Oracle® BPEL Process Analytics

User's Guide

10*g* Release 2 (10.1.2)

Part No. B15597-01

July 2005

ORACLE®

Oracle BPEL Process Analytics User's Guide, 10g Release 2 (10.1.2)

Primary Author: Ingrid Stuart

Contributor: Craig West (Analyzing Oracle BPEL Process Analytics Data with Discoverer)

Contents

5 Administering Oracle BPEL Process Analytics

6 Using the Oracle BPEL Process Analytics Web Service

List of Figures

List of Tables

Send Us Your Comments

Oracle BPEL Process Analytics User's Guide, 10*g* Release 2 (10.1.2)

Part No. B15597-01

Oracle welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most about this manual?

If you find any errors or have any other suggestions for improvement, please indicate the title and part number of the documentation and the chapter, section, and page number (if available). You can send comments to us in the following ways:

- Electronic mail: appserverdocs_us@oracle.com
- FAX: 650.506.7375   Attn: Oracle Application Server Documentation Manager
- Postal service:

    Oracle Corporation
    Oracle Application Server Documentation
    500 Oracle Parkway, M/S 1op6
    Redwood Shores, CA 94065
    USA

If you would like a reply, please give your name, address, telephone number, and electronic mail address (optional).

If you have problems with the software, please contact your local Oracle Support Services.

Preface

This guide describes how to use Oracle BPEL Process Analytics.

Audience

This manual is intended for all users of Oracle BPEL Process Analytics, which include system administrators, Oracle BPEL Process Analytics administrators, and Oracle BPEL Process Analytics end users (Dashboard users).

Readers should be familiar with the data sources with which they plan to use Oracle BPEL Process Analytics.

Documentation Accessibility

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

http://www.oracle.com/accessibility/

Accessibility of Code Examples in Documentation

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

Accessibility of Links to External Web Sites in Documentation

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

TTY Access to Oracle Support Services

Oracle provides dedicated Text Telephone (TTY) access to Oracle Support Services within the United States of America 24 hours a day, seven days a week. For TTY support, call 800.446.2398.

Structure

This guide contains nine chapters.

Chapter 1

This chapter introduces the reader to Oracle BPEL Process Analytics and introduces concepts that both the administrator and the business analyst must understand for the administrator to present data of interest to the business analyst and for the business analyst to understand the data presented to him or her.

Chapter 2

This chapter describes how to use the Admin Console section of the Oracle BPEL Process Analytics user interface to capture events of interest, model the event data, and create user accounts.

Chapter 3

This chapter describes how to use the Dashboard Configuration Console to present the modeled data in the end-user section of the interface, the Dashboard.

Chapter 4

This chapter describes how to use the Dashboard section of the Oracle BPEL Process Analytics user interface. The Dashboard is intended for user by business analysts and managers interested in analyzing business process flow.

Chapter 5

This chapter describes how to administer and monitor Oracle BPEL Process Analytics itself. It includes information about user management, log files, and setting log file and Dashboard properties.

Chapter 6

This chapter describes the Oracle BPEL Process Analytics Web service, which enables third-party applications, such as Microsoft Excel spreadsheets, portals, and so on, to access the event data collected and modeled by Oracle BPEL Process Analytics.

In addition, this chapter describes the BPEL Web service wrapper, which enables you (or an application) to invoke the Oracle BPEL Process Analytics Web service from the Oracle BPEL Process Manager (BPEL) Console.

Chapter 7

This chapter describes the steps and tools required to monitor generic event sources using the sensor development kit. In addition, it provides a step-by-step example of monitoring an Oracle database as a generic event source.

Chapter 8

This chapter describes how to use the event simulator to generate events without connecting to an actual event source.

Chapter 9

This chapter describes how to use analyze Oracle BPEL Process Analytics data using Oracle Business Intelligence Discoverer.

Related Documents

For more information, see the following manuals on related products and installation:

- *Oracle BPEL Process Manager Developer's Guide*
- *Oracle BPEL Process Analytics Installation Guide*
- *Oracle Application Server Containers for J2EE Services Guide*
- *Streams Advanced Queuing User's Guide and Reference*
- *Oracle Application Server Wireless Developer's Guide*
- *Oracle XML Developer's Kit Programmer's Guide*

Conventions

This section describes the conventions used in the text and code examples of this documentation set. It describes:

- Conventions in Text
- Conventions in Code Examples
- Conventions for Windows Operating Systems

Conventions in Text

We use various conventions in text to help you more quickly identify special terms. The following table describes those conventions and provides examples of their use.

Convention	Meaning	Example
Bold	Bold typeface indicates terms that are defined in the text or terms that appear in a glossary, or both.	When you specify this clause, you create an **index-organized table**.
Italic	Italic typeface indicates book titles or emphasis.	*Oracle Database Concepts* Ensure that the recovery catalog and target database do *not* reside on the same disk.
`UPPERCASE monospace (fixed-width) font`	Uppercase monospace typeface indicates elements supplied by the system. Such elements include parameters, privileges, datatypes, Recovery Manager keywords, SQL keywords, SQL*Plus or utility commands, packages and methods, as well as system-supplied column names, database objects and structures, user names, and roles.	You can specify this clause only for a `NUMBER` column. You can back up the database by using the `BACKUP` command. Query the `TABLE_NAME` column in the `USER_TABLES` data dictionary view. Use the `DBMS_STATS.GENERATE_STATS` procedure.

Convention	Meaning	Example
`lowercase monospace (fixed-width) font`	Lowercase monospace typeface indicates executable programs, file names, directory names, and sample user-supplied elements. Such elements include computer and database names, net service names and connect identifiers, user-supplied database objects and structures, column names, packages and classes, user names and roles, program units, and parameter values. *Note:* Some programmatic elements use a mixture of UPPERCASE and lowercase. Enter these elements as shown.	Enter `sqlplus` to start SQL*Plus. The password is specified in the `orapwd` file. Back up the datafiles and control files in the `/disk1/oracle/dbs` directory. The `department_id`, `department_name`, and `location_id` columns are in the `hr.departments` table. Set the `QUERY_REWRITE_ENABLED` initialization parameter to `true`. Connect as `oe` user. The `JRepUtil` class implements these methods.
`lowercase italic monospace (fixed-width) font`	Lowercase italic monospace font represents placeholders or variables.	You can specify the *`parallel_clause`*. Run *`old_release`*`.SQL` where *`old_release`* refers to the release you installed prior to upgrading.

Conventions in Code Examples

Code examples illustrate SQL, PL/SQL, SQL*Plus, or other command-line statements. They are displayed in a monospace (fixed-width) font and separated from normal text as shown in this example:

```
SELECT username FROM dba_users WHERE username = 'MIGRATE';
```

The following table describes typographic conventions used in code examples and provides examples of their use.

Convention	Meaning	Example
`[ ]`	Anything enclosed in brackets is optional.	`DECIMAL (`*`digits`*` [ , `*`precision`*` ])`
`{ }`	Braces are used for grouping items.	`{ENABLE	DISABLE}`
`|`	A vertical bar represents a choice of two options.	`{ENABLE	DISABLE}`   `[COMPRESS	NOCOMPRESS]`

`...`	Ellipsis points mean repetition in syntax descriptions.
	
	 In addition, ellipsis points can mean an omission in code examples or text.
		`CREATE TABLE ... AS `*`subquery`*`;`
		
		 `SELECT `*`col1`*`, `*`col2`*`, ... , `*`coln`*` FROM employees;`

Other symbols	You must use symbols other than brackets ([ ]), braces ({ }), vertical bars (|), and ellipsis points (...) exactly as shown.	`acctbal NUMBER(11,2);` `acct    CONSTANT NUMBER(4) := 3;`
Italic	Italicized text indicates placeholders or variables for which you must supply particular values.	`CONNECT SYSTEM/`*`system_password`* `DB_NAME = `*`database_name`*

UPPERCASE	Uppercase typeface indicates elements supplied by the system. We show these terms in uppercase in order to distinguish them from terms you define. Unless terms appear in brackets, enter them in the order and with the spelling shown. Because these terms are not case-sensitive, you can use them in either UPPERCASE or lowercase.	`SELECT last_name, employee_id FROM employees;`
		 `SELECT * FROM USER_TABLES;`
		 `DROP TABLE hr.employees;`

Convention	Meaning	Example
lowercase	Lowercase typeface indicates user-defined programmatic elements, such as names of tables, columns, or files.	SELECT last_name, employee_id FROM employees; sqlplus hr/hr CREATE USER mjones IDENTIFIED BY ty3MU9;
	Note: Some programmatic elements use a mixture of UPPERCASE and lowercase. Enter these elements as shown.	

Conventions for Windows Operating Systems

The following table describes conventions for Windows operating systems and provides examples of their use.

Convention	Meaning	Example
File and directory names	File and directory names are not case sensitive. The following special characters are not allowed: left angle bracket (<), right angle bracket (>), colon (:), double quotation marks ("), slash (/), pipe (|), and dash (-). The special character backslash (\) is treated as an element separator, even when it appears in quotation marks. If the file name begins with \\, then Windows assumes it uses the Universal Naming Convention.	c:\winnt"\"system32 is the same as C:\WINNT\SYSTEM32
C:\>	Represents the Windows command prompt of the current hard disk drive. The escape character in a command prompt is the caret (^). Your prompt reflects the subdirectory in which you are working. Referred to as the *command prompt* in this manual.	C:\oracle\oradata>
Special characters	The backslash (\) special character is sometimes required as an escape character for the double quotation mark (") special character at the Windows command prompt. Parentheses and the single quotation mark (') do not require an escape character. Refer to your Windows operating system documentation for more information about escape and special characters.	C:\> exp HR/HR TABLES=emp QUERY=\"WHERE job='REP'\"
HOME_NAME	Represents the Oracle home name. The home name can be up to 16 alphanumeric characters. The only special character allowed in the home name is the underscore.	C:\> net start Oracle*HOME_NAME*TNSListener

xx

1

Introduction to Oracle BPEL Process Analytics

Oracle BPEL Process Analytics is a feature of Oracle BPEL Process Manager. Oracle BPEL Process Analytics provides dashboards that measure service-level agreement, alert on exceptions, and gives greater business visibility. The Oracle BPEL Process Analytics Console is used to monitor business processes that might span multiple Oracle BPEL Process Manager processes. This console delivers useful information about service-level agreements, process metrics, and exceptions.

A business analyst can define key performance indicators (KPIs) on one or more Oracle BPEL Process Manager processes that implement a single business process and monitor the performance of these KPIs. Oracle BPEL Process Analytics offers the following benefits:

- Provides the ability to monitor one or more Oracle BPEL Process Manager processes as a single business process

- Provides the ability to define KPIs on the business processes

- Enables alignment between business operations and business strategy

- Provides a comprehensive view of related events across multiple business processes and allows the analyst to monitor these related events

- Provides the ability to notify mangers when KPIs are not performing at the optimal levels

This chapter includes the following topics:

- Sample Scenario Using Oracle BPEL Process Analytics on page 1-1

- Typical Steps in Oracle BPEL Process Manager Process Analysis on page 1-3

- Oracle BPEL Process Analytics User Interfaces on page 1-11

Sample Scenario Using Oracle BPEL Process Analytics

To understand how Oracle BPEL Process Analytics works, consider the Oracle BPEL Process Manager **process flow** for a loan procurement process, as illustrated in Figure 1–1. In this process flow, a customer requests a loan. The loan procurement company requests a credit rating on the customer. If the returned credit rating is positive, then the application is submitted to two (fictitious) loan services: United Loan Service and Star Loan Service. Each service returns an offer on the loan and the lowest offer is presented to the customer. The customer accepts or rejects the loan offer. The goal of the loan procurement company is to return the best offer for a loan request (or reject the loan request) as quickly as possible.

