow and Data Instance Cleaner settings` options.
Age of the active instances to be removed:	The age (in days, hours or minutes) that the Active instances must be before they can be deleted. This is the age of the instance at the time when the Engine Instance Cleaner runs. If you enter a value of 60 minutes, the next time that you access this pane, the value is presented in terms of hours and not minutes.

These parameters, and information about archiving instances before deleting them, are further described in section Engine Flow and Data Instance Cleaner Parameters on page 224.

Modifying the BIE Instance Cleaner Settings

To change the BIE Instance Cleaner Settings, complete the following steps on the Windows system where the HPBPI Server is installed:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Select the `Component Configuration > Business Impact Engine (BIE) > BIE Flow and Data Instance Cleaner settings` option from the Navigator pane on the HPBPI Administration Console.
3. Enter the changes that you want to make to the Engine Instance Cleaner settings as appropriate in the right-hand pane.

4. Click Apply to apply the changes to the property files.
5. Select the Status option to move to the panel where the HPBPI component status are shown.
6. Stop and restart all the HPBPI Server components using the Stop All and Start All options.

The new Business Impact Engine settings are now applied to your HPBPI system.

BIE Event Settings

This section describes the parameters that you can modify through the BIE Event settings option.

The parameter settings listed in Table 7 enable you to minimize the amount of time that the components spend in loops and possible database deadlock situations and provides settings for throughput rates.

Table 7 BIE Event Settings

Descriptive Parameter Name	**Description**
Maximum event generation number	A limit for the number of child events created from an initial incoming event. This is intended to interrupt potential infinite loops for subscriptions to data definitions.
Maximum retries for an event transaction deadlock	The maximum number of attempts by the Business Impact Engine to retry a database transaction before aborting it. It is possible for deadlocks to occur when the Engine and other applications are accessing the HPBPI database simultaneously. This parameter ensures that, in the case of a deadlock, the Engine aborts the transaction in order to break the deadlock. When this threshold is exceeded, the Business Impact Engine generates an error message and sends an exception to the Business Event Handler. The Business Event Handler then rolls back the event transaction and retries at a later time.

Table 7 BIE Event Settings

Descriptive Parameter Name	Description
Event sample count used for calculating rates	The number of Events used as a basis for calculating throughput rates. Set this parameter according to the incoming Event rate (to the Business Impact Engine). You want to use sufficient numbers of Events to give you a realistic throughput rate.
Decay period for rates when no events are received (seconds)	The time period after which the throughput rates are decremented, in stages, to zero. This is used for periods of inactivity, to maintain realistic results for throughput rates.

Modifying the BIE Event Settings

To change the BIE Event Settings, complete the following steps on the Windows system, where the HPBPI Server is installed:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Select the `Component Configuration > Business Impact Engine (BIE) > BIE Event settings` option from the Navigator pane on the HPBPI Administration Console.
3. Enter the changes that you want to make to the Engine settings as appropriate in the right-hand pane.
4. Click `Apply` to apply the changes to the HPBPI configuration.
5. Select the `Status` option to move to the panel where the HPBPI component status are shown.
6. Stop and restart all the HPBPI Server components using the `Stop All` and `Start All` options.

The new Business Impact Engine settings are now applied to your HPBPI system.

BIE Event Queue Settings

The following are the parameters that you can modify through the `BIE Event Queue settings` option.

These settings enable you to configure a business event queuing mechanism within the Business Impact Engine (BIE). When you select the option to use a queue for incoming events, all new business events are placed on a queue by the Business Event Handler. This means that the Business Event Handler does not need to wait for the BIE to process the business event and can immediately return to monitoring the event source and process the next incoming business event.

If you choose not to use a queue for business events, the Business Event Handler waits until the business event has been processed by the BIE; this includes waiting for all the necessary information from the event to be committed to the database.

The settings listed in Table 8 on page 52 control whether or not a queue is utilized for business events and the characteristics of the queuing mechanism.

Table 8 BIE Event Queue Settings

Descriptive Parameter Name	Description
Use a queue for incoming events?	Select this option if you want HPBPI to use a queue for incoming business events. Clear the check box if you do not want to use a queue. When cleared, all the remaining options for the `BIE Event Queue settings` are unavailable.
Maximum number of events in the queue	The maximum number of business events that placed on the queue. When the number of business events on the queued reaches this maximum, no further business events can be added to the queue. In this case, the Business Event Handler has to wait until the queue is reduced before it can add further events.

Table 8 BIE Event Queue Settings

Descriptive Parameter Name	Description
Save the queue to the database	Select this option if you want to write each business event to the database either as it arrives on the queue, or each time the HPBPI Server is shut down (where shut down is stopping the HPBPI Server from the Administration Console, or stopping the Windows Services.) Using this option enables you to rebuild the queue in the event of a system failure. If the business events are not written to the database, there is a risk that the events might be lost following a system failure. Be aware that there is a possible performance impact when you select the option to save each event as it arrives to the database. You need to determine whether the performance impact for writing individual events to the database is significant in your implementation.
Event queue cleaner interval (mins)	The time interval used to control the frequency that the event queue cleaner runs. Each time the event queue cleaner runs, it deletes, from the database, all business events that have been successfully processed by the BIE. This setting is available when the option to save the queue to the database for every event is selected.
Number of event handling threads	The number of individual processes available to process business events held on the event queue. A recommendation is to set this parameter to be twice the number of CPU cores that are available on your machine.

Modifying the BIE Event Queue Settings

To change the BIE Event Queue Settings, complete the following steps on the Windows system, where the HPBPI Server is installed:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Select the `Component Configuration > Business Impact Engine (BIE) > BIE Event Queue settings` option from the Navigator pane on the HPBPI Administration Console.
3. Enter the changes that you want to make to the Business Impact Engine settings as appropriate in the right-hand pane.
4. Click `Apply` to apply the changes to the HPBPI configuration.
5. Select the `Status` option to move to the panel where the HPBPI component status are shown.
6. Stop and restart all the HPBPI Server components using the `Stop All` and `Start All` options.

The new Business Impact Engine settings are now applied to your HPBPI system.

BIE Notification Settings

The following are the parameters that you can modify through the `BIE Notification settings` option.

If the Business Impact Engine initially fails to send an event notification to the Notification Server, it can try again. The settings listed in Table 9 on page 55 control at what impact level notifications are sent, and the number of retry attempts and the interval between the retries for the notifications.

Table 9 BIE Notification Settings

Descriptive Parameter Name	Description
Node impact notification service severity threshold	The minimum threshold for the severity for a node's service that can trigger an impact of impeded or blocked for the flow. When one (or more) node services reach the threshold specified a notification event is issued. Possible levels of severity are: Normal, Warning, Minor, Major and Critical. These map to other HP BTO Software products' levels of severity.
Maximum number of retries	The maximum number of attempts that the Business Impact Engine tries to send a notification event to the Notification Server, if the initial attempt fails. When the maximum number of retries is reached, the Business Impact Engine issues an error message in its log file and makes no further attempts to send the notification event.
Retry delay (seconds)	The delay (in seconds) between the Business Impact Engine's attempts to retry sending notification events to the Notification Server. Each time the Engine fails a retry attempt, it issues a warning message to the log file.

Modifying the BIE Notification Settings

To change the BIE Notification Settings, complete the following steps on the Windows system, where the HPBPI Server is installed:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Select the `Component Configuration > Business Impact Engine (BIE) > BIE Notification settings` option from the Navigator pane on the HPBPI Administration Console.
3. Enter the changes that you want to make to the Business Impact Engine settings as appropriate in the right-hand pane.
4. Click `Apply` to apply the changes to the HPBPI configuration.
5. Select the `Status` option to move to the panel where the HPBPI component status are shown.
6. Stop and restart all the HPBPI Server components using the `Stop All` and `Start All` options.

The new Business Impact Engine settings are now applied to your HPBPI system.

BIE JDBC Settings

The following are the parameters that you can modify through the `BIE JDBC settings` option.

Table 10 lists the settings that enable you to tailor the JDBC connection between the Business Impact Engine and the database.

Table 10 BIE JDBC Settings

Descriptive Parameter Name	Description
Maximum number of active JDBC Connections	The maximum number of JDBC connections that the Business Impact Engine can have active at any one time. If a connection does not become available in the required time, the connection is refused and the Engine is unable to commit the transaction. In this case, the Engine writes an error to its log file.
Maximum wait time for a JDBC Connection (seconds)	The maximum length of time (in seconds) that the Business Impact Engine waits for an available JDBC connection before reporting an error. If a connection does not become available in the required time, the connection is refused and the Engine is unable to commit the transaction. In this case, the Engine writes an error to its log file.
Maximum number of idle JDBC Connections[a]	The maximum number of JDBC connections that can be idle at any one time.
Maximum number of active JDBC Prepared Statements[1]	The maximum number of database Prepared Statements that can be active at any one time.
Maximum wait time for a JDBC Prepared Statement (seconds)	The maximum length of time (in seconds) that the Business Impact Engine waits for a Prepared Statement to be executed before reporting an error in the log file and aborting the transaction.
Maximum number of idle JDBC Prepared Statements[1]	The maximum number of Prepared Statements that can be idle at any one time.

[a] These connections are managed by Apache Commons DBCP (Database Connection Pool).

Modifying the BIE JDBC Settings

To change the BIE JDBC Settings, complete the following steps on the Windows system, where the HPBPI Server is installed:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Select the `Component Configuration > Business Impact Engine (BIE) > BIE JDBC settings` option from the Navigator pane on the HPBPI Administration Console.
3. Enter the changes that you want to make to the Engine settings as appropriate in the right-hand pane.
4. Click `Apply` to apply the changes to the HPBPI configuration.
5. Select the `Status` option to move to the panel where the HPBPI component status are shown.
6. Stop and restart all the HPBPI Server components using the `Stop All` and `Start All` options.

The new Business Impact Engine settings are now applied to your HPBPI system.

Component Configurations - Metric Engine

The Metric Engine configuration parameter settings are divided into a number of logical sections. These sections appear hierarchically in the Administration Console, under the `Metric Engine` option as follows:

- Metric Engine Java Virtual Machine (JVM) settings
- Metric Engine Events Settings
- Metric Engine Statistics settings
- Metric Engine Threshold settings
- Metric Engine Instance Cleaner settings

The parameters relating to these settings appear on the right-hand pane of the Administration Console when you select one of the Metric Engine options in the console's navigation tree. If the Metric Engine options are not visible in the navigation tree, expand the entries under Metric Engine using the usual Explorer-style navigation techniques.

Metric Engine Java Virtual Machine (JVM) Settings

This section describes the parameters that you can modify through the Metric Engine Java Virtual Machine (JVM) settings option. These are the Metric Engine parameters that enable you to modify the amount of memory heap available to the JVM and are listed in Table 11.

A heap is a storage management structure for tracking and allocating memory. In this case, the Java heap is used for allocating the Java objects used by the Metric Engine.

Table 11 Metric Engine Java Virtual Machine (JVM) Settings

Descriptive Parameter Name	Description
Initial size of the JVM heap (MB)	The initial size of the storage allocated by the JVM for Java objects.
Maximum size of the JVM heap (MB)	The maximum size of the storage allocated by the JVM for Java objects.

Modifying the Metric Engine Java Virtual Machine Settings

To change the Metric Engine Java Virtual Machine Settings, complete the following steps on the Windows system where the HPBPI Server is installed:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Select the `Component Configuration > Metric Engine > Metric Engine Java Virtual Machine (JVM) settings` option from the Navigator pane on the HPBPI Administration Console.
3. Enter the changes that you want to make to the Metric Engine Java Virtual Machine settings as appropriate in the right-hand pane.
4. Click `Apply` to apply the changes to the property files.
5. Select the `Status` option to move to the panel where the HPBPI component status are shown.
6. Stop and restart the Metric Engine component.

The new Metric Engine settings are now applied to your HPBPI system.

Metric Engine Events Settings

This section describes the parameters that you can modify for the `Metric Engine Events settings` option.

These settings provide control for the rate at which the Metric Engine processes Metric events and the maximum sizes of the resulting database transactions as Metric events are processed.

When both the Business Impact Engine and the Metric Engine are processing events simultaneously, the settings also provide some control to regulate the rate at which Metric events are generated. This control is based on the rate at which the Business Impact Engine processes its Business events. The settings enable the Metric Engine to process Metric events in line with the rate at which business events are processed in order to avoid a backlog of Metric events building up.

The values for the `Metric Engine Events settings` have been set to be suitable for most implementations. You should not change them unless you are advised to by your support representative.

The `Metric Engine Events settings` parameters are listed in Table 12.

Table 12 Metric Engine Events Settings

Descriptive Parameter Name	Description
The maximum number of pooled metric event processors (1 to19)	The number of parallel threads available to the Metric Engine for processing different Metric events.
The maximum number of metric events to process in each batch	The number of Metric events that can be processed in a single batch. This provides a balance between the Metric event processing rate and database transaction size. It also optimizes how the Metric events are processed across all available threads.
The metric events high-water mark	In cases where both the Business Impact Engine and the Metric Engine are active, this is the number of unprocessed Metric events that signal that the Metric Engine is not keeping up with the Business Impact Engine. When the high-water mark is exceeded, the Business Impact Engine temporarily stops processing Business events until the Metric Engine signals that it has cleared its backlog.
The metric events low-water mark	In cases where both the Business Impact Engine and the Metric Engine are active, this is the number of unprocessed Metric events that signal that the Metric Engine has cleared its backlog and is now keeping up with the Business Impact Engine. As a result, the Business Impact Engine can resume its processing of Business events.
The metric events idle wait time (seconds)	Each time the Metric Engine determines that there are no more Metric events to process, it waits for this time period before checking again. This prevents the Metric Engine continuously initiating searches that can return no results.

Modifying the Metric Engine Events Settings

To change the `Metric Engine Events settings` complete the following steps on the Windows system where the HPBPI Server is installed:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Select the `Component Configuration > Metric Engine > Metric Engine Events settings` option from the Navigator pane on the HPBPI Administration Console.
3. Enter the changes that you want to make to the `Metric Engine Events settings` as appropriate in the right-hand pane.
4. Click `Apply` to apply the changes to the property files.
5. Select the `Status` option to move to the panel where the HPBPI component status are shown.
6. Stop and restart the Metric Engine component.

The new settings are now applied to your HPBPI system.

Metric Engine Statistics Settings

The `Metric Engine Statistics settings` parameters control how the Metric Engine processes the staged statistical data and stores them in the user-configured statistical time intervals.

Staged statistical data are produced and stored as Metric events are processed. At intervals that you configure, the Metric Engine collects the staged statistics into the statistics table for the completed time intervals. Most of the parameters in Table 13 control the rate at which statistics are generated and these settings should not be modified unless you are instructed to do so by your support representative. The exception is the settings that controls how far back in time metric statistics are generated when the Metric Engine is restarted.

When HPBPI is restarted following a system shutdown, the metric statistics are calculated for all the metric definitions defined for flows. If you have shut your HPBPI system down for a significant amount of time (several days), then these statistic calculations can take considerable time, and impact the performance of your system. You can configure how far back in time to calculate these statistics following a restart using the `Maximum age of generated statistics on startup (days)` setting.

In Table 13, the following settings are used to control how the staged statistics are processed into the user-defined time intervals:

- `The maximum number of statistics consolidators (1 to 19)`
- `The maximum number of staged statistics' rows per transaction`
- `The maximum number of staging transactions each time`
- `The maximum number of statistics aggregators (1 to 19)`

In the description of the parameters, consolidation is the process of producing the data for specific time intervals directly from the staged data. If you have defined Business Metrics that have Groups, aggregation is also the process of taking all the individual groups' time interval data and producing the overall time interval data, where overall is the combined data for all the groups.

Table 13 lists the parameters that you can set to configure how often the Metric Engine polls the HPBPI database.

Table 13 Metric Engine Statistics Settings

Descriptive Parameter Name	Description
Statistics generation polling interval (seconds)	The time interval (in seconds) that the Statistics generator polls the metric tables in order to update the statistics that have been configured using the Metric Definer.
Maximum age of generated statistics on startup (days)	The time interval (in days) that the Metric Engine uses to calculate historical Metric statistics following a period of HPBPI shutdown.
The maximum number of statistics consolidators (1 to 19)	The number of parallel threads available to consolidate the staged statistical data, for different Metric events, into the user-defined statistical time intervals. This includes Metric events that have Groups defined.
The maximum number of staged statistics' rows per transaction	Controls the size of database transactions as the statistical data is being consolidated.

Table 13 Metric Engine Statistics Settings

Descriptive Parameter Name	Description
The maximum number of staging transactions each time	Controls the rate at which statistics are consolidated. This is a multiplier of `The maximum number of staged statistics' rows per transaction` and is applied each time the Metric Engine processes statistics.
The maximum number of statistics aggregators (1 to 19)	The number of parallel threads available to aggregate statistics that have Groups defined into their overall statistical time intervals.

Modifying the Metric Engine Statistics Settings

To change the `Metric Engine Statistics settings`, complete the following steps on the Windows system where the HPBPI Server is installed:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Select the `Component Configuration > Metric Engine > Metric Engine Statistics settings` option from the Navigator pane on the HPBPI Administration Console.
3. Enter the changes that you want to make to the `Metric Engine Statistics settings` as appropriate in the right-hand pane.
4. Click `Apply` to apply the changes to the property files.
5. Select the `Status` option to move to the panel where the HPBPI component status are shown.
6. Stop and restart the Metric Engine component.

The new settings are now applied to your HPBPI system.

Metric Engine Threshold Settings

This section describes the parameters that you can modify through the `Metric Engine Threshold settings` option.

The Metric Engine Threshold Settings are divided into two logical sections. These sections appear on the `Metric Engine Threshold settings` pane within the following categories:

- Metric Engine Threshold checking settings
- Metric Engine Threshold alert notification settings

Table 14 lists the Metric Engine Threshold settings that enable you to control how often the Metric Engine polls its data in order to identify threshold information for alarms, notifications and for statistical sampling.

Table 14 Metric Engine Threshold Checking Settings

Descriptive Parameter Name	Description
Threshold polling interval (seconds)	The time interval (in seconds) that the Threshold checker polls for metric values from the Metric database in order to determine whether an alarm needs to be generated.
Threshold minimum sample count, used for calculating relative thresholds	The minimum number of metric instances required for the calculation of the relative scope thresholds. If the number of metric instances available for sampling is less than this minimum figure, the threshold is not calculated. As a result, there are no notifications or alarms generated based on the threshold.
The maximum number of pooled threshold monitors (1 to 19)	The number of parallel threads available to the Metric Engine for processing different Business Metric Thresholds

Table 15 lists the Metric Engine Threshold settings that enable you to control the number of, and frequency at which, notifications (email, HP Operations Manager messages and script invocations) are sent out when metric threshold violations occur.

When a threshold is violated, the Metric Engine database records the detail of the violation. You can then be notified of these threshold violations in a number ways; for example, email; this is done using the Notification Server.

Where there is a significant failure in your business, for example, an application becomes unavailable, you can have the situation where thousands of thresholds are violated, or a small number of thresholds are violated many times. This can result in large numbers (or a storm) of notifications being generated. In the case of email, this can quickly fill up your email Inbox and it is possible that your email Server is unable to keep up with the rate at which the notifications are generated. This in turn creates a backlog of notifications and can severely impact the performance of the HPBPI system. In the case of HP Operations Manager messages or scripts, the backlog created depends on how your system is configured to receive the notifications.

Use the parameters in Table 15 to control the frequency and numbers of notifications that you receive. This reduces the possibility of these notification storms occurring, and still notifies users of the metric threshold violations.

The following is an example of the effect of setting these parameters:

> The parameter values set when you install HPBPI for the first time are:
>
> — 300 seconds for the polling interval
>
> — a maximum of 10 notifications for each threshold violated
>
> This means that you receive a maximum of 10 notifications for each threshold alert, every 5 minutes (300 seconds); this is for each threshold. If there are 20 thresholds violated, there are potentially 200 (10 x 20) notifications in each 5 minute polling interval.

If you know the number of threshold violations your business is likely to generate, you can set these parameters such that:

- you receive all notifications for the occasional problems.
- you have the maximum threshold set such that when there is a system failure, notification storms are prevented and you receive a summary instead.

This also has the effect of reducing the impact of notification storms on HPBPI performance.

Table 15 Metric Engine Threshold Alert Notification Settings

Descriptive Parameter Name	Description
Threshold alert notification polling interval (seconds)	The time interval (in seconds) that the Metric Engine waits between the time that is sends the last notification message at the end of one polling period, to the start of the next polling period.
Maximum number of individual alert notifications per threshold per polling interval	The maximum number of alert notifications sent to the Notification Server for each business metric threshold violation. The actual number of threshold alerts generated might be significantly more than this maximum; however, if you are using the default Velocity email templates, each email notification also has summary information, which includes all the threshold alerts generated during the polling interval.

Modifying the Metric Engine Threshold Settings

To change the Metric Engine Threshold Settings, complete the following steps on the Windows system where the HPBPI Server is installed:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Select the `Component Configuration > Metric Engine > Metric Engine Threshold settings` option from the Navigator pane on the HPBPI Administration Console.
3. Enter the changes that you want to make to the Metric Engine Threshold settings as appropriate in the right-hand pane.

4. Click `Apply` to apply the changes to the property files.
5. Select the `Status` option to move to the panel where the HPBPI component status are shown.
6. Stop and restart the Metric Engine component.

The new Metric Engine settings are now applied to your HPBPI system.

Metric Engine Instance Cleaner Settings

This section describes the parameters that you can modify through the Metric Engine Instance Cleaner settings option.

The Metric Engine instance cleaner settings section controls how often the Metric Engine instance thread is run and therefore how often Active and Completed metric instances, Metric alarm instances and statistics are deleted.

If you set a low Collection interval for your business process metric definitions, the metric data can accumulate very quickly and consume a considerable amount of disk space. In addition, if you shutdown your HPBPI system and restart it after a significant amount of time, the metric statistics are calculated for the period of the shutdown as soon as the HPBPI Server is restarted. As an example, if you set a Collection interval of five minutes and shut down your HPBPI system for a week. When you restart your HPBPI Server, the Metric Engine calculates all the metric statistics for each five-minute period since the last time the metrics were calculated. This can result in a significant amount of calculation time, and a significant amount of disk space for the results.

If these results are immediately deleted as a result of the instance cleaner settings, this can also have an impact on the overall operational performance of your machine.

The Collection interval is a parameter that you set when you define a business process metric in the Metric Definer and is described in the *Business Process Insight Online Help for the Metric Definer.*

The configuration parameters are divided into a number of logical sections. These sections appear in the Metric Engine Instance Cleaner settings pane within the following categories:

- Metric Engine Instance Cleaner settings
- Active Metric Instances settings
- Completed Metric Instances settings
- Metric Statistics settings
- Metric Alarm Instances settings

The parameters relating to these settings are described in the following sections.

Metric Engine Instance Cleaner Settings

Table 16 lists the settings that controls how often the Metric Engine instance cleaner thread is executed. This interval impacts all the other sections on the Metric Instance Cleaner pane.

Table 16 Metric Engine Instance Cleaner Settings

Descriptive Parameter Name	Description
Instance cleaner execution interval (minutes)	The time period (in minutes) that you want the Metric instance cleaner thread to run. When it runs, it deletes all eligible Active metric instances, Completed metric instances, Metric Alarm instances and Statistics.

Active Metric Instances Settings

Table 17 lists the settings that enable you to control the amount of Active metrics instance data held in the Metrics database.

Table 17 Active Metric Instances Settings

Descriptive Parameter Name	Description
Delete active metric instances from the database?	Select this check box if you want to delete Active metric instances from the Metrics database. Clear the check box if you do not want to delete the Active metric instances.
Age of active metric instances to be removed (days)	This option is available when you select the check box for deleting Active metric instances. Enter the age (in days) of the Active metric instances that you want to be deleted when the Metric Engine instance cleaner thread is run.

Completed Metric Instances Settings

Table 18 lists the settings that enables you to control the amount of Completed metrics instance data held in the Metrics database.

Table 18 Completed Metric Instances Settings

Descriptive Parameter Name	Description
Delete completed metric instances from the database?	Select this check box if you want to delete Completed metric instances from the Metrics database. Clear the check box if you do not want to delete the Completed metric instances.
Age of completed metric instances to be removed (days)	This option is available when you select the check box for deleting Completed metric instances. Enter the age (in days) of the Completed metric instances that you want to be deleted when the Metric Engine instance cleaner thread is run

Metric Statistics Settings

Table 19 lists the settings that enable you to control the amount of statistical data held in the Metrics database.

Table 19 Metric Statistics Settings

Descriptive Parameter Name	Description
Delete metric statistics from the database?	Select this check box if you want to delete statistics from the Metrics database. Clear the check box if you do not want to delete the statistics data.
Age of metric statistics to be removed (days)	This option is available when you select the check box for deleting statistics. Enter the age (in days) of the statistics data that you want to be deleted when the Metric Engine instance cleaner thread is run

Metric Alarm Instances Settings

Table 20 lists the settings that enables you to control the amount of Metric Alarm instance data held in the Metrics database.

Table 20 Metric Alarm Instances Settings

Descriptive Parameter Name	Description
Delete metric alarm instances from the database?	Select this check box if you want to delete Metric Alarm instances from the Metrics database. Clear the check box if you do not want to delete the Metric Alarm instances.
Age of metric alarm instances to be removed (days)	This option is available when you select the check box for deleting Completed metric instances. Enter the age (in days) of the Metric Alarm instances that you want to be deleted when the Metric Engine instance cleaner thread is run

Modifying the Metric Engine Instance Cleaner Settings

To change the Metric Engine Instance Cleaner Settings, complete the following steps on the Windows system where the HPBPI Server is installed:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Select the `Component Configuration > Metric Engine > Metric Engine Instance Cleaner settings` option from the Navigator pane on the HPBPI Administration Console.
3. Enter the changes that you want to make to the Metric Engine Instance Cleaner settings as appropriate in the right-hand pane.
4. Click `Apply` to apply the changes to the property files.
5. Select the `Status` option to move to the panel where the HPBPI component status are shown.
6. Stop and restart the Metric Engine component.

The new Metric Engine settings are now applied to your HPBPI system.

Component Configuration - BAC Data Sample Destinations

The BAC Data Sample Destinations options enable you to configure the details of the BAC system, or systems, where you want HPBPI data samples to be sent.

You need to configure a Data Sample Destination if you want to be able to have HPBPI flow and business metric data displayed within the BAC Dashboard.

You can have more than one destination for your data samples; for example, there might be a BAC Core Server and Dashboard both in Europe and in the United States where you want to display HPBPI data. In all cases, HPBPI sends the data samples to all configured destinations that it can contact.

The BAC Data Sample Destinations page is divided into the following logical sections:

- BAC Data Sample Destinations; see section Data Sample Destinations on page 74.
- Web Proxy; see section Web Proxy on page 77.

Data Sample Destinations

To configure a Data Sample Destination, make sure that the `BAC Data Sample Destinations` option is selected in the Administration Console navigator pane.

To create a new Destination, you need to click `Add` in the right-hand pane. The `Business Availability Center Data Samples Destination Properties` dialog is displayed.

The properties for the Data Sample Destinations are described in Table 21.

Table 21 BAC Data Sample Destinations Parameters

Descriptive Parameter Name	**Description**
Destination Name	Unique name that identifies the Destination when displayed in the HPBPI interfaces. The name can be up to 40 characters. You are advised to include some location information in the name, to make it easier to distinguish between Destinations and their location when they are displayed in HPBPI.
Description	An optional description of the Destination.
BAC Core Server hostname	The fully qualified DNS name of the machine where the BAC Core Server that is hosting the Destination is installed. This can also be an IP address.
BAC Core Server HTTP port	The port number used by the Web Server on the BAC Core Server. This is usually port 80.
Data samples send interval (seconds)	The time interval (in seconds) that you want HPBPI to use to send data samples to the named BAC Core Server.

Table 21 BAC Data Sample Destinations Parameters

Descriptive Parameter Name	Description
Connection timeout (seconds)	The period of time (in seconds) allowed for the HPBPI Server to request a connection with the BAC Core Server. If this parameter is set to zero (0), the time allowed for requesting a connection is unlimited, that is, no timeout occurs. If a timeout does occur, HPBPI attempts to resend the data sample after waiting for the interval configured for `Data samples send interval`.
Guaranteed retention period (minutes)[a]	The period of time (in minutes) that a Data Sample is guaranteed to be made available for transfer to the BAC Core Server and before it is discarded. In cases where it is not possible to make a connection to the BAC Core Server, or where the BAC Core Server's buffer tables are full, HPBPI retains copies of each Data Sample for the period specified by this parameter.
Override Data Samples Provider's log level?	Select this checkbox of you want to change the log level setting for this Data Sample Destination.
Log level for this destination	The level of logging required for this Data Sample Destination. It can be one of `Finer`, `Fine` or `Info`; see section Component Configurations - Logging on page 128 for a description of the log levels.

[a] Note that for versions of the Business Availability Center prior to version 7.5, that this parameter has no effect. This is because versions of the Business Availability Center prior to version 7.5 are able to display only the latest value of the Data Sample.

When you have added the values for you Data Sample Destinations, click OK to close the properties dialog.

You can control whether a particular Data Sample Destination is enabled for data transfer, or not, by selecting or clearing the checkbox under the `Enabled?` option on the `BAC Data Sample Destination` dialog.

Modifying the BAC Data Sample Destinations Settings

To change the `BAC Data Sample Destinations` settings, complete the following steps on the Windows system, where the HPBPI Server is installed:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Select `Component Configuration > BAC Data Sample Destinations` option from the Navigator pane on the HPBPI Administration Console.
3. Select the Data Sample Destination that you want to modify.
4. Select the `Modify` button.
5. Enter the changes that you want to make to the settings as appropriate in the dialog presented.
6. Click `OK` to dismiss the dialog.
7. Click `Apply` to apply the changes to the HPBPI configuration.
8. Select the `Status` option to move to the panel where the HPBPI component status are shown.
9. Stop and restart all the HPBPI Server components using the `Stop All` and `Start All` options.

The new settings are now applied to your HPBPI system.

Web Proxy

Use this option to set parameters for a Web Proxy that is common to all your BAC Data Sample Destinations.

The properties for Web Proxy are described in Table 22.

Table 22 Web Proxy

Descriptive Parameter Name	Description
Enable proxy?	Select this checkbox to enable your proxy. Clear the checkbox to disable the proxy.
Proxy host	Name of the host used to be for HTTP access.
Proxy port	Port number used for hosting HTTP access to the BAC Data Sample Destinations.

Modifying the Web Proxy Settings

To change the `Web proxy` settings, complete the following steps on the Windows system, where the HPBPI Server is installed:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Select `Component Configuration > BAC Data Sample Destinations` option from the Navigator pane on the HPBPI Administration Console.
3. Enter the changes that you want to make to the settings as appropriate in the right-hand pane.
4. Click `OK` to dismiss the dialog.
5. Click `Apply` to apply the changes to the HPBPI configuration.
6. Select the `Status` option to move to the panel where the HPBPI component status are shown.
7. Stop and restart all the HPBPI Server components using the `Stop All` and `Start All` options.

The new settings are now applied to your HPBPI system.

Component Configuration - Operational Service Sources

The Operational Service Options enable you to configure the sources of operational service events that you want to receive from other HP BTO Software. These components include:

- HP OpenView Internet Services (OVIS)
- HP Operations Manager for Windows (HPOM)
- HP Service Oriented Architecture (SOA) Manager
- HP Business Availability Center

The Operational Service Sources configuration page is divided into the following logical sections:

- Operational Service Sources; these sources can be any one of:
  - OVIS; see section OVIS on page 78.
  - HP Operations Manager; see section HP Operations Manager on page 82.
  - Instances of SOA Manager Service or Business Availability Center sources that you add; see section Other Service Sources on page 87.
- Proxy for Web service adapters; see section Proxy for Web Service Adapters on page 93.

OVIS

`OVIS` describes one of the pre-named service sources, which is available for you to select and modify in the table of Operational Service Sources. To enable HPBPI to be able to import operational services from OVIS, you need to configure and then enable this Service Source.

To configure interoperability with OVIS as a Service Source, make sure that OVIS is selected and click the Modify button below the list of service sources. You can then configure the details of the integration.

➤ The configuration options described in this section do not apply to the configuration of the HPBPI custom probes for OVIS; they apply only to HPBPI polling OVIS for details of operational services and impact events.

If you choose to import services, you must also configure details of the OVIS connection to enable HPBPI to communicate with OVIS.

To enable the OVIS Service Source, select the Enabled? checkbox next to the OVIS option. If the check box is not selected, operational services cannot be imported into HPBPI and HPBPI is not able to generate service impact events.

The OVIS configuration parameter settings are divided into a number of logical sections, which are described in the following sections.

Event Poll Intervals

Table 23 lists the setting that enable you to configure the time interval that HPBPI uses to for polling OVIS for alarms, service impact events and violations.

Table 23 OVIS Event Poll Intervals

Descriptive Parameter Name	Description
Alarm polling interval (minutes)	The time interval (in minutes) for the poll interval where the OVIS Event Receiver polls for alarm events from OpenView Internet Services.
SLO violation polling interval (minutes)	The time interval (in minutes) for the poll interval where the OVIS Event Receiver polls for SLO violation events from OpenView Internet Services.
SLA violation polling interval (minutes)	The time interval (in minutes) for the poll interval where the OVIS Event Receiver polls for SLA violation events from OpenView Internet Services.

OVIS Database Settings

These settings enable you to specify the details for the OVIS database in order that HPBPI can access alerts, SLA and SLO violations; the Business Impact Engine polls the OVIS database and the Business Process Dashboard queries the OVIS database.

You need to find out these details from the person responsible for managing OVIS and its database.

Table 24 lists the parameters that enable you to specify the OVIS database details.

Table 24 OVIS Database Settings

Descriptive Parameter Name	Description
Database type	Either MS SQL Server or Oracle.
Database server hostname	The fully qualified host name of the system where the Reporter database used by OVIS is installed.
Database server port number	The port number for the database used by OVIS.
Database user name	The username that HPBPI needs to specify to access the OVIS tables in the database.
Password	The password that HPBPI needs to specify to access the OVIS tables in the database.
Database name[a] (MS SQL only)	The name of the database for the OVIS data within the Microsoft SQL Server database.
Database SID[b] (Oracle only)	The database instance for the OVIS data within the Oracle database.

[a] Available only when OVIS configured for a Microsoft SQL Server database

[b] Available only when OVIS configured for an Oracle Server database

Service Import Settings

This section describes the parameters that you can modify through the `Service Import settings` option.