Figure 1–1   Loan Procurement Process Flow
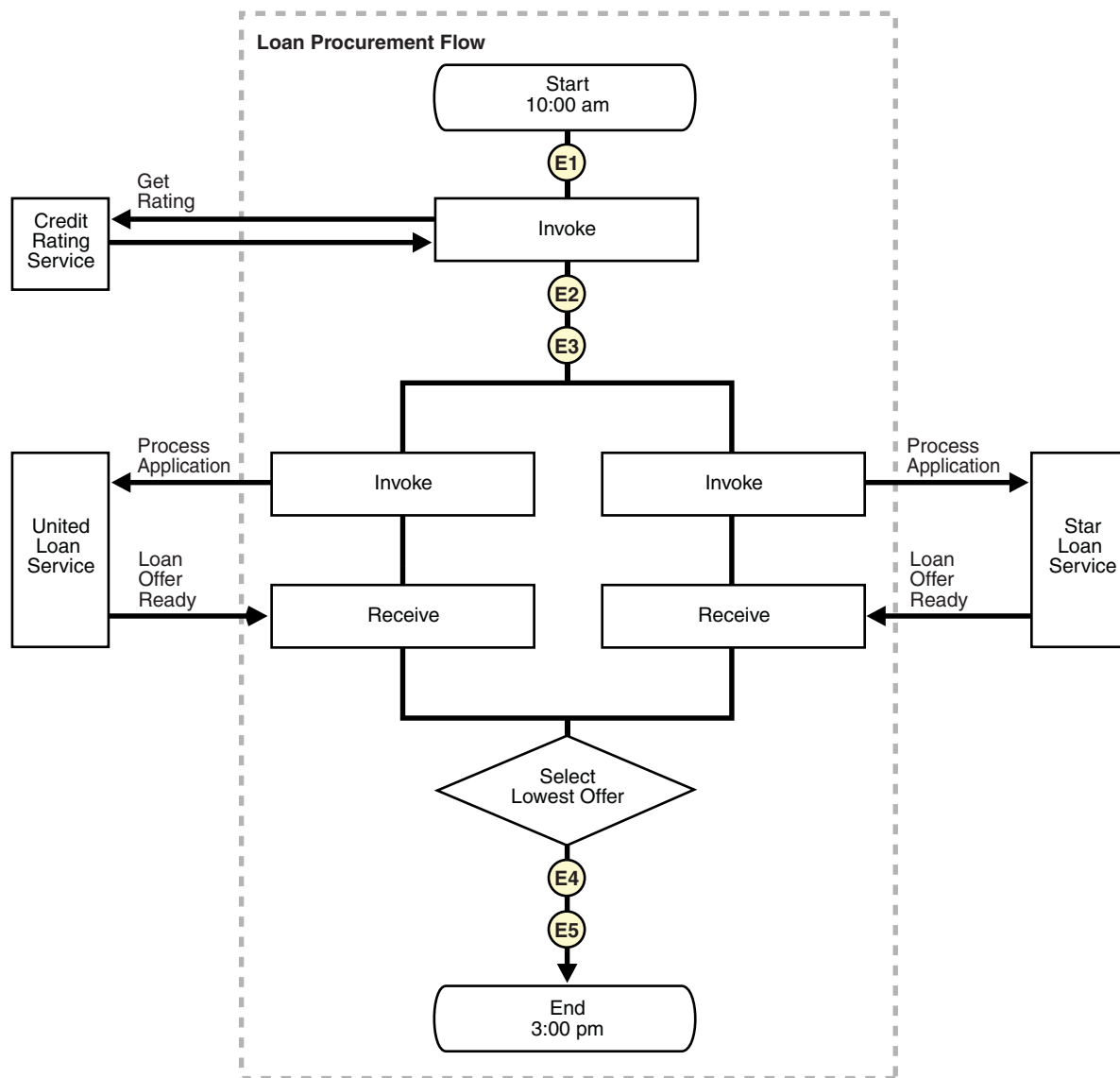
Business events are captured at five points within the process flow illustrated in Figure 1–1, as follows:

- E1: LoanRequest - A customer requests a loan

- E2 - LoanRejected - On the basis of the credit rating, the loan service rejects the loan request

- E3 - LoanApproved - On the basis of the credit rating, the loan service accepts the loan request

- E4 - LoanSelected - A loan for the customer is selected based on lowest APR

- E5 - LoanAccepted - The customer accepts the loan

Using Oracle BPEL Process Analytics, events in this process flow, can be captured, analyzed, and then processed and presented in many ways for analysis, including the following:

- A chart displaying events, as they occur.

- A table displaying the time at which a loan application was sent for processing, the time at which the loan offer was received, or if receipt of an offer is still pending.

- Line graphs displaying:

  - The average time it takes for the credit rating service to return a rating

  - The average time it takes for each loan service to return an offer

  - The average time it takes for a decision to be made on which offer to present to the customer

- Bar graphs displaying:

  - The number of loan applications processed by each loan service

  - The number of loan applications rejected by each loan service

If the analysis shows that the number of loan applications accepted by customers exceeds target goals, for example, then Oracle BPEL Process Analytics can automatically alert a business analyst of the situation, and it can invoke a Web service to take action. The Web service might increase the APR to maximize profits on a loan that has become particularly popular.

Typical Steps in Oracle BPEL Process Manager Process Analysis

The typical steps for using Oracle BPEL Process Analytics for monitoring and analyzing business processes are as follows:

1. Identify related Oracle BPEL Process Manager processes.

2. Capture events from the processes using sensors.

3. Correlate related event instances.

4. Analyze data.

5. Display analysis.

6. Respond to exceptions.

A business process can consist of one or more Oracle BPEL Process Manager processes. The Oracle BPEL Process Manager sensor framework allows users to capture important events within an Oracle BPEL Process Manager process using sensors. Oracle BPEL Process Analytics allows users to capture these events, correlate them and perform analysis on them. The administrator can configure the dashboards to view the analysis, monitor exception conditions, and take automated corrective action on exception conditions.

The following topics provide an overview of each of the steps involved in monitoring and analyzing business processes:

- Identifying Related Oracle BPEL Process Manager Processes on page 1-4

- Capturing Events from the Oracle BPEL Process Manager Processes Using Sensors on page 1-4

- Correlating Events into Meaningful Groups for Analysis on page 1-4

- Analyzing Events on page 1-5

- Presenting Data to Dashboard Users on page 1-7

- Identifying and Addressing Critical Business Conditions on page 1-9

Information about the interfaces used to perform these steps is provided in "Oracle BPEL Process Analytics User Interfaces" on page 1-11.

Identifying Related Oracle BPEL Process Manager Processes

Identify a set of related business processes that need to be monitored as a single entity. One or more Oracle BPEL Process Manager processes are implemented to perform a business activity.

Capturing Events from the Oracle BPEL Process Manager Processes Using Sensors

The Oracle BPEL Process Manager sensor framework allows the capture of important events that occur during the lifetime of a Oracle BPEL Process Manager process. Use JDeveloper BPEL Designer to define the sensors on the Oracle BPEL Process Manager process. The output of these sensors is captured as events in Oracle BPEL Process Analytics.

Correlating Events into Meaningful Groups for Analysis

To perform meaningful analysis on the events captured, related event instances must be grouped together. To determine the average amount of time it takes to process a given loan application, for example, each instance of a loan request submission must be linked to its associated acceptance or rejection event.

Composite Events and Groups

Making links between events is referred to as **correlating** the events. Events that are grouped together in this manner are referred to as composite events.

A **composite event** can include one or more events, from one event source only. Events are correlated on the basis of a common event attribute (referred to as a correlation attribute), as shown in Figure 1–2.

A **composite event group** is a collection of composite events. Each composite event in a composite event group can be associated with a different event source. In a composite event group, events are also correlated on the basis of a common event attribute.

Figure 1–2   Composite Event
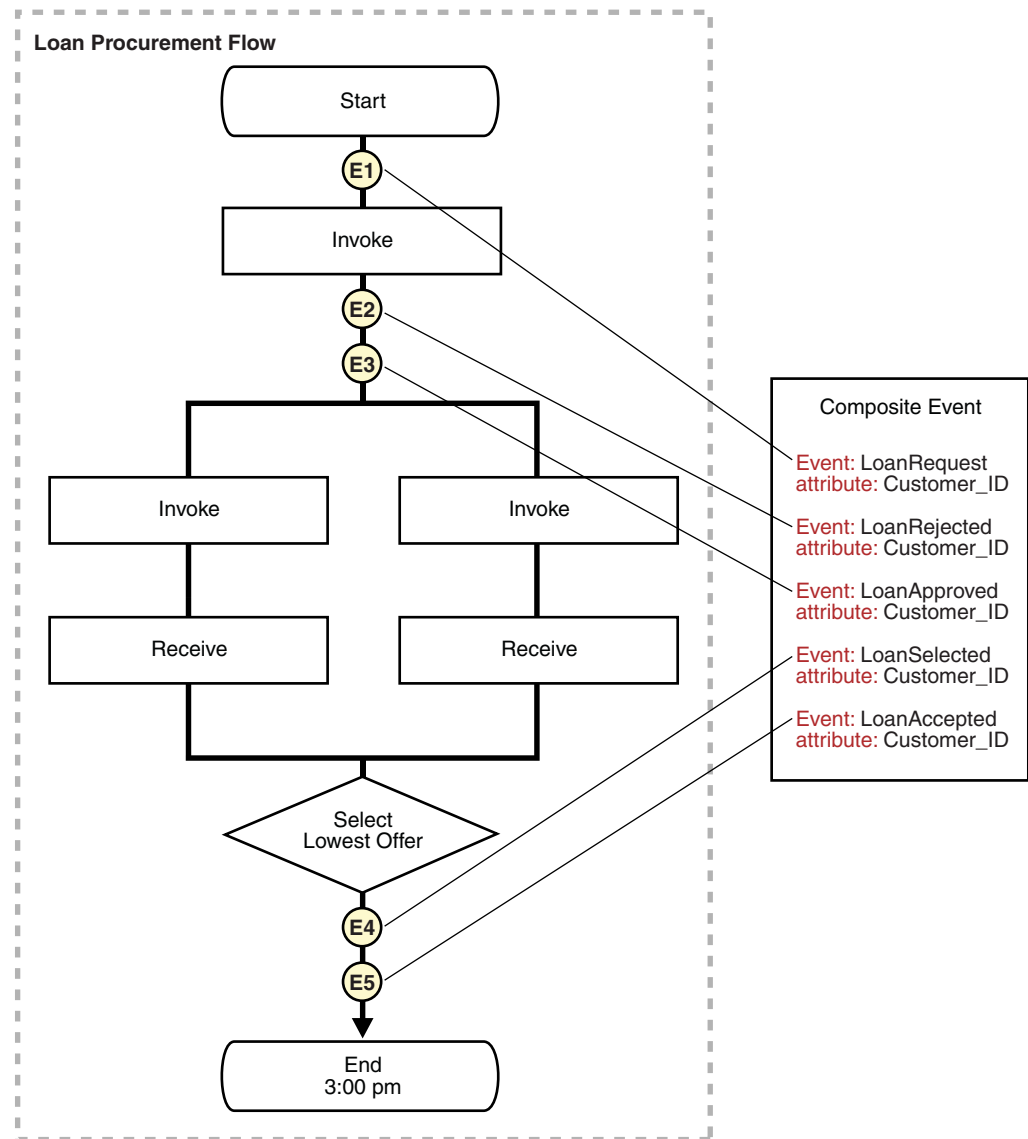
Table 1–1 presents an example of a composite event for each of the event sources that Oracle BPEL Process Analytics supports.

Table 1–1   Examples of Composite Events for Each Supported Event Type

Event Source	Example of Composite Event
Oracle BPEL Process Manager	Composite event, LoanFlow, is created by grouping sensors for activities within the Oracle BPEL Process Manager Loan Procurement process.
Applications with XSD written as required for the Oracle BPEL Process Analytics generic monitor	A composite event, SalaryIncrease, is created by grouping insertions into the EMPLOYEES and SALARY database tables and correlating them across the shared attribute, EMPLOYEE_ID. (A tool, such as XML Schema Utility (XSU) can be used to transform SQL SELECT statements into XML Schema Definition (XSD) statements.) See Chapter 7 for more information.

Analyzing Events

Once the events are captured and grouped into composite events, they can be configured for analysis. The grouping and configuration process is referred to as

modeling. Oracle BPEL Process Analytics provides wizards to help you create and model composite events into three key objects for analysis: metrics, key performance indicators (KPIs), and dimensions.

Metrics

A **metric** is an event attribute value, or a calculation on a set of event attribute values contained within a single composite event instance. The calculation can include addition, subtraction, multiplication, and division.

Using the loan flow process described in "Sample Scenario Using Oracle BPEL Process Analytics" on page 1-1, the following metrics might be defined:

- The annual percentage rate (APR) of a single instance of a loan offer
- The loan request processing time

A business analyst might find a metric useful for monitoring Oracle BPEL Process Analytics event instances to identify the pattern of critical business conditions over time. For example, a business analyst could see if there is a pattern to the rate at which approvals are taking relative to the day of the week or the season.

Key Performance Indicators

Briefly, a key performance indicator (**KPI)** consists of instances of a composite event attribute (or attributes) aggregated over a period of time, to which a mathematical function is applied. While metrics can be used to study general patterns and trends, KPIs enable an analyst to perform in-depth analysis of the event data.

Using the loan flow process described in "Sample Scenario Using Oracle BPEL Process Analytics" on page 1-1, the following KPIs might be defined:

- The total number of loans requested
- The total number of loans approved
- The average loan approval time (the average of the loan request time stamp minus the loan request approved time stamp for each of the loans requested)

For a complete discussion of KPIs, see "Modeling KPIs and Related Objects" on page 2-27.

Dimensions

Dimensions are optional constructs that the Oracle BPEL Process Analytics administrator can specify and use in a composite event definition to allow KPI values to be filtered by dimensions. A time dimension is defined by default.

For example, suppose the Oracle BPEL Process Analytics administrator defines and specifies loan provider and car model dimensions and defines a composite event using those dimensions. If a KPI is defined on that composite event, to track the number of loans approved, then a car loan processing manager can monitor the number of loans approved based on time, loan provider, and car model. Similarly, if a regional sales manager is interested in sales only in a given region, then the administrator can define a KPI on a composite event with a region dimension. When viewing the KPI, the regional sales manager can constrain the results to just the region of interest.

See Figure 1–3 and Figure 1–4 for two examples of how the types of KPIs described in this section can be presented to the business analyst, in a section of the Oracle BPEL Process Analytics Console called the Dashboard.

Presenting Data to Dashboard Users

Once an Oracle BPEL Process Analytics administrator models the data, that administrator can specify how that data is presented in the Oracle BPEL Process Analytics Dashboard. The **Dashboard** is the section of the Oracle BPEL Process Analytics Console for end users, typically business analysts. Modeled data can be presented in a variety of charts and tables to provide information about business activity, such as the following:

- A pie chart, such as shown in Figure 1–3, can show what percentage of loan requests were made, by car make and model, today.

- A bar graph, such as shown in Figure 1–4, can show the number of actual loan requests compared to budgeted values (also known as target values) for several days.

- A radar chart, such as shown in Figure 1–5, can show how the number of loan requests and loan offers, and so on, are tracking against target values.

  Note that some labels within a radar chart can be truncated. In this Figure 1–5, CountLoanOffers is truncated to CountLoanO and CountRecReject is truncated to untReqReject, and CountLoanOffers4BadCredit is truncated to CountLoanOffers4BadCr. Figure 1–5 shows that when the mouse is placed over a truncated label, a pop-up window displays the full name.

- The Real-Time Viewer, such as shown in Figure 1–6, presents events within a composite event instance as they occur in real time.

These are just a few of the charts that can be used to present business data. For more information about the types of charts that can be presented and how to read them, see Chapter 3 and Chapter 4.

Figure 1–3   Sample Pie Chart
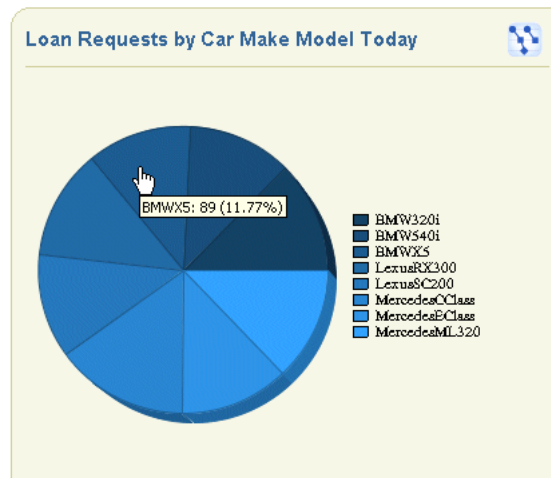
Figure 1–4   Sample Chart for Trend Analysis
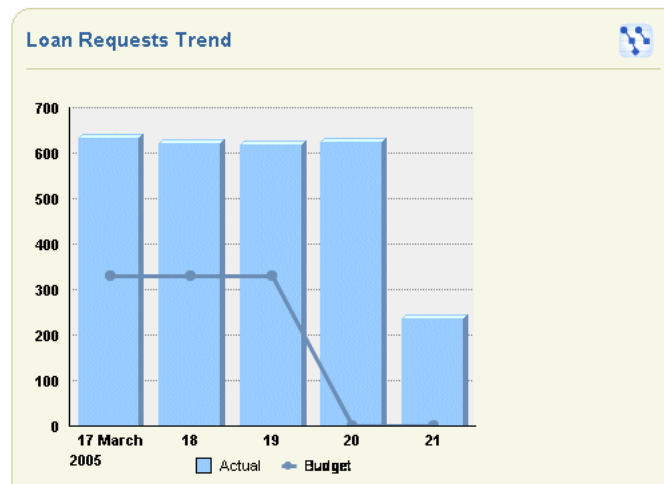
Figure 1–5   Sample Radar Chart to Track Performance Against Target Values
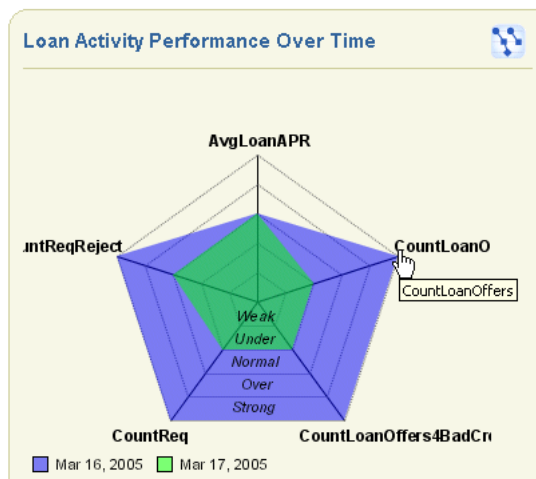
Figure 1–6   Sample Real-Time Viewer
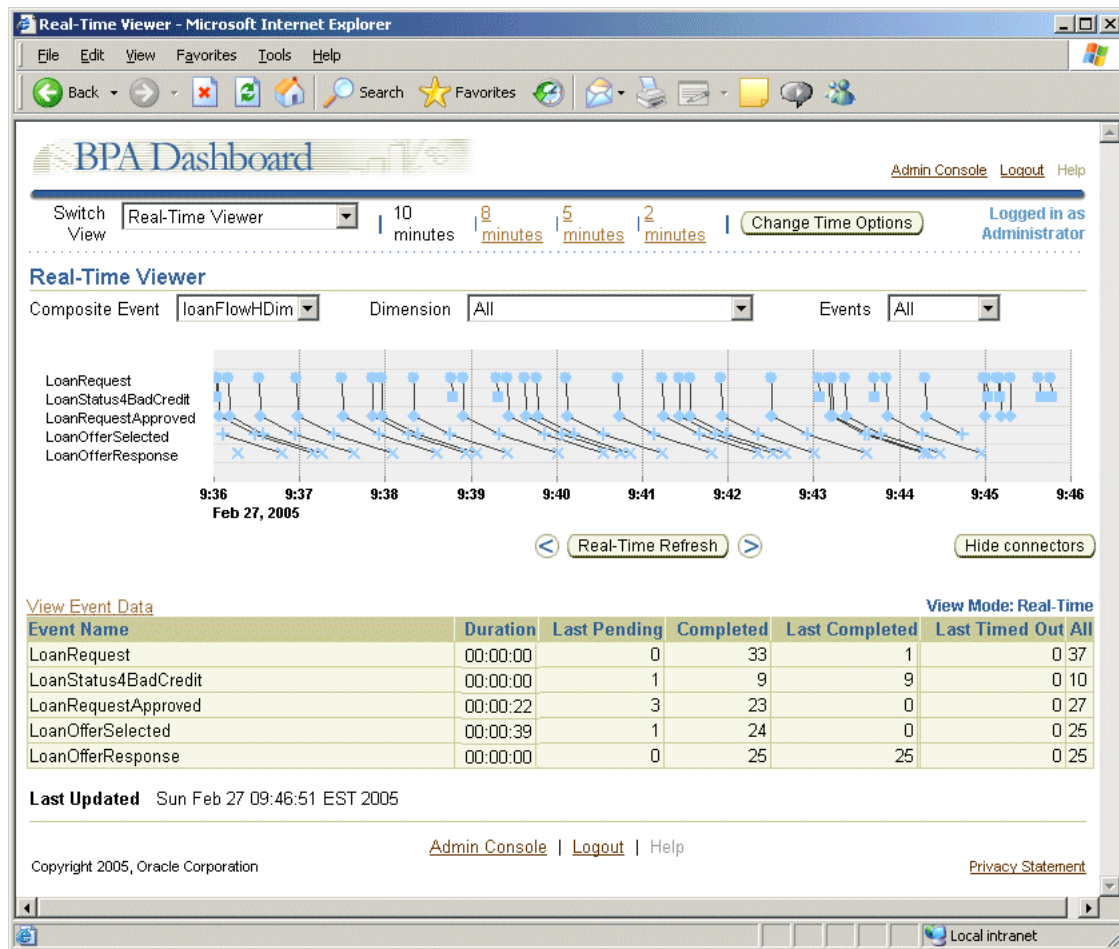
Identifying and Addressing Critical Business Conditions

The point of capturing and modeling events is to enable a business analyst to identify and respond to critical business conditions. This is accomplished using one or all of the following methods:

- The business analyst can review the real-time data as it is presented in the Dashboard, and monitor it for critical business conditions

    For example, a business analyst might review the prediction for a KPI, such as shown in Figure 1–7. This detail uses statistical analysis to predict a KPI value. By viewing this detail, the business analyst can either anticipate problems and take corrective action before problems arise, or can anticipate positive results and investigate the circumstances so that the positive results can be sustained.

Figure 1–7   Sample KPI Prediction Table

Prediction   Switch KPI AvgLoanAPR ▼

Analysis Time Grain Day ▼

Prediction Details

Predicted Value   **4.458**
Target Value   **3.914**
Confidence   **27%**
Performance Band **Normal**

KPIs Used for Prediction

Source	Confidence	Predicted Value	Correlation Factor
CountLoanOffers4BadCredit	14%	4.399	38%
CountLoanOffers4Ford	68%	4.434	-83%
CountLoanOffers4Nissan	55%	4.468	74%
CountLoanOffers4Star	24%	4.509	49%
CountLoanOffers4Toyota	8%	4.416	29%
CountLoanOffers4United	55%	4.505	-74%
OfferApprovalTime	14%	4.452	-37%
OfferSelectTime	6%	4.427	-24%
OfferSelectTime4United	14%	4.439	-37%
ReqApprovalTime	12%	4.414	-35%
SumLoanAPR	29%	4.44	-54%

- A business analyst could also review the cause and effect table for a given KPI, such as shown in Figure 1–8. This table presents the KPIs which influence (cause) or are influenced by (effect) a given KPI. By examining these KPIs, the business analyst can further determine what might be causing a critical business condition.