The `Service Import settings` must be applicable to the OVIS Server that uses the database described in section OVIS Database Settings on page 80. You cannot specify Service Import settings for one OVIS Server and have `OVIS Database settings` that are applicable to another OVIS Server.

Table 25 lists the setting that enable you to configure the location of the OVIS system from where you want to import service definitions when you are designing business flows using the HPBPI Modeler.

Table 25 OVIS Service Import Settings

Descriptive Parameter Name	Description
OVIS management server hostname	The fully qualified host name of the system where the OpenView Internet Services Management Server is installed.
OVIS HTTP server port number	The port number for the Internet Information Services (IIS) Web Server used by OVIS.

Modifying the OVIS Settings

To change the `OVIS` settings, complete the following steps on the Windows system, where the HPBPI Server is installed:

1. Start the Administration Console as described in section Administration Console Description on page 20.
   - Select the `Component Configuration > Operational Service Sources.`
   - Select `OVIS` option from the right-hand pane.
   - Click the `Modify` button.
2. Enter the changes that you want to make to the settings, as appropriate, in the `OVIS Source Properties` dialog.
3. Click `Apply` to apply the changes to the HPBPI configuration.

4. Select the `Status` option to move to the panel where the HPBPI component status are shown.

5. Stop and restart all the HPBPI Server components using the `Stop All` and `Start All` options.

The new settings are now applied to your HPBPI system.

HP Operations Manager

`HP Operations Manager` is a pre-named service source, which is available for you to select and modify in the table of Operational Service Sources. To enable HPBPI to be able to import operational services from HP Operations Manager, you need to install the HP Operations Manager Adapter on a system where HP Operations Manager is located and then you need to configure HP Operations Manager interoperability.

Specifically, this option enables you to specify whether or not you want HPBPI to interoperate with HP Operations Manager Service Navigator component (on HP-UX), or HP Operations Manager for Windows.

An HPBPI Server can connect to one HP Operations Manager Server at any one time, this can be on HP-UX (Service Navigator component), or Windows (HP Operations Manager for Windows). An HPBPI server cannot connect to both HP-UX and Windows at the same time.

To configure interoperability with HP Operations Manager:

1. Select `HP Operations Manager`

2. Click the `Modify` button below the list of service sources to configure the details of the integration.

3. Select the `Enabled?` checkbox next to the `HP Operations Manager` option to enable the interoperability. If this check box is not selected, operational services are not synchronized with those defined in the Modeler, and you are not able to deploy a Flow definition that references an HP Operations Manager Service definition.

The HP Operations Manager configuration parameter settings are described in Table 26.

Table 26 lists the HP Operations Manager parameters.

Table 26 HP Operations Manager

Descriptive Parameter Name	Description
HP Operations Adapter hostname	The fully qualified domain name for the system where the HP Operations Manager Adapter is installed and running.
Clients' maximum wait time (seconds)	The maximum time (in seconds) that a client waits for a response from the adapter before assuming that an error has occurred and causing the connection to time out. An error message is written to its log file when this maximum time is exceeded.
Clients' reconnect interval (seconds)	The time (in seconds) that a client waits before retrying a connection to the adapter following a previously failed connection. A warning message is written to its log file each time the client fails to reconnect.

If you want to enable or disable HP Operations Manager synchronization; see section Enabling or Disabling HP Operations Manager Settings for Interoperability on page 84.

If you want to modify the Adapter hostname, refer to section Modifying HP Operations Manager Adapter Hostname on page 84.

If you want to modify the client maximum wait time and reconnect intervals, refer to section Modifying the Client Maximum Wait and Reconnect Intervals on page 86.

Enabling or Disabling HP Operations Manager Settings for Interoperability

This setting enables you to configure whether or not you want to enable or disable synchronization with the HP Operations Manager Adapter services.

To change modify this setting, complete the following steps:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Select the `Component Configuration > Operational Service Sources` option from the Navigator pane on the HPBPI Administration Console.
3. Select or clear the checkbox adjacent to the `HP Operations Manager` entry in the `Service Source` table as appropriate.
4. Click `Apply` to apply the changes to the HPBPI configuration.
5. Select the `Status` option to move to the panel where the HPBPI component status are shown.
6. Stop and restart all the HPBPI Server components using the `Stop All` and `Start All` options.

The new setting is now applied to your HPBPI system.

Modifying HP Operations Manager Adapter Hostname

This is the hostname of the system where the HP Operations Manager Adapter is installed. As the adapter has to be installed on the same system as the HP Operations Manager Server, this is also the hostname of the system where either the HP Operations Manager Service Navigator component or HP Operations Manager for Windows are installed (according to your configuration). If you want to change this hostname for any reason, complete the following steps on the Windows system, where the HPBPI Server is installed:

1. Make sure that the HP Operations Manager Adapter is installed and running on the new system.
2. Make sure that the HP Operations Manager services required by your HPBPI flows are available on the new server.

3. Use the HPBPI Modeler to undeploy all the parent flows that link to the affected HP Operations Manager services.

   You do this by individually selecting the Flow Definitions from the navigator pane in the HPBPI Modeler and selecting `Undeploy` from the `File` menu. Chapter 3, HPBPI Modeler Administration discusses undeployment in more detail.

4. Start the Administration Console on the HPBPI Server system as described in section Administration Console Description on page 20.

5. Using the Administration Console, stop the Business Impact Engine and the Model Repository using the `Stop` button on the `Status` option.

6. Start the Administration Console as described in section Administration Console Description on page 20.

   - Select the `Component Configuration > Operational Service Sources.`
   - Select `HP Operations Manager` option from the right-hand pane.
   - Click the `Modify` button.

7. Enter the new name for the `HP Operations Manager Adaptor hostname` in the `HP Operations Manager Source Properties` dialog.

   This should be the fully qualified hostname for the system where either Service Navigator is installed, or HP Operations Manager for Windows is installed.

8. Select `Apply` and the changes are made to the configuration files.

9. Select the `Status` option where the component states are listed.

10. Stop and all the HPBPI Server components using the `Stop All` button.

    The HPBPI Modeler issues a warning stating that it has lost its communication link with the HPBPI Server and prompting you to close the modeler. All unsaved changes are saved when the HPBPI Modeler shuts down.

11. Restart the all the HPBPI Server components using the `Start All` button.

12. Restart the HPBPI Modeler so it re-establishes its connection to the Model Repository.

13. Use the File|Link to HP Operations Manager Services... menu option in the Modeler to re link the HP Operations Manager services on the new HP Operations Manager Server.

14. Use the HPBPI Modeler to redeploy the previously undeployed Flow and Service definitions to the new HP Operations Manager system.

    This is in order that the new services are registered with the HP Operations Manager system.

You have now completed the configuration for the hostname for the HP Operations Manager Adapter system.

Modifying the Client Maximum Wait and Reconnect Intervals

To change these settings, complete the following steps on the Windows system, where the HPBPI Server is installed:

1. Start the Administration Console as described in section Administration Console Description on page 20.

2. Start the Administration Console as described in section Administration Console Description on page 20.

    - Select the Component Configuration > Operational Service Sources.
    - Select HP Operations Manager option from the right-hand pane.
    - Click the Modify button.

3. Enter the changes that you want to make to the settings as appropriate in the HP Operations Manager Source Properties dialog.

4. Click Apply to apply the changes to the HPBPI configuration.

5. Select the Status option to move to the panel where the HPBPI component status are shown.

6. Stop and restart all the HPBPI Server components using the Stop All and Start All options.

The new settings are now applied to your HPBPI system.

Other Service Sources

In addition to the two pre-named Service Sources (`OVIS` and `HP Operations Manager`), you can add instances of a SOA Manager Operational Service sources and HP Business Availability sources.

SOA Manager Service Sources

In order to access SOA Manager as a source of operational services, you also need to install the SOA Manager adapter. If you want to connect to more than one SOA Manager system, you must install the adapter multiple times and you can then create an instance of the SOA Manager service source for each adapter installation.

To add a new SOA Manager Service source:

1. Click Add to add a new Service Source

   The `Add Operational Service Source` dialog is presented.

2. Select `Service Oriented Architecture Manager`

The Service Oriented Architecture Manager Source Properties dialog is presented. The properties for the SOA Manager Service Source are described in Table 27.

Table 27 HP SOA Manager Service Source

Descriptive Parameter Name	Description
Service Source Name	Unique name that identifies the service source within the HPBPI interfaces. The name can be up to 40 characters and must not include the following strings: • OVIS • OVSN • OVIS Services • OVO These names are reserved for use by HPBPI in all upper and lower case options. You are advised to include the name of the machine where the adapter is located in order to distinguish between adapter instances in the HPBPI Modeler and other HPBPI interfaces.
Description	An optional description of the source adapter.
Hostname	The name of the host where the SOA Manager adapter is installed and running.
Port	Port Number used to publish the SOA Manager adapter as a Web Service. This is the Axis port number as configured when you install the SOA Manager adapter.
Status Event Poll Interval (seconds)	The time interval (in seconds) that you want be used for this service source to poll the SOA Manager adapter for status events from SOA Manager.

Table 27 HP SOA Manager Service Source

Descriptive Parameter Name	Description
Connection timeout (seconds)	The period of time allowed for the HPBPI Server (SOA Manager adapter client) to request Service Definition and Service status information from the SOA Manager adapter. If this parameter is set to zero (0), the time allowed for the connection is unlimited, that is, no timeout occurs. If a timeout does occur, the SOA Manager adapter retries requests for Service status information when it next polls SOA Manager for service information. In the case of Service Definition information, the Model Repository parameter `Background service import` determines the timing of next request for Service Definitions.
Override Service Adapter’s log level?	Select the Check Box if you want to change the log level setting for this adapter instance.
Log level for this adapter	The level of logging required for the adapter instance. It can be one of `Finer`, `Fine` or `Info`; see section Component Configurations - Logging on page 128 for a description of the log levels.

HP Business Availability Center

If you want to use HP Business Availability Center (BAC) as the source for the status of IT Operational Services, you need to add an operational service source for each BAC system that you want to connect to.

To add a new BAC source:

1. Click `Add` to add a new Service Source

   The `Add Operational Service Source` dialog is presented.

2. Select the `HP Business Availability Center` option.

The `Business Availability Center Source Properties` dialog is presented. The properties for the HP Business Availability Center Sources are described in Table 28

Table 28 HP Business Availability Center Source

Descriptive Parameter Name	Description
Service Source Name	Unique name that identifies the service source within the HPBPI interfaces. The name can be up to 40 characters. You are advised to include the name of the machine where BAC is located to make it easier to identify the service source name from within the HPBPI interfaces.
Description	An optional description of the Business Availability Center source.
BAC Centers Server hostname	The DNS name of the host where the Business Availability Center Server is installed and running. This can be an IP address. Where the Business Availability Center is deployed across a number of machines, this is the DNS name of the Business Availability Center Gateway Server.
BAC Centers Server HTTP port	HTTP port number used by the HPBPI Business Process Dashboard to link to the Business Availability Center Dashboard. This is the HTTP port number for the Business Availability Center Server or Business Availability Center Gateway Server; the terminology depends on the version of Business Availability Center you are using. This is usually port 80.
BAC View name	The name of the Business Availability Center Dashboard view that contains the business process information (service status information in HPBPI terms), required for display in the HPBPI Dashboard. The default view for the Business Availability Center dashboard is the Business Processes view.

Table 28 HP Business Availability Center Source

Descriptive Parameter Name	Description
User name	A user account name with sufficient privileges to access the Business Availability Center Dashboard view specified in `View name`.
Password	The account password for the user account identified by `User name`. The password text is protected as it is typed in the Administration Console.
Status event poll interval (seconds)	The time interval (in seconds) that you want be used for this service source to poll the Business Availability Center Core Server or Business Availability Center Gateway Server servlet engine for status events.
Connection timeout (seconds)	The period of time allowed for the HPBPI Server to request Service status information from the Business Availability Center Core Server or Business Availability Center Gateway Server servlet engine. If this parameter is set to zero (0), the time allowed for the connection is unlimited, that is, no timeout occurs. If a timeout does occur, the HPBPI Server waits for the interval configured for `Status event poll interval` before issuing another request for status events.
Override Service Adapter's log level?	Select the Check Box if you want to change the log level setting for this Business Availability Center Core Server connection.
Log level for this adapter	The level of logging required. It can be one of `Finer`, `Fine` or `Info`; see section Component Configurations - Logging on page 128 for a description of the log levels.

Modifying the Service Source Settings

To change the Service Source settings, complete the following steps on the Windows system, where the HPBPI Server is installed:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Select Component Configuration > Operational Service Sources option from the Navigator pane on the HPBPI Administration Console.
3. Select the Service Source that you want to modify.
4. Select the Modify button.
5. Enter the changes that you want to make to the settings as appropriate in the dialog presented.
6. Click OK to dismiss the dialog.
7. Click Apply to apply the changes to the HPBPI configuration.
8. Select the Status option to move to the panel where the HPBPI component status are shown.
9. Stop and restart all the HPBPI Server components using the Stop All and Start All options.

The new settings are now applied to your HPBPI system.

Proxy for Web Service Adapters

You can set parameters for a Web proxy if you have one for your organization. This Web proxy can then be used by the instances of any Web Service adapters that you create from the Operation Service Source option. The Web proxy applies only to adapters that use Web services, which currently includes only the SOA Manager adapter.

The properties for the `Proxy for web service adapters` are described in Table 29.

Table 29 Proxy for Web Service Adapters

Descriptive Parameter Name	Description
Enable proxy?	Select this checkbox to enable your proxy. Clear the checkbox to disable the proxy.
Proxy host	Name of host used for `HTTP` access. Use this option to configure Web proxy details as an alternative to accessing the SOA Manager service sources that you are configuring.
Proxy port	Port number used by host for `HTTP` access. Use this option to configure Web proxy details as an alternative to accessing the SOA Manager service sources that you are configuring.

Modifying the Web Service Proxy Settings

To change the `Proxy for Web Service Adapter` settings, complete the following steps on the Windows system, where the HPBPI Server is installed:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Select `Component Configuration > Operational Service Sources` option from the Navigator pane on the HPBPI Administration Console.
3. Enter the changes that you want to make to the settings as appropriate in the right-hand pane.
4. Click `OK` to dismiss the dialog.
5. Click `Apply` to apply the changes to the HPBPI configuration.

6. Select the `Status` option to move to the panel where the HPBPI component status are shown.

7. Stop and restart all the HPBPI Server components using the `Stop All` and `Start All` options.

The new settings are now applied to your HPBPI system.

Component Configurations - Model Repository

The Model Repository configuration has one section as follows:

- Background service import

The parameters relating to these settings appear on the right-hand pane of the Administration Console when you select one of these options in the console's navigation tree.

Background Service Import

This section describes the parameters that you can modify through the `Background service import` option.

These settings enable you to configure the intervals for polling OVIS, HP Operations Manager and SOA Manager for Service Definitions as a background activity.

Services are imported into the Model Repository as a background activity at the time period specified and each time the Model Repository is started. The services are imported only when you have also enabled service definition import for OVIS (as described in section OVIS on page 78), or SOA Manager (as described in section Other Service Sources on page 87). HP Operations Manager Services are references to the Services and are not imported into the Model Repository.

This then makes the services available automatically to you through the Modeler without needing to refresh the services manually through the Modeler interface.

Table 30 lists the Background Service Import parameters that you can modify.

Table 30 Background Service Import

Descriptive Parameter Name	**Description**
Enable background service definition import?	A check box to enable or disable importing service definitions into the Model Repository. When checked, background import is enabled. If you uncheck this check box, you can use the menu options within the HPBPI Modeler to import the services that you need defined.
Service import period (minutes)	The time interval (in minutes) for importing Service definitions when background importing is enabled.

Modifying the Background Service Import Parameters

To change the `Background service import` option, complete the following steps on the Windows system, where the HPBPI Server is installed:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Select the `Component Configuration > Model Repository> Background service import` option from the Navigator pane on the HPBPI Administration Console.
3. Enter the changes that you want to make to the settings as appropriate in the right-hand pane.
4. Click `Apply` to apply the changes to the HPBPI configuration.

5. Select the `Status` option to move to the panel where the HPBPI component status are shown.

6. Stop and restart all the HPBPI Server components using the `Stop All` and `Start All` options. You can stop and restart the Model Repository component; however, you are advised to use `Stop All` and `Start All` to ensure that the HPBPI components are stopped and started in the correct order.

The new settings are now applied to your HPBPI system.

Component Configurations - Business Event Handler

The Business Event Handler options enable you to configure the parameters for the Adaptor (based on openadaptor) that sends and receives events between the Business Impact Engine and the Business Event Handler.

These parameters are listed in Table 31.

Table 31 Business Event Handler Settings

Descriptive Parameter Name	Description
Log package information?	A check box that when checked means that the Business Event Handler logs package information.
Log thread information?	A check box that when checked means that the Business Event Handler logs thread information.
Log time information?	A check box that when checked means that the Business Event Handler logs time information.
Maximum number of socket source threads	Maximum number of threads that can be created for the adapter socket source. When this threshold is exceeded, the request for a thread is refused and the Business Event Handler reports a `Connection Refused` error in its log file.
Maximum number of retries to deliver events into Business Impact Engine	The maximum number of times the Business Event Handler attempts to deliver an event to the Business Impact Engine (using RMI) before rolling back the transaction for the business event.

Table 31 Business Event Handler Settings

Descriptive Parameter Name	Description
Business Impact Engine event retry delay (seconds)	Time (in seconds) between each attempt to send an event from the Business Event Handler to the Business Impact Engine using RMI.
Maximum number of retries to deliver events from Hospital	The maximum number of times the Business Event Handler attempts to deliver an event from the Event Hospital to the Business Impact Engine. The Business Impact Engine marks particular categories of event errors to result in the event being sent to the event Hospital and marked to be automatically discharged; for example, events that are received out of sequence and where there is no flow or data instance created for the event. If the Business Event Handler does not succeed in delivering the event to the Engine in within the specified number of retries, the event remains in the event Hospital.
Hospital event poll interval (seconds)	Time (in seconds) that the Business Event Handler polls the Event Hospital looking for events that have been marked as ready for discharge and that can be delivered to the Business Impact Engine.

Modifying the Business Event Handler Parameters

To change these settings, complete the following steps on the Windows system, where the HPBPI Server is installed:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Select the `Business Event Handler` option from the Navigator pane on the HPBPI Administration Console.
3. Enter the changes that you want to make to the settings as appropriate in the right-hand pane.
4. Click `Apply` to apply the changes to the HPBPI configuration.

5. Select the `Status` option to move to the panel where the HPBPI component status are shown.
6. Stop and restart all the HPBPI Server components using the `Stop All` and `Start All` options. You can stop and restart the `Business Event Handler` component; however, you are advised to use `Stop All` and `Start All` to ensure that the HPBPI components are stopped and started in the correct order.

The new settings are now applied to your HPBPI system.

JMS Business Event Handler

In addition to the parameters for the Business Event Handler, you also have the option to enable the JMS Business Event Handler for your implementation.

HPBPI supports the JMS message type `javax.jms.TextMessage`, where the content of the text is an XML file in a particular format. This means you must be integrating with an application that sends and receives this message type if you want to use the JMS Business Event Handler.

Table 32 describes the parameter that you can use to enable the JMS Business Event Handler for your implementation.

Table 32 Enabling the JMS Business Event Handler

Descriptive Parameter Name	Description
Add JMS Business Event Handler to Status Page and enable configuration?	A check box that when checked means that the JMS Business Event Handler options are added to the Administration Console under the JMS Business Event Handler menu. An entry for the JMS Business Event Handler is also added to the `Status` pane to enable you to start and stop it. You need to stop and restart all the HPBPI components using the `Stop All` and `Start All` options and also stop and restart the Administration Console for the new entries to be added to the navigation menu.

Removing the JMS Business Event Handler Options

To remove the JMS Business Event Handler options, complete the following steps on the Windows system, where the HPBPI Server is installed:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Stop the JMS Business Event Handler:

   Select the `Status` page and click the `Stop` button for the JMS Business Event Handler.
3. Select the `Business Event Handler` option from the Navigator pane on the HPBPI Administration Console.
4. Clear the checkbox for the following option:

   ```
   Add JMS Business Event Handler to Status Page and enable
   configuration?
   ```
5. Click the `Apply` button to apply your changes to the HPBPI configuration.
6. Close and restart the Administration Console.
7. The JMS Business Event Handler options have now been removed from the `Navigation` pane.

The new settings are now applied to your HPBPI system.

Component Configurations - Business Process Dashboard

The `Business Process Dashboard` option enables you to manage the parameters of the Business Process Dashboard.

The configuration parameters are divided into a number of logical sections. These sections appear hierarchically in the Administration Console in the `Business Process Dashboard settings` pane within the following categories:

- General settings
- Link to HPSD service calls and incidents

The parameters relating to these settings appear on the right-hand pane of the Administration Console when you select one of the `Business Process Dashboard` options in the console's navigation tree. If the Business Process Dashboard options are not visible in the navigation tree, expand the entries using the Explorer-style navigation techniques.

You can also control the security settings for the Dashboard as described in section Component Configurations - Security on page 123.

General Settings

This option enables you to configure parameters related to the behavior of the Business Process Dashboard. These parameters are listed in Table 33.

Table 33 Business Process Dashboard General Settings

Descriptive Parameter Name	Description
Page refresh delay (seconds)	The time interval (in seconds) that the Business Process Dashboard page uses to set the refresh interval for the Web browser.
Maximum number of retries on database deadlock	The maximum number of attempts by the Business Process Dashboard to retry a database transaction before aborting it. It is possible for deadlocks to occur when the Business Process Dashboard and other applications are accessing the HPBPI database simultaneously. This parameter ensures that, in the case of a deadlock, the Business Process Dashboard aborts the transaction in order to break the deadlock and generates an error message. which is displayed on a Web page. If the Business Process Dashboard aborts the transaction, use the `Refresh` button on your browser to reload the page. This is particularly significant if you are developing your own customized dashboard, and you are using Microsoft SQL Server.
Database deadlock retry delay (seconds)	The time (in seconds) between each attempt to retry a database transaction.
Show service state for a completed node?	A check box indicating whether or not to show the status of services for nodes that have already been completed in the business flow.

Table 33 Business Process Dashboard General Settings

Descriptive Parameter Name	Description
Show superseded flows?	A check box indicating whether or not to show superseded flows through the Dashboard.
HPBPI server time zone	This parameter is available only when you have selected a Dashboard Only installation type. The timezone for the machine where the HPBPI Server is installed. This enables you to make sure that, in cases where the remote Business Process Dashboard is located in a different time zone, the times displayed from a remote Business Process Dashboard are correct. When selecting a time zone option, you are advised to select a location/city style option, for example, Pacific/Midway. Selecting this option, rather than a GMT offset (for example, GMT+8) allows a daylight saving rule to be applied.

Modifying the General Settings

To change these settings, complete the following steps on the Windows system, where the HPBPI Server is installed:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Select the `General Settings` option from the Navigator pane on the HPBPI Administration Console.
3. Enter the changes that you want to make to the settings as appropriate in the right-hand pane.
4. Click `Apply` to apply the changes to the HPBPI configuration.
5. If you have modified the `Show service state for a completed node?` parameter, you need to stop and restart the Servlet Engine as described in step 6. If you have not modified this parameter, continue at step 8.
6. Select the `Status` option to move to the panel where the HPBPI component status are shown.
7. Stop and restart the Servlet Engine component.

8. If you have modified the `Page refresh delay (seconds)` parameter, and the parameter was previously set to zero (0), you need to refresh the Web browser window where the Business Process Dashboard is running.

9. The new settings are now applied to your HPBPI system.

Link to HPSD Service Calls and Incidents

This option enables you to link the HPBPI Server to an HP Service Desk implementation from the Business Process Dashboard. If this integration is correctly configured, the Business Process Dashboard can link HPBPI Services to the related HPSD Service Calls and Incidents.

The HPSD integration configuration parameter settings are divided into logical sections. These sections appear on the `Link to HPSD service calls and incidents` pane within the following categories:

- Link to HP Service Desk Service Calls and Incidents on page 105
- Refresh Service Desk Content on page 106
- HP Service Desk Custom Field Configuration on page 107
- Open Incident Status Configuration on page 108
- Open Servicecall Status Configuration on page 108

Link to HP Service Desk Service Calls and Incidents

Figure 34 lists the parameters that you can set to enable HPBPI to integrate with a specific HPSD system to obtain the Service Call and Incident information for flow instances.

Table 34 Link to HPSD Service Calls and Incidents

Descriptive Parameter Name	Description
Enable link?	Select this check box to enable the HPBPI Business Process Dashboard to access HPSD Incident reports and Service Calls, which are then available from the Dashboard Service Health page.
HP Service Desk application server hostname	The fully qualified domain name for the machine where the HPSD application server that you want to access is running.

Table 34 Link to HPSD Service Calls and Incidents

Descriptive Parameter Name	Description
HP Service Desk user name	The name of the user account set up within the HPSD implementation for HPBPI. This is the user name that HPBPI uses to access HPSD. You are recommended to create an HPSD user account specifically for HPBPI and this user account should: • be a concurrent user • have the role `Helpdesk` Note that the user name that you specify for HPBPI is the HPSD login name for the user and not the display name.
HP Service Desk password	The password associated with the user account.

You can now optionally configure the type of integration you require with HPSD. The *Business Process Insight Reference Guide* provides details of the different configuration options and how they relate to the parameters listed in Table 35 on page 107.

If you want to configure HPSD custom field information or the Incident and Service Call status information that is presented see the following sections.

Refresh Service Desk Content

Click the `Refresh Service Desk Content` button to obtain the latest settings for custom fields, Incident and Service Call status information from the HPSD machine that you have configured for HPBPI. When you refresh the settings the parameters described in the following sections are updated:

- HP Service Desk Custom Field Configuration on page 107
- Open Incident Status Configuration on page 108
- Open Servicecall Status Configuration on page 108

When you first open the `Link to HPSD service calls and incidents` pane, the contents of the drop-down lists for Custom Field, Incident and Service Call information are disabled. The data shown in the drop-down lists show the current settings.

The content for the Custom Field, Incident and Service Call drop-down lists is populated from the Service Desk using the `Refresh Service Desk Content` button. Each time you refresh the Service Desk content, make your selection and click the `Apply` button, the contents need to be re-populated before further options can be selected.

HP Service Desk Custom Field Configuration

If you plan to enable custom field support, you can optionally select to refresh the custom field information presented through the Administration Console to make sure it is up to date. Click the `Refresh Service Desk Content` button to do this; see also section Refresh Service Desk Content on page 106.

Table 35 lists the parameters that you can modify for the custom field configuration.

Table 35 HPSD Custom Field Configuration

Descriptive Parameter Name	Description
HP Service Desk integration type	An indication of whether you want to configure HPBPI to use the already defined HP Operations Manager and OVIS service definitions within HPSD (`Automatic`), or whether you want to enable the use of HPSD custom fields (`Custom only`). There are three options: • `Automatic` • `Custom only` • `Both`
Select HP Service Desk custom field to use	This option available only when you choose to enable `Custom only` or `Both` as an integration type. The custom field that you want to be used within HPSD to hold HPBPI Service names. The list of fields presented includes those already assigned, and those that are currently available (inactive). If there are no fields currently available, you need to find out whether any of the assigned fields can be reused for HPBPI.

Open Incident Status Configuration

Use the parameters listed in Table 36 to set the lower and upper bounds for the HPSD Incidents that you want to see reported from the Business Process Dashboard. The Lower Incident and Upper Incident parameter values must be logical in that the lower Incident Status must be lower than the higher Incident Status. If the selected values are not logical, you are presented with an error message. You can select the same value for the upper and lower status.

Table 36 Open Incident Status Configuration

Descriptive Parameter Name	Description
Lower Incident Status	The lowest HPSD Incident status setting that you want to be reported as open within the Business Process Dashboard.
Upper Incident Status	The highest HPSD Incident status setting that you want to be reported as open within the Business Process Dashboard.

Open Servicecall Status Configuration

Use the parameters listed in Table 37 to set the lower and upper bounds for the HPSD Service Calls that you want to see reported from the Business Process Dashboard. The Lower Service Call and Upper Service Call parameter values must be logical in that the lower Service Call Status must be lower than the higher Service Call Status. If the selected values are not logical, you are presented with an error message. You can select the same value for the upper and lower status.

Table 37 Open Servicecall Status Configuration

Descriptive Parameter Name	Description
Lower Servicecall Status	The lowest HPSD Service Call status setting that you want to be reported as open within the Business Process Dashboard.
Upper Servicecall Status	The lowest HPSD Service Call status setting that you want to be reported as open within the Business Process Dashboard.

Modifying the HPSD Service Calls and Incidents Settings

To change these settings, complete the following steps on the Windows system, where the HPBPI Server is installed:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Select the `Link to HPSD Service Calls and Incidents` option from the Navigator pane on the HPBPI Administration Console.
3. Enter the changes that you want to make to the settings as appropriate in the right-hand pane.

   If you are changing the user name for the HPBPI Service Desk user, make sure the user has the correct profile characteristics as described in the *HPBPI Installation Guide*.
4. Click `Apply` to apply the changes to the HPBPI configuration.
5. Select the `Status` option to move to the panel where the HPBPI component status are shown.
6. Stop and restart the Servlet Engine component.
7. The new settings are now applied to your HPBPI system.

Component Configurations - MS SQL Server Access

The MS SQL Server Access parameters are available only when you select MS SQL as the database from within the HPBPI installation procedure. If you select Oracle as your database, the MS SQL Server parameters are not available through the Administration Console.

The `MS SQL Server Access` option enables you to view the current configurations for the SQL Server access and modify the password for the database access. The database is used by the following components:

- Business Impact Engine
- Metric Engine
- Business Event Handler
- Notification Server
- Business Process Dashboard
- Intervention Client
- OVIS, where you have installed the HPBPI OVIS Custom Probes

These parameters are listed in Table 38.

Table 38 MSSQL Access

Descriptive Parameter Name	Description
MS SQL Server hostname	This is a non-modifiable field and shows the fully qualified domain name of the system where the Microsoft SQL Server Database is installed.
MS SQL Server port number	This is a non-modifiable field and shows the port number that the Microsoft SQL Server Database is expecting connections on.
MS SQL Server database	This is a non-modifiable field and shows the name of the database configured for HPBPI.

Table 38 MSSQL Access

Descriptive Parameter Name	Description
MS SQL Server login user	This is a non-modifiable field and shows the name of the user who has permission to read and write the HPBPI database files.
MS SQL Server login password	The password for the user with permission to read and write the HPBPI database files. This field can be used to re-synchronize the HPBPI version of the password with the database password after the database password is changed using the database management tools.

To change the password for the database, follow the instructions on section Changing the Password Details on page 111.

Changing the Password Details

This section describes how to change the password for the HPBPI database user that you created during the installation.

Complete the following steps to make the changes to the database password:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Stop all the HPBPI components using the `Stop All` button.
3. Select the `MS SQL Server Access` option on the HPBPI Administration Console and make the changes to the following fields:
   - `MSSQL login password`
   - `Confirm password`

4. Click `Apply` to apply the changes to the HPBPI configuration.
5. Return to the `Status` option and restart all the HPBPI components, using the `Start All` button.

Make sure that you also make this change on systems that have remote HPBPI components installed; for example, remote Business Process Dashboards.

You have now completed the tasks to change the database password details used by HPBPI components.

Component Configurations - Oracle Server Access

The Oracle Server Access parameters are available only when you select Oracle as the database from within the HPBPI installation procedure. If you select MS SQL Server as your database, the Oracle Server parameters are not available through the Administration Console.

The `Oracle Access` option enables you to manage the properties of the connection to the Oracle database. The database is used by the following components:

- Business Impact Engine
- Metric Engine
- Business Event Handler
- Notification Server
- Business Process Dashboard
- Intervention Client
- OVIS, where you have installed the HPBPI OVIS Custom Probes

These parameters are listed in Table 39.

Table 39 Oracle Server Access

Descriptive Parameter Name	Description
Oracle Server hostname	This is a non-modifiable field and shows the fully qualified domain name of the system where the Oracle Database is installed.
Oracle Server port number	This is a non-modifiable field and shows the port number that the Oracle Database is expecting connections on.
Oracle Server SID	This is a non-modifiable field and shows the name of the database sid configured for HPBPI.

Table 39 Oracle Server Access

Descriptive Parameter Name	Description
Oracle Server login user	This is a non-modifiable field and shows the name of the user who has permission to read and write the HPBPI database files.
Oracle Server login password	The password for the user with permission to read and write the HPBPI database files. This field can be used to resynchronize the HPBPI version of the password with the database password after the database password is changed using the database management tools.

If you are changing the password for the database user with permission to read and write the HPBPI database tables, follow the instructions on section Changing the Password Details on page 114.

Changing the Password Details

To change the password for the HPBPI database user, complete the following steps:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Stop all the HPBPI components using the `Stop All` button on the `Status` option.
3. Select the `Oracle Access` tab on the HPBPI Administration Console and make the changes to the following fields:
   - `Oracle Server login password`
   - `Confirm password`

4. Click `Apply` to apply the changes to the HPBPI configuration.
5. Return to the `Status` option and restart all the HPBPI components, using the `Start All` button.

Make sure that you also make this change on systems that have remote HPBPI components installed; for example, remote Business Process Dashboards.

You have now completed the tasks to change the database password details used by HPBPI components.

Component Configurations - HP Operations Manager Adapter

This section describes the configuration options to add the HP Operations Manager Adapter for Windows to the list of installed components that can be managed from this system. The Adapter needs to be installed on the same system as the HPBPI Server, you cannot manage adapters running remotely using this option. (You can manage only those HPBPI components that are running on the same system as the system where the Administration Console is running.)

This option is available only if you have completed a `Server and Modeler` or `Server Only` installation type.

Table 40 describes the HP Operations Manager Adapter options.

Table 40 HP Operations Manager Adapter

Descriptive Parameter Name	Description
Add HP Operations Manager Adapter to Status Page?	Select this option to add the HP Operations Manager Adapter to the list of HPBPI components on the Status pane. You need to stop and restart all the HPBPI components using the `Stop All` and `Start All` options and also stop and restart the Administration Console for the new entry to be added to the navigation menu.

Modifying HP Operations Manager Adapter Settings

To change the HP Operations Manager Adapter settings, complete the following steps:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Select the `HP Operations Manager Adapter` option in the Navigator pane.
3. Make the required changes.

4. Click `Apply` to apply your modifications.
5. Stop and restart all the HPBPI components.
6. Exit and restart the Administration Console.

The `HP Operations Manager Adapter` entry on the `Status` pane is added or removed as appropriate.

Component Configurations - Port Numbers

This section describes the port numbers used by the HPBPI components. You might need to modify these if there is a port number clash, or if you have specific requirements with personal firewalls or other security processes.