Figure 1–8   Sample KPI Cause/Effect Table

Cause KPI Name	Influence	Band
OfferSelectTime	70.1%	● Over
ReqApprovalTime	38.44%	● Strong
CountLoanOffers4BadCredit	10.7%	● Weak
Effect KPI Name	**Influence**	**Band**
CountReqReject	65.8%	● Under

For a complete description of the KPI Prediction table and the KPI Cause/Effect table, see Chapter 4.

- An administrator can set up notifications, called **explicit alerts,** that are sent to the business analyst if a KPI or metric exceeds a predetermined threshold value.

  An explicit alert can be sent as an e-mail message, a phone message, a fax, a Short Message Service (SMS) message, a pager message, or as an Internet instant message.

  For a complete description of alerts and how they are defined, see "Creating User Accounts" on page 2-37.

- If the administrator configures it into the Dashboard, the user can review the Alert View table, such as shown in Figure 1–9, that lists explicit alerts.

  The Dashboard user can click a link in the alert table to get detailed information about that alert.

Figure 1–9   Sample Alert View Table

For a complete description of alert tables, see "Viewing Alerts in Multiple Pane Views" on page 4-33.

After a critical business condition is identified, it can be addressed manually or programmatically, as follows:

- Programmatically

  For events that trigger explicit alerts, a Web service can be configured such that it is called to respond to the condition that caused the alert to be triggered. In the case of the loan flow process, for example, an alert could be defined such that a Web service is called to decrease the APR for car loans if the target number of car loans was not accepted for the preceding month. See Chapter 6 for more information about using Web services.

- Manually

  – The business analyst can use the variety of charts and graphs within the Dashboard to help determine the cause of, or anticipate, a problem and take action manually.

  – Upon receiving an explicit alert through an e-mail message, a phone message, or whatever method is used for explicit alert delivery, the recipient can take appropriate action.

Oracle BPEL Process Analytics User Interfaces

Oracle BPEL Process Analytics provides a Console that is divided into three main sections: an Admin Console, a Dashboard Configuration Console, and a Dashboard. Access to these sections of the Console is governed by the privileges of the account holder.

In addition, the Oracle Enterprise Manager Application Server Control Console enables a system administrator to configure, start, and stop Oracle BPEL Process Analytics, and set Oracle BPEL Process Analytics properties. See Chapter 5 for more information.

The following topics provide a brief introduction to the parts of the Oracle BPEL Process Analytics Console:

- Introduction to the Admin Console on page 1-11
- Introduction to the Dashboard Configuration Console on page 1-12
- Introduction to the Dashboard on page 1-13

Introduction to the Admin Console

The Admin Console, accessible by accounts granted the Oracle BPEL Process Analytics Admin User privilege, provides wizards and other tools that enable an administrator to:

- Specify connection parameters for the sources from which events will be captured

- Create composite events.

- Model KPIs and metrics

- Set up alerts to send a notification to one or more users to inform them that critical measurements are not meeting expectations

- Configure data for presentation to end users in the Dashboard

- Create user accounts and grant them privileges to view the event source data

Figure 1–10 shows the Admin Console Welcome page in the Oracle BPEL Process Analytics user interface.

See Chapter 2 and Chapter 3 for more information about the tasks the administrator performs using the Admin Console.

Figure 1–10 Oracle BPEL Process Analytics Admin Console Welcome Page
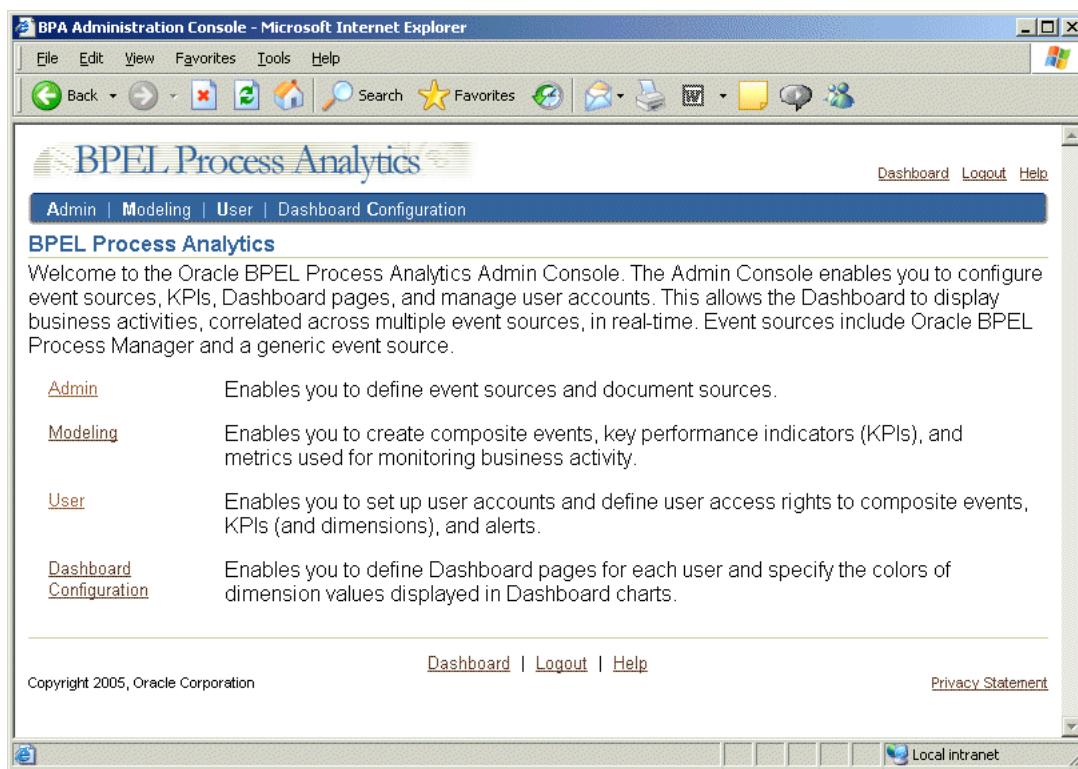
Introduction to the Dashboard Configuration Console

The Dashboard Configuration Console provides wizards that enable a user with Oracle BPEL Process Analytics admin privileges to specify how data modeled using the Admin Console should be displayed in the Dashboard. Tasks accomplished through the Dashboard Configuration Console include:

- Specifying the types of charts and tables to present to Dashboard users

- Creating Dashboard pages and specifying their layout

- Granting users access to Dashboard pages

Figure 1–11 shows a sample page in the Dashboard Configuration Console. See Chapter 3 for a detailed description of the tasks the administrator performs using the Dashboard Configuration Console.

Figure 1–11   Sample Dashboard Configuration Console Page

Introduction to the Dashboard

The Dashboard is the interface through which Oracle BPEL Process Analytics end users view and analyze modeled data. These users are typically business analysts or high-level business managers.

Using the Dashboard, a user views the data collected by Oracle BPEL Process Analytics and modeled by the administrator. The Dashboard interface presents the events and performance measurements, in real time, within tables and charts, for the business analyst to assess.

Figure 1–12 shows a sample Dashboard page. See Chapter 4 for a detailed description of the tasks the business administrator performs using the Dashboard.

Figure 1–12   Sample of Dashboard Charts

2
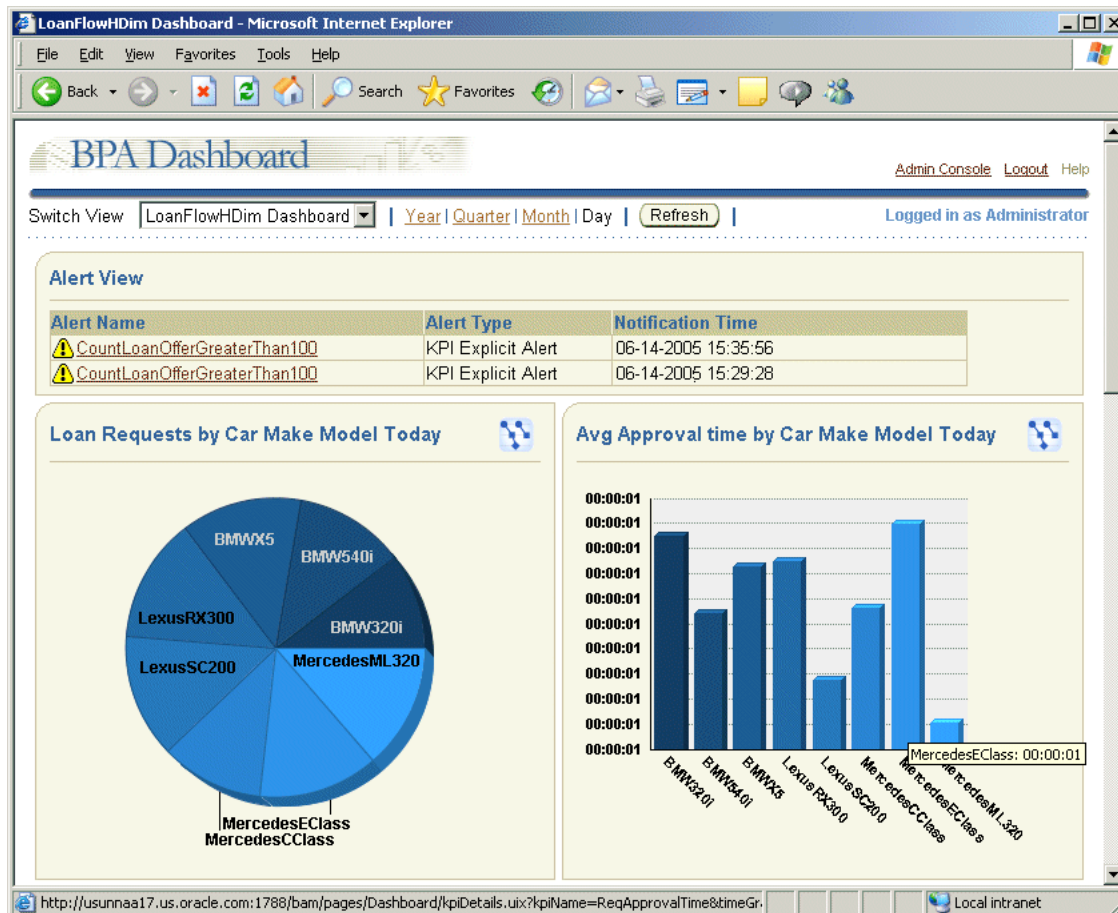
Using the Admin Console to Model Data and Create Users

The Admin Console is the section of the Oracle BPEL Process Analytics Console through which the Oracle BPEL Process Analytics administrator configures event sources, models events, creates users, and then configures that data for presentation to Dashboard users. This chapter provides information about using the Admin Console to configure event sources, model events, and create users. Chapter 3 describes how to use the Dashboard Configuration Console. It is assumed that you are an Oracle BPEL Process Analytics administrator if you are reading this chapter.

This chapter includes the following topics:

- Introduction to the Admin Console on page 2-1
- Getting Started with the Admin Console on page 2-2
- General Procedure for Using the Admin Console on page 2-5
- Creating and Managing Event Source Connections and KPI Document Sources on page 2-6
- Creating and Managing Modeled Objects on page 2-18
- Creating and Managing User Accounts on page 2-35

Introduction to the Admin Console

The Admin Console enables you to perform the tasks necessary to present useful data in the Dashboard. It is divided into four key areas:

- Admin

  This section of the Admin Console enables you to specify the event sources to monitor. Data cannot be modeled until the event sources, from which the event instances and event metadata will be fetched, are defined.

  This section of the Admin Console also enables you to specify the source system where documents that provide contextual information about KPIs are stored.

- Modeling

  This section of the Admin Console enables you to create, modify, or delete the metadata that will be used to monitor business activity across event sources within the enterprise, including composite events, KPIs, metrics, and dimensions. The modeled data must be assigned to a user before that data can be configured in the Dashboard for that user to view.

- User

This section of the Admin Console enables you to define Admin Console and Dashboard user accounts, and specify the data to which they will have access through the Dashboard. A user must be granted Admin User privileges to access the Admin Console.

The User section of the Admin Console is also where you can create, modify, or delete KPI and metric alerts, as well as alert delivery channels for each user account.

- Dashboard Configuration

The Dashboard Configuration Console of the Admin Console enables you to specify how modeled data will be presented to Dashboard users. You can select various charts and tables in which to present the data, how those charts and tables will be presented on the Dashboard page, and which users will be able to view each of the Dashboard pages.

Figure 2–1 shows the Welcome page of the Admin Console, which introduces and provides access to each of these major areas.

Figure 2–1   Oracle BPEL Process Analytics Admin Console Welcome Page
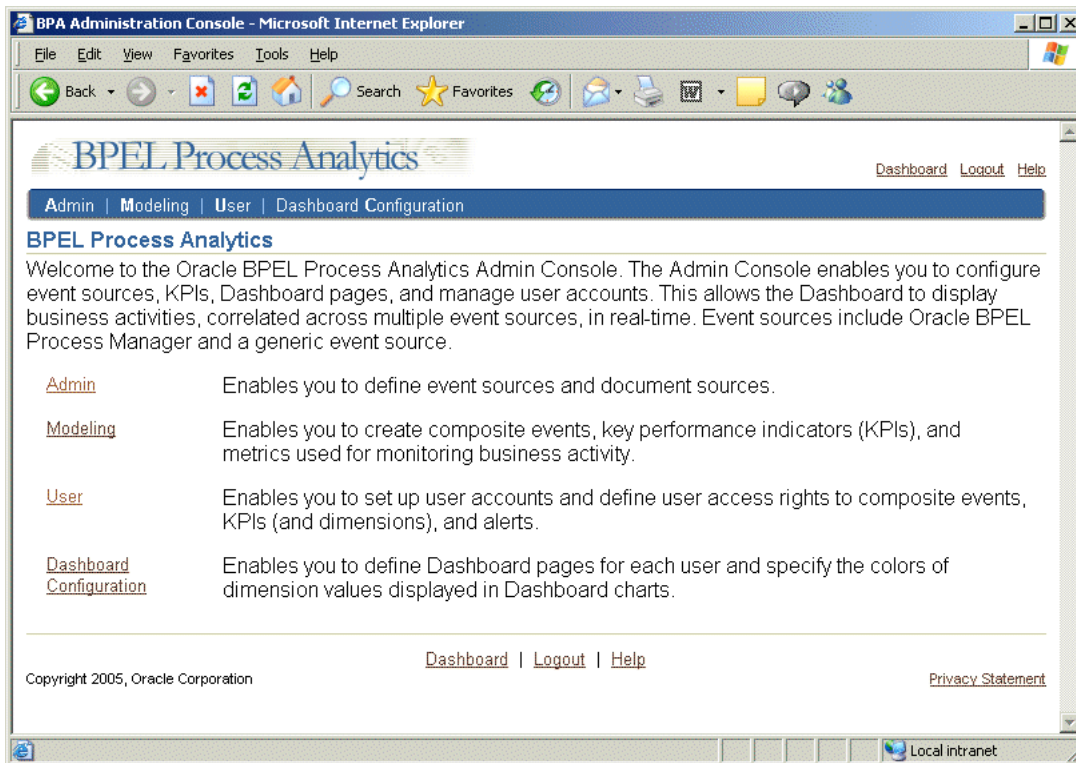
Getting Started with the Admin Console

To get started with the Admin Console, start the Oracle BPEL Process Analytics Server and log in as the Administrator. You are then ready to begin specifying event sources and modeling data.

> **Note:** Oracle recommends that only one user log in to the Oracle BPEL Process Analytics Console as the Administrator at a time. Unpredictable results may occur when multiple Administrators use the Admin Console concurrently.

Be aware that two users cannot log in to the console

When you start the Oracle BPEL Process Analytics Server, all of the components required to use Oracle BPEL Process Analytics are started. You can also start and stop each component individually using Oracle Enterprise Manager 10*g* Application Server Control Console. See "Starting, Stopping, or Restarting Components" on page 5-9 for more information.

> **Note:** Be sure to close your Web browser after you log out of the Oracle BPEL Process Analytics Console, to ensure that others cannot view Console pages using the Web browser's Back button.

This section includes the following topics:

- Starting the Server and Opening the Console on Microsoft Windows Systems on page 2-3
- Stopping the Server on Microsoft Windows on page 2-4
- Starting the Server and Opening the Console on Unix Systems on page 2-4
- Stopping the Server on Unix on page 2-4
- Logging In to the Admin Console on page 2-5

Starting the Server and Opening the Console on Microsoft Windows Systems

To start Oracle BPEL Process Analytics from the Microsoft Windows desktop and open the Console, follow these steps, where *OAS_Home* is the name of the Oracle home in which you installed Oracle BPEL Process Analytics:

1. On the desktop, click **Start,** then choose **All Programs,** then ***Oracle – OAS_ Home_Name,*** then **Oracle BPEL Process Analytics**, and then click **Start BPA Server.**

    A command window opens, as shown in Figure 2–2. It indicates that Oracle Process Manager and Notification Server (OPMN) is starting the processes required by Oracle BPEL Process Analytics - the engine and the monitor framework. The **engine** starts all of the services required by Oracle BPEL Process Analytics; the **monitor framework** starts the event source monitors, and captures and instruments events from the event sources.

    When the command window closes, it indicates that all of the required processes have been started. Wait until the window closes before proceeding to step 2.

Figure 2–2   Start BPA Server Command Window
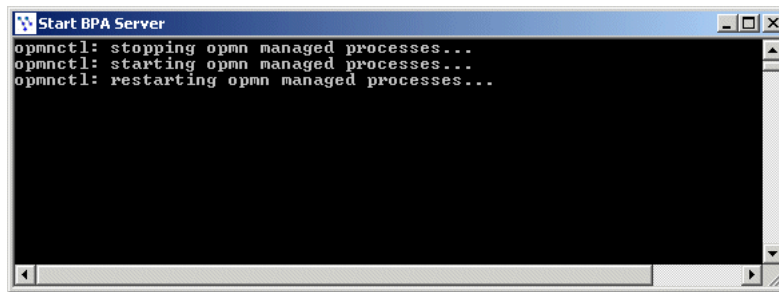
2. On the desktop, click **Start,** then choose **All Programs,** then *Oracle – OAS_ Home_Name,* then **Oracle BPEL Process Analytics,** and then click **BPA Console.**

    The login window opens, as shown in Figure 2–3.

Stopping the Server on Microsoft Windows

To stop the Oracle BPEL Process Analytics Server on the Microsoft Windows desktop, click **Start,** then choose **All Programs,** then **Oracle BPEL Process Analytics,** and then click **Stop BPA Server.**

Starting the Server and Opening the Console on Unix Systems

To start Oracle BPEL Process Analytics on a Unix system and open the Console, follow these steps:

1. From the Oracle Process Manager and Notification Server (OPMN) `bin` directory in the Oracle home where you installed Oracle BPEL Process Analytics, issue the `opmnctl startall` command.

    ```
    > opmnctl startall
    ```

2. Open a Web browser and enter the following URL, where *system* is the system on which you installed Oracle BPEL Process Analytics and *port* is the Oracle HTTP Server port for Oracle BPEL Process Analytics. (For information on determining the port number, see "Determining Port Numbers for Event Source Specifications and URLs" on page 5-10.)

    ```
    http://system:port/bam/pages/login.uix
    ```

    The login page opens, as shown in Figure 2–3.

Stopping the Server on Unix

To stop the Oracle BPEL Process Analytics server on a Unix system, follow these steps:

1. Set the current directory to the Oracle Process Manager and Notification Server (OPMN) `bin` directory in the Oracle home where you installed Oracle BPEL Process Analytics.

2. Issue the `opmnctl stopall` command.

    ```
    > opmnctl stopall
    ```

Logging In to the Admin Console

When the Oracle BPEL Process Analytics Console opens, you are presented with the Login page, as shown in Figure 2–3.

Enter `Administrator` for in the **Username** box and the password for the Administrator account in the **Password** box, and then click **Login.**

By default, two accounts are created when Oracle BPEL Process Analytics is installed: an Administrator and a Manager account. For each account, the password is the same as the user name. The Administrator account has full access to the Oracle BPEL Process Analytics Console, which means access to both the Admin Console and the Dashboard. The Manager account is the default account for a business manager and has access to the Dashboard only.

Although the default password for the Administrator account is `Administrator`, this value was should have been changed by the installer after Oracle BPEL Process Analytics was installed. See "Changing the Administrator and Other User Passwords" on page 5-7 for details.

After you log in, the interface opens and presents an overview of the Admin Console, as shown in Figure 2–1.

Figure 2–3   Oracle BPEL Process Analytics Console Login Page
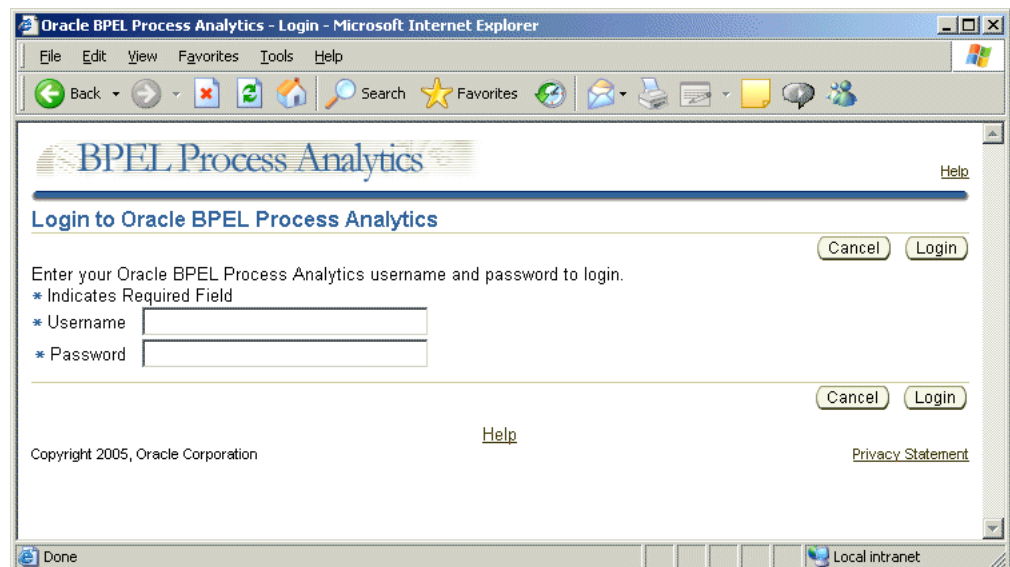
General Procedure for Using the Admin Console

> **Note:**   Be aware that Oracle BPEL Process Analytics uses Oracle Application Server Java Authentication and Authorization Service (JAAS) Provider in Oracle Application Server Containers for J2EE (OC4J) to restrict access to the Oracle BPEL Process Analytics Console. See Chapter 5 "Administering Oracle BPEL Process Analytics" for more information.

The basic procedure for using the Admin Console is as follows:

1.   Specify the event sources.

For Oracle BPEL Process Analytics to present the event metadata for modeling, it must be connected to the event source or sources that generate the events you want to monitor. See "Creating, Modifying, and Deleting Event Source Connections" on page 2-7 for more information.

2. If desired, create one or more dimensions.

   If you plan to use dimensions to filter KPI values, then the most efficient approach is to specify the dimensions before creating the composite event on which the KPI will be based. However, a composite event can be modified to add a dimension, if needed. Oracle BPEL Process Analytics provides a wizard to assist you in creating dimensions. See "Creating Dimensions" on page 2-34 for more information.