The port number parameters are listed in Table 41

Table 41 Port Numbers

Descriptive Parameter Name	Description
RMI Registry	The port number for SUN's implementation of the Java RMI Activation System Daemon (RMID), which is used by HPBPI as a reliable RMI registry. The RMI registry holds the directory of port numbers used by the HPBPI components to receive RMI requests. If this port number is in use by another application, the HPBPI Server components are unable to start.
Administration Console Server	The port number that the server component of the Administration Console uses to listen for incoming administration requests.
Business Impact Engine SOAP Receiver	The port number used by the Business Impact Engine to receive incoming SOAP requests.
Business Impact Engine XML Receiver	The port number used by the Business Impact Engine to receive XML requests.
Business Impact Engine RMI Receiver	The port number used by the Business Impact Engine to receive incoming RMI requests.
Metric Engine RMI Services	The port number used by the Metric Engine to receive connections to RMI services, for example, a service that enables the Business Impact Engine to determine whether or not there is backlog of Metric Events queued at the Metric Engine.
Business Event Handler Source	The port number used by the Business Event Handler Engine Adapter to receive incoming Events into the Business Event Handler.

Table 41 Port Numbers

Descriptive Parameter Name	Description
Business Event Handler Control	The port number used to stop and start the Business Event Handler Engine Adapter programmatically; see the *Business Process Insight Integration Training Guide - Business Events*, for more details.
Model Repository	The port number used by the Model Repository to listen for incoming RMI requests.
HP Operations Manager Adapter Request	The port number used by the HP Operations Manager Adapter to receive incoming RMI requests from the Business Impact Engine. Note that the port numbers entered on the HPBPI Server system for the HP Operations Manager Request must match the numbers entered for the HP Operations Manager port numbers on the system where the adapter is installed.
HP Operations Manager Adapter Reply	The port number used by the HPBPI Operations Adapter when sending HP Operations Manager responses related to the change in status of HP Operations Manager Services, and for the Business Impact Engine to listen for responses. Note that the port numbers entered on the HPBPI Server system for the HP Operations Manager Reply must match the numbers entered for the HP Operations Manager port numbers on the system where the adapter is installed.
Servlet Engine HTTP	The port number that the Tomcat Web Server uses to receive HTTP requests.
Servlet Engine HTTPS	The port number that the Tomcat Web Server uses to receive HTTP over SSL encrypted requests.
Servlet Engine Shutdown	The port number used by the Tomcat Servlet Engine to shutdown.

Table 41 Port Numbers

Descriptive Parameter Name	Description
Servlet Engine AJP 1.3	The AJP 13 protocol is a packet-based protocol that allows a Web server to communicate with the Tomcat JSP/Servlet Container over a TCP connection.
Servlet Engine Admin	The port number used by the Tomcat Servlet Engine administration tools.
Web Services Provider	The port number used by the HP Operations Dashboard to access the HPBPI Business Process Dashboard pages and display them within the HP Operations Dashboard.

To change the `HP Operations Manager Adaptor Request` or `HP Operations Manager Adaptor Reply` port numbers; refer to section HP Operations Manager Adapter Port Numbers on page 120.

For all other port number changes, refer to section Modifying HPBPI Port Numbers on page 122.

HP Operations Manager Adapter Port Numbers

The HP Operations Manager Adapter comprises a client and a server component. The client component is embedded in the Model Repository and the Business Impact Engine components and the server component runs on the same system as HP Operations Manager Adapter.

If you need to alter the port numbers used by the adapter server or client, complete the following steps:

1. Start the Administration Console on the system where the HPBPI Server is installed as described in section Administration Console Description on page 20.
2. Stop the HPBPI components using the `Stop All` button on the `Status` option.

3. Start the HPBPI Administration Console on the system where the HP Operations Manager Adapter Server is installed.

   If the adapter is installed on HP-UX, use the following command to start it:

   ```
   HPBPI-install-dir/bin/biaadmin.sh
   ```

4. Stop the `HP Operations Manager Adaptor Server` component using the `Stop` button.
5. On the system, where the HP Operations Manager Adapter is installed, select the `Port Numbers` option from the Administration Console.
6. Modify one or both of the following port numbers as appropriate:
   - `HP Operations Manager Adaptor Request`
   - `HP Operations Manager Adaptor Reply`
7. On the system where the HPBPI server is installed, select the `Port Numbers` option from the HPBPI Administration Console.
8. Modify one, or both of the following port numbers as appropriate:
   - `HP Operations Manager Adaptor Request`
   - `HP Operations Manager Adaptor Reply`

   Note that the port numbers entered on the HPBPI Server system for the HP Operations Manager Request and Reply must match the numbers entered for the HP Operations Manager port numbers on the system where the adapter is installed.

9. Use the HPBPI Administration Console to restart the HPBPI component on the system where HP Operations Manager adapter is installed.
10. Use the HPBPI Administration Console to restart the HPBPI components on the system where the HPBPI Server is installed.

You have now completed the tasks to reconfigure the port numbers for the HP Operations Manager Adapter.

Modifying HPBPI Port Numbers

If you want to modify the HP Operations Manager Adapter port numbers, refer to section HP Operations Manager Adapter Port Numbers on page 120.

The HPBPI components use a number of different port numbers for connections. There should not be any need to modify these port numbers unless you know that other applications are already using them, or if you are using personal firewall software that filters incoming connections based on port number. You must stop all the HPBPI components before you can modify the HPBPI port numbers.

To change the port number for the HPBPI Server components, complete the following steps:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Using the `Status` option, stop the HPBPI components using the `Stop All` button.

   You cannot modify any of the port numbers until you have stopped the HPBPI components.
3. Select the `Port Numbers` option in the Navigator pane.
4. Make the changes to the port numbers that you want to modify.
5. Click `Apply` to apply your modifications.
6. Restart, using `Start All`, the HPBPI components that you previously stopped.

The port number changes have now been applied.

Component Configurations - Security

The Security option enables you to set the type of authorization and HTTP security that you want to use for the HPBPI components.

The parameters relating to security appear on the right-hand pane of the Administration Console when you select Security in the Administration Console's navigation tree.

The right-hand pane is divided into the following logical sections:

- Authorization Settings on page 123
- HTTP Over SSL Settings on page 125

Authorization Settings

The Authorization settings enable you to configure whether you want to use Select Access or the Servlet Engine as the method of authorization for your HPBPI components. These authorization settings apply to all the HPBPI components.; it is not possible to use Select Access as the authorization method for some components and the Servlet Engine for others. The only exception to this is for the Dashboards, you can choose not to use authorization for the Dashboards at all. If you do want the Dashboards to be authorized, they have to use the authorization method selected for the rest of the HPBPI components.

Table 42 lists the parameters that you configure to use either Select Access or the Servlet Engine as the method of authorization for your HPBPI components.

If you want to choose Select Access as your method of authorization, you must first configure all the HPBPI components within Select Access. If you attempt to set Select Access as your method of authorization without first configuring Select Access, the HPBPI Administration Console does not accept the configuration change. This is to prevent the situation where you could be blocked from accessing any of the HPBPI components.

Be aware that you are not able to configure the Dashboard to use the Servlet Engine authentication and have the remaining HPBPI components use Select Access.

Table 42 Authorization Settings

Descriptive Parameter Name	Description
Authorization method	Use this option to select which method, or product, that you want to use to authorize your HPBPI components. The method can be using either the Tomcat Servlet Engine Realm mechanism, or HP Select Access authorization. Information about using the Servlet Engine or Select Access for authorization can be found in this guide as follows: • Chapter 8, Select Access Authorization • Chapter 9, Servlet Engine Authentication
Enforcer configuration file	If you choose Select Access as your preferred method for authorization, you need to specify the location of the Servlet Enforcer configuration file that you have created. This field is not available if you choose Servlet Container as your method of authorization.
Use authorization for the Dashboards?	Check the box if you want to secure access to the Business Process Dashboards using the method of security selected under the `Authorization method` options. When this option is not checked, no authorization is required for the Dashboard. Following a new installation, when you select this options and you have Servlet Engine as the method of authorization, the built-in user name is `dashboard` and the password is `console`; refer to Chapter 9, Servlet Engine Authentication for more details of modifying the Servlet Engine authorization settings for the Dashboard. Note that administrator username and password can also be used to access the Business Process Dashboard.

HTTP Over SSL Settings

HTTP over SSL (secure socket layer) is a secure version of HTTP. SSL enables the data from a client, such as a Web browser, to be encrypted prior to transmission to prevent unauthorized access to the data being transmitted.

Refer to Chapter 9, Servlet Engine Authentication for more information about HPBPI and HTTPS.

Table 43 lists the properties that you can configure to encrypt the data that you send to the HPBPI Web-based clients.

Table 43 HTTP Over SSL Settings

Descriptive Parameter Name	Description
Enable HTTP over SSL for Dashboards?	Select the checkbox to enable HTTP over SSL from the Web Server to the Web Browser, or Web Browsers, where the Business Process Dashboard is running. The setting applies to all Web Browsers where the Dashboard is running for a specific installation of the Servlet Engine. Clear the checkbox if you do not want to use HTTP over SSL.
Enable HTTP over SSL for Repository Explorer?	Select the checkbox to enable HTTP over SSL from the Web Server to the Web Browser, or Web Browsers, where the Repository Explorer is running. The setting applies to all Web Browsers where the Repository Explorer is running for a specific installation of the Servlet Engine. Clear the checkbox if you do not want to use HTTP over SSL.
Enable HTTP over SSL for Metric Definer?	Select the checkbox to enable HTTP over SSL from the Web Server to the Web Browser, or Web Browsers, where the Metric Definer is running. The setting applies to all Web Browsers where the Metric Definer is running for a specific installation of the Servlet Engine. Clear the checkbox if you do not want to use HTTP over SSL.

Table 43 HTTP Over SSL Settings

Descriptive Parameter Name	Description
Enable HTTP over SSL for Intervention Client?	Select the checkbox to enable HTTP over SSL from the Web Server to the Web Browser, or Web Browsers, where the Intervention Client is running. The setting applies to all Web Browsers where the Intervention Client is running for a specific installation of the Servlet Engine. Clear the checkbox if you do not want to use HTTP over SSL.
Enable HTTP over SSL for Notification Server Admin?	Select the checkbox to enable HTTP over SSL from the Web Server to the Web Browser, or Web Browsers, where the Notification Server Administration Console is running. The setting applies to all Web Browsers where the Notification Server Administration Console is running for a specific installation of the Servlet Engine. Clear the checkbox if you do not want to use HTTP over SSL.
SSL Keystore	This is the location of the signed certificate file required for the HTTPS connection.
SSL Keystore Password	This is the password for the signed certificate that you supply.
Enable Certificate-based Client Authentication	Select the checkbox to enable certificate-based client authentication (from the Web Browser to the Web Server). Clear the checkbox if you do not want to use this form of authentication.

Modifying the Security Parameters

To change these settings, complete the following steps on the Windows system, where the HPBPI Server is installed:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Select the `Security` option from the Navigator pane on the HPBPI Administration Console.
3. Enter the changes that you want to make to the settings as appropriate in the right-hand pane.
4. Click `Apply` to apply the changes to the HPBPI configuration.

   If the Administration Console returns an error and prevents you from making the changes, make sure that you have configured HP Select Access as described in Chapter 8, Select Access Authorization.
5. Select the `Status` option to move to the panel where the HPBPI component status are shown.
6. Stop and restart all the HPBPI Server components using the `Stop All` and `Start All` options. If you do not start all the components, some of the security settings can be compromised.

The new settings are now applied to your HPBPI system.

Component Configurations - Logging

This section describes the parameters that you can set for the log HPBPI component log levels and the component startup settings. Section Logging Parameters on page 128 describes the logging parameters and section Component Startup Parameters on page 132 describes the component startup parameters.

Logging Parameters

HPBPI includes a number of third-party products, which use their own logging mechanisms, for example:

- the Business Event Handler component, which is based on the openadaptor framework, uses Log4j as its logging mechanism.
- Hibernate, which is used by the Business Impact Engine and other HPBPI Server components, uses Jakarta Apache Commons logging.

All the HP owned HPBPI components are based on the HP Operations Manager mechanism for logging, which is Java J2SE logging.

Log4j and Java J2SE logging are two alternative implementations and Apache Commons is an layer above a specific logging implementation. In the case of HPBPI, Apache Logging is a layer over the Java J2SE logging implementation.

This Logging option enables you to set logging levels for the HPBPI components, and for the Business Event Handler; the Business Event Handler logging is based on openadaptor. There is no mechanism for setting the logging levels for Hibernate using the Administration Console, as the logging for the components using Hibernate is usually sufficient.

The logging parameters, which can be set through the Logging option, are listed in Table 44. For your HPBPI implementation, you see only those logging parameters that relate to the components that you have installed.

Table 44 Logging

Descriptive Parameter Name	**Description**
Change the log viewer application	
Path for the log viewer application	The full path name for the application that is being used to display the log files. If the application is defined as being on your `PATH`, you can enter the application name without its full path name.
Logging Levels - HP Operations Manager Logging	
The Administration Console log level	The log level set for the Administration Console.
The Administration Console Server log level	The log level set for the Administration Console Server.
The Metric Engine log level	The log level set for the Metric Engine
The Business Impact Engine log level	The log level set for the Business Impact Engine.
The Model Repository log level	The log level set for the Model Repository.
The Notification Server log level	The log level set for the Notification Server
The HP Operations Manager Adapter log level[a]	The log level set for the HP Operations Manager Adapter.
The Servlet Engine log level	The log level set for the Tomcat Servlet Engine.
The Web Services Provider log level	The log level for the Web Services interface to Operations Dashboard.

Table 44 Logging

Descriptive Parameter Name	Description
The Service Adapters' log level	The log level for all the Service Sources, unless overridden by specific instances of the service source. You override the log level on the configuration page for the individual Service Source. Each entry in the log file is identified by the name that you give the service source in the Administration Console.
The BAC Data Samples Destination log level	The log level for all the Data Sample Destinations, unless overridden by specific instances of the Destination. You override the log level on the configuration page for the individual Data Sample Destination. Each entry is identified by the name that you give the Destination in the Administration Console.
Logging Levels - Log4j Logging	
The Business Event Handler log level	The logging level set for the Business Event Handler.

a. Available only when the HP Operations Manager Adapter is installed.

The following are the logging levels that you can set for the `HP Operations Manager Logging` in ascending level of log detail provided:

1. `Info`, which reports runtime informational messages and is the default level of logging.
2. `Fine`, which provides more detail and is used when debugging or diagnosing problems.
3. `Finer`, which provides the greatest detail and is used for tracing and debugging low level problems.

The following are the logging levels that you can set for `Log4j Logging` in ascending level of detail:

1. `Info`, which reports at the informational level and is the default level of logging.
2. `Trace`, which should be used for debugging purposes.

➤ When setting logging levels to report more detail, make sure that these are temporary settings as the detail logged can consume a significant amount of disk space and impact the performance of the system where HPBPI is installed.

Modifying the Log File Parameter Values

To modify the log levels for the HPBPI server components complete the following steps:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Stop the server components using the `Stop All` button.
3. Select the `Logging` option in the Navigator pane.
4. Make the changes to the log levels that you want to modify.
5. Click `Apply` to apply your modifications.
6. Restart the HPBPI components that you previously stopped.

The logging level changes have now been applied.

⚠ The log levels set when HPBPI is installed are suitable for most error reporting. If you need to increase the granularity of the logging information provided, make sure that you set the log levels for a temporary period only. When set the `Fine` and `Finer` logging levels, the log files become large very quickly and can have a detrimental effect on your system's performance.

Component Startup Parameters

Use this parameter to configure the time allocated to start individual components.

HPBPI components have a configured time allocated for starting. If they fail to start within the period specified by the `Startup timeout (seconds):` parameter, they are assumed to have failed.

On some machines, it can take some time for components to start; for example, because the machine has a slow processor, or has many applications running. If this is the case in your implementation, you can increase the value of the startup time to allow more time for the component startup.

Table 45 describes the startup setting.

Table 45 Component Startup Settings

Descriptive Parameter Name	Description
Startup timeout (seconds):	The time, in seconds, that is allocated to start each HPBPI component. All components are allocated the same value for their startup time.

Modifying the Component Startup Parameter Values

To modify the component startup parameter for the HPBPI server components, complete the following steps:

1. Start the Administration Console as described in section Administration Console Description on page 20.
2. Stop the server components using the `Stop All` button.
3. Select the `Logging` option in the Navigator pane.
4. Make the changes to the startup time.
5. Click `Apply` to apply your modifications.
6. Restart the HPBPI components that you previously stopped.

The startup times are now modified.

License Management

You use the License Management software (Autopass) to manage the HP Operations Manager software license keys. When you purchase an HP BTO Software product, for example HPBPI, you receive an Entitlement Certificate. This Entitlement Certificate contains information that you need to retrieve and install HPBPI license keys.

The License Management utility is not an HPBPI component. HPBPI provides access to the utility to make it easier for you to request and install the HPBPI license keys. You can also request your license keys using email or facsimile (fax). Requesting license keys is described as part of the License Key installation in the License Management online help.

The License Management utility provides a number of features, including:

- License key installation
- License key reporting
- License key backup
- License key removal
- License key recovery

Details of these features and how to use them are provided in the License Management online help.

To install your HPBPI license keys complete the following steps:

1. Make sure that you have your Entitlement Certificates, which contain the order numbers that you need to retrieve your license keys.
2. Open the HPBPI Administration Console.
3. Select `License Manager` from the `File` menu on the Administration Console to start the Autopass License Management software.

   The License Management utility opens at the screen to `Retrieve/ Install License Key`.

4. Follow the on-screen instructions and the instructions in the License Management online help to install the HPBPI license keys.

5. Select the `Report License Key` option from the License Management Navigation window to check that the HPBPI license keys are successfully installed.

   When the keys are successfully installed, you should see the HPBPI product number with an `Expiration Date` of `Forever`.

If you have problems retrieving or installing the license keys, contact your local support representative. The Administration Console provides access to the Support Web site from the `Support Online` option on the `Help` menu.

3 HPBPI Modeler Administration

This chapter describes the administration tasks related to the HPBPI Modeler.

The chapter describes specific tasks relating to the management of your definitions. For details of creating definitions and using the HPBPI Modeler refer to the *Business Process Insight Integration Training Guide - Modeling Flows*. You can also find information about using the HPBPI Modeler in the Modeler online help.

This chapter describes the following:

- How to start and stop the Modeler component; see section Starting and Stopping the Modeler on page 137.
- The configuration parameters relating to the HPBPI Modeler connection to the Repository Server; see section Changing the Details of the Repository Server on page 138.
- Deploying definitions after they have been defined; see section Deploying Flows and Dependencies on page 140.
- How to undeploy flow definitions from the Model Repository using the Modeler; see section Undeploying Definitions from the HPBPI Modeler on page 141.
- How to access the log files associated with the Modeler; see section HPBPI Modeler Log Files on Windows on page 143.
- Using the Modeler to export and import flow definitions; see section Exporting and Importing Flow Definitions on page 144.

- Using the Modeler to import BPEL processes; see section Importing BPEL Processes on page 146.
- Copy properties between Data and Event definitions; see section Copying Properties Between Data and Event Definitions on page 147.
- Exporting definitions in order to recover unsaved changes; see section Exporting Definitions to Recover Unsaved Changes on page 148.

Starting and Stopping the Modeler

Before trying to start the HPBPI Modeler, make sure that the `Model Repository` component is started on the HPBPI Server system. Use the HPBPI Administration Console to start the Model Repository as described in section Starting and Stopping the HPBPI Server Components on page 27.

Starting the Modeler

To start the HPBPI Modeler, complete the following steps:

1. On the system where the Modeler is installed, start the HPBPI Modeler as follows:

   ```
   Start|Programs|HP|HP Business Process Insight|Modeler
   ```

   You are presented with an `HPBPI Modeler` dialog.

2. Enter details of the username and password to connect to the Model Repository. On an new installation, the username is `admin` and the password is `hpbpi`.

   The HPBPI Modeler uses the same username and password as is configured for the Repository Explorer. You can modify the Repository Explorer login credentials using the Tomcat Realm configuration or Select Access. Chapter 8, Select Access Authorization describes how to use Select Access for authorization and Chapter 9, Servlet Engine Authentication describes the `tomcat-users.xml` file that you can edit to change the login credentials for the Repository Explorer.

   You can also modify the details of the Repository Server that you connect to when you start the HPBPI Modeler. This is described in section Changing the Details of the Repository Server on page 138.

3. Click `OK`, and the HPBPI Modeler opens.

Stopping the Modeler

To stop the HPBPI Modeler, select `Exit` from the `File` menu in the Modeler window. You are prompted to save any currently unsaved changes to the models that you have been editing.

Changing the Details of the Repository Server

Table 46 describes the HPBPI Modeler parameters that you can modify that relate to its connection to the Repository Server. These parameters are available from the `File|Options` menu in the HPBPI Modeler, or through the `Options...` button on the HPBPI Modeler login dialog box. You can modify the parameters on the login dialog only; the Server options within the `File|Options` menu are for reference only.

Table 46 Repository Server Configuration Parameters

Descriptive Parameter Name	Description
Hostname	The hostname of the system where the HPBPI Server (and Repository Server) is installed.
RMI Registry Port Number	The port number used for the Java Remote Method Invocation (RMI) demon. The port number is required in order that Java processes can communicate with each other. You need to change this option if you modify the port number through the Administration Console on the HPBPI Server. The port number used by the HPBPI Server is shown within the HPBPI Administration Console under the `Port Number` option as `RMI Registry`.

Table 46 Repository Server Configuration Parameters

Descriptive Parameter Name	Description
Modeler Port Number	The port number used by the HPBPI Server when it needs to communicate with the Modeler. This port number can be zero (0), in which case, the HPBPI Server selects a port number at random; however, if you want to ensure the port number fits with your organization's firewall policies or other similar policies, you need to specify the port number explicitly.
Servlet Engine HTTP Port	The port number used to communicate with the Servlet Engine on the HPBPI Server system in order to access the Model Repository. You need to change this option if you modify the port number through the Administration Console on the HPBPI Server. The port number used by the HPBPI Server is shown within the Administration Console under the `Port Number` option as `Servlet Engine HTTP`.

Deploying Flows and Dependencies

This section explains how to deploy a Flow and any associated Data definitions, Event definitions and Services.

Before deployment, you have either created a design-time definition in the Model Repository that the Business Impact Engine has no knowledge of, or have undeployed a definition and made changes to it. When you deploy the definition to the Engine, the Engine then creates:

- a Java-compiled definition for the flow in its database
- database entries for the definition
- a mapping file

To deploy the Flow, Data, Event and Service definitions to the Business Impact Engine, complete the following steps:

1. Make sure the Modeler is started.
2. Select the Flow from the Navigator pane in the left-hand pane.
3. Select `File|Deploy` from the menu options.
4. You are prompted to save all currently unsaved changes, if you have not saved them already. Click `Yes`.

   The Modeler checks the state of all the definitions that you selected to deploy. A dialog box appears to prompt you to confirm that you want to deploy all currently undeployed changes to the business flow and all its dependencies. Click `Deploy` to confirm your request.

   The deployer shows its progress as it deploys the Services, Flows, Data and Event definitions.
5. Click `Close` when the Deployer indicates that it has finished deploying the definitions.

You have now deployed the flow and its dependencies.

Undeploying Definitions from the HPBPI Modeler

There are circumstances where you might want to undeploy one or more of your business flows. In addition, you might need to undeploy individual definitions from the HPBPI Model Repository; for example, you might need to undeploy Event definitions when an Event becomes out of date.

You can undeploy Flows, Data and Event definitions, you cannot undeploy services. This is because services are identifiers for operational services that exist in other applications, for example OVIS. The deployment status for services is marked, in the Modeler interface, as `Deployed or Not Applicable`.

In the case of HP Operations Manager services, a warning is shown if there is no equivalently named service in HP Operations Manager. The warning is removed when the service is created in HP Operations Manager and the HPBPI Modeler has successfully imported the revised service definitions.

If you undeploy a definition, existing instances using the definitions are still running in the Business Impact Engine continue to run until completion, or until the Instance Cleaner deleted them. New instances of any undeployed definitions are no longer created. The definition is removed by the Model Cleaner as described in section Modifying the BIE Model Cleaner Settings on page 45.

You can undeploy a definition that is referenced by another definition, so you need to make sure that you maintain consistency within your business flows.

To undeploy a Flow, Data or Event definition, complete the following steps:

1. Start the HPBPI Modeler as follows:

   `Start|Programs|HP|HP Business Process Insight|Modeler`

   You are presented with an `HPBPI Modeler` dialog, where you enter details of the location of the Model Repository.

2. Enter details of the username and password to connect to the Model Repository. On an new installation, the username is `admin` and the password is `hpbpi`.

   You can modify these credentials using the Tomcat Realm configuration. Chapter 9, Servlet Engine Authentication describes the `tomcat-users.xml` file that you can edit to change the login credentials for the Model Repository.

   You can also modify the details of the Repository Server that you connect to when you start the HPBPI Modeler. This is described in section Changing the Details of the Repository Server on page 138.

3. Click `OK`, and the HPBPI Modeler opens.

4. Select the definition that you want to undeploy from the Navigator pane in the HPBPI Modeler.

5. Select `Undeploy` from the `File` menu and follow the instructions provided.

   The status of the definition changes from `Yes` to `No, but can be deployed`.

The `Summary` Window in the HPBPI Modeler now shows the revised status of the definition.

If you have modified a definition, it is no longer seen as being deployed within the HPBPI Modeler. In this case, you are unable to undeploy it. If you want to know what the deployed status of definitions is (as opposed to a development status), use the Intervention Client to show you the status of the definitions. The Intervention Client is described in the *Business Process Insight Administration Guide*.

You can also view your definitions using the Repository Explorer, see Chapter 5, Repository Explorer.

HPBPI Modeler Log Files on Windows

The HPBPI Modeler log files are called `bia_modeller`*`n`*`_`*`n`*`.log` and are located in the following directory on the Windows system where the HPBPI Modeler is installed:

`HPBPI-install-dir``\data\log`

The log file is a text file that you can open using a text editor. Using the log files to solve problems with your HPBPI system is described in the *Business Process Insight Problem Solving Guide*.

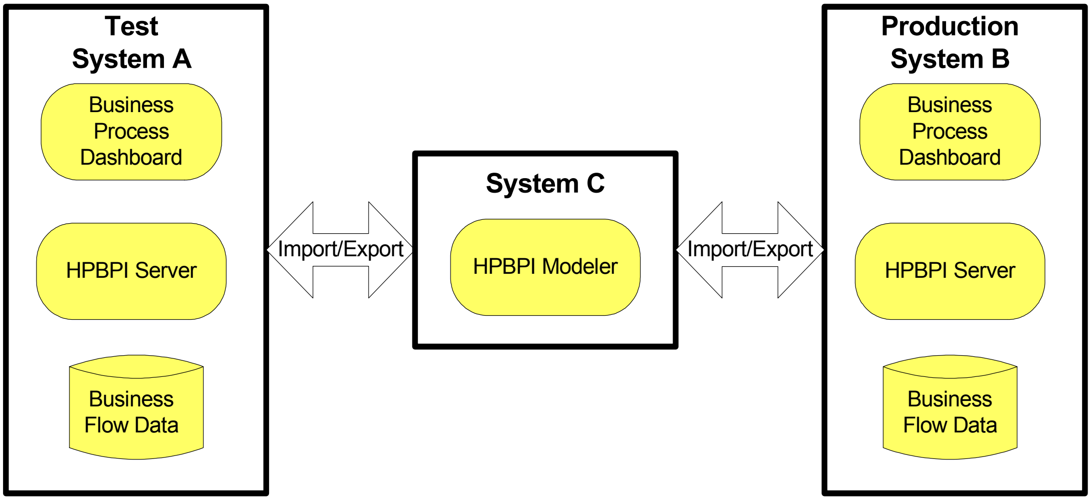
There are no logging settings specifically for the Modeler. The Modeler uses `The Model Repository log level` parameter value for its logging levels. Refer to section Component Configurations - Logging on page 128 for details of setting the logging levels for the Model Repository.

Exporting and Importing Flow Definitions

The HPBPI Modeler has an option to enable you to export and import business flows, including their dependencies (Data, Service and Event definitions) to a `.zip` file that contains XML files. This enables you to create business flows using one Modeler for testing and then move the flows to a production system when they are fully verified. You can also export and import any business process metric definitions that you have defined for your flows as described in Exporting and Importing Metric Definitions on page 151.

The following figure shows an example scenario for using the import and export function of the HPBPI Modeler.

Figure 2 Remote HPBPI Modeler



You use the `File|Import Definitions...` and `File|Export Definition...` menu options from the HPBPI Modeler to export and import your business Flow definitions.

Make sure that you have a naming policy for Flow, Data and Event definitions across your organization to minimize the possibility of a name clash when the business flows are imported for deployment to the Business Impact Engine. You have the option to rename your definitions when you import them; however, you might prefer to have a standard naming scheme.

Exporting Definitions

Use the `File|Export Definition...` option to export the business flows to an XML .zip archive file. When you have exported the file, you can copy the file to the system where the HPBPI Modeler that you want to import the file to is installed. Finally, use the Import option to import the Flow definitions as described in section Importing Definitions on page 145.

Importing Definitions

If you have previously exported a Flow definition and you now want to import it into the HPBPI Modeler, complete the following steps:

1. Select the following option from the `File` menu:

   `Import Definitions...`

2. Navigate, or browse, to the location of the previously exported definition file and click `Open`.

3. Click `Next` to move to the next option. You are presented with a dialog that lists the content of the definition file that you are importing.

   From this dialog, you can:

   — import a listed definition and overwrite any existing definition of the same type that has the same name. If you provide a new name for the definition, the definition is overwritten and renamed.

   — create a new copy of a listed definition. For this option, you provide a new name for the definition in the `New Name` column.

   — choose not to import a listed definition.

   In the case of Service definitions, existing services are referenced automatically.

   When you have tailored your import requirements, click `Import`.

4. Click `Close` to complete the import.

When the Flow definitions are successfully imported they are shown as being undeployed in the HPBPI Modeler. You now need to deploy the imported Flows and their dependencies to the target HPBPI Server system.

Importing BPEL Processes

You can use the HPBPI Modeler to import Business Process Execution Language (BPEL) definitions.

You import the structure of a BPEL definition using the HPBPI Modeler; however, you cannot import the associated BPEL logic. When you have imported the BPEL definition, you can use the structure as the basis of a Business Flow, which you can then modify to suit your reporting and monitoring requirements. You also need to create the Data and Event definitions required to monitor the imported BPEL definition.

Details of importing BPEL definitions, and the options available for tailoring the definitions when importing them are provided in the *Business Process Insight Integration Training Guide - Importing BPEL*. There are also instructions on using the Modeler commands in the HPBPI Modeler Online Help.

Copying Properties Between Data and Event Definitions

Using this option enables you to create properties in your Data definition that match the properties of an Event definition that it subscribes to. Similarly, it can be used to create properties of an Event definition that match those in a Data definition.

Using the copy function saves the need to define identical properties for your Data and Event definitions multiple times. To copy properties from one definition to another definition, complete the following steps:

1. Open the definition whose properties you want to define.

   The properties that you select in the following steps are copied to this definition.

2. Select the Properties tab for the definition.
3. Select the Copy Properties option from the Tools menu.

   You are presented with a Copy Properties dialog that has a tree definition of all the Data and Event definitions currently defined.

4. Select the definition whose properties you want to copy.

   Note that you are not able to copy a property that already exists in the selected definition.

5. Select the properties within the definition that you want to copy. You can use the `Select All` option to select all the properties.
6. Click `OK` and the selected properties are copies to the definition that you opened in step 1.

You have now copied properties from one definition to another definition.

Exporting Definitions to Recover Unsaved Changes

If the HPBPI Modeler loses its connection to the Repository Server for any reason, for example, the HPBPI Server has been shutdown in order to reconfigure a component, you cannot continue and modify your Flow definitions.

If you have made changes to your definitions, but not saved the changes, you have the opportunity to export these changes to a recovery file before closing the HPBPI Modeler.

➤ If you close the HPBPI Modeler without exporting your unsaved changes, the changes are lost.

As soon as the HPBPI Modeler realizes that it has lost its connection to the Repository Server, it presents a dialog for you to enter the name of the file where you want to save your modifications. The next time that you start the HPBPI Modeler and the Modeler successfully connects to the Repository Server, you are presented with a dialog that enables you to select the file where you saved your changes in order to import them back into the Repository Server.

If the HPBPI Modeler loses its connection to the Repository Server, and all your definitions are up to date, you are informed that the HPBPI Modeler must exit, and that no data is lost.

When you have exported the unsaved modifications, the file that is created is the same as the files created using the `Export` option and you can import the file at a later time if required or preferred.

The file used to store the definitions for recovery purposed is not deleted after the recovery process is complete. It is therefore available in the future if you wanted to import your modifications again for any reason.

4 HPBPI Metric Definer Administration

This chapter describes the administration tasks related to the HPBPI Metric Definer. Using the Metric Definer is described in the *Integration Training Guide - Defining Business Metrics*.

Managing the Metric Engine parameters is described in section Component Configurations - Metric Engine on page 59 of this guide and the architecture for the Metric Engine and Metric Definer is described in the *Business Process Insight Reference Guide*.

The following sections describe:

- how to view the log files associated with the business process metric definer, and the Metric Engine; see section HPBPI Metric Definer Log Files on page 150.
- importing and exporting metric definitions; see section Exporting and Importing Metric Definitions on page 151.
- exporting files for the Business Availability Center

Starting and stopping the Metric Definer is described in Chapter 2, HPBPI Component Administration along with other HPBPI Server components; the Metric Engine and the Metric Definer are components of the HPBPI Server.

HPBPI Metric Definer Log Files

There are two sets of log files that you need to be familiar with when managing your business process metrics:

- Metric Engine log files

  The Metric Engine log files are managed in the same way as other HPBPI Server components and are described in section Component Configurations - Logging on page 128.

- Metric Definer log files

  The Metric Definer logs messages to the HPBPI Servlet Engine log file. These messages in the Servlet Engine log file are prefixed with the string `Metric Definer`. The Servlet Engine is considered as a component of the HPBPI Server and section Component Configurations - Logging on page 128 describes the Servlet Engine log file.

Exporting and Importing Metric Definitions

The HPBPI Metric Definer has options that enable you to export and subsequently import the business process metrics, metric thresholds and filters defined for a selected business flow. You might want to export a business process metric definition to enable you to create business process metrics for testing and then move these metrics to a production system when they are fully verified.