3. Create one or more composite events.

   Before you can model metrics or KPIs, you must create the composite events on which they will be based. Oracle BPEL Process Analytics provides a wizard to assist you in creating composite events. See "Creating Composite Events" on page 2-23 for more information.

4. If desired, create one or more metrics.

   The Admin Console provides a wizard to assist you in creating metrics. See "Creating Metrics" on page 2-26 for more information.

5. If desired, create one or more KPIs.

   The Admin Console provides a wizard to assist you in creating KPIs. See "Creating Key Performance Indicators" on page 2-29 for more information.

6. Add user accounts.

   You must specify a user name and password for the Dashboard user accounts and designate which composite events, metrics, and KPIs each account can view using the Dashboard. Optionally, you can specify alerts for the user. See "Creating User Accounts" on page 2-37 for more information about creating user accounts.

7. Configure the Dashboard.

   For simplicity, a number of Dashboard views are defined by default, with only minor adjustments required by you, the administrator. In addition, you can create custom Dashboard pages to provide a more tailored presentation to the Dashboard user or users. See Chapter 3"Using the Admin Console to Configure the Dashboard" for more information.

Creating and Managing Event Source Connections and KPI Document Sources

The Admin section of the Admin Console provides pages and wizards to assist you in setting up and managing the sources from which events are to be captured, as well as pages for setting up KPI context documents. This section includes instructions on performing each of the following tasks:

- Creating, Modifying, and Deleting Event Source Connections on page 2-7

- Information Required to Create or Modify Event Source Connections on page 2-7

- Viewing Existing Event Sources on page 2-16

- Modifying Event Sources on page 2-17

- Deleting Event Sources on page 2-17

- Defining KPI Context Documents on page 2-17

Creating, Modifying, and Deleting Event Source Connections

You must specify the connection information for each event source that contains events that you want to model for presentation in the Oracle BPEL Process Analytics Dashboard. When you complete an event source specification, a connection is created between the Admin Console and the specified event source. This connection allows you to browse the event metadata when you create composite events and provides the connection information needed to allow the Dashboard to be populated with real-time data.

> **Note:** Multiple instances of Oracle BPEL Process Analytics should not be configured to monitor the same event source. See "Running Multiple Instances of Oracle BPEL Process Analytics" on page 5-14 for more information.

To create, modify, or delete an event source connection:

1. In the Admin Console, click **Admin** to open the Admin page.

2. Click **Event Source** to open the Event Sources page, then do one of the following, depending on which operation you want to perform:

   - To create an event source specification

     Click the **Create** button corresponding to the type of event source you want to specify. A page that is specific to the type of event source for which you are creating a connection opens. Specify the event source connection parameters requested.

     See "Information Required to Create or Modify Event Source Connections" on page 2-7 for a detailed description of the event source connection parameters.

   - To modify an event source specification

     Click the **pencil** icon that corresponds to the event source that you want to modify. The Edit Event Source page opens. Update the event source connection parameters, as needed.

     See "Information Required to Create or Modify Event Source Connections" on page 2-7 for a detailed description of the event source connection parameters.

   - To delete an event source specification

     Click the **trash can** icon that corresponds to the event source that you want to delete. A confirmation page opens. In the confirmation page, review the list of objects that will be deleted as a result of deleting this event source. Click **Yes** if you want to delete the event source; otherwise, click **No**.

Information Required to Create or Modify Event Source Connections

Depending on the type of event source to which it is connecting, Oracle BPEL Process Analytics requires different information, and may have some restrictions on how events are generated, as described in the following topics:

- Restrictions and Recommendations For Oracle BPEL Process Manager Sensors on page 2-8

- Oracle BPEL Process Manager Event Source Configuration Values on page 2-11

- [Generic Event Source Configuration Values](#) on page 2-13
- [Event Simulator Configuration Values](#) on page 2-16

Restrictions and Recommendations For Oracle BPEL Process Manager Sensors

The following topics describe restrictions and recommendations on using Oracle BPEL Process Manager sensors that will be captured by Oracle BPEL Process Analytics:

- [Only Activity Sensors Are Supported](#) on page 2-8
- [Activity Sensors Can Be Associated with messageType Variable Only](#) on page 2-8
- [Database Sensor Action Can Be Used for Debugging Sensor Output](#) on page 2-8
- [Designing a Sensor for Oracle BPEL Process Analytics](#) on page 2-9
- [Define Sensors for End Events Last](#) on page 2-10
- [Process Versioning Is Not Supported](#) on page 2-11

For complete information about defining and deploying activity sensors, see *Oracle BPEL Process Manager Developer's Guide*.

Only Activity Sensors Are Supported Oracle BPEL Process Manager supports the following types of sensors: variable, activity, and fault. However, Oracle BPEL Process Analytics supports activity sensors only.

Activity Sensors Can Be Associated with messageType Variable Only Each Oracle BPEL Process Manager activity sensor must be associated with a variable. The state of the variable is sent to Oracle BPEL Process Analytics when the sensor fires. Oracle BPEL Process Analytics supports sensors defined on variables of type messageType only. For example:

```
<variable name="inputVariable" messageType="client:pcshopRequestMessage"/>
<variable name="outputVariable" messageType="client:pcshopResponseMessage"/>
```

As the example shows, the messageType variables correspond to input and output parameters of the Oracle BPEL Process Manager invoke activity.

Oracle BPEL Process Analytics does not support sensors defined on variables that are primitives, such as the following:

```
<variable name="finalresult" type="xsd:string">
<variable name="resultReason" type="xsd:string">
```

Database Sensor Action Can Be Used for Debugging Sensor Output The Oracle BPEL Process Manager database sensor action results in the sensor data being displayed in the Oracle BPEL Process Manager Console. This is useful for debugging and to check that sensors are being fired as expected. The Oracle BPEL Process Manager administrator must add the database sensor action when defining sensors, to enable the display of sensor data in the Oracle BPEL Process Manager Console. Figure 2–4 shows the Oracle BPEL Process Manager Designer window used for defining the database sensor action.

Figure 2–4   Database Sensor Action

Designing a Sensor for Oracle BPEL Process Analytics Follow these steps to create an Oracle BPEL Process Manager sensor for Oracle BPEL Process Analytics to monitor:

1. Create the sensor actions

   The JMS Queue sensor action is required. The Database sensor action is optional, but is helpful for debugging purposes. See "Database Sensor Action Can Be Used for Debugging Sensor Output" on page 2-8 for more information.

2. Create the desired sensors.

   When you create the sensor, select Completion for the evaluation time, as shown in Figure 2–5.

   Be aware that sensors must be defined on messageType variables. See "Activity Sensors Can Be Associated with messageType Variable Only" on page 2-8 for more information.

   Choose only the part name from the Variable XPath Builder, as shown in Figure 2–6. Do not select attributes; the attributes are selected when the sensor is modeled in the Oracle BPEL Process Analytics Admin Console.

3. Add the sensor actions to the sensors.

4. Save the sensors and sensor actions.

5. Deploy the sensors.

Figure 2–5   Create Activity Sensor
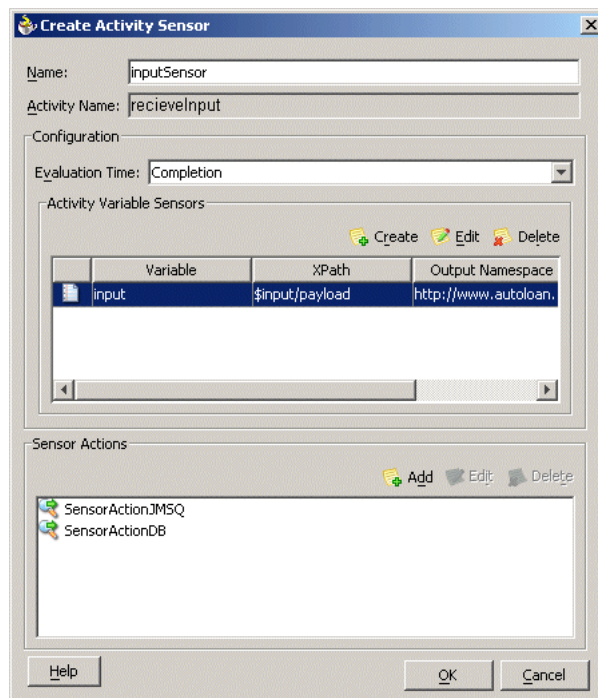
Figure 2–6   Variable XPath Builder
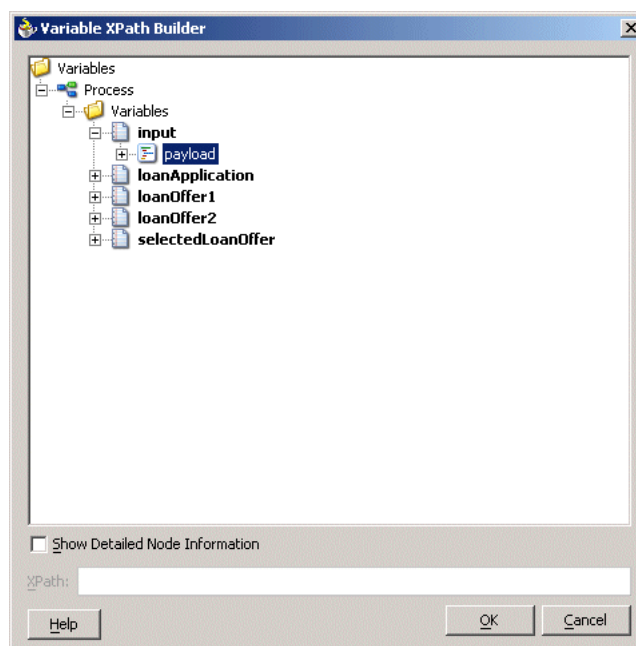
Define Sensors for End Events Last If an activity in an Oracle BPEL Process Manager process is an end activity, and there is more than one sensor defined on the same activity with one event as the end event, then define the sensor for the end event sensor after defining all the other sensor. Otherwise, a situation can occur where Oracle BPEL Process Analytics shows the other sensor events (not corresponding to the end event) as Not Completed or Pending in the Dashboard. This, in turn, will distort time-based KPIs, if any are defined for the other sensor events. See

"Understanding the Real-Time Viewer Table" on page 4-15 for information about Not Completed and Pending events.

Process Versioning Is Not Supported Oracle BPEL Process Manager supports process versioning, which enables users to deploy different versions of the same process and specify any existing version for process deployment.

Oracle BPEL Process Analytics, however, does not support process versioning. If processes are deployed with multiple versions, Oracle BPEL Process Analytics uses the latest version. Therefore, if you create a composite event with one version of a process, and a new version of the process is created later, runtime errors will result when you create a composite event based on the newer version of the process.

Oracle BPEL Process Manager Event Source Configuration Values

For Oracle BPEL Process Analytics to capture events from an Oracle BPEL Process Manager event source, the Oracle BPEL Process Manager administrator must insert sensors that capture the events of interest and then publish those events to a Java Message Service (JMS) queue. Oracle BPEL Process Analytics connects to the JMS queue to capture the events and uses the Web Services Description language (WSDL) URL to browse the event metadata.

See "Restrictions and Recommendations For Oracle BPEL Process Manager Sensors" on page 2-8 for information about the types of sensors Oracle BPEL Process Analytics supports.

> **Note:** After you configure Oracle BPEL Process Manager as an Oracle BPEL Process Analytics event source, you may receive one or more error messages such as the following:
>
> ```
> <BaseCubeSessionBean::logError> Error while invoking bean "process
> manager":
> Cannot find process content file. Cannot find the content file
> "sensor.xml" for process "TaskActionHandler" (revision "").
> Please check that the path "sensor.xml" is correct and refers to a
> file that has been included as a part of the deployment archive.
> ```
>
> This message is a result of Oracle BPEL Process Analytics requiring a sensor.xml file for every deployed Oracle BPEL Process Manager process that it will monitor. If such a message is returned for a process that you do not intend to monitor using Oracle BPEL Process Analytics, you can safely ignore it.