These definitions can be exported to a `.zip` archive file, which contains XML files describing the definitions. The `.zip` archive contains a copy of the metric definition and some (but not all) information about the parent Flow definition. The information relating to the Flow definition is that required to uniquely identify the Flow definition that the business process metric applies to. The `.zip` archive file does not contain all the information relating to files that you might have created for custom business process metrics; it contains the custom metric definition, but you need to separately manage and export any SQL you have created for the custom metric definitions.

Be aware that the `.zip` archive created from the Metric Definer is not the same as the `.zip` archive created from the HPBPI Modeler. You cannot interchange the `.zip` archives between the two applications, you must import the archive created from the appropriate application. This is because the `.zip` archive from the Metric Definer contains sufficient information about the business flow to enable the Metric Definer to import the metric; it does not contain a full definition of the business flow.

➤ You cannot import a metric definition until the flow definition that the metric applies to is deployed to the Business Impact Engine. Refer to section Exporting and Importing Flow Definitions on page 144 for details of exporting and importing flow definitions.

Exporting a Business Process Metric

To export one or more business process metrics, use the Export or Export All options from the File menu in the Metric Definer Navigation frame.

When you select a business process metric to export, all the associated metric threshold and filters are also exported. The export is based around the identifier of the business flow that the business process metric is linked to. All the information, including metric thresholds and filters, is exported to XML files, which are then packaged into an overall .zip archive file. The .zip file can then be copied to the destination system in order to be imported.

When you export a business process metric, you also need to make a copy of any additional files that you might have created that are associated with the business process metric; for example, you might have defined a custom metric. In the case of a custom metric, you need to make sure that you have a copy of any SQL files that you have created for the custom metric.

The name of the exported file is based on the Flow name, machine name and time that the metric is exported. This is in order to provide a unique name for the .zip archive file. If any of the elements of this constructed file name contain non-ASCII characters, the elements are excluded from the file name. This is because Internet Explorer allows only .zip archive file names that contain ASCII characters.

Importing a Business Process Metric

Before importing a business process metric, you must have already imported and deployed the business flow that the business process metric is linked to.

In addition, if you have defined custom metrics, and you are moving the definitions to a new machine, make sure that you also copy any files that you have created for the custom metric to the new machine. For example, you might have created SQL files for your custom metric and this SQL file is not exported as part of the .zip file. If you do not copy the files relating to any custom metrics before importing the Metric definition and creating the appropriate stored procedures, the import fails.

To import the business process metric, complete the following steps:

1. Select the following option from the `File` menu on the Metric Definer navigation frame:

   `Import`

   The right-hand pane provides a browse button that you can use to navigate to the file where the file that you want to import is located.

2. Select the file that you want to import and click `Open`.

3. Click the `Import Definitions` button.

   The Metric Definer provides a summary of the metric definitions contained in the zip file and how they will be applied to the flows that are currently defined on your system. You need to check that the metric definitions in the zip file are being applied to the correct flow definitions before importing the file. As an example, you need to make sure that you have not renamed a flow since a metric definition was exported.

4. Select the checkbox next to each set of business process metrics that you want to import from the zip archive file.

5. Click `Import Definitions`.

   The Metric Definer displays a feedback message indicating whether or not it successfully imported the selected metric definition.

6. Click `OK` and the Metric Definer returns to the Metrics and Thresholds page for the flow.

If you import metric definitions which are named identically to metric definitions that are already defined within the Metric Definer, they are imported and renamed (appended with a number), in order to make them unique. Threshold names are not changed as they need only be unique for the Metric definition that they apply to.

Exporting Files to the Business Availability Center

You can export HPBPI Business Flow data into an XML file, which can then be imported into the Business Availability Center as an XML File source adapter. If you want to have the Business Availability Center Dashboard, or a My BAC page, report on HPBPI Business Processes, you need to import an XML entities file into the Business Availability Center with the required XML definitions of the HPBPI entities.

When you create the XML File source adapter within the Business Availability Center, the HPBPI data then becomes available within the Business Availability Center Dashboard to be viewed as Business Processes and BPI Monitors.

The required information is exported as a .zip archive. When you have exported the required information and created the .zip archive, you then need to copy the .zip archive file to the destination system, unpack the archive and select the files that you want to import into the Business Availability Center.

The *Business Process Insight Reference Guide* and the Business Availability Center documentation provide more details of the HPBPI and the Business Availability Center integration, how you select the data that you want to export and why you might want to export Business Flow data.

5 Repository Explorer

This chapter describes the HPBPI Repository Explorer. The Repository Explorer is a Web-based interface that enables you to browse and manage the contents of the Model Repository. The Model Repository holds the data for the business flows that you have defined using the HPBPI Modeler.

For details of creating definitions using the HPBPI Modeler, refer to the *Business Process Insight Integration Training Guide - Modeling Flows*.

This chapter describes the following:

- Repository Explorer Overview on page 156
- Starting and Stopping the Repository Explorer on page 164.
- Exporting Flow Definitions on page 166
- Repository Explorer Log Files on page 170.

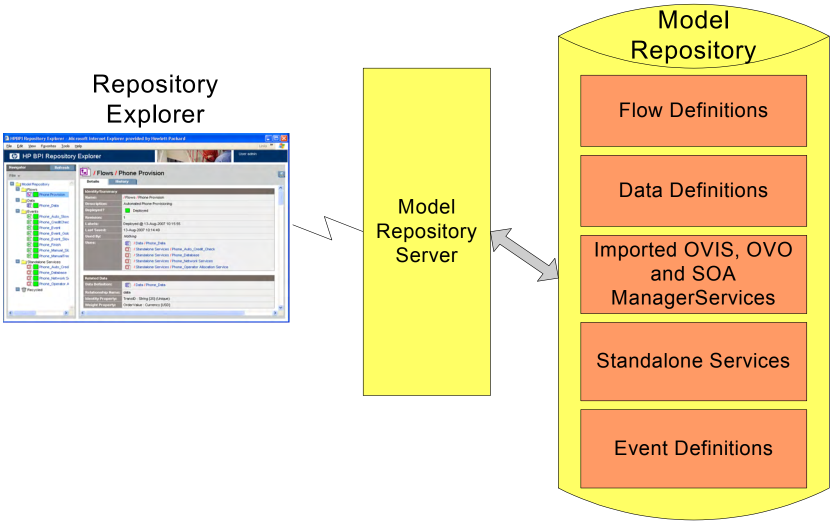
Repository Explorer Overview

Using the HPBPI Modeler, you can create, modify and subsequently deploy a definition. When you have deployed the definition, you might then go on and revise the definition and significantly change it. This means that, unless you have exported the version of the definition that you previously deployed, you no longer have a copy of it within the HPBPI Modeler. You can therefore be in the position of having a deployed flow, but no copy of the source for the flow.

The Repository Explorer enables you to always be able to access the source to versions of flows that you have created and deployed.

The Repository Explorer is a view on the Model Repository data held in the HPBPI database. Access to the data is provided through the Repository Server as shown in Figure 3.

Figure 3 Repository Explorer Architecture



Specifically, the Repository Explorer enables you to:

- browse the definitions that you have created.

  The HPBPI Modeler is optimized for editing and not for browsing and therefore does not present the information relating to Flow, Data, Event and Service definitions in an easy-to-view way. In addition, the Modeler does not enable you to access data relating to superseded versions of definitions. Using the Repository Explorer, you can easily browse the current and earlier revisions of a definition to quickly see how they were defined.

- export the latest version of a definition.

  You might want to do this in order to take a copy of the latest revision of the definitions. The latest version of the definition is exported to a `.zip` file. Exporting the definitions through the Repository Explorer, or the HPBPI Modeler, also ensures that the definitions are consistent within themselves as the definition and all its dependencies are exported.

- export the latest version of all definitions.

  You might want to do this in order to move the definitions from one HPBPI Server to another HPBPI Server, for example, from a development system to a staging system. The latest versions of all the definitions within the Model Repository are exported to a `.zip` file, including all the dependencies.

- export a superseded version of a definition.

  You might want to do this if you have not saved a particular version of a definition, when using the HPBPI Modeler to edit it, and you need to keep a record of it.

- remove definitions.

  You can delete any definition using the Repository Explorer, provided it is not the current, or any superseded, version of the definition that has instances currently running in the Business Impact Engine.

- restore a definition that has been deleted using the HPBPI Modeler or Repository Explorer.

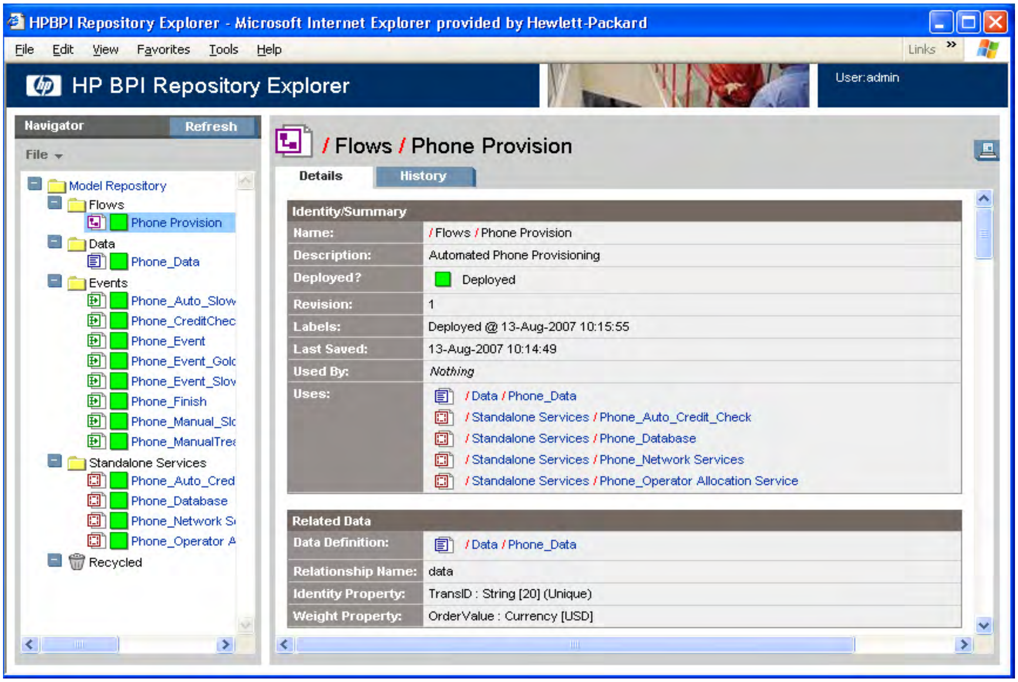
- permanently remove deleted definitions from the recycle folder.

- print definitions.

  There is a print option within the Repository Explorer that enables you to print the information listed in the `Details` or `History` views.

Figure 4 shows an example of the layout of the HPBPI Repository Explorer when you open it in a Browser Window.

Figure 4 Repository Explorer



Identity/Summary	
Name:	/ Flows / Phone Provision
Description:	Automated Phone Provisioning
Deployed?	🟩 Deployed
Revision:	1
Labels:	Deployed @ 13-Aug-2007 10:15:55
Last Saved:	13-Aug-2007 10:14:49
Used By:	*Nothing*
Uses:	/ Data / Phone_Data / Standalone Services / Phone_Auto_Credit_Check / Standalone Services / Phone_Database / Standalone Services / Phone_Network Services / Standalone Services / Phone_Operator Allocation Service

Related Data	
Data Definition:	/ Data / Phone_Data
Relationship Name:	data
Identity Property:	TransID : String [20] (Unique)
Weight Property:	OrderValue : Currency [USD]

The Repository Explorer comprises:

- A left-hand Navigator frame that lists the deployed Flow, Data, Event and Service definitions. The Navigator frame also contains a Recycled folder where definitions that have been deleted from the HPBPI Modeler are listed.

  There are also a number of menu options on the Navigator frame.

- A right-hand frame that displays the appropriate details of the definition that you have selected in the Navigator frame. The right-hand frame includes tabbed pages for Details and History.

  The right-hand frame also contains a print icon, which enables you to print the contents of the right-hand pane.

The features offered by the Repository Explorer are described in more detail in the following sections.

Navigator Frame

The Navigator frame lists all the definitions that have been created in the Model Repository, using the HPBPI Modeler; the deployment status of each definition is also displayed in the Navigator frame. The Repository Explorer uses the same icons as are used within the HPBPI Modeler to show whether a definition is deployed or undeployed.

The following are the menu options available within the Navigator frame:

- `File`
- `Refresh`

  Click the `Refresh` button to refresh the Navigator frame to be up to date with the latest definitions that have been recently entered into the Model Repository using the HPBPI Modeler.

The following are the options are accessible from the `File` menu:

- `Export`

  Enables you to export the selected definition to a zip file; see section Exporting Flow Definitions on page 166.

- `Export All`

  Enables you to export the latest revision of all the definitions in the Navigator frame to a zip file; see section Exporting Flow Definitions on page 166.

- Restore

  This option is available when you select a definition from the Recycled folder. It restores the definition back to its original location, in the Model Repository, at the point where it was deleted in the HPBPI Modeler; see section Restoring a Definition on page 169.

  The Navigator frame also includes a Recycled folder. Under the Recycled folder is a list of definitions that have been deleted from the HPBPI Modeler. When you delete a definition using the HPBPI Modeler the definition is moved to this Recycled folder, which is accessible only from the Repository Explorer.

- Cleanup Recycled

  This option deletes all definitions in the Recycled folder and undeploys definitions that are deployed. If a definition is still in use by the Business Impact Engine, it is not deleted or undeployed. When all instances of these definitions have completed, these entries can then also be removed using the Cleanup Recycled option and the entries no longer appear in the interfaces. The instances of definitions are deleted by the Business Impact Engine using the Engine Instance Cleaner.

When you select a definition in the Navigator frame, the right-hand frame opens at the Details tab; see section Details Tab on page 160. You also have the option of selecting the History tab from the right-hand frame; see section History Tab on page 162.

Details Tab

The Details tab provides a page listing the details of the definition that you have selected in the Navigator frame. Much of the information listed under the Details tab can be determined from the HPBPI Modeler; however, the Repository Explorer provides the information in a structured form, which makes it easier to reference and print. The information listed under the Details tab varies according to type of definition selected in the Navigator frame:

- Flows
- Data
- Events
- OVIS Services

- OVSN Services
- SOA Manager Service Source
- Standalone Services

The following is an overview of all the sections for all the definitions:

- Identity/Summary

  This section appears for all the definitions and lists an overview, or summary, of the definition.This includes information that you have entered relating to the definition using the HPBPI Modeler, and information relating to the revision of the definition.

  This section also lists the revision number of the definition and details of the other definitions that the selected definition uses and is used by.

- Related Data

  This section appears only for a Flow definition and lists details of the Data definition that has been defined as the Related Data definition for the selected Flow definition.

- Flow Diagram

  This section appears only for a Flow definition and shows the flow diagram of the selected flow. It is the same flow diagram as is shown through the HPBPI Modeler.

- Nodes

  This section appears only for a Flow definition and lists details of the individual Nodes that make up the Flow definition, including:

  - Name of the Node
  - Type of Node (Start, End or Activity)
  - Progression Rules (Start and Complete conditions)
  - Details of any operational services that the Node relies on.

- Checked Arcs

  This section appears only for a Flow definition and lists details of the arcs that have been defined as Checked Arcs for the Flow definition.

- Properties

  This section appears for Data and Event definitions and lists details of the Properties that have been defined for the selected definition.

- Subscriptions

  This section appears only for a Data definition and lists details of the Event subscriptions related to the Data definition.

- ToDo List

  This section appears for all the definitions and lists the content of the current to-do list for the selected definition.

History Tab

The History tab provides a page tabulating the details of the revision history related to the selected definition in the Revision History table.

These details include:

- Revision number
- Deployment status
- Name
- Date the revision was created
- Description (from the Description field in the definition)
- Label, which is the date and time that the revision was deployed.

Each entry in the table has a check box which you can select (and clear). When you select the check box for one of the definitions in the table, the options available depend on how many definitions are selected. The Export option is available for single options only. If you select more than one definition, or do not select any definitions, the Export option is not available.

If you want to export a definition on the Revision History table, select the definition, click the `Export` button and then follow the instructions that are presented to you.

There is also a delete option available for each revision of the definition. You cannot delete a definition that is currently being processed by the Business Impact Engine, if you do try to delete such a definition, the Repository Explorer issues a warning message and does not allow the delete action to progress.

You can select any version of a definition to delete, including the current version of a definition. In the case of the current definition, the Repository Explorer issues a warning message for you to confirm the delete action and then continues. Be aware that you cannot recover any definition that is deleted using the Repository Explorer.

Starting and Stopping the Repository Explorer

Before trying to start the HPBPI Repository Explorer, make sure that the `Model Repository` and `Servlet Engine` components are started on the HPBPI Server system. Use the HPBPI Administration Console to start them, as described in section Starting and Stopping the HPBPI Server Components on page 27.

Opening the Repository Explorer

To start the HPBPI Repository Explorer, complete the following steps:

1. Make sure that the `Model Repository` and `Servlet Engine` components are started.
2. Type the following URL into a Web Browser:

   `http://`*hostname*`:44080/ovbpirepositoryexplorer`

   where:

   — *`hostname`* is the fully qualified domain name of the system where the HPBPI Server is installed and running. You can use `localhost` as the hostname if you are starting the Repository Explorer on the system where the server components are installed and running.

   — `44080` is the port number for the Servlet Engine, identified by the `ServletEngine HTTP` port number. Use the port number configured for your system.

   You are prompted for a username and password.
3. Enter details of the username and password to connect to the Model Repository. On a new installation, the username is `admin` and the password is `hpbpi`.

   You can modify these credentials using the Tomcat Realm configuration. Chapter 9, Servlet Engine Authentication describes the `tomcat-users.xml` file that you can edit to change the login credentials for the Model Repository.
4. Click `OK`, and the HPBPI Repository Explorer opens.

Stopping the Repository Explorer

You close the Repository Explorer by closing the browser Window where the Explorer is running.

If you want to increase the level of security for the Repository Explorer pages, you can use the Web Server authentication mechanisms to lock the Web Browser screen after a certain length of time. Chapter 9, Servlet Engine Authentication provides details of the default authentication that is provided for the HPBPI Web interfaces.

Exporting Flow Definitions

The Repository Explorer can be used to export any revision of a definition from the Model Repository.

⚠ Note that this definition is compatible with the Modeler and not the Metric Definer; it does not contain any business process metric information.

You might have multiple versions of a definition where you have deployed the Flow definition multiple times. You can also export the latest version of a definition using the HPBPI Modeler; however, you cannot export earlier revisions of a definition using the Modeler, you can do this only through the Repository Explorer.

In addition to exporting individual versions of a definition, you can also choose to export the latest revision of all definitions from the Model Repository; you might want to do this to move all the latest definitions to another HPBPI Server.

Exporting a Single Definition

To export a definition, complete the following steps:

1. Select the definition that you want to export from the left-hand Navigator pane.

   If you want to export a superseded version of the definition, continue at step 3; otherwise, continue at step 2.

2. Click the `Export` option from the `File` menu in the Navigation pane.

   Continue at step 6.

3. Select the `History` tab in the right-hand pane.
4. Select the revision of the definition that you want to export from the list.
5. Click the `Export` button in the right-hand pane.
6. The right-hand pane lists the definition that will be exported.
7. Click the `Download` button to continue to export the definition.

   You are presented with your browser's `File Download` dialog.

8. Select `Save`, to save the definition as a `.zip` file.

   You are presented with a `Save As` dialog, where you can specify a file name and directory location for the `.zip` file.

9. Enter the details of the file name and click `Save`.

   The file is saved and a `Download complete` dialog is displayed.

10. Click `Close` and the export is complete.

Exporting All Definitions

To export all definitions, complete the following steps:

1. Click the `Export All` Option from the `File` menu in the Navigation pane.
2. The right-hand pane lists the definitions that will be exported.
3. Click the `Download` button to continue to export the definition.

   You are presented with your browser's `File Download` dialog.

4. Select `Save`, to save the definition as a `.zip` file.

   You are presented with a `Save As` dialog, where you can specify a file name and directory location for the `.zip` file.

5. Enter the details of the file name and click `Save`.

   The file is saved and a `Download complete` dialog is displayed.

6. Click `Close` and the export is complete.

Undeploying a Flow Definition

You cannot use the Repository Explorer to undeploy a specific definition. You must use the HPBPI Modeler to do this. You can use the `Cleanup Recycled` option within the Repository Explorer to undeploy and delete all current revisions of definitions held in the `Recycled` folder; see section Navigator Frame on page 159.

Be aware that undeploying an Event definition, which is referenced by a Data definition that has current active instances, means that these data instances might not be able to progress. If this is the case, you can use the Intervention Client to delete the individual Data definition instances.

Restoring a Definition

The Repository Explorer can be used to restore definitions that have been deleted using the HPBPI Modeler. These definitions are listed in the Recycled folder on the Navigator frame.

You can access the `Restore` option from the `File` menu on the Navigator frame. The steps for restoring a previously deleted definition are as follows:

1. Select the definition that you want to restore from the Recycled folder.
2. Select the `Restore` option from the `File` menu in the Navigator frame.

   The definition is restored to the Model Repository and is removed from the Recycled folder in the Repository Explorer. The restored definition appears under the appropriate definition list in the Repository Explorer.

Repository Explorer Log Files

The Repository Explorer uses the Repository Server to access its data and the Repository Server logging is recorded under Model Repository in the Administration Console. Refer to section Component Configurations - Logging on page 128 for details of the logging settings that are used for HPBPI components.

In addition, you can find log messages for the Repository Explorer in the Servlet Engine log file.

6 Notification Server Configuration

This chapter describes the HPBPI Notification Server Web Administration Console and how to use it to configure subscriptions for alerts such as flow impact and metric threshold alerts.

The chapter also describes how to configure email templates and scripts, which can be used when notification alerts are received by the Notification Server.

The Notification Server is the component responsible for sending email alerts and HP Operations Manager messages for events. These messages provide details of the flow impact, metric threshold and out-of-sequence events, plus SLO and SLA email violations. You can receive these alerts through your email client or through an HP BTO Software client according to your requirements.

You also have the option to execute a script when an impact event is received; for example, the script might update a file, or send an SMS message.

The Notification component can be considered in two parts:

1. The server, which is responsible for receiving the impact, SLO and SLA events from the Business Impact Engine and converting them into the appropriate email messages and HP Operations Manager Messages for delivery. The architecture for the Notification Server is described in the Architecture chapter; refer to the *Business Process Insight Reference Guide*.
2. The Notification Server Web Administration Console, which enables you to create subscriptions for users to receive email notifications, SLOs and SLAs. It also enables you to create subscriptions for HP Operations Manager messages and scripts. Using this Web Administration Console is the topic described in this chapter. The Web Administration Console uses Tomcat as its Servlet Engine to manage these Web pages. Tomcat is installed as part of the HPBPI Server.

The Notification Server components are shown in Figure 5 on page 173.

Specifically, this chapter describes:

- Configuring subscriber accounts for email alerts; see section Adding Users for Email Subscriptions on page 174. This includes:
    - subscriber accounts to receive specific flow impact events that are generated by HPBPI.
    - subscriber accounts to receive specific metric threshold events that are generated by HPBPI.
    - subscriber accounts to receive specific out-of-sequence events that are generated by HPBPI.
    - subscriber accounts to receive OVIS SLOs and SLA violation notifications.
- Configuring subscriptions to send messages to HP Operations Manager; see section "Adding HP Operations Manager Message Subscriptions" on page 182.
- Configuring the templates used by the Notification Server to send alerts.

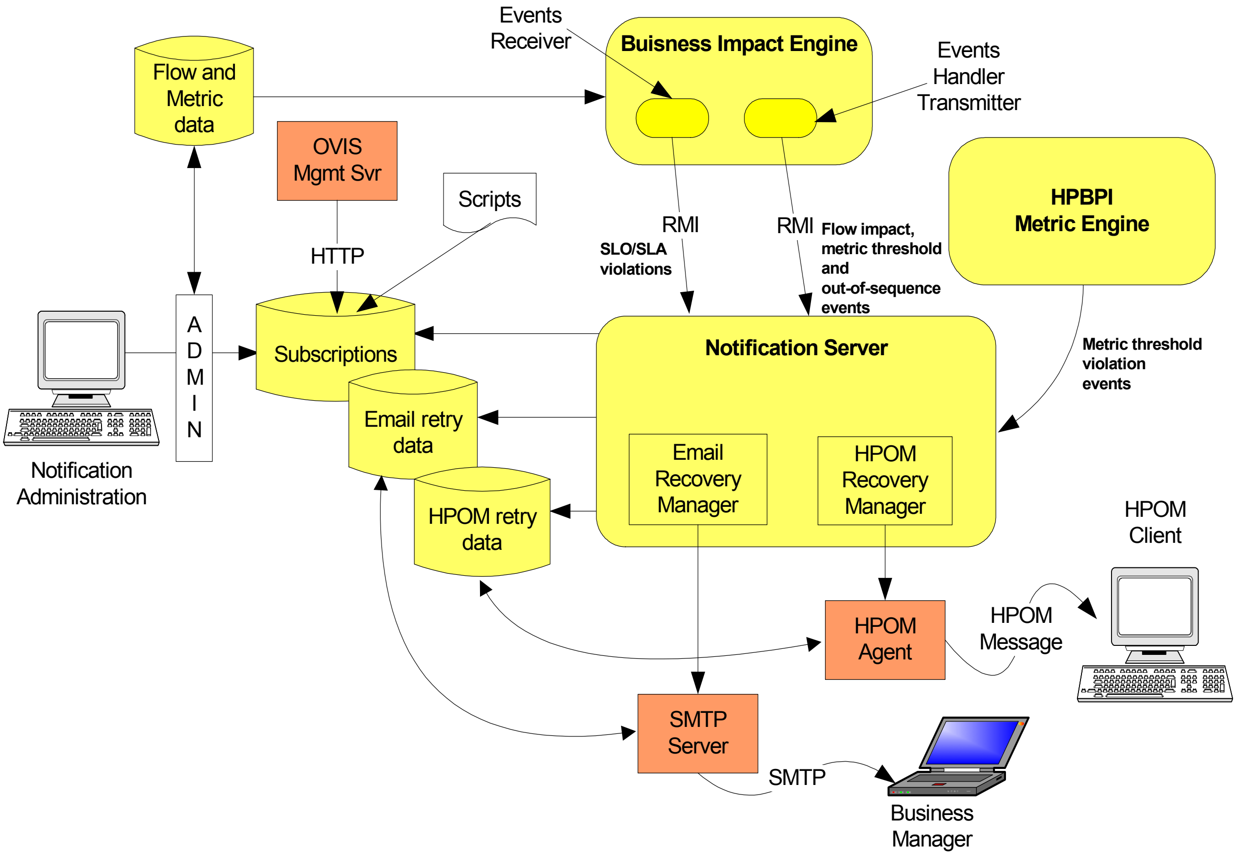
  The Notification Server uses templates to format the email and HP Operations Manager notifications. These templates contain instructions about how to transform the event information such as event name and event time into a format of your choice, subject to the restrictions of the transformation tool you are using. There are two types of templates that you can define to format a notification, these are Velocity templates and XSLT style sheets. Creating new templates for your user subscriptions is described in section Creating Notification Server Templates on page 188.
- Configuring scripts that the Notification Server runs when it receives a specific notification event; see section Creating Scripts on page 206.

You can also change the login username and password for the Notification Server Administration Console; see section Chapter 9, Servlet Engine Authentication.

Follow the instructions in the section appropriate to the task that you want to complete.

The Figure 5 shows an architectural overview of the Notification Server.

Figure 5 Notification Server Architecture



Adding Users for Email Subscriptions

Complete the following steps to configure the Notification Server to send email notifications of impacted flows to a specified email account.

1. Make sure the `Servlet Engine` component is started using the HPBPI Administration Console.

2. Start the HPBPI Notification Server Administration as follows:

   a. Open a new Web browser window

   b. Type the following URL:

      `http://`*`hostname`*`:44080/ovbpinotifyadmin`

      where:

      – *`hostname`* is the fully qualified hostname of the system where the HPBPI Server is running.

      – `44080` is the port number for the Servlet Engine, identified by the `ServletEngine HTTP` port number. Use the port number configured for your system.

      If you are running the Web Browser on the same system as the HPBPI Server, you can use `localhost` in the URL.

   c. You are presented with a dialog to enter the login credentials for the Notification Server Administration Console. Enter a User Name and Password for the Console. By default the User Name is `admin` and the Password is `hpbpi`. Chapter 9, Servlet Engine Authentication describes how to change the username and password credentials for the Notification Server Administration Console.

3. Select `Email Subscriptions` from the menu.

4. Click the `New User` button to add a new user.

   You are presented with a `User Information` screen.

5. On the `User Information` screen:

   a. Enter a `User Name`

      This is a name that you want to assign to the user; it can be any name that identifies the user subscription.

   b. Enter an `Email Address`

      This is the email address for the user named in the previous step. This must be a valid email account.

   c. Click `OK`.

      The new user now has an account on the Notification Server and is listed in the User List with the other users.

You now have the option to:

- add another new user as described in the above steps.
- subscribe the new user account to the events that you want it to receive. To do this refer to section Configuring the Events Received by Notification Server Users on page 176.
- return to the main menu, where you can add more email subscriptions or add HP Operations Manager message subscriptions.
- delete an entry in the User List. To do this select the checkbox next to the `User Name` entry that you want to delete and then click the `Delete` button.
- logout of the Notification Server Administration Console. You do this by closing the Web Browser Window where the console is running.

Configuring the Events Received by Notification Server Users

When you have added the user accounts for the users to receive notifications, you then need to configure these user accounts to subscribe to HPBPI events that you want them to receive. To do this, select the `Subscriptions` link on the Users List screen, under the `Email Subscriptions` column, for the user that you want to configure. This link takes you to the `Email Subscriptions` screen.

You are presented with three options:

- `Flow Subscriptions`; see section "Flow Subscription" on page 176.

  Select this option if you want to add user accounts that you want to configure to subscribe to flow impact, out-of-sequence, or metric threshold alerts.

- `OVIS Service Level Agreement Subscriptions`; see section "OVIS Service Level Agreement Subscription" on page 179.

  This option is available only when you choose to enable OVIS interoperability using the HPBPI Administration Console. Select this option if you want to add accounts to subscribe to SLA alerts.

- `OVIS Service Level Objectives Subscriptions`; see section "OVIS Service Level Objectives Subscription" on page 180.

  This option is available only when you choose to enable OVIS interoperability using the HPBPI Administration Console. Select this option if you want to add accounts to subscribe to SLO alerts.

Flow Subscription

To add a flow impact, out-of-sequence, or metric threshold alert subscription to a user account on the Notification Server, complete the following steps:

1. Select `Flow Subscriptions` from the list of options on the `Email Subscription` screen for the user account that you are modifying.

   You are taken to the `Flow Subscriptions` screen.

2. Click the `New Subscription` button.

   You are taken to the `New Event Subscription` screen where you can add the email subscription details.

3. Select the Event Name and the Flow Name for your new event subscription:

    — Event Name:

      You can select from: `Flow Impacted` events, `Flow Metric Threshold` Events, `Flow Out Of Sequence` events, or `All` events. If you select `All` you subscribe to all types of event.

    — Flow Name:

      You can select all flow names to report on, or you can choose a specific flow that is deployed to the Business Impact Engine. There is a drop down list provided, which lists all the deployed flows for you to select from, including the `All` option.

4. Click `Next` to move to the next screen, where you select the Minimum Severity Level and the Template for your new subscription:

    — Minimum Severity Level:

      This is the minimum severity level for which you want to receive email notifications; for example, if you select Critical you will receive email notifications only for alerts that are critical. If you select Minor, you receive email notifications for Minor, Warning, Major and Critical alerts.

    — Template:

      Select the template that is appropriate to your Event Name, for example, you might have created your own template for the event. If you have selected `Flow Impacted` as the Event Name, choose the `EMAIL-FlowImpactedDefault.vm` template. If you have selected `Flow Metric Threshold` as the Event Name, choose the `EMAIL-MetricThresholdDefault.vm` template. If you have selected `Flow Out of Sequence`, choose the

EMAIL-OutOfSequenceDefault.vm template. If you have selected All for the Event Name, choose the EMAIL-GenericDefault.vm template, as it provides the most detailed information.

➤ If you create two subscriptions for the same event type, each with different templates, HPBPI uses the template from the more restrictive subscription to format the alert. For example, if you create a subscription for all flow impact events using one template and a subscription for a specific flow impact event using a different template; the template for the specific flow impact event is the one used to format the alert.

5. Click OK to commit the subscription to the administration database.

   The new Event Subscription is now added to the list of Flow Subscriptions for the user.

You now have the option to:

- add another subscription as described above.
- return to the Email Subscription page; click the Email Subs button or link.
- delete an entry in the subscription list. To do this select the checkbox next to the entry that you want to delete and then click the Delete button.
- logout of the Notification Server Administration Console. You do this by closing the Web Browser Window where the console is running.

OVIS Service Level Agreement Subscription

This option is available when you choose to enable OVIS interoperability.

To add an SLA event subscription to a user account on the Notification Server, complete the following steps:

1. Select OVIS Service Level Agreement Subscriptions from the list of options on the Email Subscriptions screen for the user account that you are modifying.

   You are taken to the OVIS SLA Subscription screen.

2. Click the New Subscription button.

   You are taken to the OVIS SLA Subscriptions Details screen where you are asked to select the customer for the SLA subscription. The Customer details are those defined in the OVIS database.

   ➤ The Notification Server presents only those OVIS Customers that have Service Level Agreements defined. You can use the Reload OVIS Config option to update the list.

3. Select one of the defined customers, or select All if you want to filter for all SLA violations defined in OVIS.
4. Click OK to move to the next screen, where you are asked to select a service level agreement.
5. Select an SLA by name, or select All if you want the user to receive alerts for all SLA violations.
6. Click OK to move the next screen where you can select a template for the email message that contains the details of the SLA violation.
7. Select a template that is appropriate to the SLA subscription, for example, the default SLA template (EMAIL-OVIS-SLA-Default.vm), or select a custom template, if you have created one.
8. Click OK to complete the subscription and move to the next screen that lists the SLA subscriptions for the user account that you are modifying.

You now have the option to:

- add another OVIS SLA Subscription as described above.
- return to the Email Subscriptions page, in which case, click the `Email Subs` button, or link.
- delete an entry in the OVIS SLA Subscription list. To do this select the checkbox next the entry that you want to delete and then click the `Delete` button.
- logout of the Notification Server Administration Console. You do this by closing the Web Browser Window where the console is running.

OVIS Service Level Objectives Subscription

To add an SLO subscription to a user account on the Notification Server, complete the following steps:

1. Select `OVIS Service Level Objective Subscriptions` from the list of options on the `Email Subscriptions` screen for the user account that you are modifying.