Table 2–1 describes the values Oracle BPEL Process Analytics requires to create the JMS queue and form a connection.

Table 2–1   Oracle BPEL Process Manager Event Source Information

Page Element	Description
Name	A unique name for this connection to the Oracle BPEL Process Manager event source. The name must be an alphanumeric string of 50 characters or fewer. If you are updating an existing event source, then this value cannot be changed.
Description	A description of this event source connection. The description must be a string of 50 characters or fewer.

Table 2–1   (Cont.) Oracle BPEL Process Manager Event Source Information

Page Element	Description
JMS Queue Hostname	The name of the system on which the Java Message Service (JMS) queue associated with Oracle BPEL Process Manager is running.
Event Source Installation	The Oracle BPEL Process Manager installation type. If the installation type was BPEL Process Manager for OracleAS Middle Tier, then select **Midtier.** If the installation type was Oracle BPEL Process Manager for Developers, then select **Standalone.**
ORMI port (Standalone)	The OC4J Remote Method Invocation (ORMI) port of the JMS queue associated with Oracle BPEL Process Manager.
	Specify a value for this field when the value specified for the **Event Source Installation** field is **Standalone.** (If the Event Source Installation is specified as Midtier, then this field is ignored.)
	The default value is 23791.
OPMN port (Midtier)	The Oracle Process Manager and Notification Server (OPMN) ONS Request port of the Oracle Application Server middle tier where Oracle BPEL Process Manager is installed.
	Specify a value for this field when the value specified for the **Event Source Installation** field is **Midtier.** (If the Event Source Installation is specified as Standalone, then this field is ignored.)
	This port number is dynamically assigned and must be updated each time Oracle Application Server is restarted. Its initial default value is 6003.
	See "Determining Port Numbers for Event Source Specifications and URLs" on page 5-10 for information about using the Oracle Enterprise Manager 10*g* Application Server Control to determine port numbers.
Administrator User Name	The user name for the administrator account of the Oracle Application Server Containers for J2EE (OC4J) instance associated with Oracle BPEL Process Manager. By default, this value is admin.
Administrator Password	The password for the administrator account of the OC4J instance associated with Oracle BPEL Process Manager.
JMS Queue Factory	The name of the connection factory for the JMS queue. By default, this value is jms/QueueConnectionFactory.
JMS Queue Name	The name of the JMS queue configured on the Oracle BPEL Process Manager database from which Oracle BPEL Process Analytics will receive events. By default, this value is jms/demoQueue.
Application Initial Context	The application initial context of the JMS queue. By default, this value is com.evermind.rmi.RMIInitialContextFactory.

Table 2–1   (Cont.) Oracle BPEL Process Manager Event Source Information

Page Element	Description
WSIL URL	The URL for the Oracle BPEL Process Manager Web Services Inspection Language (WSIL) document from which Oracle BPEL Process Analytics will browse event metadata.
	This page element is prefilled; however, you must replace the string `<HOST>:<PORT>` with the name of the system on which Oracle BPEL Process Manager is installed and with the WSIL port number.
	If the Event Source Installation is specified as Midtier, the default port number is 80.
	If the Event Source Installation is specified as Standalone, the default port number is 9700.
	See "Determining Port Numbers for Event Source Specifications and URLs" on page 5-10 for information about using the Oracle Enterprise Manager 10*g* Application Server Control to determine port numbers.

Generic Event Source Configuration Values

For Oracle BPEL Process Analytics to capture events from a generic event source, the administrator for the event source must specify the connection parameters and publish the events of interest to one of the following types of queues:

- An Oracle Java Message Service (OJMS) queue preconfigured by Oracle BPEL Process Analytics, called `bam_genericmonq`.

   Table 2–2 describes the connection parameter values Oracle BPEL Process Analytics requires to specify a generic event source using the `bam_genericmonq` queue.

- A user-specified OJMS queue.

   Table 2–3 describes the connection parameter values Oracle BPEL Process Analytics requires to specify a generic event source using a user-specified OJMS queue.

- A user-specified Java Message Service (JMS) queue

   Table 2–4 describes the values Oracle BPEL Process Analytics requires to specify a generic event source using a user-specified JMS queue.

Details about publishing events for the generic monitor are described in Chapter 7.

Table 2–2   Generic Event Source Information - Using the Preconfigured bam_genericmonq

Page Element	Description
Name	A unique name for this connection to the event source. The name must be an alphanumeric string of 50 characters or fewer. If you are updating an existing event source, then this value cannot be changed.
Description	A description of this event source connection. The description must be a string of 50 characters or fewer.
JMS Queue Hostname	The host name of the server where Oracle BPEL Process Analytics is installed.
Event Source Installation	The event source installation type. For the bam_genericmonq this must be the Oracle Application Server middle tier, so select **Midtier.**

Table 2–2   (Cont.) Generic Event Source Information - Using the Preconfigured bam_genericmonq

Page Element	Description
ORMI port (Standalone)	This field is ignored when the events source installation specified is **Midtier.**
OPMN port (Midtier)	The Oracle Process Manager and Notification Server (OPMN) ONS Request port of the Oracle Application Server middle tier where Oracle BPEL Process Analytics is installed.
	The initial default value is 6003. To verify this value for your installation, see "Determining Port Numbers for Event Source Specifications and URLs" on page 5-10 for information about using the Oracle Enterprise Manager 10*g* Application Server Control to determine port numbers.
Administrator User Name	The user name for the administrator account associated with the OC4J where the preconfigured OJMS queue is configured. By default, the user name is `admin`.
Administrator Password	The password for the administrator account associated with the OC4J where the preconfigured OJMS queue is configured.
JMS Queue Factory	The queue factory name for the JMS queue. Accept the default value: `java:comp/resource/BAM/QueueConnectionFactories/qcf`
JMS Queue Name	The name of the JMS queue to which you want to publish the generic events. Accept the default value: `java:comp/resource/BAM/Queues/bam_genericmonq`
Application Initial Context	The initial context factory for application-client contexts. Accept the default value: `com.evermind.server.rmi.RMIInitialContextFactory`

Table 2–3   Generic Event Source Information - Using a User-Specified OJMS Queue

Page Element	Description
Name	A unique name for this connection to the event source. The name must be an alphanumeric string of 50 characters or fewer. If you are updating an existing event source, then this value cannot be changed.
Description	A description of this event source connection. The description must be a string of 50 characters or fewer.
JMS Queue Hostname	The host name of the system where the user-specified OJMS queue is running.
Event Source Installation	The event source installation type. Select **Midtier.**
ORMI port (Standalone)	This field is ignored when the events source installation specified is **Midtier.**

Table 2–3 (Cont.) Generic Event Source Information - Using a User-Specified OJMS

Page Element	Description
OPMN port (Midtier)	The Oracle Process Manager and Notification Server (OPMN) ONS Request port of the Oracle Application Server middle tier where the event source is installed.
	The initial default value is 6003. To verify this value for your installation, see "Determining Port Numbers for Event Source Specifications and URLs" on page 5-10 for information about using the Oracle Enterprise Manager 10*g* Application Server Control to determine port numbers.
Administrator User Name	The user name for the administrator account associated with the OC4J where the OJMS queue is configured.
Administrator Password	The password for the administrator account associated with the OC4J where the OJMS queue is configured.
JMS Queue Factory	The queue factory name for the OJMS queue. Replace the default string with the following, where *JMS-provider* is the name of the JMS provider you configured:
	`java:comp/resource/JMS-provider/QueueConnectionFactories/qcf`
JMS Queue Name	The name of the JMS queue to which you want to publish the generic events. Replace the default string with the following, where *JMS-provider* is the name of the JMS provider you configured and *JMS-queue-name* is the name of the OJMS queue you configured:
	`java:comp/resource/JMS_provider/Queues/OJMS-queue-name`
	Note: You must configure the data source and a resource provider for the queue as described in *Oracle Application Server Containers for J2EE Services Guide.*
Application Initial Context	The initial context factory for application-client contexts. Accept the default value:
	`com.evermind.server.rmi.RMIInitialContextFactory`

Table 2–4 Generic Event Source Information - Using a User-Specified JMS Queue

Page Element	Description
Name	A unique name for this connection to the event source. The name must be an alphanumeric string of 50 characters or fewer. If you are updating an existing event source, then this value cannot be changed.
Description	A description of this event source connection. The description must be a string of 50 characters or fewer.
JMS Queue Hostname	The host name of the system where the JMS queue is running.
Event Source Installation	The event source installation type. Select **Standalone.**
ORMI port (Standalone)	The OC4J Remote Method Invocation (ORMI) port of the JMS queue.
	See the `rmi.xml` file for your standalone installation of OC4J for the port number.
OPMN port (Midtier)	This field is ignored when the events source installation specified is **Standalone.**

Table 2–4   (Cont.) Generic Event Source Information - Using a User-Specified JMS

Page Element	Description
Administrator User Name	The user name for the administrator account associated with the OC4J where the JMS queue is configured.
Administrator Password	The password for the administrator account associated with the OC4J where the JMS queue is configured.
JMS Queue Factory	The queue factory name for the JMS queue. Replace the default string with this string: `jms/QueueConnectionFactory`
JMS Queue Name	The name of the JMS queue from which Oracle BPEL Process Analytics will receive events. Specify the following string where *JMS-queue-name* is the name of the JMS queue: `jms/JMS-queue-name`
Application Initial Context	The initial context factory for application-client contexts. You can accept the default value: `com.evermind.server.rmi.RMIInitialContextFactory` If you prefer to use a different context factory, refer to standard texts on Java 2 Platform, Enterprise Edition (J2EE) for information about specifying an initial context factory.

Event Simulator Configuration Values

For Oracle BPEL Process Analytics to capture events generated by an event simulator, you must set up an event simulator as described in Chapter 8"Simulating Events".

Table 2–5 describes the values Oracle BPEL Process Analytics requires to capture events generated by an event simulator.

Table 2–5   Event Simulator Event Source Information

Page Element	Description
Name	A unique name for this connection to the event source. The name must be an alphanumeric string of 50 characters or fewer. If you are updating an existing event source, this value cannot be changed.
Description	A description of this event source connection. The description must be a string of 50 characters or fewer.
Event Simulator Event Source Data Directory	The directory specification for a simulated composite event. See "Setting Up the Directory Structure for the Required Files" on page 8-2 for more information.

Viewing Existing Event Sources

You can view a summary page that presents a list of all of the existing event sources.

To view existing event sources:

1. In the Admin Console, click **Admin** to open the Admin page.

2. Click **Event Source** to open the Event Sources page. This page lists all previously created event sources.

3. Click an entry in the **Event Source Name** column to view the event source configuration details.

4. Click **Help** (or see "Information Required to Create or Modify Event Source Connections" on page 2-7) for information about the configuration details.

5. Click **OK** when you are done viewing the event source configuration details.

Modifying Event Sources

If connection parameters for an event source change (or are entered incorrectly), you can modify any or all of the connection parameters, except the event source name.

To modify an event source:

1. In the Admin Console, click **Admin** to open the Admin page.

2. Click **Event Source** to open the Event Sources page.

3. Click the **pencil** icon that corresponds to the event source that you want to modify. The Edit Event Source page opens.

4. Click **Help** (or see "Information Required to Create or Modify Event Source Connections" on page 2-7) for information about the page element values.

5. Edit the values that you want to modify.

6. Click **Finish**.

Deleting Event Sources

If an event source is no longer being used, then you can delete it. When you delete an event source, any objects that depend on that event source are also deleted. The Admin Console presents a list of the dependent objects for you to review before you finalize the delete operation.