   You are taken to the `OVIS SLO Subscriptions` screen.

2. Click the `New Subscription` button.

   You are taken to the `OVIS SLO Subscription Details` screen where you are asked to select the customer for the SLO subscription. The Customer details are those defined in the OVIS database.

3. Select one of the defined customers, or select `All` if you want to filter for all SLO violations defined in OVIS.
4. Click `OK` to move to the next screen, where you are asked to select a service group.
5. Select the Service Group name from the drop-down list provided, or select `All` if you want to filter on all OVIS Service Groups.
6. Click `OK` to move to the next screen, where you are asked to select an objective metric for the SLO.
7. Select an objective metric name from the drop-down list, or select `All` to filter on all objective metrics.
8. Click `OK` to move to the next screen.

9. Select a template that is appropriate to SLO subscription, for example, the default SLO template (`EMAIL-OVIS-SLO-Default.vm`), or select a custom template, if you have created one.

10. Click `OK` to move to the next screen that lists the SLO subscriptions for the user.

You now have the option to:

- add another OVIS SLO Subscription as described above.
- return to the Email Subscription page; click the `Email Subs` button or link.
- delete an entry in the OVIS SLO Subscription list. To do this select the checkbox next the entry that you want to delete and then click the `Delete` button.
- logout of the Notification Server Administration Console. You do this by closing the Web Browser Window where the console is running.

Adding HP Operations Manager Message Subscriptions

The Notification Server uses Velocity and XSLT templates to create HP Operations Manager messages before forwarding the message to the HP Operations Manager agent.

The HP Operations Manager agent then applies one or more templates to filter out unwanted messages and to make any additional changes to the HP Operations Manager message format.

Before configuring HPBPI to send events to HP Operations Manager, ensure that the HP Operations Manager Agent and Message Interface template are installed and set up on the system where HPBPI and the Notification Server are installed. Refer to the *VantagePoint Operations for Unix Administrator's Reference Guide*.

Complete the following steps to configure the Notification Server to send HP Operations Manager message notifications to the HP Operations Manager Agent for flow impact, out-of-sequence or metric threshold notification events:

1. Make sure the `Servlet Engine` component is started.
2. Start the HPBPI Notification Server Administration as follows:
    a. Open a new Web browser window
    b. Type the following URL:

       `http://`*`hostname`*`:44080/ovbpinotifyadmin/index.jsp`

       – where *`hostname`* is the fully qualified hostname of the system where the HPBPI Server is running.

       – `44080` is the port number for the Servlet Engine, identified by the `ServletEngine HTTP` port number. Use the port number configured for your system.

       If you are running the Web Browser on the same system as the HPBPI Server, you can use `localhost` in the URL.

    c. You are presented with a dialog to enter the login credentials for the Notification Server Administration Console. Enter a User Name and Password for the Console. By default the User Name is `admin` and the

Password is hpbpi. Chapter 9, Servlet Engine Authentication describes how to change the username and password credentials for the Notification Server Administration Console.

3. Select HP Operations Manager Message Subscriptions from the menu.

   You are taken to the HP Operations Manager Subscriptions screen.

4. Click the New Subscription button.

   You are presented with the New Event Subscription Screen where you can enter the subscription details for the HP Operations Manager messages.

5. From the New Event Subscription screen, select the Event Name and the Flow Name for your new event subscription:

   — Event Name

     You can select to filter for Flow Impacted events, Flow Metric Threshold events, Flow Out Of Sequence events, or All events. If you select All, you receive HP Operations Manager impact alerts for all HPBPI notification events.

   — Flow Name

     You can select all flow names to report on, or you can choose a specific flow that you have defined. There is a drop down list provided, which lists all the deployed flows for you to select from, including the All option.

6. Click Next to move to the next screen, where you select the Minimum Severity Level and the Template for your new subscription:

   — Minimum Severity Level

     This is the minimum severity level for which you want to receive email notifications; for example, if you select Critical you will receive email notifications only for alerts that are critical. If you select Minor, you receive email notifications for Minor, Warning, Major and Critical alerts.

   — Template

     Select the template that is appropriate to your Event Name, for example you might have created a custom template. If you have selected Flow Impacted as the Event Name, choose the OVO-FlowImpactDefault.vm template. If you have selected Flow

Metric Threshold as the Event Name, choose the OVO-MetricThresholdDefault.vm template. If you have selected Flow Out of Sequence, choose the template for the OVO-OutOfSequence.vm for the template. If you have selected All for the Event Name, choose the OVO-GenericDefault.vm template, as it provides the most detailed information.

➤ If you create two subscriptions for the same HP Operations Manager message subscription, each with different templates, HPBPI uses the template from the more restrictive subscription to format the alert. For example, if you create a subscription for all flow impact events using one template and a subscription for a specific flow impact event using a different template; the template for the specific flow impact event is the one used to format the alert.

7. Click OK to commit the subscription to the Notification Server Administration database.

   The new HP Operations Manager Message Subscription is now added to the list of HP Operations Manager Subscriptions.

You now have the option to:

- add another HP Operations Manager Message Subscription as described above.
- return to the Main Menu; click the Menu button, or link.
- delete an entry in the HP Operations Manager Message Subscription list. To do this select the checkbox next the entry that you want to delete and then click the Delete button.
- logout of the Notification Server Administration Console. You do this by closing the Web Browser Window where the console is running.

You can configure the template set used by HP Operations Manager and you can filter the events received through HP Operations Manager by customizing the HP Operations Manager Message Interface template. Details of how to do this are provided in the HP Operations Manager documentation.

Script Subscription

You can configure the Notification Server to run a script when it receives specific notification events. Section "Creating Scripts" on page 206 describes how you can create these scripts and provides details of the variables that you can include in your scripts to report details of the alert as required.

This section describes how to configure the Notification Server to run the scripts for specific notification alerts that it processes for flow impact, out-of-sequence or metric threshold notification events:

1. Make sure the `Servlet Engine` component is started.
2. Start the HPBPI Notification Server Administration as follows:
   a. Open a new Web browser window
   b. Type the following URL:

      `http://`*`hostname`*`:44080/ovbpinotifyadmin/index.jsp`

      where:

      – *`hostname`* is the fully qualified hostname of the system where the HPBPI Server is running.

      – `44080` is the port number for the Servlet Engine, identified by the `ServletEngine HTTP` port number. Use the port number configured for your system.

      If you are running the Web Browser on the same system as the HPBPI Server, you can use `localhost` in the URL.

   c. You are presented with a dialog to enter the login credentials for the Notification Server Administration Console. Enter a User Name and Password for the Console. By default the User Name is `admin` and the Password is `hpbpi`. Chapter 9, Servlet Engine Authentication describes how to change the username and password credentials for the Notification Server Administration Console.
3. Select `Script Subscriptions` from the menu.

   You are taken to the `Script Subscriptions` screen.

4. Click the `New Subscription` button.

   You are presented with a `New Subscription` screen where you can enter the subscription details for the script.

   If there are no scripts defined, the `New Subscriptions` button is disabled and you cannot access the `Script Subscription` screen.

5. From the `New Script Subscription` screen, enter the following details:

   — Event Name

     You can select to filter for `Flow Impacted` events, `Flow Metric Threshold` events, `Flow Out Of Sequence` events, or `All` events. If you select `All`, you receive HP Operations Manager impact alerts for all HPBPI notification events.

   — Flow Name

     You can select all flow names to report on, or you can choose a specific flow that you have defined. There is a drop down list provided, which lists all the deployed flows for you to select from, including the `All` option.

6. Click Next to move to the next screen where you select the `Minimum Severity Level` and the `Script Name`:

   — Minimum Severity Level

     This is the minimum severity level for which you want to receive email notifications; for example, if you select Critical you will receive email notifications only for alerts that are critical. If you select Minor, you receive email notifications for Minor, Warning, Major and Critical alerts.

   — Script Name

     Select the script that you want the Notification Server to run when it received a notification event for the specified event and flow. All the scripts that you have defined and saved in the `notify-scripts` are presented in the drop-down list for you to select from.

7. Click OK to commit the subscription to the administration database.

   The new Script Subscription is added to the list of Script Message Subscriptions.

You now have the option to:

- add another New Script Subscription as described above.
- return to the Main Menu; click the `Menu` button, or link.
- delete an entry in the Script Subscription list. To do this select the checkbox next the entry that you want to delete and then click the `Delete` button.
- logout of the Notification Server Administration Console. You do this by closing the Web Browser Window where the console is running.

Creating Notification Server Templates

The Notification Server uses templates to format email messages and messages sent to HP Operations Manager. This template contains instructions on how to format event information, for example, event name and event time. You might want customize an email template if you want to generate email messages for different locales.

This section describes how HPBPI uses these templates and some guidance in how you can modify them. You need to be familiar with either XSLT, or Velocity to make the changes; XSLT is a standard for transforming XML documents. It is up to you which format you choose.

The Notification Server uses the Apache Velocity template engine to process Velocity templates. There is a user guide available with the template engine that provides details of creating and using Velocity templates. Details of Velocity templates, including a user guide for creating them, can be found at the following location:

`http://jakarta.apache.org/velocity/`

You configure the template used for each event when you create the event subscription through the Notification Server Web Administration Console. The Notification Server then uses the template that you have defined to format all notifications related to the specified event.

Creating Email Templates

You create a custom email template for a specific event. These custom email templates must have a file name that starts with the string `EMAIL-` and a file extension of `.vm` for Velocity templates and `.xsl` for XSLT templates.

The Notification Server reads the custom templates from the following directories:

- *HPBPI-install-dir*`/data/conf/bia/notify-templates/flows`
- *HPBPI-install-dir*`/data/conf/bia/notify-templates/slos`
- *HPBPI-install-dir*`/data/conf/bia/notify-templates/slas`

If you want to modify these templates, make a copy of the appropriate example template from the files in the following directory:

HPBPI-install-dir`/examples/bia/NotifSvrTemplates`

Make sure that you add your template to the correct directory as the Notification Server Administration Console presents the templates in the context of the selection, based on the contents of the above directories. For example, when creating an SLA subscription, you are offered the list of templates from the following directory:

HPBPI-install-dir/data/conf/bia/notify-templates/slas

You also need to make sure that you copy the template from the examples directory to the data directory as the examples directory is updated and files replaced if you reinstall HPBPI.

The Notification Server uses an email template to create XML that describes an email. Then the notification server uses the XML to create an email.

The email XML is very simple; it contains a subject, content type, and body. The actual schema is described in the file email.xsd, which is located at:

HPBPI-install-dir/misc/bia

The Notification Server applies the Velocity or XSLT templates that you define to an alert from the Business Impact Engine. From this the Notification Server is able to create an email XML document based on your templates.

HP Operations Manager Templates

The Notification Server uses a template to create XML that describes the HP Operations Manager message. It then uses this XML to create an HP Operations Manager message.

You configure the templates that are used to send HP Operations Manager messages through the Notification Server Administration Console.

Custom HP Operations Manager template files must start with the string OVO- and a file extension of .vm for Velocity templates and .xsl for XSLT templates. Store custom templates in the following directory:

HPBPI-install-dir/data/conf/bia/notify-templates/flows

The following is an example of an HP Operations Manager message XML document:

```
<?xml version="1.0" encoding="UTF-8"?>
<OVOMessage xmlns:xsi="http://www.w3.org/2001/XMLSchema
   instance"xsi:noNamespaceSchemaLocation="ovomessage.xsd" >
<Severity>critical</Severity>
<Application>HPBPI application</Application>
<Object>FLOW_IMPACT or FLOW_OUT_OF_SEQUENCE</Object>
<MessageGroup>NOTIFICATIONS</MessageGroup>
<MessageText>A FLOW_OUT_OF_SEQUENCE alert occurred at 7.10pm.</
MessageText>
<Option>example_option_variable_1=this is an example option
value</Option>
<Option>example_option_variable_2=this is another example option
value</Option>
</OVOMessage>
```

The example XML document contains an element for each message attribute:

- message severity
- application
- object
- message group
- message text

Optionally, the XML document, can contain message options. An option has the form *variable=value*. Refer to the *VantagePoint Operations for Unix Administrator's Reference Volume I* for more information about HP Operations Manager messages. The notification server applies Velocity and XSLT templates to an alert from the Business Impact Engine to create an HP Operations Manager message XML document.

The schema for the message is available at the following location:

`*bpi-install-dir*/misc/bia/ovomessage.xsd`

Velocity Templates

This section describes the list of the HPBPI alert methods that have been defined for use within your Velocity template. An alert variable in the Velocity context contains information from an Business Impact Engine event.

A Velocity template is a file that contains the XML describing an email or HP Operations Manager message.

In addition to text, the file can contain Velocity formatting directives. The following is an example of an email Velocity template:

```
<?xml version="1.0" encoding="UTF-8"?>
<Email xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
    xsi:noNamespaceSchemaLocation='email.xsd'>
<Subject>
$alert.getSeverity() $alert.getEventName() Alert
</Subject>
<ContentType>text/plain</ContentType>
<Body>
A "$alert.getEventName()" alert has occurred.

Severity: $alert.getSeverity()
Event Group: $alert.getEventGroup()
Event Name: $alert.getEventName()
Event Time: $alert.getEventTime()
Flow Name: $alert.getFlowName()
Flow Guid: $alert.getFlowGuid()
</Body>
</Email
```

A formatting directive starts with a hash (#) or a dollar ($); these formatting directives are described more fully in the *Velocity Users' Guide*. A formatting directive is used to insert event information into the XML document. In the case of HPBPI this event information is provided through the alert variables.

The `$alert` variable contains information from a Business Impact Engine event, for example, the service name for the event. The `$alert` variable contains methods such as `getFlowName()` and `getServiceName()`, which can be used to insert the event flow name and the event service name into the XML. (`getServiceName()` inserts the service name).

If the Velocity templates that you create contain non-ASCII characters, you must encode and save the template as UTF8. This UTF8 needs to be created without a byte order mark (BOM). BOMs are added automatically by some Windows editors; for example, Notepad and Wordpad. Choose an editor that enables you to exclude the BOM.

If you include a BOM in your template encoding, it is not recognized as an XML file.

When the Notification Server generates an email notification message, it adds data to the message, where the display language is determined by the locale of the HPBPI Server. If your email recipients are not in the same locale as the

HPBPI Server, you can use the Velocity templates to specify the locale for the language used for the email notifications. This enables you to provide email notifications in several languages if required.

The java Locale() class is used to identify the language string used within the alert methods. The definition for the Locale() class can be found at the following URL:

http://java.sun.com/j2se/1.4.2/docs/api/java/util/Locale.html

The Locale() class takes a parameter comprising the language code and the country code as described in the definition for the Locale() class; for example: ko_KR for Korea and en_CA English-speaking Canadian.

Methods for All Alerts

The following table lists the alert methods that can be used for all alerts: Flow Impact, Flow Out-Of-Sequence, Business Metric, Metric Threshold, SLO and SLA.

Table 47 Velocity Template Alert Methods for all Alerts

Alert Method Name	Alert Method Description
$alert.getVersion()	Returns a STRING containing the version number for the HPBPI event.
$alert.getSeverity(*locale*)	Returns a STRING containing the severity of the HPBPI event. Possible values are `Critical`, `Major`, `Minor`, `Warning`, and `Normal`. The language used for these values is determined by the locale for the HPBPI Server or, if specified, the locale of the country and language specified in *locale*, for example: • $alert.getSeverity("ja"), for a Japanese locale • $alert.getSeverity("ko"), for a Korean locale The value returned is also used to determine the color of the text in the email notification.
$alert.getEventGroup()	Returns a STRING containing the name of the event group for the HPBPI event, that is, `NOTIFICATIONS`.

Table 47 Velocity Template Alert Methods for all Alerts

Alert Method Name	Alert Method Description
$alert.getEventName()	Returns a STRING containing the name of the HPBPI event, that is, `FLOW_IMPACT`, `FLOW_METRIC_THRESHOLD_ALERT` or `FLOW_OUT_OF_SEQUENCE`. In the case of an SLO, this is the objective identifier. In the case of an SLA, this is the SLA identifier.
$alert.getEventTime(*locale*)	Returns a STRING containing the time of the HPBPI event, formatted using the locale for the HPBPI Server or, if specified, the locale of the country and language specified in *`locale`*.
$alert.getEventDataListSize()	Returns the number of event data items in the HPBPI event.
$alert.getEventDataNames()	Returns an ARRAYLIST of the names of the event data items in the HPBPI event.
$alert.getEventDataByName (*`Stringname, locale`*)	Returns a STRING containing the value of the data item with the given name using the locale for the HPBPI Server or, if specified, the locale of the country and language specified in *`locale`*.
$alert.getEventDataByIndex (*`index, locale`*)	Returns a STRING containing the value of the data item located at the supplied index using the locale for the HPBPI Server or, if specified, the locale of the country and language specified in *`locale`*.

Methods Specific to Flow Impact Alerts

The following table lists the alert methods that can be used for Flow Impact alerts. Using any of these methods for other alerts results in a message containing erroneous data.

Table 48 Velocity Template Alert Methods for Flow Impact Alerts Only

Alert Method Name	Alert Method Description
$alert.getFlowName()	Returns a STRING containing the flow name for the HPBPI event.
$alert.getFlowGuid()	Returns a STRING containing GUID for the flow definition in the HPBPI event.
$alert.getServiceName()	Returns a STRING containing the name of the service for the HPBPI event.
$alert.getServiceGuid()	Returns a STRING containing the service GUID for the HPBPI event.
$alert.getFlowNodeString()	Returns a STRING containing node names for the HPBPI event. An example of a string returned by this method is: `Bill Customer`, `Check Stock`, `Remove Item`.
$alert.getFlowNodeList()	Returns an ARRAYLIST of node names in the HPBPI event. You can use the methods defined in the ArrayList class. For example, you can access the first element of an array list named flowNodeList using `$flowNodeList.get(0)`
$alert.getFlowNodeListSize()	Returns the number of elements in the flow node list in the HPBPI event.
$alert.getRootCause()	Returns a STRING containing the labels of the root cause services in the HPBPI event. (Note that this string cannot be localized.) An example of a string returned by this method is: `Oracle Financial System-&gt;Oracle System Europe-&gt;Network Resources`

Table 48 Velocity Template Alert Methods for Flow Impact Alerts Only

Alert Method Name	Alert Method Description
$alert.getFlowNodeStatusList()	Returns an ARRAYLIST of the status of the Nodes in the HPBPI event. You can use the methods defined in the ArrayList class. For example, you can access the first element of an array list namesFlowNodeStatusList using `$flowNodeStatusList.get(0).`
$alert.getFlowNodeStatusString (*locale*)	Returns a STRING containing the labels for the node status in the HPBPI event. For example, `Critical`, `Major`, `Minor`, `Warning` and `Normal`. These status correspond to the HP Operations Manager severity levels. The language used for these values is determined by the locale for the HPBPI Server or, if specified, the locale of the country and language specified in *`locale`*, for example: • $alert.getFlowNodeStatusString("ja"), for a Japanese locale • $alert.getFlowNodeStatusString("ko"), for a Korean locale
$alert.getFlowNodeStatusList Size()	Returns the number of elements in the flow node status list in the HPBPI event.

Methods Specific to Out-of-Sequence Alerts

The following table lists the methods for out-of-sequence events only. Using any of these methods for other alerts results in a message containing erroneous data.

Table 49 Velocity Template Methods for Out-of-Sequence Alerts Only

Alert Method Name	Alert Method Description
$alert.getFlowName()	Returns a STRING containing the flow name for the HPBPI event.
$alert.getFlowGuid()	Returns a STRING containing GUID for the flow definition in the HPBPI event.
$alert.getFlowInstanceGuid()	Returns a STRING containing GUID for the flow instance in the HPBPI event.
$alert.getSourceNode()	Returns a STRING containing the name of the node that is the source node for the out-of sequence alert.
$alert.getDestinationNode()	Returns a STRING containing the name of the node that is the destination node for the out-of sequence alert.
$alert.getFlowIdentifier()	Returns a STRING containing the identifier for the impacted flow instance.

Methods Specific to Metric and Threshold Alerts

The following table lists the metric and threshold alert notification events only. Using any of these methods for other alerts results in a message containing erroneous data.

Table 50 Velocity Template Methods for Metric and Threshold Alerts Only

<table>
<tr><th>Alert Method Name</th><th>Alert Method Description</th></tr>
<tr><td>$alert.getThresholdAlertGuid()</td><td>Returns a STRING containing the internal unique identifier (GUID) of the threshold alert event raised for a specific threshold violation.</td></tr>
<tr><td colspan="2">The following six methods are duration periods that correspond to the Threshold alert notification polling interval (seconds) as set in the Administration Console for Metric Engine Threshold Alert Notification Settings.
It is called the actual period because it can differ from the requested configuration interval according to the other priorities that the machine, and the JVM, have at the times the requests to start and stop monitoring are made.</td></tr>
<tr><td>getThresholdAlertNotificationPeriodDurationHours()</td><td>Returns a STRING containing the duration of actual alert notification period in terms of hours.</td></tr>
<tr><td>getThresholdAlertNotificationPeriodDurationHours(locale)</td><td>Returns a STRING containing the duration of actual alert notification period in terms of hours, using the locale for the HPBPI Server or, if specified, the locale of the country and language specified in locale.</td></tr>
<tr><td>getThresholdAlertNotificationPeriodDurationMinutes()</td><td>Returns a STRING containing the duration of actual alert notification period in terms of minutes.</td></tr>
<tr><td>getThresholdAlertNotificationPeriodDurationMinutes(locale)</td><td>Returns a STRING containing the duration of actual alert notification period in terms of minutes, using the locale for the HPBPI Server or, if specified, the locale of the country and language specified in locale.</td></tr>
<tr><td>getThresholdAlertNotificationPeriodDurationSeconds()</td><td>Returns a STRING containing the duration of actual alert notification period in terms of seconds.</td></tr>
</table>

Table 50 **Velocity Template Methods for Metric and Threshold Alerts Only**

Alert Method Name	Alert Method Description
getThresholdAlertNotificationPeriodDurationSeconds(*locale)*	Returns a STRING containing the duration of actual alert notification period in terms of seconds, using the locale for the HPBPI Server or, if specified, the locale of the country and language specified in `locale`.
$alert.getThresholdAlertNotificationPeriodStart()	Returns a STRING containing the start time of the `Threshold Alert Notification` period. Start time is the time from and including the time specified.
$alert.getThresholdAlertNotificationPeriodEnd()	Returns a STRING containing the end time of the `Threshold Alert Notification` period. End time is the time up to but not including the time specified.
$alert.getThresholdAlertStatusChangeTime()	Returns a STRING containing the time that the `Metric_Fact_Alerts` table was updated with the Threshold alert notification.
$alert.getThresholdAlertRaisedTime()	Returns a STRING containing the time that the event notification was sent to the Notification Server.
$alert.getThresholdGuid()	Returns a STRING containing the internal unique identifier (GUID) for the metric threshold defined for the business process metric in the HPBPI event.
$alert.getThresholdMessage()	Returns a STRING containing the metric threshold message in the HPBPI event. This is the message that you have defined to be displayed when the event is triggered.
$alert.getThresholdName()	Returns a STRING containing metric threshold name in the HPBPI event.

Table 50 **Velocity Template Methods for Metric and Threshold Alerts Only**

Alert Method Name	Alert Method Description
$alert.getThresholdNumber NotificationsInPeriod()	Returns a STRING containing the total number of notifications sent for the threshold in the `Threshold alert notification polling interval`.
$alert.getThresholdNumber AlertsInPeriod()	Returns a STRING containing the total number of threshold alerts generated for the threshold in the `Threshold alert notification polling interval`. The number of threshold alerts generated might be different to the number of notifications sent, depending on how you have configured the threshold alert notification settings.
$alert.getThresholdNumber NormalAlertsInPeriod()	Returns a STRING containing the total number of Normal threshold alerts generated for the threshold in the `Threshold alert notification polling interval`. This number is a subset of `getThresholdNumber AlertsInPeriod()`.
$alert.getThresholdNumber WarningAlertsInPeriod()	Returns a STRING containing the total number of Warning threshold alerts generated for the threshold in the `Threshold alert notification polling interval`. This number is a subset of `getThresholdNumber AlertsInPeriod()`.
$alert.getThresholdNumber MinorAlertsInPeriod()	Returns a STRING containing the total number of Minor threshold alerts generated for the threshold in the `Threshold alert notification polling interval`. This number is a subset of `getThresholdNumber AlertsInPeriod()`.

Table 50 Velocity Template Methods for Metric and Threshold Alerts Only

Alert Method Name	Alert Method Description
$alert.getThresholdNumber MajorAlertsInPeriod()	Returns a STRING containing the total number of Major threshold alerts generated for the threshold in the `Threshold alert notification polling interval`. This number is a subset of `getThresholdNumber AlertsInPeriod()`.
$alert.getThresholdNumber CriticalAlertsInPeriod()	Returns a STRING containing the total number of Critical threshold alerts generated for the threshold in the `Threshold alert notification polling interval`. This number is a subset of `getThresholdNumber AlertsInPeriod()`.
$alert.getFlowMetricGuid()	Returns a STRING containing the business process metric GUID in the HPBPI event.
$alert.getFlowMetricName()	Returns a STRING containing the business process metric name in the HPBPI event.
$alert.getFlowMetricValue()	Returns a STRING containing the business process metric value in the HPBPI event.

Methods Specific to OVIS SLO Violation Methods

The following table lists the OVIS SLO Violation methods that can be used for HPBPI. Using any of these methods for other alerts results in a message containing erroneous data.

Table 51 Velocity Template Methods for SLO Violations

SLO Method Name	SLO Method Description
$alert.getOvisCustomer()	Returns a STRING containing the name of the OVIS Customer defined in the OVIS Configuration Manager that triggered the violation.
$alert.getOvisServiceGroup()	Returns a STRING containing the name of the OVIS Service Group defined in the OVIS Configuration Manager that triggered the violation.
$alert.getOvisHost()	Returns a STRING containing the name of the host that OVIS is monitoring and that triggered the violation; for example, the name of a Web site.
$alert.getOvisProbe()	Returns a STRING containing the location of the probe that relates to the violation.
$alert.getOvisViolationTime (*locale*)	Returns a STRING containing the date and time that the violation was reported within OVIS. The date and time are displayed in the locale of the HPBPI Server or, if specified, the locale of the country and language specified in *locale*
$alert.getOvisObjectiveName()	Returns a STRING containing the name of the SLO metric relating to the violation.
$alert.getOvisObjectiveId()	Returns a STRING containing the unique identifier for the SLO metric relating to the violation.
$alert.getOvisObjective Operation()	Returns a STRING containing the arithmetical operation that is defined for the SLO that has been violated.

Table 51 Velocity Template Methods for SLO Violations

SLO Method Name	SLO Method Description
$alert.getOvisObjectiveValue()	Returns a STRING that is the metric value recorded for the SLO that has been violated.
$alert.getOvisObjective Threshold()	Returns a STRING that is the metric threshold value assigned to the SLO that has been violated.

Methods Specific to OVIS SLA Violations

The following table lists the OVIS SLA Violation methods that can be used for HPBPI. Using any of these methods for other alerts results in a message containing erroneous data.

Table 52 Velocity Template Methods for SLA Violations

SLA Method Name	SLA Method Description
$alert.getOvisAgreementName()	Returns a STRING containing the name of the SLA that has been violated.
$alert.getOvisAgreementId()	Returns a STRING containing the unique identifier for the SLA that has been violated.
$alert.getOvisCustomer()	Returns a STRING containing the Customer details relating to the SLA that has been violated.
$alert.getOvisViolationTime()	Returns a STRING containing the time that the SLA was violated.
$alert.getOvisAgreement ConformanceThreshold()	Returns a STRING that contains the conformance threshold for the SLA, expressed as a percentage.
$alert.getOvisAgreement Conformance()	Returns a STRING that contains the conformance for the SLA, expressed as a percentage.

Table 52 Velocity Template Methods for SLA Violations

SLA Method Name	SLA Method Description
$alert.getOvisAgreementSamples()	Returns an INTEGER that contains the number of samples taken over the measurement period for the violation.
$alert.getOvisAgreementConformingSamples()	Returns an INTEGER that contains the number of samples that conformed to the SLA over the measurement period.

XSLT Templates

An XSLT template is a file containing XML that transforms event XML into email or HP Operations Manager message XML. An example of an email XSLT template is shown below.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
version="1.0"
        xmlns:env="http://schemas.xmlsoap.org/soap/envelope/"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:etype="http://www.hp.com/openview/bia/eventtype"
        xmlns:edata="http://www.hp.com/openview/bia/eventdata">
  <xsl:output method="xml"
       indent="no" />

<xsl:template match="/">
<Email xsi:noNamespaceSchemaLocation='email.xsd'>
<Subject>
<xsl:apply-templates select="//env:Body/*/etype:Severity/*"/><xsl:text> </
xsl:text><xsl:value-of select="//etype:EventName" /> Alert</Subject>
<ContentType>text/plain</ContentType>
<Body>
A "<xsl:value-of select="//etype:EventName" />" alert has occurred.

Severity: <xsl:apply-templates select="//env:Body/*/etype:Severity/*"/>
Event Group: <xsl:value-of select="//etype:EventGroup" />
Event Name: <xsl:value-of select="//etype:EventName" />
Event Time: <xsl:value-of select="//etype:EventTime" /><xsl:text>
</xsl:text>
<xsl:for-each select="//edata:DataItem" >
<xsl:choose>
<xsl:when test="edata:Name = 'RootCause'" >
<xsl:text>Root Cause: </xsl:text>
```

```
<xsl:for-each select="//edata:Value/Services/Service" >
<xsl:text>-&gt;</xsl:text><xsl:value-of select="Label" />
</xsl:for-each>
</xsl:when>
<xsl:otherwise>
<xsl:value-of select="edata:Name" /><xsl:text>: </xsl:text><xsl:value-of
select="edata:Value" />
</xsl:otherwise>
</xsl:choose>
<xsl:text>
</xsl:text>
</xsl:for-each>
</Body>
</Email>
</xsl:template>

<xsl:template match="etype:Critical" >
<xsl:text>Critical</xsl:text>
</xsl:template>

<xsl:template match="etype:Major" >
<xsl:text>Major</xsl:text>
</xsl:template>

<xsl:template match="etype:Minor" >
<xsl:text>Minor</xsl:text>
</xsl:template>

<xsl:template match="etype:Warning" >
<xsl:text>Warning</xsl:text>
</xsl:template>

<xsl:template match="etype:Normal" >
<xsl:text>Normal</xsl:text>
</xsl:template>

</xsl:stylesheet>
```

The style sheet is applied to the event sent by the Business Impact Engine. The style sheet contains email XML text as well as XSL elements. XSL elements format information from the form of the event and insert it into the email XML.

The schemas for the event are located in the files `eventtype.xsd` and `eventdata.xsd` in the following directory:

`HPBPI-install-dir``/misc/bia`

The message header contains a business event type, which is described in `eventtype.xsd`. The message body contains a business event type followed by business event data; Business event data is described in `eventdata.xsd`.

Creating Scripts

You can define a script, which is a Windows `.bat` file, to be run when a notification event is received by the Notification Server; for example, this script might send an SMS message to a recipient directly (bypassing the email server), or it might update an entry in a database or spreadsheet.

You can create a script to complete any action that you want, and you can also include any of the environment variables that have been defined for use within your scripts. These environment variables identify configuration information relating to the HPBPI data available for notification alerts and are defined in section "Environment Variables for Scripts" on page 209.

These environment variables are included in your scripts using the following notation:

`%`*`env_name`*`%`

where `%`*`env_name`*`%` is an environment variable name, for example, `%OVBPI_FLOWNAME%`.

The following examples show how you can create a script and configure the Notification Server to run the script when a specific notification event is received.

Example Script to Write a String to a File

The following is an example of the steps that you need to complete in order to configure the Notification Server to write the string `Hello World` to a file on receipt of a specific flow impact event:

1. Create the script that you want to be executed when the notification event is received, for example:

   ```
   echo "hello world" >>c:\notif.txt
   ```

   This script writes the string "`hello world`" to the file `notif.txt` in the root of your `c:` drive.

2. Save the script to the following location with a unique file name and a `.bat` file extension:

   ```
   HPBPI-install-dir/data/conf/bia/notify-scripts
   ```

   The Notification Server looks in this directory for scripts and presents these scripts to your through the Web Administration Console for the Notification Server when you are configuring subscriptions. Note that if the script that you create does not have a `.bat` file extension, it is ignored.

3. Follow the instructions in section "Script Subscription" on page 185 to create a subscription for the script selecting `Flow Impacted` as the `Event Name` and the name of the flow for the `Flow Name`.

   If you have the Web Administration for the Notification Server administration open at the `New Event Subscription` page, you might need to click the `Refresh` option on your Web Browser. This refreshes the page and adds the new script to the list.

Example Script to Write Flow Name and Blocked Instances to a File

The following is an example of the steps that you need to complete in order to configure the Notification Server to take the flow name and the number of blocked instances from the event data and write the information to a file on receipt of a specific flow impact event:

1. Create the script that you want to be executed when the notification event is received, for example:

   ```
   echo %OVBPI_FLOWNAME%,
   %OVBPI_NOOFBLOCKEDINSTANCES%>>c:\notif2.txt
   ```

   This script writes the values of the flow name and the number of blocked instances to the file `notif2.txt` in the root of your `c:` drive.

2. Save the script to the following location with a unique filename:

   HPBPI-install-dir`/data/conf/bia/notify-scripts`

   The Notification Server looks in this directory for scripts and presents these scripts to your through the Web Administration Console for the Notification Server when you are configuring subscriptions.

3. Follow the instructions in section "Script Subscription" on page 185 to create a subscription for the script selecting `Flow Impacted` as the `Event Name` and the name of the flow for the `Flow Name`.

   If you have the Web Administration for the Notification Server administration open at the `New Event Subscription` page, you might need to click the `Refresh` option on your Web Browser. This refreshes the page and adds the new script to the list.

Environment Variables for Scripts

The following tables list the environment variables that can be used within your scripts to report data on flow impact, flow out-of-sequence and metric threshold alerts.

Table 53 on page 209 lists the environment variables that are available for all HPBPI alerts.

Table 53 Environment Variables for All Alerts

Variable	Description
OVBPI_EVENTGROUP	name of the event group, which is always NOTIFICATIONS.
OVBPI_EVENTNAME	name of the event: FLOW_IMPACT, FLOW_METRIC_THRESHOLD_ALERT or FLOW_OUT_OF_SEQUENCE.
OVBPI_SEVERITY	severity of the HPBPI event. Possible values are Normal, Warning, Minor, Major or Critical

Table 54 on page 210 lists the environment variables available for flow impact alerts. Using these environment variables for other alerts results in erroneous data being returned.

Table 54 Environment Variables for Flow Impact Alerts

Variable	Description
OVBPI_FLOWGUID	internal unique identifier (GUID) for the flow definition in the HPBPI event. This identifier is useful to add to a URL in order to link directly to a flow definition page within the Business Process Dashboard.
OVBPI_FLOWNAME	name of the flow in the HPBPI event
OVBPI_NOOFBLOCKEDINSTANCES	number of blocked flow instances at the time of the FLOW_IMPACT event. A blocked instance is an instance of the flow that is active at a node and cannot proceed as there is a problem in an underlying operational service.
OVBPI_NOOFIMPEDEDINSTANCES	number of at risk flow instances at the time of the FLOW_IMPACT event. An at risk instance is where one or more flow instance has the potential to be blocked in the future.
OVBPI_SERVICEGUID	internal unique identifier (GUID) for the service in the HPBPI FLOW_IMPACT event.
OVBPI_SERVICENAME	name for the service in the HPBPI FLOW_IMPACT event.
OVBPI_ROOTCAUSE	the label of the root cause service, or services, in the HPBPI FLOW_IMPACT event.

Table 55 on page 211 lists the environment variables available for threshold alerts. Using these environment variables for other alerts results in erroneous data being returned.

Table 55 Environment Variables for Threshold Alerts

Variable	Description
OVBPI_FLOWGUID	internal unique identifier (GUID) for the flow definition in the HPBPI FLOW_METRIC_THRESHOLD_ALERT event. This identifier is useful to add to a URL in order to link directly to a flow definition page within the Business Process Dashboard.
OVBPI_FLOWIDENTIFIER	identifier for the flow instance in the HPBPI FLOW_METRIC_THRESHOLD_ALERT event. This property contains a value only if the threshold alert is an instance alert, for example, Absolute duration. It does not contain a value if the property is for a statistical alerts, for example, a Backlog count. This the property that you have nominated to be the identifier for the flow.
OVBPI_FLOWINSTANCEGUID	internal unique identifier (GUID) for the flow instance in the HPBPI FLOW_METRIC_THRESHOLD_ALERT event. This property contains a value only if the threshold alert is an instance alert, for example, Absolute duration. It does not contain a value if the property is for a statistical alerts, for example, a Backlog count.
OVBPI_FLOWMETRICGUID	internal unique identifier (GUID) for the flow metric in the HPBPI FLOW_METRIC_THRESHOLD_ALERT event.
OVBPI_FLOWMETRICNAME	name of the flow metric in the HPBPI FLOW_METRIC_THRESHOLD_ALERT event.

Table 55 Environment Variables for Threshold Alerts

Variable	Description
OVBPI_FLOWMETRICVALUE	value of the flow metric in the HPBPI FLOW_METRIC_THRESHOLD_ALERT event.
OVBPI_FLOWNAME	name of the flow in the HPBPI FLOW_METRIC_THRESHOLD_ALERT event
OVBPI_THRESHOLDALERTGUID	internal unique identifier of the threshold identified in the HPBPI FLOW_METRIC_THRESHOLD_ALERT event.
OVBPI_THRESHOLDMESSAGE	message defined for the HPBPI FLOW_METRIC_THRESHOLD_ALERT event. This is the message that you have defined to be displayed when the event is triggered.
OVBPI_THRESHOLDNAME	name of the metric threshold in the HPBPI FLOW_METRIC_THRESHOLD_ALERT event.

Table 56 on page 212 lists the environment variables for the Out-of-Sequence alerts. Using these environment variables for other alerts results in erroneous data being returned.

Table 56 Environment Variables for Out-of-Sequence Alerts

Variable	Description
OVBPI_FLOWIDENTIFIER	identifier for the flow instance in the HPBPI event. This the property that you have nominated to be the identifier for the flow.

Table 56 Environment Variables for Out-of-Sequence Alerts

Variable	Description
`OVBPI_FLOWINSTANCEGUID`	internal unique identifier (GUID) for the flow instance in the HPBPI event.
`OVBPI_SOURCENODE`	name of the node that is the source node for the out-of-sequence alert.
`OVBPI_DESTINATIONNODE`	name of the node that is the destination node for the out-of-sequence alert.

7 Intervention

This chapter describes how you can make changes to the business flow data, and delete completed instances of Flows and Data definitions using HPBPI tools and configuration options. Specifically this chapter describes:

- modifying, or removing individual instances, services or HPBPI Metrics; see section Intervention Client on page 217
- regularly pruning completed entries from the database; see section Engine Flow and Data Instance Cleaner Parameters on page 224.

Tools and Parameter Settings

There are two different ways of clearing HPBPI data from the database and you need to select the method that is most appropriate to your requirements.

You can use:

- the Intervention Client to remove individual flow instances or data instances from the database.

  The purpose of the Intervention Client is to enable you to access deployed Flow and Data definitions, plus Flow instances, Data instances and Services. This then enables you to modify or delete (with the exception of Services) instances of these definitions in order to correct:

  — potential errors within your HPBPI system.

  — errors in the system that is being monitored.

  — errors that exist within deployed Flow definitions.

  You can delete individual instances using the Intervention Client. However, if you want to delete numbers of completed or active instances regularly you can use the Engine Instance Cleaner parameters as described in section Engine Flow and Data Instance Cleaner Parameters on page 224.

  ➤ You cannot delete services using the Intervention Client; however, you can modify the status of service definitions using the Intervention Client.

- the Engine Instance Cleaner parameters for regular removal of completed instances from the database.

  The instance cleaner parameters are not designed to enable individual instances to be manipulated nor do they enable you to name the flow definitions or data definitions whose instances are to be deleted; these functions are provided by the Intervention Client.

Intervention Client

The Intervention Client enables you to access, or intervene in, business flow instances; this, in turn, enables you to make changes to the flow instances. The Intervention Client also enables you to delete Flow and Data definitions, and clear the average and total counts associated with a specific definition.

The following are some examples of the uses of the Intervention Client:

- where a flow instance has progressed through a flow, but one or more nodes that should be complete still appear as active.

  You can use the Intervention Client to change the state of the active nodes to `Complete`. Nodes can get into this state if you have manually progressed a Flow instance, for example, when an application that usually provides the service is not available and you have substituted another application to complete the step.

- when the start and complete conditions for a node are incorrect and therefore cannot be satisfied. In this case, you might want to delete all instances of the flow and any associated flow data.
- where you want to progress a specific flow instance to its completion, for example, there might be and order processing instance waiting for an automated system to enter credit card information, and the automated system is not operational. In this case, you can modify a data instance to add credit card information manually and progress the order for a customer.
- to remove erroneous flows to tidy up your system. You might need to do this if you have been developing new flows and want to remove older versions, which are no longer required.
- where you want to search for all flow instances over a certain age and remove them. You can also do this using the instance cleaner parameters tools (see section Engine Flow and Data Instance Cleaner Parameters on page 224); however, you might not want to remove all the flow instances, just for one specific flow.
- to delete a orphaned Data definition that is not linked to a Flow definition, or perhaps where the Data definition was linked to a Flow that has been deleted.

- to test your HPBPI system. You might want to delete all the active Data definitions to reset the system.
- to change the status of services that are not behaving as they should be.
- reset the totals and averages for Flow and Data definitions. You might want to do this periodically (daily, weekly, monthly) so you can see the totals and averages in terms of a known period.

Accessing the Intervention Client

You access the Intervention Client as follows:

1. Open a new Web browser window
2. Type the following URL:

   ```
   http://hostname:44080/ovbpiintclient/index.jsp
   ```

   where:

   — *hostname* is the fully qualified hostname of the system where the HPBPI Server is running.

   — `44080` is the port number for the Servlet Engine, identified by the `ServletEngine HTTP` port number. Use the port number configured for your system.

   If you are running the Web Browser on the same system as the HPBPI Server, you can use `localhost` in the URL.

3. You are presented with a dialog to enter the login credentials for the Intervention Client. Enter a User Name and Password for the Client. By default the User Name is `admin` and the Password is `hpbpi`. Section Security and the Intervention Client on page 223 describes how to change the username and password credentials for the Intervention Client.

Using the Intervention Client

On the home page for the Intervention Client, you are offered the following menu options:

- Flow Definitions
- Data Definitions
- Services

Typically, you have identified an anomaly using the Business Process Dashboard, and are therefore using the Intervention Client to solve a specific problem. In this case, you have the identifier of the definition that you want to modify and you can enter this identifier directly on the search page to access the definition that you want to modify. Alternatively, you can use the Intervention Client's filtering capability to search for the instance that you want.

When you select one of the options (Flow Definitions, Data Definitions or Services), you are presented with a list of definitions within the selected category. The actions that you can complete at this point depend on the definition that you have selected:

- for Flow definitions, refer to section Flow Definitions on page 220.
- for Data definitions, refer to section Data Definitions on page 222.
- for Services, refer to section Services on page 223.

⚠ If you make changes to the values of data properties within a flow, the progression rules associated with the data properties are evaluated and the flow progresses, based on the new values.

You need to be aware of the following when using the Intervention Client:

- When selecting the option to delete a Flow and associated Data definition, make sure that the Data definition is not also a dependency for other Flow definitions. If the Intervention Client detects that by deleting the Data definition other deployed flows are affected, it presents you with a page that lists all the flows that also depend on the Data definition, where appropriate.
- When you select the option to delete a data instance, or a flow instance including all its data instances, you can potentially leave other flow instances in an indeterminate state.

This is a similar scenario to the previous bullet describing Flow and Data definitions; however, in the case of instances, the Business Impact Engine cannot determine any relationships between the data instance that is being deleted and other flow instances that might have a dependency on it.

If you do delete a data instance and there are flow instances that depend on it, the flow can no longer progress.

If you want to delete a significant number of flow instances or data instances, the quickest way to achieve this is to delete the Flow definition or Data definition; however, this deletes all instances of the Flow and Data definitions, including any active instances. You can then redeploy the Flow and Data definitions using the HPBPI Modeler.

Flow Definitions

When you select the Flow Definitions option from the Intervention Client menu, you are presented with a list of Flow definitions that are currently superseded and Flow definitions that are deployed in the Business Impact Engine.

The Flow definitions within the Modeler are not impacted by the actions of the Intervention Client, so you can redeploy the Flow; however active and completed instances of the flow are potentially impacted by changes made through the Intervention Client.

At this stage, you can choose one of the following options:

- Delete Flow and Metric Definitions and instances
- Delete Flow, Metric and Data Definitions and all related instances
- Clear Average Time and Total Instance Count (Flow and Nodes)

  This clears (or removes) the values for TotalFlows and AvrgTime from the Flows table in the Business Impact Engine database. The *HPBPI Reference Guide* describes the database tables in detail.

Alternatively, you can select the `Search` option to identify specific instances of the definition.

From the search page, you can enter a Flow instance identifier directly, or you can filter for instances according to the following criteria:

- flow instances with a specific identifier and value for the Weight parameter.
- flow instances within specified time periods.
- flow instances that are in a particular state, for example: `Active` or `Completed`.

When you have selected your search criteria, click the `Search` button and you are presented with a `Flow Instance List`, where all the flow instances matching your search criteria are listed. If no criteria are listed, all Flow instances are returned.

From this list, you can select instances from the list and delete them as required.

In addition, you can edit a selected Flow instances. To do this select the `Edit` option, and a flow diagram is displayed. The flow diagram shows the status of each node in the flow, for example, whether it is in the started or completed state. You can progress flow instances from this option, by manually activating or completing nodes in the flow.

There are three icons:

- the tick indicates that the node instance has completed at least once
- the cog wheel indicates that the node instance is started
- no icon indicates that the node instance is not started or completed

Data Definitions

When you select the Data Definitions option from the Intervention Client menu, you are presented with a list of Data Definitions that are currently deployed in the Business Impact Engine.

At this stage, you can choose one of the following options:

- Delete Data Definition and instances
- Clear Average Time and Total Instance Count

  This clears (or removes) the values for TotalInstances and AvrgTime from the Data Objects table in the Business Impact Engine database. The *HPBPI Reference Guide* describes the database tables in detail.

Alternatively, you can select the `Search` option to identify specific instances of the definition.

From the search page you can filter based on the values of the properties of the Data instances.

Following a request to search for a specific Data instance, you are presented with a `Data Instance List` page where you can delete or edit the Data Instances that match the search criteria.

When you choose to edit a Data instance, you are presented with a `Data instance` screen where you have access to the properties of the Data instance in order to confirm that it is the definition that you are interested in.

When modifying the properties of a Data instance, make sure that you enter the correct `Type` for the property, for example, if the property is an Integer, a numeric value for the property must be specified.

The Data definitions within the Modeler are not impacted by the actions of the Intervention Client, so you can redeploy the Data definition; however all active instances of the Data definition, and possibly the associated Flow definition, are impacted by changes made through the Intervention Client.

Services

When you select the Services option from the Intervention Client menu, you are presented with a list of services that are currently deployed in the Business Impact Engine. From this Window you can edit a specific service by selecting the `Edit` option. From the Service screen you can modify the status of a service.

Any changes that you make affect only the HPBPI system, the changes have no effect on the Service as it might be defined in HP Operations Manager or OVIS.

Security and the Intervention Client

The Intervention Client requires you to provide a username and password in order to access the administration screens. Following a new HPBPI installation the username and password are:

- Username: `admin`
- Password: `hpbpi`

You can change the username and password used for the Intervention Client; see Chapter 8, Select Access Authorization or Chapter 9, Servlet Engine Authentication.

Engine Flow and Data Instance Cleaner Parameters

The Engine Instance Cleaner parameters enable you to control the numbers of Completed and Active Flow and Data instances that are stored in the database. There are separate parameters for deleting Completed and Active Metric instances from the database; see section Metric Engine Instance Cleaner Parameters on page 231.

Using the Engine instance cleaner parameters, you can control:

- how often the Engine Instance Cleaner thread marks Completed and Active Flow and Data instances as candidates to be deleted.
- how often the Engine Instance Cleaner thread is run
- how often active and completed flow and data instances are deleted from the database (if at all)
- the time and the age of the instances that you delete

Following a new HPBPI installation, the Engine Instance Cleaner thread does not run and Completed and Active instances therefore accumulate in the database.

This section describes the behavior of these parameters and the SQL procedures, which you can modify to enable you to archive the data, before it is deleted, to a location of your choice. By archiving the data you have it available for your reporting tools, but it is no longer stored in the HPBPI database tables and therefore does not impact the performance of your system.

If you configure these parameters to delete Completed and Active instance data from the database, and you do not archive the instance data, you cannot customize the HPBPI Business Process Dashboard to monitor Completed and Active instances.

In summary, the instance cleaner parameters are intended for regular removal of Completed and Active instances as they occur during normal operations. They are not intended to be used for the removal of individual completed instances, or removal of instances in states other than the `COMPLETED` or `ACTIVE` state. Use the Intervention Client to remove individual instances; see section Intervention Client on page 217.

Configuring the Engine Instance Cleaner Parameters

You modify the instance cleaner parameters using the HPBPI Administration Console as described in section BIE Flow and Data Instance Cleaner Settings on page 45.

These parameters enable you to mark instances as being candidates to be deleted. Instances are initially marked to be deleted and not immediately deleted to reduce the impact of the database activity on other Business Impact Engine activities. If you are using a remote database, this might not be an issue for your implementation. Once the instances are candidates to be deleted, they are deleted periodically by the instance cleaner thread as specified by the instance cleaner parameter values. You can specify the frequency that you want them deleted and the numbers of instances that you want deleted in one database Delete statement, using the parameters.

In addition to modifying the instance cleaner parameters, you can modify the stored procedures invoked by the instance cleaner thread and add SQL commands to archive the Business Impact Engine data to a location of your choice; see section Archiving Completed and Active Flow Instances on page 228.

The following are some examples of how you can set the instance cleaner parameters to achieve specific outcomes, or requirements.

Example for Deleting Completed Instances

If you want to delete complete flow instances from the database that are more than one day old and once a day, set the instance cleaner parameters as follows:

- From the `Engine Flow and Data Instance Cleaner settings:`
  - Set the `Mark instances for deletion:` option to `Once a Day`.

    Enter `01:00` as the time for the instances to be marked. Pick a time when the database is not busy processing other HPBPI data.
  - Set the Delete interval: option to 1 minute.

    This is the interval at which the Flow and Data instance cleaner thread runs and attempts to delete any Flow and Data instances that are marked as being candidates to be deleted.
  - Set the Delete batch size to 500

    This is the number of instances that the instance cleaner thread attempts to delete each time it runs.
- From the `Completed Flow and Data Instances settings:`
  - Set the `Delete completed instances from the database` option to `As scheduled above`

    The Completed instances are then deleted on the basis of the times set for the Engine Instance Cleaner thread.
  - Set the `Age of the completed instances to be removed (minutes)` to two days, which is expressed in minutes (2880 minutes).

Example for Deleting Active and Completed Instances

If you want to delete Completed flow instances that have an age of more than 30 days, and Active flow instances that have an age of more than 50 days, once a day, set the instance cleaner parameters as follows:

- From the `Engine Flow and Data Instance Cleaner settings:`
  - — Set the `Mark instances for deletion:` option to `Once a Day`.

    Enter `01:00` as the time for the instances to be marked. Pick a time when the database is not busy processing other HPBPI data.
  - — Set the Delete interval: option to 1 minute.

    This is the interval at which the Flow and Data instance cleaner thread runs and attempts to delete any Flow and Data instances that are marked as being candidates to be deleted.
  - — Set the Delete batch size to 1000

    This is the number of instances that the instance cleaner thread attempts to delete each time it runs.
- From the `Completed Instances settings:`
  - — Set the `Delete completed instances from the database` option to `As scheduled above`
  - — Set the `Age of the completed instances to be removed (minutes) to` 30 days, which is expressed in minutes (43200 minutes).
- From the `Active Instances settings:`
  - — Check the `Delete active instances from the database?` check box so the active instances are deleted on the basis of the Engine Instance Cleaner thread settings.
  - — Set the `Age of the active instances to be removed to` 50 days.

Example for Deleting Completed Instances as Soon as They Are Complete

If you want to delete Completed instances as soon as they are complete, set the instance cleaner parameters as follows:

- Set the `Engine Flow and Data Instance Cleaner settings` as required for the Active instances, as these settings are ignored in the case of the Completed Instances setting `Immediately on completion`.
- From the `Completed Instances settings`:
  - Set the `Delete completed instances from the database` option to `Immediately on completion`
  - `Age of the completed instances to be removed (minutes)` is also not applicable and is not an editable field.

  Be aware that using this setting has the effect of removing data that might have a value for monitoring purposes. If your flow instances do not have long lives, then there is going to be very little data in the database to be reported on. Make sure that you do not need the data that you are deleting from the database.
- Set the `Active Instances settings` as required.

Archiving Completed and Active Flow Instances

You have the option to archive Completed and Active instances before they are deleted from the database by the instance cleaner thread. This enables you to remove data from the HPBPI database to improve performance, but still keep the data for reporting purposes.

You cannot archive Completed instances if you choose the option to `Delete completed instances from the database Immediately on completion`. This is because for this option, the instances are deleted as soon as the event that completes the instance is received.

To archive the instance data, you need to make changes to the stored procedures or functions provided with HPBPI to delete the flow and data instances. There are stored procedures for Microsoft SQL Server and functions for the Oracle Server, both contain the SQL commands required to delete data from the HPBPI database and also provide a return code to the calling program. The following are the stored procedures/functions:

- `bia_DeleteDataInstances`

  This procedure deletes the data instances in the `COMPLETED` state and that meet the criteria specified in the configuration. This stored procedure also provides the following SQL Cursor that you can use:

  `completedInstances_cursor`

  This stored procedure is available for Microsoft SQL Server only.

  You can add your custom SQL statements to the stored procedure in order to archive the business flow data before it is deleted.

- `bia_DeleteFlowInstances`

  This procedure deletes all flow instances in the `COMPLETED` state that meet the criteria specified by the configuration. This stored procedure also deletes the associated node instances, node started and node completed times, and any metrics that are defined for the flow instance.

- `DeleteActiveDataInstances`

  This procedure deletes the data instances in the `ACTIVE` state and that meet the criteria specified in the configuration. This stored procedure also provides the following SQL Cursor that you can use:

  `activeInstances_cursor`

  This stored procedure is available for Microsoft SQL Server only.

  As for the `bia_DeletedDataInstances` procedure, you can add your custom SQL statements to the stored procedure in order to archive the business flow data before it is deleted.

- `DeleteActiveFlowInstances`

  This procedure deletes all flow instances in the `ACTIVE` state that meet the criteria specified by the configuration. This stored procedure also deletes the associated node instances, node started and node completed times, and any metrics that are defined for the flow instance.

These stored procedures or functions are contained in the following SQL file:

HPBPI-install-dir`\misc\bia\EngineStoredProcedures_script.sql`

There are also examples of this file located at:

- *HPBPI-install-dir*\examples\bia\EngineScripts
- the examples directory on the distribution media

The example files are named as follows, according to the database:

- EngineStoredProcedures_script.mssql
- EngineStoredProcedures_script.oracle

⚠ Be aware that changes to the SQL file EngineStoredProcedures_script.sql are lost if you reinstall or upgrade HPBPI. Make sure that you have made backup copies of your modifications, which you can then reapply following a reinstallation.

You need to make your changes to the stored procedures, which have been created in the database for HPBPI, to add the SQL to archive the Completed or Active instances.

You are also advised to make changes to the following SQL script and make a backup copy of the script to ensure that you have a copy of the changes:

HPBPI-install-dir\misc\bia\EngineStoredProcedures_script.sql

The instance cleaner parameters are executed at the frequencies specified in the Engine Instance Cleaner Settings. The value of the Age of the completed instances to be removed and Age of the active instances to be removed parameters is passed to the stored procedure, or function, when it is called.

➤ It is possible for two data instances to be created, each representing the same business flow: one in the Business Impact Engine database and one in the archive database. This occurs when an out-of-sequence event is received by the Engine after the Completed or Active instances are archived and deleted. An out-of-sequence event is where an event is received by the Business Impact Engine for an activity in a flow instance that is earlier than the latest event received for the same flow instance.

In summary, you need to ensure that your archive code can take account of the fact that there might be more than one instance that represents the same business flow when this scenario exists.

Metric Engine Instance Cleaner Parameters

The Metric Engine Instance Cleaner parameters enable you to control the numbers of metric instances that are stored in the database. These include:

- Active metric instances
- Completed metric instances
- Metric statistic instances
- Metric alarm instances

There are separate parameters for deleting Completed and Active business impact Engine instances from the database; see section Engine Flow and Data Instance Cleaner Parameters on page 224.

Using the Metric Engine instance cleaner parameters, you can control:

- how often the Metric Engine Instance Cleaner thread is run.
- how often active and completed metric instances are deleted from the database (if at all).
- how often metric statistics settings are deleted from the database (of at all).
- how often metric alarm settings are deleted from the database (of at all)
- for all metric instances, the time and the age of the instances that you delete.

Following a new HPBPI installation, the Metric Engine Instance Cleaner thread does not run and the different metric instances therefore accumulate in the database.

Configuring the Metric Engine Instance Cleaner Parameters

You modify the Metric Engine instance cleaner parameters using the HPBPI Administration Console as described in section Metric Engine Instance Cleaner Settings on page 68.

These parameters enable you to delete the Metric Engine instances periodically, using the instance cleaner thread, as specified by the instance cleaner parameter values.

You can control how often the metric instance cleaner thread is executed and the age of the specific metric instances that you want to delete. Each type of metric instance can be separately controlled.

The following are some examples of how you can set the Metric Engine instance cleaner parameters to achieve specific outcomes, or requirements.

Example for Deleting Completed Metric Instances

If you want to delete completed metric instances from the database, which are more than one day old and once a day, set the metric engine instance cleaner parameters as follows:

- From the `Metric Engine Instance Cleaner settings:`
  - Set the `Instance cleaner execution interval (minutes)` to 5.
- From the `Completed Metric Instances settings:`
  - Select the `Delete completed metric instances from the database?` option in order to activate the `Age of completed metric instances to be removed (days)` option.
  - Set the `Age of completed metric instances to be removed (days) option` to 2.
- Set the other option as appropriate for your implementation.

8 Select Access Authorization

You can secure access to some of the HPBPI components using HP Select Access. HP Select Access is part of the HP Identity Management suite of products; it provides policy-based authentication and authorization for your applications; for example, HPBPI Repository Explorer. Using HP Select Access enables you to centralize and automate access control for your HPBPI, and other HP BTO Software, components.

Be aware that some HPBPI components can also be authenticated using Tomcat Realm Configuration as described in Chapter 9, Servlet Engine Authentication. You can use Tomcat as an alternative to using Select Access. The security option that you select depends on your particular environment and what policies you have in place already.

HPBPI provides Select Access support for the following components:

- Repository Explorer
- Intervention Client
- Notification Server
- Metric Definer
- HPBPI Business Process Dashboard
- HPBPI Service Desk Process Insight Dashboard
- HPBPI Model Repository, and therefore the HPBPI Modeler.

  Any changes to the Model Repository in terms of security options also impact the Flow Simulator, which is a contributed component. Refer to the instructions provided with the Flow Simulator for more details.

Select Access controls access to applications through its Policy Builder, Policy Validator and Enforcer plugins. The Policy Builder is a graphical interface that enables you to define the authentication methods and authorization polices; it is part of the Select Access Administration Server.

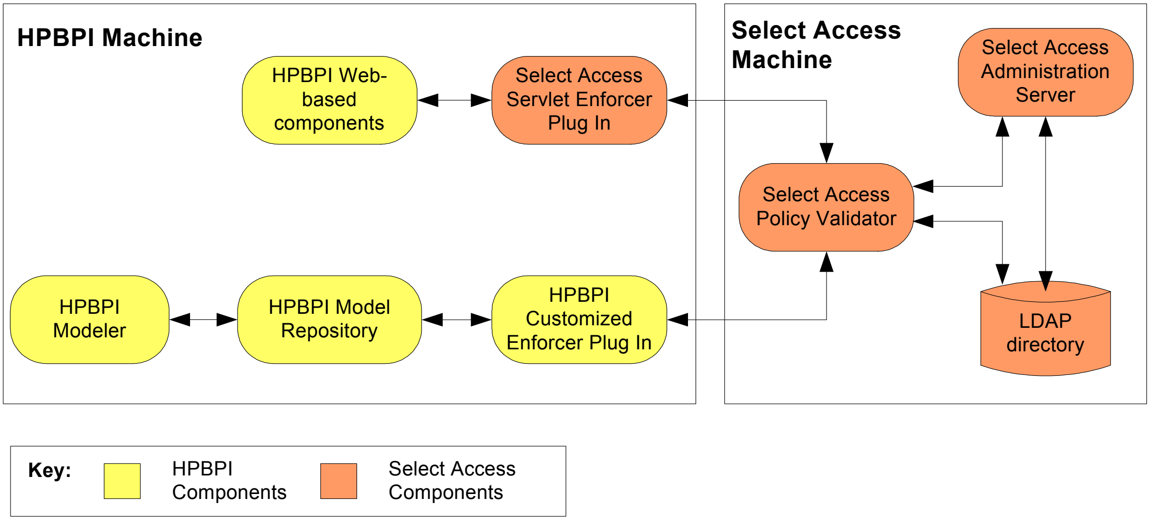
This chapter is structured as follows:

- The following sections describe how to configure Select Access for HPBPI:
  - Select Access Enforcer Plugin on page 235
  - Select Access Components Required for HPBPI on page 238
  - HPBPI Select Access Identity Tree Configuration on page 246
  - Configure the Select Access Authentication Service on page 247
  - Configure Identity Access to HPBPI Components on page 248
  - Enabling Select Access Authorization within HPBPI on page 249
- Section HPBPI Web Client Customized Login Forms on page 251 explains how you can use the HPBPI customized Web login forms in place of the Web forms provided with the Servlet Enforcer Plugin.
- Section Using Select Access to Control Access to Individual Flows on page 253 explains how you can use Select Access to control who has access to particular flows when using the Business Process Dashboard.
- Section Using Certificate-based Client Authentication on page 255 explains how Select Access can be used to further refine the security for HPBPI Web clients, when you have configured your Servlet Engine (Web Server) to send and receive HTTPS requests.

Select Access Enforcer Plugin

The Enforcer Plugin needs to understand the protocols in the incoming access request. Select Access provides a number of predefined Enforcer plugins, one being the Servlet Enforcer Plugin, which is used for HTTP protocols and is suitable for all the HPBPI Web-based components. There is also a customized Enforcer Plugin that can be used to accept requests from the HPBPI Model Repository; this is an RMI request. Figure 6 shows the Enforcer Plugins used by HPBPI.

Figure 6 Select Access Components for HPBPI


HPBPI Machine
HPBPI Web-based components
Select Access Servlet Enforcer Plug In
HPBPI Modeler
HPBPI Model Repository
HPBPI Customized Enforcer Plug In
Select Access Machine
Select Access Administration Server
Select Access Policy Validator
LDAP directory
Key:
HPBPI Components
Select Access Components

The Enforcer plugins and the Policy Validator evaluate a request for access to a components as follows:

1. The Enforcer plugin consolidates the HPBPI component request to access a particular resource and sends an XML document containing the relevant information to the Policy Validator.
2. The Policy Validator accepts incoming XML documents from the Enforcer plugin and works out what additional data is required to complete the validation between the HPBPI component and the resource.
3. The Policy Validator accesses the required policy information and identifies the rules and conditions that apply to the request. This information is cached in case it is required again.

4. The Policy Validator identifies whether there are conditional rules for access and, if there are, it evaluates the rules until it is able to reach a conclusion on whether access can be granted, or not.

5. The Policy Validator communicates its decision to the Enforcer plugin, which then enforces the policy decision.

The Select Access components can be on the same system as HPBPI; however, to optimize performance, you are advised to have the Select Access server components on a separate system; these are the Select Access components shown on the right-hand side of Figure 6.

The Select Access Servlet Enforcer plugin is a Select Access component and therefore needs to be separately installed. It should be installed on the same system as any of the HPBPI components that require the Servlet Engine in order to operate; these components are the components listed earlier in this chapter. The Model Repository does not require the Servlet Engine and therefore does not use the Select Access Servlet Enforcer plugin. The Model Repository requires a customized Enforcer plugin, specifically designed for RMI connections. This customized plugin is installed with HPBPI.

The following are the HPBPI installation options that require the Servlet Enforcer to be installed if you want to use Select Access for authorization:

- Server and Modeler
- Server Only
- Dashboards Only

The custom Enforcer plugin is installed with the appropriate HPBPI components; it does not have to be separately installed.

Select Access enables you to define access policies, which are definitions of whether a specified identity that you define is able to access a specific HPBPI resource. For more information on Select Access policies, refer to the Select Access documentation.

You define these policies using the Policy Builder GUI, which is a component of Select Access and is installed with the Select Access Policy Validator.

The following is a high-level overview of the tasks that you need to complete in order to use Select Access as the method of authorization for HPBPI components:

1. Make sure that Select Access is installed and configured and identify and create the access control policies that you want to use throughout your organization.
2. Install the Select Access Servlet Enforcer on the appropriate HPBPI system, or systems.
3. Configure the Select Access Servlet Enforcer.
4. Set up the HPBPI resources (HTTP and RMI) within the Select Access Policy Builder.
5. Create the identities that you want to access the HPBPI resources that you have configured
6. Enable Select Access Authorization within the HPBPI Administration Console.

The following sections describe these steps in more detail.

Select Access Components Required for HPBPI

Before configuring any HPBPI components to use access control through Select Access, you must ensure that the following Select Access components are installed and configured:

- Administration Server
- Policy Validator
- Servlet Enforcer plugin; note that is can also be referred to as the Generic Enforcer Plugin in the Select Access documentation. The Servlet Enforcer plugin secures the Servlet Engine used by HPBPI Web-based components.

The Administration Server and the Policy Validator can be installed on any system; however, the Servlet Enforcer plugin must be installed on the same machine as the HPBPI component or components that you want to place under Select Access Authorization.

Section Select Access Servlet Enforcer Configuration on page 239 describes the configuration requirements for the Servlet Enforcer plugin that are specific to HPBPI components.

The Servlet Enforcer plugin does not need to be installed on machines running the HPBPI Modeler, where the Modeler is installed on a system remote from the HPBPI Server. The Modeler is authorized through the Model Repository and RMI, and the Model Repository requires the customized Enforcer plugin.

Refer to the Select Access documentation for details of how to install and configure the required Select Access components.

The customized Enforcer required for the HPBPI Model Repository is installed as part of HPBPI and does not need any additional configuration.

Select Access Servlet Enforcer Configuration

Follow the instructions on the Select Access documentation to install the Servlet Enforcer plugin on the appropriate HPBPI system.

You can complete a `Typical` installation for the Servlet Enforcer Plugin; HPBPI has no special installation requirements. You then have the option to complete the Servlet Enforcer Plugin configuration when prompted at the end of the installation, or halting the installation and configuring the plugin later. There are some specific HPBPI configuration requirements for the Servlet Enforcer plugin, which are:

- Enabling Web session cookies; see section Enabling Web Cookies for Servlet Enforcer Plugin on page 240.
- Entering the full domain name, in addition to all other names and aliases that a machine is known by when configuring the HPBPI resources within the Policy Builder. This is described as one of the tasks in section HPBPI Select Access Resource Configuration on page 241.
- Excluding some file types if you decide to use the customized login forms provided with HPBPI; see section HPBPI Web Client Customized Login Forms on page 251.

When you do configure the plugin, make sure that you have the details of the name and port number for the Select Access Administration Server, as this needs to be specified as part of the Servlet Enforcer Plugin configuration.

During the Servlet Enforce Plugin configuration, you are asked to specify the name and location of a Policy Enforcer file for HPBPI. Make a note of the name and location that you specify as you need to enter the same details into the HPBPI Administration Console later.

If you want to configure the Servlet Enforcer separately from the installation, or if the Servlet Enforcer plugin is already installed, you access the configuration on the system where it is installed from the start menu as follows:

```
Start|All Programs|HP|HP Select Access|Setup Tool
```

You can also access the Setup Tool from within the Policy Builder.

This is one of the places where the Servlet Enforcer Plugin is referred to as the Generic Enforcer Plugin.

Enabling Web Cookies for Servlet Enforcer Plugin

You need to use the `Setup Tool` to configure the Enforcer Plugin to enable Web cookies for HPBPI components as follows:

1. Locate the `Tuning Parameters` dialog in the Select Access `Setup Tool`. This dialog can also be accessed from the Policy Builder `Tools` menu option. (`Tools|Component Configuration`)

   Make sure that you select the correct Servlet Enforcer Plugin configuration file.

2. On the `Tuning Parameters` dialog, make sure that the option to `Enable Web session cookies` is selected.

The Enforcer Plugin file is then updated with this new information.

If you reconfigure the Select Access components at any time, you need to reconfigure the Servlet Enforcer Plugins on each HPBPI system to pick up the changes.

HPBPI Select Access Resource Configuration

When you have configured the Servlet Enforcer Plugin, the next step is to add the HPBPI components that you want to be authorized using Select Access to the Select Access Policy Builder Resources Tree. The following instructions are an example of the configuration required for the HPBPI components. For full details of adding components to the Resources Tree, you should follow the Select Access documentation instructions.

The following instructions also assume that you are adding Select Access authorization for all the eligible HPBPI components:

1. Open the Select Access Policy Builder on the Select Access server system; for example:

   `https://hostname:9986/admin`

2. Add the name of the machine where you have installed the Servlet Enforcer for HPBPI as follows:

   Right click `Resource Access` and select `New|Folder`.

3. Enter `HTTP` as the folder name for the new resource.
4. The new `HTTP` resource now appears in the Resources Tree.
5. Right click the newly created `HTTP` resource name and select `New|Resource Server`.
6. Enter the fully qualified domain name of the machine where the Servlet Enforcer Plugin for HPBPI is installed (as returned from an `nslookup` *`hostname`* command) as the name of the resource.

7. Click the Add button to add additional Servers.

   You need to add all possible names and IP addresses for the machine where the Enforcer Plugin for HPBPI is installed; for example, the short domain name, IP address and the fully qualified domain name, plus any additional aliases. For each Server Hostname entry that you create, you also need to specify the protocol and a port number as follows:

   — Select the HTTP from the Protocol field.

   — Enter the Port # to be the port number for the Servlet Engine HTTP port for HPBPI. Unless modified it, this is: 44080. If you have modified the port number, you need to confirm the port number within the HPBPI Administration Console.

   If you have configured HPBPI to use HTTPS, you need to enter all the Server host names again. For each Server Hostname entry:

   — Select the HTTPS from the Protocol field.

   — Enter the Port # to be the same as the port number for the Servlet Engine HTTPS port; unless modified this is: 44443.

You now need to add a Network Resource Server and Protocol for the HPBPI Model Repository. To do this you first need to create a resource called rmi in the same way as you created the HTTP resource.

1. Right click Resource Access and select New|Folder.

2. Enter rmi as the folder name for the new resource.

   The new rmi resource now appears in the Resources Tree at the same level as the HTTP entry.

3. Right click the newly created rmi resource name and select New|Resource Server.

4. Enter the fully qualified domain name of the machine where the HPBPI Server (Model Repository) installed (as returned from an `nslookup` *`hostname`* command) as the name of the resource.
5. Click the `Add` button to add additional Servers.

   You need to repeat the process of adding all possible Server Hostnames for the machine where the HPBPI Server (Model Repository) is installed.

   For each Server Hostname that you add:

   — Select `rmi` from the `Protocol` field.

   — Enter the `Port #` to be the port number for the `RMI Registry` port for HPBPI. Unless modified it, this is: 44000. If you have modified the port number, you need to confirm the port number within the HPBPI Administration Console.

   All the Resources that you have configured now appear as Resource Access entries in the Resources Tree under the `http` and `rmi` entries.

Each time you make changes using the Policy Validator, you need are prompted to clear the validation cache to make sure the new entries are used.

You can also force the validation cache to clear, using the following command:

```
Tools | Clear Validator Cache(s)
```

The next step in configuring HPBPI authorization using Select Access is to add the HPBPI components as entries under the configured Resource Access entries in the Resources Tree.

Add the HPBPI Web-Based Components

The HPBPI Web-based components that you need to add are those components that require the Servlet Engine in order to operate; these are:

- Repository Explorer
- Intervention Client
- Notification Server
- Metric Definer
- HPBPI Business Process Dashboard
- HPBPI Service Desk Process Insight Dashboard

You can allow Select Access to discover the HPBPI Web-based applications; however, this example, describes how to add the Web-based applications individually.

To add the applications:

1. Right-click the resource name where the Servlet Enforcer Plugin is installed (under `http` in the Resources Tree) and select `New|Resource`.

2. Enter the name of the `http` resource that you want to add. This name is derived from the name of the Servlet Engine `.war` file for each of the HPBPI Web applications. You derive the name from the stem of the file name, that is, you omit the file extension (`.war`) when you add the resource. The following is the list of names that you need to enter in Select Access:

Web Component	**Select Access Resource Name**
Repository Explorer	hpbpirepositoryexplorer
Intervention Client	hpbpiintclient
Notification Server	hpbpinotifyadmin
Metric Definer	hpbpimetricdefiner
HPBPI Dashboard	hpbpi-bpd
HPBPI Service Desk Process Insight Dashboard	hpbpi_sd_dashboard

   Click `OK` to add each `http` Resource to the Resources Tree.

3. If you have not been prompted to already, make sure that you clear the validator cache.

   Now add the `rmi` resources as described in section Add the HPBPI RMI Components on page 245.

Add the HPBPI RMI Components

The HPBPI Model Repository (for HPBPI Modeler client) is the only rmi-based component that you need to add.

To add the Model Repository, complete the following steps:

1. Right-click the resource name where the Servlet Enforcer Plugin is installed (under `rmi` in the Resources Tree) and select `New|Resource`.
2. Enter the name of the HPBPI Modeler `rmi` resource that you want to add as follows:

   ```
   OVBPIModelRepository
   ```

3. Click `OK` to add the Resource to the Resources Tree.
4. If you have not been prompted to already, make sure that you clear the validator cache.

Now continue and configure the user accounts that you want to be able to access to the HPBPI components as described in section HPBPI Select Access Identity Tree Configuration on page 246.

HPBPI Select Access Identity Tree Configuration

So far you have added the HPBPI components as Select Access Resources. The next step is to configure `Identities` (or users) within the Policy Builder `Identity Tree`. These are the identities that you want to configure to be able to access these HPBPI components. For example, you might want to add an identity called `Admin` with a suitable password to log into the components.

The following steps are an example of how to add a Select Access Identity:

1. Right click at the point in the `Identity Tree` where you want to add the new identity.
2. Select `New | Identity`
3. Add the details of the identity that you want to create. Make sure that you add a `User Id`, as this is the identifier that you will use as the login name when you start the HPBPI components that you configure to use the identity.

You now need to configure the authorization service that you want for the resource; see section Configure the Select Access Authentication Service on page 247.

Configure the Select Access Authentication Service

When you have configured the users that you want to be able to access the HPBPI components, you can enable the authentication service that you want to use for each of the HPBPI components that you have added to the Resources Tree.

The Authentication Service that you select must already have been configured within Select Access; this is done using the Policy Builder. Refer to the HP Select Access Policy Builder Guide for details of creating Authentication Services.

The following is an example of how to configure an authentication service for HPBPI:

1. Right-click the Select Auth icon adjacent to the HPBPI resource that you want to configure.
2. Select the option to `Enable Select Auth` from the menu.

   A dialog for `Authentication Properties` is displayed.
3. Click the `Add` button to add the authentication service that you want to use for the resource. There is a set of available services to select from in the left-hand pane of the new dialog.
4. Select the authentication service that you want to use for the resource, for example, `Password` and move it from the available services to selected services using the `Add >` button.
5. Click `OK`.

Full details of the authentication services available using Select Access are documented in the *HP Select Access Policy Builder Guide*.

The HPBPI resources that appear under `http` in the Resources Tree support all the Select Access authentication services; however, the HPBPI resources that appear under `rmi`, support only the password-based authentication, such as `password` and `Kerberos`.

Configure Identity Access to HPBPI Components

You have set up the authorization requirements for the HPBPI components that you have added to the Select Access Resources Tree in the Policy Builder.

You now need to grant the user, or users, that you have configures access to these components. To do this you need to update the entries in the Policy Administration grid as follows:

1. Right-click at the intersection of the Identity and the HPBPI Resource where you want to enable authentication.
2. Select `Allow Access` from the menu offered.

   The intersection point should now show a tick, rather than the cross that was previously displayed.

If the grid location still shows a cross, it might be because you have not granted an authorization service for the resource. (As described in section Configure the Select Access Authentication Service on page 247.)

When you have granted access to all the Identities for all the HPBPI Resources, make sure that you have cleared the validator cache as follows:

```
Tools | Clear Validator Cache(s)
```

Select Access configuration is now complete for this example.

You now need to enable Select Access authorization within HPBPI to complete the configuration.

Enabling Select Access Authorization within HPBPI

You enable Select Access authorization for HPBPI using the HPBPI Administration Console as follows:

1. Start the HPBPI Administration Console from the Start Programs menu.
2. Select the `Security` option from the left-hand pane.
3. The HPBPI options for configuring authorization are shown in the right-hand pane.

   Full details of all the Security options are documented in Chapter 2, HPBPI Component Administration.
4. To enable authorization for all the HPBPI components that you have configured using Select Access, with the exception of the Business Process Dashboard, select the `Use Select Access` authorization method under `Authorization settings`.

   !! Any HPBPI component not configured within Select Access or not assigned an authorization service within Select Access is inaccessible. When you attempt to logon to these components, you are presented with an error indicating that access is denied and that HPBPI is unable to authorize user credentials.
5. Enter the name of the Policy Enforcer configuration file that you set up when you configured the Servlet Enforcer Plugin. (You can browse to the location of the file.)
6. To separately enable Select Access authorization for the Business Process Dashboard, check the `Use authorization for Dashboards?` option under the `Authorization settings`.
7. Click the `Apply` button to apply your changes.
8. Stop and restart all the HPBPI components from the Administration Console `Status` option to complete the configuration.

   You must stop and restart all the HPBPI components when you change the security options, if you do not, some of the security settings can be compromised.

Select Access authorization is now complete. When you next start the components that you have configured using Select Access, you must use the credentials, or authorization method, configured through Select Access.

Displaying User Details Within the HPBPI Web Clients

When using Servlet Engine authentication, the HPBPI Web-based clients show the user details for the user that is currently logged in to the client. This information is displayed in the top right-hand corner of the Web page.

When you configure HPBPI to use Select Access Authentication this information is displayed only if you enable the Personalization option in the Select Access Policy Builder as follows:

1. Start the Policy Builder and locate the HPBPI Resource that you want to enable the service for in the `Resources Tree`.
2. Right-click the Select Auth icon adjacent to the HPBPI resource that you want to configure.
3. Select the option `Select Auth Properties` from the menu.

   A dialog for `Authentication Properties` is displayed.
4. Select the Personalization tab on the dialog.
5. Select the checkbox `Store identity attributes in:`
6. Click the `Add` button to add new `Identity Data`.
7. Select `uid, userid` from the drop down menu under `Directory Attribute Name`.
8. Type `USERID` as the `Environment Variable Name`.
9. Click `OK`.

The user information is now displayed in the HPBPI Web-based clients when Select Access is configured as the method of authorization.

HPBPI Web Client Customized Login Forms

Select Access provides default forms, which are presented when you log into the HPBPI Web clients after configuring Select Access authorization.

You can change these forms and use a set of login forms, which are provided with HPBPI. These forms are based on the HPBPI Web client style.

To use the HPBPI customized forms, complete the following steps:

1. Locate the directory where the Servlet Enforcer Plugin default forms are located as follows:

   enforcer-plugin-install-dir`\content`

2. Make a copy of the following (or all) Select Access files in the `content` directory, before you overwrite them with the HPBPI-specific files:

   ```
   accepted.html
   deny.html
   login_form.html
   password_change_form.html
   password_expired_form.html
   password_expiry_warning_form.html
   password_history_match_form.html
   password_invalid_length_form.html
   password_missing_chars_form.html
   password_username_match_form.html
   registration_form.html
   winauth_form.html
   ```

3. Locate the HPBPI customized Select Access forms in the HPBPI installation directory:

   bpi-install-dir>\data\conf\bia\SelectAccessForms

4. Copy all the HPBPI files located in the `SelectAccessForms` directory to the `content` directory under the Servlet Enforcer Plugin installation:

   enforcer-plugin-install-dir`\content`

You now need to configure the Servlet Enforcer Plugin to ignore specific files as follows:

1. Locate the `Ignored Filenames` dialog in the Select Access Setup Tool. This dialog can also be accessed from the Policy Builder `Tools` menu option. (`Tools|Component Configuration`)

   Make sure that you select the correct Enforcer Plugin configuration file.

2. From the `Ignore Filenames` dialog select the following check boxes:

   — `Ignore GIF images (*.gif)`

   — `Ignore JPEG images (*.jpg and *.jpeg)`

3. From the `Ignore Filenames` dialog add `*.css` to the list of `Ignored Filenames` using the `Add` button.

The Enforcer Plugin file is then updated with this new information.

When you next login to one of the HPBPI Web-based components, the new HPBPI-specific files are used for authorizing the HPBPI Web-based components in place of the Select Access default forms.

Using Select Access to Control Access to Individual Flows

In addition to securing access to the HPBPI components using HP Select Access, you can also use Select Access to control which Business Flows are presented to particular users through the Business Process Dashboard.

The process for controlling access to flows is very similar to the process that you have already used to secure access to the HPBPI clients, using the Select Access Policy Builder.

You can control which Business Flows are presented to a specified user through the Business Process Dashboard; however, you cannot control which Services are presented. This means that when viewing HPBPI Business Flows through the Dashboard, you will always see the full list of Services, and some of them might not relate to the Business Flows that are listed.

Before configuring access to your Business Flows, you need to make sure that you have configured authorization for the Business Process Dashboard using Select Access as described in Configure the Select Access Authentication Service on page 247.

In order to set up authorization for your Business Flows, you create a `Flows` resource under the entry for each Dashboard resource in the Policy Builder. This `Flows` resource enables you to choose the method that you want to use for access to individual Business Flows through the Business Process Dashboard.

Select Access has an option to enable authorization settings to be inherited. This means you set the authorization for `Flows` resource to allow access to all Business Flows, or deny access to all Business Flows. You can then selectively allow or deny access to specific flows on an individual user basis by adding specific flows under the `Flows` resource and configuring them individually.

If you deny a user access to the `Flow` resource, the Business Process Dashboard does not present any flows, unless you explicitly add them to the Resources Tree under the `Flows` resource and enable access to them for each user.

If you give full access to the `Flows` resource, all business flows are shown through the Business Process Dashboard, unless you explicitly add them to the Resources Tree under the Flows resource and deny the user access to them.

The method that you choose depends on the policy for allowing access to the Business Flows within your organization.

When you have an entry for the Business Process Dashboard, you can continue and add Business Flows as follows:

1. Open the Select Access Policy Builder and find the entry for `hpbpi-bpd` under the required machine name in the Resources Tree.
2. Right-click the `hpbpi-bpd` entry and select `Add Resource` from the menu.
3. Enter `Flows` as the resource name.
4. In the next step you need to add the flows that you want to selectively allow or deny access for. The flows that you add and the options that you select depend on the method of authorization that you are using at the `Flow` resource level.
5. Right-click the `Flows` resource and select `New|Resource` from the menu options.
6. In the `New Resource` dialog type the name of the resource, which must match the name of the Business Flow as it is defined in the Model Repository. This is the name that is shown in the HPBPI Modeler.

   The name is case insensitive.

   When you have populated the Resources Tree with the appropriate business flows, you can set the authorization levels as required for your implementation.

You have now completed the steps to control access to your Business Flows. When different users start the Business Process Dashboard, they should only see the Business Flows that have been configured for them through the Select Access Policy Builder.

Using Certificate-based Client Authentication

If you have enabled certificate-based client authentication for HPBPI Web clients, you can also use Select Access to further refine the level of authorization for the HPBPI Web clients. Enabling certificate-based client authentication is described in section Tomcat and HTTP Over SSL on page 265.

Certificate authentication is another service offered by Select Access. Select Access can validate a Web client certificate against a CA Certificate in its LDAP server. Select Access can also validate a user Certificate in LDAP server if appropriately configured. This enables you to configure a more refined way of controlling access to resources for those users that have a valid certificate.

The following are the steps that you need to complete in order to use the Select Access Certificate authentication service with HPBPI components. You complete these steps using the Select Access Policy Builder:

1. Set up a new service that the Policy Validator can use to authenticate identities with a certificate.

   The LDAP server requires two identity entries; one to represent the CA Certificate and one to represent the client Certificate. The identity for the CA certificate must contain an entry with a `certificationAuthority` object class and a `caCertificate` attribute. If it does not, it is unable to locate the certificate authority policy required to complete the authentication. Refer to the Select Access documentation for details of how to configure Select Access to support Certificates.

2. Make sure that you enter Certificates in the LDAP server for the CA Certificate using DER encoding.

3. Make sure that the Identity entries for the LDAP server match the `cn`, `ou`, `o` and `c` attributes supplied when creating the Certificate.

4. If you configure the client Certificate to be stored in the Select Access LDAP server, you need to enter the Certificate using DER format.

5. Configure the required HPBPI HTTP resources in the Select Access Resource Tree and enable the `Certificates` authentication service. You do this in the same was as you configured the Select Access configuration service in section Configure the Select Access Authentication Service on page 247.

6. Configure the individual Identities in the Identity Tree to allow or disallow access to the HPBPI resource as previously described.

If you want to use this level of authentication with HPBPI components, you also need to make sure that the `Authorization settings` are set as required in the HPBPI Administration Console. These settings appear under the `Security` option.

9 Servlet Engine Authentication

This chapter describes how to use the Memory Realm mechanism within the Servlet Engine (Tomcat) to modify the login details for the HPBPI Web-based interfaces. It also describes how to configure the use of HTTP over SSL for HPBPI.

- Tomcat Realm Mechanism on page 259
- Tomcat and HTTP Over SSL on page 265

You can also use HP Select Access to authorize access to HPBPI components. This is described in Chapter 8, Select Access Authorization.

The HPBPI components that can be configured for authentication using the Servlet Engine are:

- Model Repository, which affects both the Repository Explorer and the Modeler.

  Setting authentication for the Model Repository also impacts any Contributed components that use the repository.
- Intervention Client
- Notification Server Administration Console
- Metric Definer
- Business Process Dashboard
- Service Desk Process Insight Dashboard

With the exception of the Dashboards, each of these components requires you to provide a username and password in order to access the Web screens. Following a new HPBPI installation, the username and password for these components are predefined as:

- Username: `admin`
- Password: `hpbpi`

In the case of the Business Process Dashboard, you do not need to supply a username and password unless you have enabled the option to use authorization for the Dashboard in the Administration Console. This setting is available in the Security settings; see Component Configurations - Security on page 123.

In addition, you can use the following username and password for the Business Process Dashboard and the Service Desk Process Insight Dashboard. Use these credentials when you want users to have access to the Dashboards only, and not the remaining HPBPI Web-based components:

- Username: `dashboard`
- Password: `console`

Tomcat Realm Mechanism

HPBPI uses Tomcat as its Servlet Engine to present JSPs to the user, through a Web browser; starting the Servlet Engine is described in section Starting and Stopping the HPBPI Server Components on page 27. For authentication, HPBPI uses the Tomcat Realm mechanism, specifically it uses the Memory Realm mechanism, which is more fully described in the Tomcat documentation at the following URL:

```
http://jakarta.apache.org/tomcat/tomcat-5.0-doc/realm-howto.html
```

Within the Tomcat documentation, a Realm is described as a database of usernames and passwords, which identify valid users of a Web application (or set of Web applications), plus a list of roles associated with each valid user.

Within HPBPI the following roles have been defined:

- `ovbpi-model-repository`, which is the role for the Repository Explorer and the Modeler; see Chapter 3, HPBPI Modeler Administration and Chapter 5, Repository Explorer.
- `ovbpi-notify-admin`, which is the role for the Notification Server Web Administration Console; see Chapter 6, Notification Server Configuration.
- `ovbpi-int-client`, which is the role for the Intervention Client; see Chapter 7, Intervention.
- `ovbpi-metric-definer`, which is the role for the Web-based Metric definition interface; see Chapter 4, HPBPI Metric Definer Administration.
- `ovbpi-dashboard`, which is the role for the HPBPI Business Process Dashboard; refer to the *HPBPI Integration Training Guide - Customizing the Business Process Dashboard* for more information.
- `ovbpi-sdpi-dashboard`, which is the role for the HPBPI Service Desk Process Insight interface; refer to the *HPBPI Integration Training Guide - Monitoring Service Desk* for more details of the Service Desk Process Insight Dashboard.
- `ovbpi-event-injector`, which is for a contributed tool that is provided on the HPBPI media.

There is an XML file called `tomcat-users.xml` that defines the credentials for these roles. The file is located at:

HPBPI-install-dir/nonOV/jakarta-tomcat-5.0.19/conf/
tomcat-users.xml

Following a new installation, the file has the following content:

```
<?xml version='1.0' encoding='utf-8'?>
<tomcat-users>
  <role rolename="ovbpi-model-repository"/>
  <role rolename="ovbpi-dashboard"/>
  <role rolename="ovbpi-int-client"/>
  <role rolename="ovbpi-notify-admin"/>
  <role rolename="ovbpi-metric-definer"/>
  <role rolename="ovbpi-sdpi-dashboard"/>
  <role rolename="ovbpi-event-injector"/>
  <user username="dashboard" password="console"
roles="ovbpi-dashboard,ovbpi-sdpi-dashboard"/>
  <user username="admin" password="hpbpi"
roles="ovbpi-notify-admin,ovbpi-int-client,ovbpi-event-injector,
ovbpi-metric-definer,ovbpi-model-repository,ovbpi-dashboard,ovbp
i-sdpi-dashboard"/>
</tomcat-users>
```

Following a new installation, there are two sets of login credentials specified. One set for the Dashboards and one set for the remaining Web-based interfaces. Again, following a new installation, the Dashboard does not require you to login; you can change this behavior from the HPBPI Administration Console.

You are strongly advised to assign different credentials to the different roles in order to increase the security of your system. This is particularly the case for the Intervention Client, from where you can delete active flow instances and data instances.

Modifying Login Credentials

To assign different credentials, you need to edit the XML file that defines the roles and the login credentials. The following is an example of how you can do this:

```
<?xml version=’1.0’ encoding=’utf-8’?>
<tomcat-users>
  <role rolename="ovbpi-event-injector"/>
  <role rolename="ovbpi-int-client"/>
  <role rolename="ovbpi-notify-admin"/>
  <role rolename="ovbpi-metric-definer"/>
  <role rolename="ovbpi-model-repository"/>
  <role rolename="ovbpi-dashboard"/>
  <role rolename="ovbpi-sdpi-dashboard"/>

  <user username="inject" password="inject"
```

```
    roles="ovbpi-event-injector"/>
  <user username="interv" password="secret"
    roles="ovbpi-int-client"/>
  <user username="notif-admin" password="admin"
    roles="ovbpi-notify-admin"/>
  <user username="metric" password="secret"
    roles="ovbpi-metric-definer"/>
  <user username="model-repository" password="admin"
    roles="ovbpi-model-repository"/>
  <user username="bpi-dashboard" password="admin"
    roles="ovbpi-dashboard"/>
  <user username="spdi-dashboard" password="admin"
    roles="ovbpi-spdi--dashboard"/>
</tomcat-users>
```

In this example, the following clients now have a separate username and password:

- The `ovbpi-event-injector` role for the contributed tool has a username and password of `inject`.
- The `ovbpi-int-client` role for the Intervention Client has a username of `interv` and a password of `secret`.
- The `ovbpi-notify-admin` role for the Notification Server Web Administration Console has a username `notif-admin` with a password of `admin`.
- The `ovbpi-metric-definer` role for the Metric Definer interface has a username of `metric` and a password of `secret`.
- The `ovbpi-model-repository` role for the Model Repository (used within the HPBPI Modeler) has a username `model-repository` with a password of `admin`.
- The `ovbpi-dashboard` role has a username `bpi-dashboard` and a password `admin`.
- The `ovbpi-spdi-dashboard` role has a username `spdi-dashboard` and a password `admin`.

To make changes to the `tomcat-users.xml` file, complete the following steps:

1. Make sure that you have logged out of the HPBPI Modeler, Notification Server Administration Console and the Intervention Client.
2. If you are using the Event Injector contributed component, you must also close the Event Injector.
3. Open `tomcat-users.xml` in a text editor:

   ```
   bpi-install-dir/nonOV/jakarta-tomcat-5.0.19/conf/
   tomcat-users.xml
   ```
4. Make your required changes to the file.
5. Save and close the `tomcat-users.xml` file.
6. Make your changes to the `Security` settings in the Administration Console as required; for example, for the Business Process Dashboard.
7. Stop and restart all the HPBPI components from the `Status` pane of the HPBPI Administration Console for the changes to take effect.

   You must stop and restart all the HPBPI components when you change the security options, if you do not, some of the security settings can be compromised.

Memory Realm is a simple implementation of the Tomcat 5 Realm interface. It is described in the Tomcat documents as a demonstration implementation and not designed for production use.

Modifying Servlet Engine Files

If you intend to modify the Servlet Engine configuration files, you need to be aware that changes made directly to the Servlet Engine files are overwritten by HPBPI configuration files. As an example, if you want to configure another form of Tomcat Realm security, you would expect to make the changes in the `web.xml` and `server.xml` files under the following directory:

`bpi-install-dir``\nonOV\jakarta-tomcat-5.0.19\conf`

However, before making any changes to these files, you need to be aware of how HPBPI manages configuration changes to the Servlet Engine files. HPBPI overwrites the Servlet Engine configuration files in order to add HPBPI-specific configuration details and as a result of your making changes to configuration parameters using the Administration Console. HPBPI overwrites the Servlet Engine configuration files whenever a change is made to any parameter using the Administration Console, it does not have to be a Servlet Engine specific change.

If you want to make changes directly to the Servlet Engine configuration files, you need to make the modifications in the HPBPI-specific configuration files and then make sure that HPBPI propagates these modifications to the Servlet Engine files.

HPBPI has a number of configuration files and the Servlet Engine files related to `web.xml` and `server.xml` are located under the following directory:

`bpi-install-dir``\newconfig\DataDir\conf\bia\tomcat\conf`

These files are database specific and you need to modify the files that are appropriate to your configuration.

If you are using Oracle, you need to modify the following files:

```
bpi-install-dir\newconfig\DataDir\conf\bia\tomcat\conf\
                                      server.engine.oracle.xml
bpi-install-dir\newconfig\DataDir\conf\bia\tomcat\conf\
                                      server.views.oracle.xml
```

If you are using SQL Server, you need to modify:

```
bpi-install-dir\newconfig\DataDir\conf\bia\tomcat\conf\
                                        server.engine.mssql.xml
bpi-install-dir\newconfig\DataDir\conf\bia\tomcat\conf\
                                        server.views.mssql.xml
```

When you have made the required changes to these files, you need to artificially make a change to any parameter using the Administration Console and apply the change. As a result of this change, the Administration Console updates all the relevant configuration files, including the Servlet Engine files, and your modifications are then propagated to these files.

Tomcat and HTTP Over SSL

HTTP over SSL (secure socket layer) is a secure version of HTTP. SSL enables the data from a client, such as a Web browser, to be encrypted prior to transmission to prevent unauthorized access to the data being transmitted.

Using SSL, the Web Server and the Web browser (client) establish a secure connection, which enables them to trust each other and send data knowing that the connection is secure. In order to do this, a level of authentication is required and this is provide through Certificates. You obtain Certificates from a Certificate Authority (CA).

You can configure HTTPS for the Tomcat Servlet Engine installed with HPBPI and then enable the use of HTTPS for your HPBPI Web applications using the Administration Console. Note that the use of SSL is between the HPBPI Servlet Engine and the HPBPI Web applications only. Other HP BTO Software products can also be configured to use SSL (for example OVIS); however, HPBPI can not be configured to use SSL as a means of communicating with these other HP BTO Software products.

There are two aspects of configuring HTTPS for HPBPI, one aspect is related to the Servlet Engine configuration and one to the Web client configuration. These are described in the following sections.

Servlet Engine Configuration

In order to configure the Tomcat Servlet Engine to support HTTPS, you need to obtain an SSL certificate. SSL uses public key cryptography, which is based on key pairs. These key pairs contain one public key and one private key. These keys are used by the Servlet Engine to enable HTTPS on the machine where the HPBPI components are running. When installed the certificate encrypts the data that is sent from the Servlet Engine to the Web client, or Web browser.

There is more than one form of keystore, which is a repository of certificates used for identifying a client or a server. The Java Keytool generates a key based on the JKS format of keystore. There is also a PKCS12 format of keystore; however, this cannot be used with HPBPI. You can use only the JKS format as generated from the `keytool` command.

The following are ways that you can obtain, or generate, a certificate:

1. Using the Java Keytool

   Keytool is a key and certificate management utility that ships with your Java distribution. It enables users to administer their own public/private key pairs and associated certificates for use in self-authentication or data integrity and authentication services, using digital signatures.

   Use the `keytool` command, from a Command Window, to generate a Java Key.

2. From a Certificate Authority

   You can obtain SSL certificates from Certificate Authorities (CA) such as Verisign.

   In order to obtain and SSL certificate from a CA, you first need to generate a Certificate Signing Request (CSR) for the Certification Authority to sign. You can do this using the Java Keytool. When generating this certificate make sure that you supply the fully qualified domain name of the machine where the HPBPI is installed as the common name (cn) attribute.

   The CSR certificate and the certificate chain received from the CA need to be stored in a JKS keystore file.

When you have obtained all the required information and configured the Servlet Engine to send HTTPS requests, you can use the Administration Console to enable HTTPS as the protocol to be used to send data to one or more of the HPBPI Web Clients. This is done through the `Security` options; see section Component Configurations - Security on page 123.

Web Client Certificate-Based Authentication

You can also use the Servlet Engine to provide certificate-based client authentication, based on HTTPS, for your Web applications:

- Business Process Dashboard
- Repository Explorer
- Metric Definer
- Intervention Client
- Notification Server Administration

In order to use client authentication, you first need to configure the Servlet Engine to be able to send HTTPS requests to the Web Browser, as described in section Servlet Engine Configuration on page 265. You need to configure the Servlet Engine on each system where it installed. The following installation types also install and use the Servlet Engine:

- Server and Modeler
- Server Only
- Dashboards Only

You then need to obtain a client certificate, which is signed by a CA, plus a CA root certificate.

In order for the Servlet Engine to be able to receive encrypted data from the Web Browser, you need to enable client authentication to ensure only Web browsers presenting a valid certificate can authenticate with the Servlet Engine.

Each CA provides a root certificate, which you install, or import, for your Web Server (in this case the Tomcat Servlet Engine). The Servlet Engine uses this certificate to determine whether or not it trusts the data being sent from the Web Browser. You obtain this root certificate from your CA. When you have the certificate, you import it into the keystore for your Java distribution. The root certificate needs to be imported so it can become a trusted certificate, enabling a chain of authorization to be established between the Web Server and the Web Browser.

The keystore location for the Servlet Engine is as follows:

```
Java-home\jre\lib\security\cacerts
```

The Administration Console allows you to select a keystore file from any location; however, the Tomcat Servlet Engine requires the keystore to be located on a local hard drive. The keystore must not be located on a virtual or network drive.

You import the certificate using the `keytool` utility that comes with your Java distribution, using the following command:

```
keytool -import -keystore java-home/jre/lib/security/cacerts
-storepass changeit -alias rootca -file rootca.pem -trustcacerts
```

where:

- *`java-home`* is the directory where your Java distribution is installed.
- `changeit` is the default password for the Java security keystore for a new Java installation.

Type `yes` when prompted if you trust this certificate

When you have imported the root certificate on the Web Server, you then need to import both the root certificate and the client certificate on each Web Browser where you intend to use the HPBPI Web-based interfaces (or those for which you want to enable HTTPS). Follow the instructions in you Web Browser documentation for installing Certificates.

When you have completed the configuration to set up your Web Browser to Web Server HTTPS connection, you can use the Administration Console to enable certificate-based client authentication. This is done through the `Security` options; see section Component Configurations - Security on page 123. Client authentication is used only for those clients where you have also enabled HTTPS as described in section Servlet Engine Configuration on page 265.

10 High Availability Using Microsoft Clusters

A highly available system is one that enables applications and services to continue after hardware and software failures, thus reducing the time when the system is not available for use; also known as downtime. You can have planned downtime, where you have set aside time for maintenance tasks, such as upgrading, and you can have unplanned downtime. Unplanned downtime is where the system fails due to events such as hardware and software failures.

High availability in terms of HPBPI is where your HPBPI system is always available when it is planned to be available and fail over mechanisms are in place to minimize, or eliminate, unplanned downtime.

A Microsoft Server Cluster can provide some of the high availability capabilities, but it cannot provide a complete solution for high availability with no single point of failure. You need to make sure that your HPBPI deployment includes other protection mechanisms for disks, networks, power supplies, memory, controllers and systems; for example:

- dual power supplies to HPBPI systems
- dual paths from system to disks
- dual network controllers
- uninterruptable power supplies (UPS)
- disk mirroring and (or) the use of redundant arrays of inexpensive disks (RAID)

In other words, to achieve the maximum availability of your system, you need to build in redundancy at all levels.

You can install HPBPI in a Microsoft Server Cluster environment in order to provide a high-availability solution for HPBPI. By using the cluster technology when one node in a cluster fails or is taken offline another node in the cluster takes over the failed node's operations. When HPBPI functions are moved from one node in the cluster to another node, users can experience some temporary interruption in their use of HPBPI, but this interruption is minimal.

Modes of High Availability

There are two modes of high availability that are offered by Microsoft:

- Active / Passive

  In this mode, one node in the cluster hosts an application at any given time; this can be likened to having a standby system in place.

- Active / Active

  In this mode, different instances of the same application are running concurrently on different nodes in the cluster. In this mode, when a node fails the remaining nodes in the cluster need to take on additional load and this can impact performance.

HPBPI must be deployed in the Active/Passive mode within the cluster. You cannot use the Active/Active mode with HPBPI.

By supporting Microsoft Server Clusters in an Active/Passive mode, HPBPI Services are automatically restarted on an alternative machine, if the machine on which they are currently running fails for any reason.

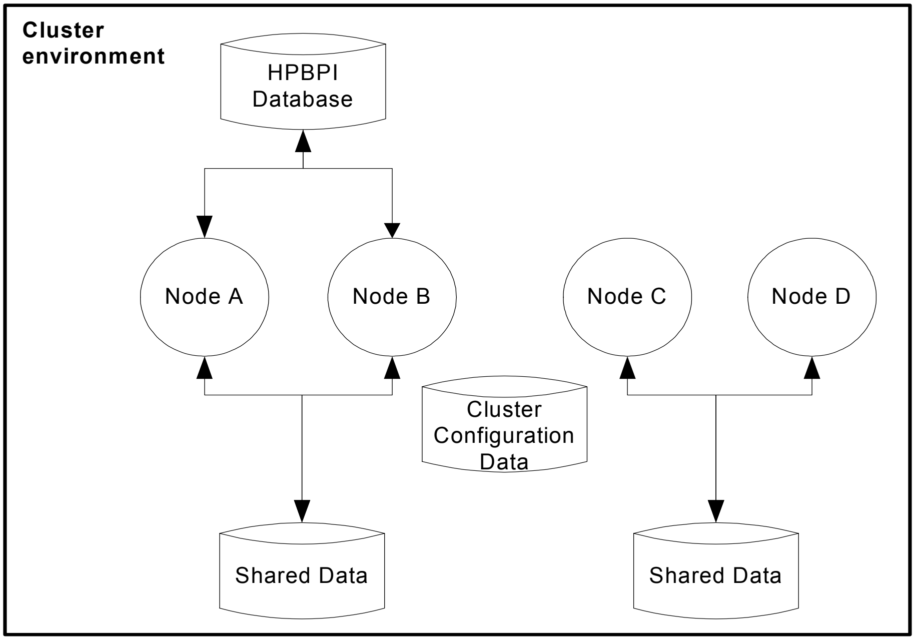
A Microsoft Server Cluster can support up to eight nodes; however, in an Active/Passive mode, HPBPI requires only two nodes to be configured within the cluster. You can configure additional passive nodes for HPBPI to keep to a minimum the single points of failure, but it is not needed in order for HPBPI to operate within the Server Cluster.

Each node within the cluster can host one or more resource groups. A resource group is a collection of components that are managed and monitored as a single unit within the cluster; for example, a public file share, a Web server, and a database application can all be managed as resources. HPBPI is managed as a Resource Group within the cluster. The use of Resource Groups enables you to manage the order in which resources are started within the cluster, and dependencies between resources. This then enables you to specify specific dependencies for HPBPI; for example, the dependency between HPBPI and its database.

The nodes and the resources within the cluster are constantly monitored and in the event of a failure on the node where HPBPI is running, the workload of the failed node is re-started on another designated node. Note that there is an interruption to service while HPBPI is in the process of failing over and following the failover, the HPBPI Modeler needs to be restarted as it has lost its connection to the Model Repository.

In Figure 7 Node A and Node B are both configured to run HPBPI in Active/ Passive mode. Node A is currently hosting HPBPI. If Node A fails for any reason, the cluster can start HPBPI on Node B.

Figure 7 HPBPI Operating on Two Nodes in a Cluster



In order for the failover to be successful, both Node A and Node B must have access to the HPBPI data. This includes the installation data and configuration files on disk, plus all the Flow, Metric and Events data in the database. In Figure 7 it is only Nodes A and B that have this access. Node C and D are also operating in the cluster but are not available for HPBPI failover.

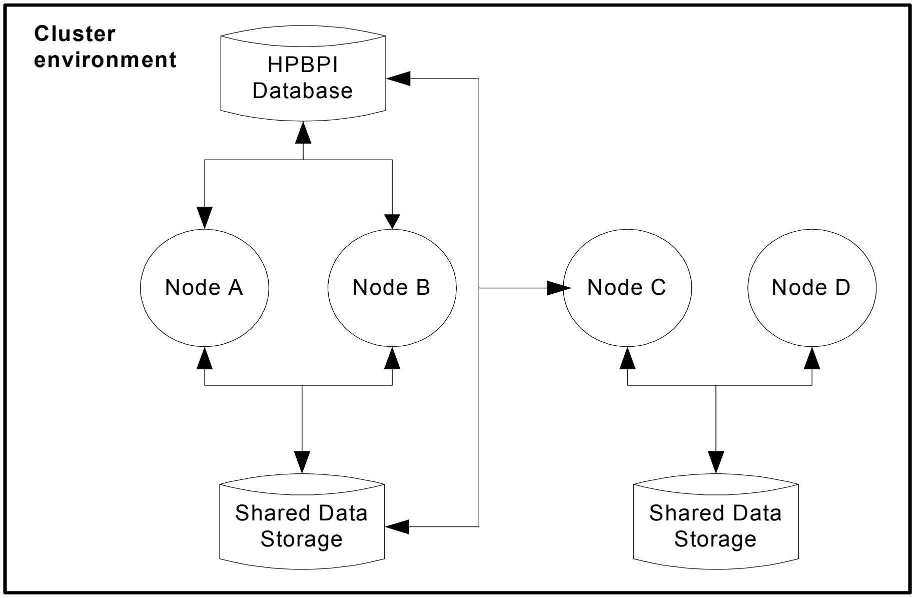
The HPBPI shared data device is a resource that needs to be managed by the cluster and must therefore reside within the bounds of the cluster. The HPBPI database might or might not be managed by the cluster, depending on how you have configured your system. In this case, the device where the database is located might or might not be managed as part of the cluster.

The documentation for the Microsoft Server Cluster provides details of the hardware configuration require, plus recommendations for using technology such as RAID for the shared storage to minimize the single points of failure.

There is also a cluster configuration database that must be accessible to all the nodes in the cluster and which contains information such as the membership of the cluster, the configuration data and quorum information.

You can have additional nodes in the cluster made available for HPBPI failover if you want to minimize the your single points of failure. In Figure 8 HPBPI has also been installed and configured on Node C, which means that either Node B or Node C can be used as a failover system if needed. Note that Node C is also available as a failover node for another application whose data is stored on a different device to that used by HPBPI.

Figure 8 HPBPI Operating on Three Nodes in a Cluster



Configuring HPBPI Within a Microsoft Cluster

This section describes the HPBPI-specific tasks that you need to complete in order to have HPBPI operate within the Microsoft Server Cluster environment. The section assumes that the Server Cluster has been configured to meet your organizational requirements; it does not provide any information relating to the installation and setup of the cluster. Refer to the Microsoft Server Cluster documentation for this information.

In order to have HPBPI operate within a cluster environment, it is necessary to install OVPBI on each node in the cluster that you want HPBPI to run on. This does not mean you have to install HPBPI on every node in the cluster; see section Modes of High Availability on page 271.

You need to install HPBPI multiple times, even though each node has access to the same shared install directory and database. This is to ensure that the HPBPI Windows Services are created on every node where HPBPI is to run. You also need to install and register the HPBPI license on each node on the cluster where HPBPI is to run; see section Licenses for HPBPI in a Cluster Environment on page 278 for more information about licensing HPBPI in a cluster environment.

If you do not install on each node, it is not possible to configure the HPBPI Windows Services as generic service resources for the cluster. This is because the Cluster Administrator checks that each generic service exists on every node where the resource is configured in order to restart it in the event of a failover.

Groups

HPBPI installs a number of Windows services, therefore one way to configure HPBPI within the cluster is to set up the services using the Generic Service resource type; for example, resources can be grouped as self-contained resource groups, called Groups. How you configure a resource within the cluster is outside the bounds of this document. You need to read the information provided by Microsoft to determine the best configuration for your requirements.

The following are suggested scenarios for grouping the HPBPI resources:

1. Configure a self-contained resource group for HPBPI resources and have the database outside the cluster environment. The means that if there is a catastrophic cluster failure, the HPBPI data is still accessible for browsing and for analysis.

   This configuration enables a specific disk drive (or partition), IP address and virtual server name to be assigned to HPBPI and HPBPI can then be moved between nodes as an independent resource.

2. Configure a self-contained resource group for HPBPI resources and have the database installed as a cluster application. You then configure the HPBPI resources to be in the same resource group as the database.

   In this case, HPBPI can share the disk, virtual server name and IP address of the database. This enables you to define specific dependencies between HPBPI and the database and the database and HPBPI can be treated as a single entity and operates as described in scenario 1. In addition, you need only allocate one disk and host name.

3. Configure a self-contained resource group for HPBPI resources and have the database installed as a cluster application. You then configure the HPBPI resources to be in a different resource group to the database.

   By allocating different resource groups to HPBPI and the database, each can failover independently of the other, which minimizes the potential downtime of the HPBPI system. In addition, HPBPI can be moved between nodes as an independent resource as in scenario 1.

➤ Note that when a specific physical disk partition is assigned to HPBPI, it cannot be shared by another resource group.

You also need to assign the following resource types to HPBPI:

- An IP Address resource
- A Network Name resource

These resource types become the virtual server name and address that HPBPI clients must use to access the HPBPI Server within the cluster. This virtual server name is valid for whichever node in the cluster HPBPI is configured to run on. As an example, you configure the Business Process Dashboard to use the virtual server name in its URL. This enables the Dashboard to access the HPBPI data on whichever physical node in the cluster HPBPI is currently running on.

Having all the HPBPI resources in the same resource group enables you to define dependencies between the HPBPI services, and the disk and network resources. This means that the cluster manager does not try to start HPBPI unless all the required resources are available.

When you have identified how you want to manage the HPBPI resources within the cluster, use the Cluster Administrator to create the new resource groups.

Summary of Configuration Requirements on page 276 provides a summary of the steps that you need to take to set up HPBPI within a Microsoft cluster environment.

Summary of Configuration Requirements

The following is a summary of the steps that you need to complete in order to install and configure HPBPI on an existing Microsoft Server Cluster:

1. Make a decision on how you are going to configure the HPBPI Services as resource groups; that is, whether they have a dedicated resource group, or whether the HPBPI services are added to an existing resource group.
2. If HPBPI is to have a dedicated resource group, allocate a shared physical disk for HPBPI.

   If HPBPI is to share an existing resource group, you can choose to allocate a different (and unique) shared physical disk for HPBPI, or use the shared physical disk already defined for the group. As an example, there might be a resource group already allocated for the database and you can use the same shared physical disk or allocate a different one.
3. Allocate an IP address and network name for HPBPI. If HPBPI is sharing a resource group, these details already exist. If you are creating a resource group specifically for HPBPI, you need to allocate a unique IP address and network name.
4. If you are configuring a new resource group for HPBPI, use the Cluster Administrator to create a new resource group. In this new group, create the resources for the allocated physical disk, the HPBPI IP address and network name.

5. Ensure that the version of the J2SE required by HPBPI is accessible to each node of the cluster where you want to install and run HPBPI.

   To avoid multiple installations, you can install the J2SE on the shared disk, where it is accessible to all the HPBPI instances operating in the cluster. You can also choose to install it on a disk that is local to each node, in this case, the installation path must be the same on every node, for example `c:\Program Files\java`.

6. From a node in the cluster where the physical disk resource that you want to use for HPBPI is active, install HPBPI. You need to provide the following, cluster specific, information as part of the installation:

   a. When the HPBPI installer prompts for the installation directory, enter the drive letter of the allocated physical disk for HPBPI.

   b. When the HPBPI installer prompts for the database host name and the database is installed within the cluster environment, use the virtual network name allocated to the database; do not use `localhost` as the hostname.

7. For each of the other nodes where you want HPBPI to be able to run (or failover to), use the Cluster Administrator to make the disk for HPBPI active on that node. When the disk is active, you can install HPBPI on the node. These subsequent installations are identified and treated as reinstallations by the HPBPI installation procedure. The installer detects the previous installation on the shared disk and presents as defaults the installation details that it finds, including the database details. This means that any existing data in the database is preserved and it is therefore possible to add new nodes to the cluster at a later time.

8. Use Cluster Administrator to create the following new `Generic Service` resources in the HPBPI resource group for the following services:

   — OVBPIAdminServer

   — OVBPIBACDataSamplesProvider

   — OVBPIEngine

   — OVBPIEventHandler

   — OVBPIMetricEngine

   — OVBPIModelRepository

   — OVBPINotificationServer

   — OVBPIServiceAdapters

- OVBPIServletEngine
- OVBPIWebServicesProvider

If you have HPBPI adapters that also have services, you can create generic services for these in the same way.

9. Set the OVBPIAdminServer resource to be dependent on the shared disk and virtual network name resources used by HPBPI. If the OVBPIAdminServer service is in the same resource group as the database, it must also be made dependent on the database server resource.
10. For each of the HPBPI services, set the resource name to be the same as the service name.
11. Set all the remaining HPBPI resources to be dependent on the OVBPIAdminServer resource.
12. On each node in the cluster where HPBPI is installed, modify the shortcuts for the HPBPI Web clients and change the hostname to be the virtual server name.
13. Make sure that any HPBPI adapters access HPBPI using the virtual server name.

You can test the cluster installation and configuration by checking that the HPBPI group can be made online using the Cluster Administrator and that the HPBPI Modeler and Business Process Dashboard can access HPBPI using the configured virtual network name.

Licenses for HPBPI in a Cluster Environment

You need to obtain and install a separate HPBPI licence on every node of the cluster where HPBPI is installed.

In the Active/Passive mode of cluster configuration, HPBPI is operational on one node only at any one time; the active node. You therefore need purchase only one set of the required business process licenses for this Active/Passive cluster configuration. However, as HPBPI license keys are linked to a specific Node Name, and you need install HPBPI license keys on each cluster node where HPBPI in installed. You therefore need to obtain additional licenses for failover (passive) nodes in the cluster.

For this purpose, a set of additional complimentary license products has been created for HPBPI, and you can order and use these licenses for the failover nodes in the cluster (for each of the passive nodes you want to configure). You need to order these additional licenses using the product numbers that correspond to the licensed product, or products, that you have already purchased.

To obtain these complimentary licenses, you need to contact the HP Password Delivery Service. You do this after you order and receive your initial HPBPI licenses. Your request for additional licenses needs to include the original order number.

The following URL takes you to the HP Password Delivery Service Web page:

http://www.webware.hp.com

In the left-hand menu is a link for contacting the Password Delivery Center, where you can find contact details for your region.

Full details of the complimentary product numbers can be found in the HPBPI product Data Sheet. The Data Sheet is available from the HPBPI product page of the Management Software Web site:

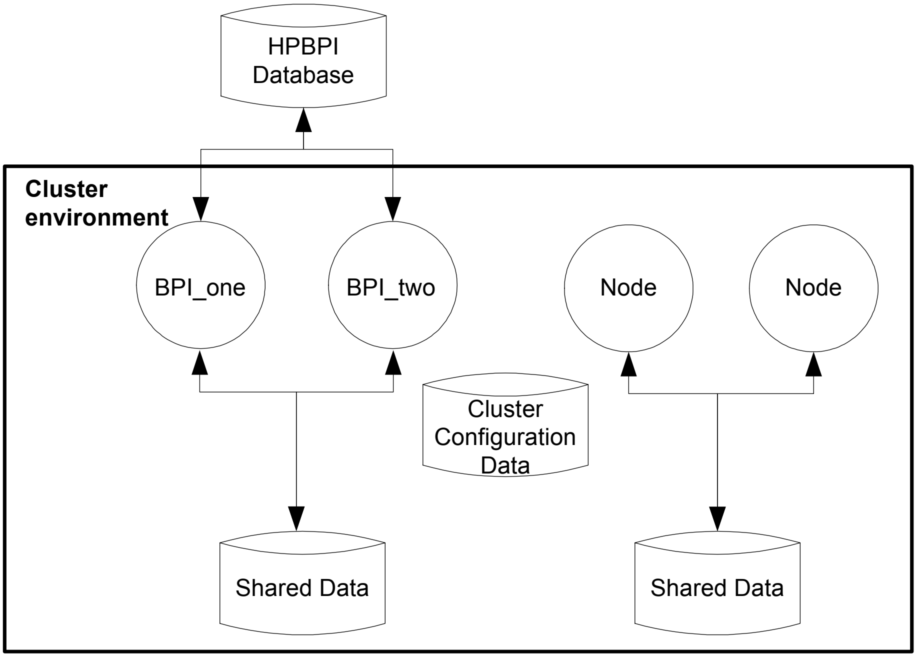
http://www.hp.com/managementsoftware

Section Cluster License Example on page 280 provides an example of an HPBPI cluster configuration.

Cluster License Example

HPBPI is to be installed on two nodes in your cluster. The virtual server name for this HPBPI resource BPI_apple. The node names of the physical nodes in the HPBPI cluster are BPI_one and BPI_two; see Figure 9.

Figure 9 Cluster Licensing



You purchase one HPBPI LTU for Enterprise Foundation on node BPI_one. The following are examples based on the HPBPI version 2.10 product numbers:

```
BB185AA         HP BPI Enterprise Foundation LTU
```

You also need to request one complimentary license for BPI_two as follows:

```
BB185AA-COMP    HP BPI Enterprise Foundation LTU
```

When referencing these nodes from within the cluster set up, you use the actual server names (BPI_one and BPI_two).

Use the Cluster Administrator to configure the HPBPI Resource Group on each node in turn. When the Resource Group is online, you can request and install the appropriate HPBPI license using the Administration Console on that node.

Reinstalling HPBPI In a Cluster

You can reinstall HPBPI in the cluster using the instructions in the *Business Process Insight Installation Guide*. You need only install HPBPI on one node in the cluster to reinstall, as this process updates the files and directories on the shared disk.

The only time you need to install HPBPI on all the nodes in the cluster is if there is a change to the local information, for example, HPBPI service definitions or shortcuts.

Removing HPBPI From the Cluster

Use the instructions in the *Business Process Insight Installation Guide* to remove the HPBPI files from the shared disk.

Note that you can use the uninstall procedure only for the first node in the cluster where you want to remove HPBPI. This is because the all the files and directories on the shared disk have already been removed.

The following are still present on each of the other nodes in the cluster where HPBPI is installed:

- HPBPI component shortcuts
- HPBPI services
- personal configuration files for the HPBPI Modeler and the HPBPI Administration Console

These need to be removed as required; see the *Business Process Insight Problem Solving Guide* for information about removing HPBPI Services.

Other Considerations

The following are other points that you need to be aware of when configuring HPBPI to operate within a cluster environment:

- When HPBPI is installed on each node of the cluster, the URLs used for the shortcuts to the Web applications contain the hostname of the last node to be installed. This is because the configuration files are overwritten at each installation and the name of the host being installed is added to the files.

  You need to modify these URLs manually to include the virtual server name following the last HPBPI installation.

- When you shut down the HPBPI components using the HPBPI Administration Console, the cluster server attempts to automatically restart them. You can modify this behavior from the Cluster Administrator tool; there is a property that specifies that specific resources are not restarted if they are shut down. However, if you use the Cluster Administrator and not the HPBPI Administration Console to start and stop the HPBPI components, you do not have this problem.

- If you have created a customized adaptor, using openadaptor, and the adaptor is running on a machine outside the cluster environment, there is additional information that you need to be aware of. This information is appropriate only for adaptors running outside the cluster and sending business events to the Business Event Handler installed within the cluster; the information covers configuring the adaptor to retry in order to re-establish a connection with the HPBPI cluster when there is a cluster failover. The information that you need is covered in the *Business Process Insight Integration Training Guide - Business Events* in the section describing Service Wrappers.

11 Backup and Recovery

This chapter describes the backup and recovery procedures for HPBPI.

It is important to back up your HPBPI system to ensure that the HPBPI files and data can be recovered in the event of a media corruption. It is also important to back up your system to ensure that any flows being processed and their associated data, services, and events can be recovered to a known state.

There are general high reliability configurations, which are recommended for all systems, for example, using mirror or Redundant Array of Inexpensive Disk (RAID) storage. Restoring or recovering from a backup should be a last resort. Your goal should be to make sure that the backups are never required.

This section does not cover how to set up your system for high reliability. The section focuses on the recommended procedures for backing up and recovering HPBPI data. The section also describes the behavior of the HPBPI system when specific components are shutdown, and provides recommendations for minimizing the effect of the shutdown.

If you have a system shutdown or failure, you want to keep your HPBPI system as synchronized as possible during the shutdown period. This section describes techniques for design that can help achieve this.

The following list describes some key considerations that you need to be aware of when designing your HPBPI system for effective backup and subsequent recovery:

- For planned component shutdowns:
  - Shut your HPBPI system down during quiet periods where possible.
  - Use high reliability tools where appropriate.
- Design adapters to minimize the number of events lost during a system failure.

  For most adapters, it is possible to design them such that they process events from the last time the adapter was executed. The *Business Process Insight Integration Training Guide - Business Adapters* describes how you do this.
- Minimize the number of Check Sequence arcs that you use in your flows.

  When the HPBPI system is recovered following a shutdown, it is likely to receive a burst of out-of-sequence events. You are recommended to use Check Sequence arcs only where essential. Overuse can cause the HPBPI system to generate an excess of alerts when it restarts. The *Business Process Insight Online Help for the Modeler* describes Check Sequence arcs in more detail.

Files and Directories to Backup

The following list describes the HPBPI data that is persisted within your HPBPI system and that therefore needs to be part of your backup policy:

- All HPBPI schema objects in the database.

  You specified the database details when you installed HPBPI, you need these details in order to complete your backup.

  The database for HPBPI should be backed up using the utilities and procedures provided as part of your database management system.

- Event definitions

  HPBPI stores details about Event definitions that have been deployed to the Business Impact Engine in a series of files; these files are located as follows:

  `HPBPI-install-dir\data\repository\events`

- Configuration information

  HPBPI includes configuration information, which controls how HPBPI is setup and is relatively static. This configuration information is located in the following directory:

  `HPBPI-install-dir\data\conf\bia`

- Template configuration files

  The following directory contains the template definitions for most of the configuration files for the HPBPI Server and presented (managed) within the Administration Console. When you configure a component using the Administration Console, all the configuration files in `data\conf\bia` are regenerated, based on the content of the template files located at:

  `HPBPI-install-dir\newconfig`

- Log files

  These do not need to be backed up to restore a consistent system; however, you are advised to back up the log files, which are located at:

  `HPBPI-install-dir\data\log`

- Data files

  These are the files (Class files, Java files and so on) that contain the details of the deployment status of your definitions. HPBPI includes configuration information relating to the flows and data that are deployed within the database. This information changes whenever the components of a flow are deployed to the Business Impact Engine. This information might be static in a production system, but variable in a development system where business flows are being designed, developed and tested and are therefore in a state of change.

  These data files are located at:

  `HPBPI-install-dir\data\datafiles`

- Business Event Handler

  Events received by the Business Impact Engine from the Business Event Handler are not acknowledged by the Engine until they have been successfully received. Out-of-sequence events and events that contain errors (for example, missing data) are stored in the Business Event Handler hospital.

  The Business Event Handler includes database tables for the hospital and event store. By default these tables are created as part of the same database schema as the rest of the HPBPI data, and are therefore backed up with the HPBPI data.

  For more information about the HPBPI database schemas and tables, refer to the *HPBPI Reference Guide*.

  Deployed Event definitions are located, as text files, within the HPBPI installation directory as follows:

  `HPBPI-install-dir\data\repository\events`

- Application adapter files and data

  Application adapters are those developed using the Business Event Handler component to access business data from applications within your organization. The *Business Process Insight Integration Training Guide - Business Events* describes how to build application adapters.

  It is usual for application adapters to be located close to (or co-located with) their data source, which means that they are likely to be running on a different system to the system where HPBPI is installed.

  Any data used by that application adapter that needs to be synchronized with the application data (for example buffer tables populated using database triggers or adapter history files), should be backed up as part of the backup procedures for the application. Where possible the buffer table should exist within the same database as the application data that is being monitored. This makes it easier to backup the data together and keep it synchronized.

- HP Operations Manager Adapter

  Configuration files for the HP Operations Manager adapter are located on the system where the adapter and HP Operations Manager are installed.

  For HP-UX, these files are located at:

  ```
  HPBPI_install_dir/data/conf/bia
  HPBPI_install_dir/misc/bia
  ```

  For Windows, these files are located at:

  ```
  HPBPI_install_dir\data\conf\bia
  HPBPI_install_dir\misc\bia
  ```

  Log files for the HP Operations Manager adapter are also located on the system where HP Operations Manager is installed as follows:

  `HPBPI-install-dir/data/log` (HP-UX)

  `HPBPI-install-dir\data\log` (Windows)

With the exception of the application adapter files and data, you must back up all of the data described above at the same time in order to maintain its consistency. The HP Operations Manager adapter data must be backed up when the application data is backed up.

How often you back up the HPBPI data depends on how frequently the data is changing. The data can be backed up while the HPBPI system is operational; however, you need to ensure that you do not deploy or undeploy and definitions during the backup.

As the HPBPI database cannot be synchronized with the Event definition files and data files, it is not possible to recover the HPBPI system to the point of failure from a backup. You must use the appropriate database management system tools to restore your system to the most recent backup and not to the point of failure.

As you cannot recover to the point of failure, if you have installed a database solely for HPBPI, you can consider not backing up all the transaction log files. This will reduce the amount of disk space required by the database for recovery purposes. In the case of Microsoft SQL Server, you could use the Simple Recovery model, rather than the Full Recovery model. The Simple Recovery model enables you to recover to the most recent backup. The Full Recovery model enables you to recover to the point of failure.

In the case of Oracle, you need to use ARCHIVELOG to enable online backups. If you do not want to use ARCHIVELOG, you can stop HPBPI and complete an offline backup.

How the HPBPI System Behaves When Components are Shutdown

This section describes the behavior of the HPBPI components when they are shutdown. It provides additional information that helps you understand the behavior of the system for backup and recovery purposes.

Engine Shutdown

If the Business Impact Engine component is shutdown, it is no longer able to receive and process events. In addition, the status of OVIS and HP Operations Manager Services that you have linked to from HPBPI are no longer updated.

As a result of the Engine shut down, the Business Event Handler component stops accepting business events until the Business Impact Engine is restarted. When the Engine is restarted, any business events waiting to be processed are automatically submitted to the Business Impact Engine.

Database Shutdown

If the database is shut down for any reason, the Business Impact Engine can no longer communicate with it and also shuts down, as described in section Engine Shutdown on page 289. In addition, all other HPBPI components that use the database are shut down.

Notification Server Shutdown

When the HPBPI system determines that a notification is required, the Business Impact Engine passes a notification alert to the Notification Server. All notification alerts are held by the Business Impact Engine, in the database, until the Notification Server has confirmed that it has accepted responsibility for them.

If the Business Impact Engine shuts down before the alert is sent to the Notification Server, the alert is retried when the Business Impact Engine restarts and the service status is shown as usual through the Business Process Dashboard. Note that the service status shown is the latest status. No notifications are generated for any interim changes that occur when the

Business Impact Engine is shut down. For example, if the Engine shuts down when a service is in the state Critical, and the service subsequently changed from Critical to Normal and then back to Critical, when the Engine restarts, the service status is shown as Critical. In this case, the information about the state change to Normal and then back to Critical is not recorded.

Email Server Shutdown

If the Notification Server cannot make a connection to the email server to send the alert to the designated user, the Notification Server stores the alert in the database and resends it at a later time when a connection is established. The Notification Server continues the attempt to resend the alert until is reaches the value for the Maximum retries to SMTP server parameter. When this maximum is reached, the Notification Server stops retrying and logs an error in its log file.

HP Operations Manager Adapter Shutdown

If the HP Operations Manager Adapter is shutdown for any reason the HPBPI system can no longer receive operational events from HP Operations Manager, nor can it deploy flows that depend on services that are monitored using the adapter.

When the adapter is restarted, the Business Impact Engine queries the current state of all the services it is monitoring, which means that no operational events are lost. The latest state of the service is reported, but any changes in state while the HPBPI system was shut down are not reported.

Business Event Handler - HPBPI Adapter Shutdown

If the Business Event Handler shuts down for any reason, the effect of the shutdown depends on the design of the individual adapters that are connected to it.

For database adapters, where you are using buffer tables, make sure that the buffer tables are backed up and synchronized with the application data that is being monitored.

For file adapters, make sure that the current position in the file is backed up with the source application data.

This means that when the Business Event Handler HPBPI adapter is restarted, the application adapters can be also restarted and any events marked as `DISCHARGE_WAIT` can then be sent back into the HPBPI system.

Servlet Engine Shut Down

If the Servlet Engine is shut down for any reason, all the currently open HPBPI clients using Browser Windows also stop. You might need to restart the Web Browser Windows and the HPBPI clients when the Servlet Engine is restarted.

Completing an Online Backup for HPBPI

Before starting an online backup, you need to ensure that your database is configured such that it is capable of completing an backup online. For example, in the case of an Oracle Server, you need to make sure that the Oracle instance you have specified for HPBPI is running in `ARCHIVELOG` mode. If it is not, you cannot complete an online backup.

The following steps describe the process for completing an online backup:

1. Ensure no one is using the HPBPI Modeler.
2. Close the HPBPI Modeler.
3. Use the HPBPI Administration Console to shut down the Model Repository component. This prevents changes being added to the repository whilst the backup is taking place.

   Do not make any further administration changes using the Administration Console until after the online backup is complete.
4. Make sure that no changes are made to the HP Operations Manager Adapter configuration during the online backup.
5. Backup the following HPBPI directories and their subdirectories on the system where the HPBPI Server is installed:

   ```
   bpi-install-dir\data\repository
   bpi-install-dir\data\conf\bia
   bpi-install-dir\data\datafiles
   bpi-install-dir\data\log
   bpi-install-dir\newconfig
   ```

   If you prefer, you can backup the entire *`HPBPI-install-dir`*`\data` directory.

➤ Any log files currently locked by HPBPI processes cannot be backed up, so there are some error messages in the backup process referring to these files. Older log files are successfully backed up.

6. Back up the HPBPI data in the database using the database management system backup utilities.

7. Backup the following HPBPI directories and their subdirectories on the system where the HP Operations Manager adapter is installed:

   ```
   HPBPI-install-dir\data\conf\bia
   HPBPI-install-dir\misc\bia
   HPBPI-install-dir\data\log
   bpi-install-dir\newconfig
   ```

8. Use the HPBPI Administration Console to restart the Model Repository component on the system where the HPBPI Server is installed.

9. Restart the HPBPI Modeler.

Your backup is now complete.

It is possible that the Engine Model Cleaner process is running at the same time as the online backup is completed. This is not a problem, but can result in inconsistencies between the data stored in the Engine's database and its configuration files. If you see warnings in the Business Impact Engine log file indicating that the Engine cannot find a definition in the database, it is as a result of this scenario.

Restoring your HPBPI Data from a Backup

This section describes how to restore your HPBPI system following:

- a complete system failure
- an HPBPI file system failure
- a database failure

In all cases, you are restoring both the database and the file system for the HPBPI components.

The following are points to note when restoring the database and the file system:

- Flow instances created after the last database backup are lost.
- Flow instances completed after the last database backup remain active and are unlikely to complete.
- Flow, Data, Service or Event definitions created since the last database backup are lost and need to be recreated.
- Modifications to Flow, Data, Service or Event definitions since the last database backup are lost and need to be reapplied.
- Flow, Data, Service or Event definitions deployed since the last database backup are lost and need to be redeployed.
- Metric definitions that are new or have been modified since the last database backup need to be recreated or reapplied.
- Email users and subscriptions configured through the Notification Server Administration Console after the last backup are lost and need to be re-added.
- Scripts and files created since the last backup need to be recreated; for example, any Velocity templates or Notification Server scripts that you have recently created might need to be redone.
- Email users and subscriptions deleted since the last database backup reappear and will need to be deleted again.
- Flow, Data, Service and Event definitions that have been undeployed reappear and need to be undeployed again.

Where appropriate, you are advised to restore the database files, HPBPI Event definition files and HPBPI configuration files to maintain consistency. If you do this, you also need to re-enter any Event definitions created since the backup was completed.

Steps to Restore your HPBPI System

This section describes the steps for restoring your HPBPI system.

If you have had a severe system failure, you are recommended to reinstall HPBPI including all its dependencies, for example, the database management system (Microsoft SQL Server or Oracle Server) and the J2SE, before restoring any files.

To restore the HPBPI data, complete the following steps:

1. Ensure no one is using the HPBPI Modeler.
2. Close the HPBPI Modeler to prevent new and modified definitions being created.
3. Use the HPBPI Administration Console to shut down all the HPBPI components.

   Do not make any further administration changes using the Administration Console until after the recovery is complete.
4. Restore the files that you previously backed up under:

   ```
   HPBPI-install-dir\data
   HPBPI-install-dir\newconfig
   ```

   If your HPBPI system has been running since the last backup you are likely to receive warnings for log files that already exist; this is expected. However, other error and warning messages should be fully investigated before continuing.

   If you are using an Oracle database continue at step 6. If you are using a Microsoft SQL Server database, continue at step 5.

5. Use the Microsoft SQL Server database utilities to restore the HPBPI database to the last backup. Be aware that when restoring files following the reinstallation of the database, the system identifiers for the database users are not the same. As a result, the newly created HPBPI user is not able to access the data in the restored database.

   For Microsoft SQL Server you use the Query Analyzer to execute an SQL command similar to the following (if you are not using the default schema name or database user defined within HPBPI, substitute the correct schema name or user for your database):

   ```
   use hpbpiSchema;
   exec sp_change_users_login 'Update_One','hpbpiuser',
   'hpbpiuser';
   ```

   You also need to ensure that the default database for your database user is set correctly; for example: `hpbpiSchema`.

   ➤ As the HPBPI database cannot be synchronized with the HPBPI file data relating to Event definitions and configuration information, it is not possible to recover to the point of failure from a backup. You must use a database management system command to restore your system to the most recent backup and not to the point of failure.

   Continue at step 7.

6. Use the Oracle Server database tools to restore the HPBPI database to the last backup.

   ➤ As the HPBPI database cannot be synchronized with the HPBPI file data relating to Event definitions and configuration information, it is not possible to recover to the point of failure from a backup. You must use a database management system command to restore your system to the most recent backup and not to the point of failure.

7. Restart the Business Impact Engine, Metric Engine and Model Repository from the HPBPI Administration Console.

8. Restart the HPBPI Modeler.

9. Open a Metric Definer Window.

10. Re-apply any business flow data that has been entered through the HPBPI Modeler, since the last backup.
11. Re-apply any metric definition data that has been entered using the Metric Definer, since the last backup.

    You have now restored your HPBPI data.

12. Undeploy any Flow, Data, Service or Event definitions that are deployed and that were undeployed before you restored your HPBPI system.
13. Use the HPBPI Administration Console to restart the remaining HPBPI components.
14. Delete any email users and subscriptions that have reappeared since you restored your HPBPI system.

Now go to section Completing the HPBPI Recovery Procedure on page 298 to complete the recovery procedure.

Completing the HPBPI Recovery Procedure

This section describes points to note after you have completed an HPBPI recovery from backup.

After you have completed the steps described in section Restoring your HPBPI Data from a Backup on page 294, you have recovered to the latest backup. You have therefore potentially missed events that are relevant to your business flows. This means that HPBPI can raise erroneous events, particularly in cases where you have Check Sequence arcs defined.

You do also have the option to take a differential backup, which is faster. However, you are advised to adopt the backup and recover policies adopted for other applications within your organization, using the information provided in this section, which is specific to HPBPI.

Index