To delete an event source:

1. In the Admin Console, click **Admin** to open the Admin page.

2. Click **Event Source** to open the Event Sources page.

3. Click the **trash can** icon that corresponds to the event source you want to delete. A confirmation page opens.

4. In the confirmation page, review the list of objects that will be deleted as a result of deleting this event source.

5. Click **Yes** if you want to continue with the operation to delete the event source; otherwise, click **No**.

Defining KPI Context Documents

A **KPI context document** is a document that you think may be of interest to the Dashboard user when evaluating a KPI.

You make the documents accessible to the Dashboard user by placing them in a directory accessible to a Web-based Distributed Authoring and Versioning (WebDav) server and then specifying a carefully-worded description of the KPI when you define it. Oracle BPEL Process Analytics uses the words in the KPI description as keywords for searching the documents and determining which of them, if any, are likely to provide information about that KPI.

Any text document (plain or HTML) can serve as a KPI context document, as well as documents generated by the following applications:

■ Microsoft Word

- Microsoft Excel

- Microsoft Output

See Figure 4–24 for an example of how documents configured as KPI context documents are presented in the Dashboard. If they exist, KPI context documents are included in the Dashboard automatically.

Defining documents as KPI context documents and making them available to Dashboard users is a three-part process, as follows:

1. Place the documents in a directory that is accessible through the WebDAV protocol.

   For information about the WebDAV protocol, see

   http://www.webdav.org/

2. When you specify (or modify) a KPI, ensure that the description contains meaningful keywords. Oracle BPEL Process Analytics uses the words in the KPI description to search for and identify context documents that appear to be a match on the basis of those keywords.

3. Specify the document as a KPI context document, as follows:

   a. In the Admin Console, click **Admin** to open the Admin page.

   b. Click **Document Source** to open the Context Document Sources page.

   c. Click **Set Host** to open the Document Source Server Information page. Enter values as described in the following list:

      – Host

        Enter the name of the system hosting the documents you want to serve as context documents for your KPIs.

      – User

        Enter the name of the account that Oracle BPEL Process Analytics will use to access the system.

      – Password

        Enter the password of the account that Oracle BPEL Process Analytics will use to access the system.

      – Confirm Password

        Reenter the password.

   d. Click **Apply.** The Context Document Sources page displays.

   e. Click **Add Path** to open the Path page.

   f. In the **Path** box, enter the path to the directory (accessible through the WebDAV protocol) that contains the documents that will serve as context documents for your KPIs.

   g. Repeat steps c through f if there are additional hosts and paths to specify.

Creating and Managing Modeled Objects

The Modeling section of the Admin Console provides access to the pages of the Oracle BPEL Process Analytics Console that allow you to create and view each of the Oracle BPEL Process Analytics objects.

For information about modeling objects, see the following topics:

- Approaches to Modeling on page 2-19
- Required Date Format on page 2-19
- Modeling Composite Events and Composite Event Groups on page 2-20
- Modeling Metrics on page 2-25
- Modeling KPIs and Related Objects on page 2-27
- Modeling Dimensions on page 2-32

Approaches to Modeling

There are two approaches to consider when deciding how to model event data using Oracle BPEL Process Analytics:

- Top-down approach – This approach includes these steps:
  1. Identify all KPIs that need to be monitored within your enterprise.

     Using the loan procurement process illustrated in Figure 1–1 as an example, some KPIs to consider are the total number of loan applications processed, the total number of loans accepted, and the average time it takes for each loan service to return an offer.
  2. Identify activities that allow the KPI to be monitored.
  3. Use Oracle BPEL Process Analytics to create a composite event for each activity that needs to be monitored. This can be a specific event in a business process, multiple events in a business process, or can be events that span multiple business processes.
  4. Configure the Dashboard for the business analyst.
- Bottom-Up Approach – This approach includes these steps:
  1. Create a composite event based on all of the activities within a business process, such that one composite event maps to one business process.
  2. Define the metrics and KPIs based on the data available in the composite event.
  3. Configure the Dashboard for the business analyst.

When using Oracle BPEL Process Analytics for the first time, you might find the bottom-up approach to be the easiest. The first time you model event data, you might want to identify one business process of interest, model the KPIs and metrics for that process, and then use the default Dashboard views to display those composite events, KPIs, and metrics. (See "Default Dashboard Pages" on page 3-8 for information about Dashboard default pages.) After the first business process is modeled and the data is available for review in the Dashboard, you will have a better idea of what additional business processes you want to model and how you want to model them.

Typically, modeling data and setting up the Dashboard using Oracle BPEL Process Analytics is an iterative process.

Required Date Format

Oracle BPEL Process Analytics requires that event date attributes to conform to the XML Schema dateTime data type, as described by the following format:

```
yyyy-MM-dd'T'HH:mm:ss[.SSS] Z | [+|-]hh:mm
```

The values in this format have the following meanings:

- yyyy - year, expressed as a number

- MM - month, expressed as a number

- dd - day in month, expressed as a number

- T - a literal value indicating the start of the time element

- HH - hour in day, expressed as a number from 0 to 23

- mm - minute in hour, expressed as a number

- ss - second in minute, expressed as a number

- SSS - millisecond, expressed as a number

- Z - Coordinated Universal Time (UTC)

- [+ |-]hh:mm - the local time zone, specified as the number of hours and minutes ahead of, or behind UTC, respectively.

The following examples demonstrate this format:

- April 22, 2005 at noon Eastern Standard Time (EST) is formatted as:

  ```
  2005-04-22T12:00:00-05:00
  ```

- June 22, 2004 at 4:45 pm UTC can be formatted using either of the following:

  ```
  2004-06-22T16:45:00Z
  2004-06-22T16:45:00+00:00
  ```

> **Note:** Oracle recommends that all dates include the time zone. If a time zone is not provided, the time zone is assumed to be the time zone in which the Oracle BPEL Process Analytics server is running.

Modeling Composite Events and Composite Event Groups

Composite events are the prerequisite building blocks for creating metrics and KPIs. You may recall from the section entitled "Correlating Events into Meaningful Groups for Analysis" on page 1-4, that a composite event is a collection of one or more events (and their attributes) within a business process that a business analyst wants to monitor and analyze. The events within a composite event are correlated on the basis of a common event attribute. Thus, an instance of a purchase order request can be matched with the corresponding purchase order confirmation instance, for example.

Within the context of the Admin Console, a composite event can contain events from one and only one event source. This restriction is required because each type of event source uses a different structure for its event metadata. When you create a composite event, you browse event metadata, and select the event attributes of interest.

This restriction, however, does not prevent you from creating KPIs and metrics on events that span event sources. Oracle BPEL Process Analytics overcomes the restriction by providing an object called a composite event group. A **composite event group** is a collection of correlated composite events.

To create a composite event group, first you correlate the events from each event source into a separate composite event, then you correlate these composite events to create a composite event group (as shown in Figure 2–7). You can then create KPIs and metrics on the composite event group.

Figure 2–7   Composite Event Group
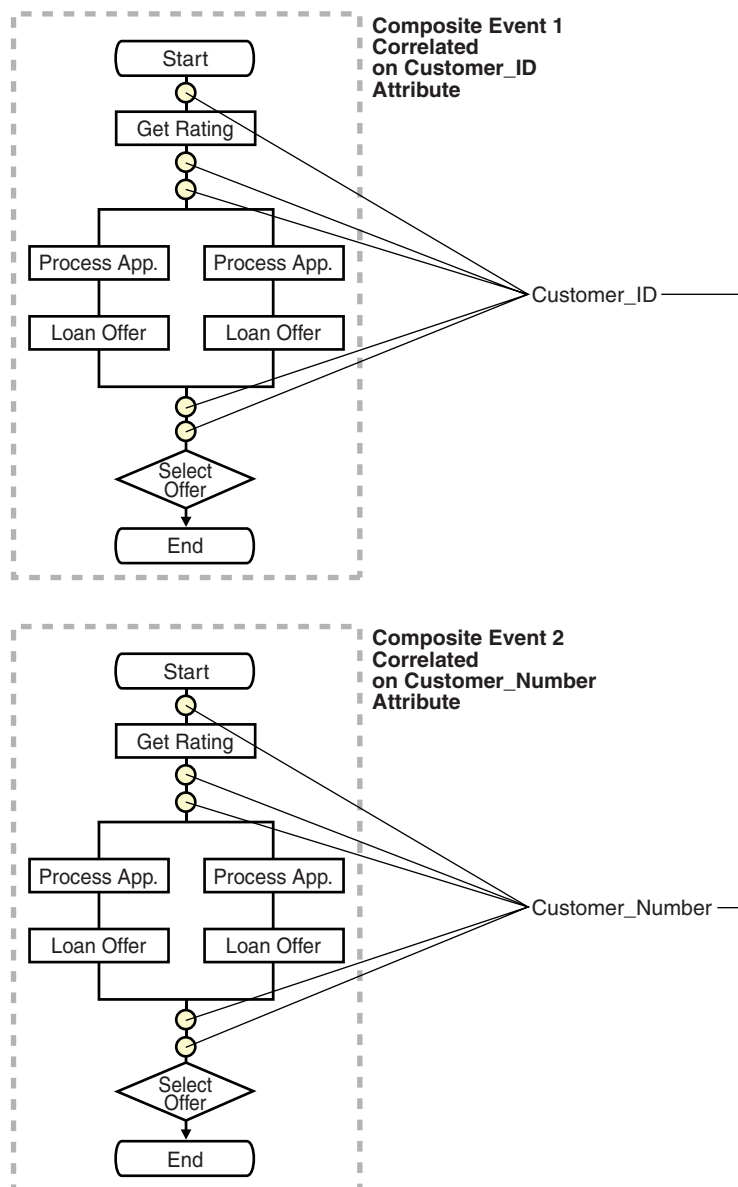
Note that a composite event group can contain just one composite event, if desired. You might include just one composite event in a composite event group to serve as a placeholder as you develop your Oracle BPEL Process Analytics configuration, or you might choose to do this so you can specify one set of dimensions for the standalone composite event and another set of dimensions for the composite event group. See "Modeling Dimensions" on page 2-32 for information about dimensions.

When composite events and composite event groups are presented in the Dashboard, a distinction between composite events and composite event groups is not made.

Oracle BPEL Process Analytics provides pages and wizards that enable you to view, update, and delete currently defined composite events and composite event groups, and create new composite events and composite event groups.

The following topics provide step-by-step instructions on viewing and modeling composite events and composite event groups:

Viewing Existing Composite Events and Composite Event Groups

You might find it useful to review existing composite events and composite event groups as you proceed with data modeling. You can view composite events and composite event groups, as well as access details about them from the Composite Events and Groups page. See the following topics for more information:

■ Viewing a List of Existing Composite Events and Groups on page 2-22

■ Viewing Details About a Composite Event Group on page 2-22

■ Viewing Details About a Composite Event Within a Composite Event Group on page 2-22

■ Viewing Details About a Composite Event on page 2-23

Viewing a List of Existing Composite Events and Groups

1. In the Admin Console, click **Modeling** to open the Modeling page.

2. Click **Composite Events** to open the Composite Events and Groups page.

This page displays the name, type (the event source for the composite event, or in the case of a composite event group, the value `Group`), and the administrator-specified description for each composite event and each composite event group.

Viewing Details About a Composite Event Group

To view details about a composite event group:

1. In the Admin Console, click **Modeling** to open the Modeling page.

2. Click **Composite Events** to open the Composite Events and Groups page.

3. In the Composite Event and Groups page, click the name of a composite event group to display the list of composite events contained in that group.

4. Click **Show Properties** to display the following composite event group properties:

   ■ Name of the composite event group

   ■ Administrator-defined description of the composite event group

   ■ Timeout value for the composite event group, expressed in seconds

   ■ Dimensions defined on the composite event group

   ■ Composite events in the composite event group

   ■ End events for the composite event group

Viewing Details About a Composite Event Within a Composite Event Group

To view details about a composite event within a composite event group:

1. In the Admin Console, click **Modeling** to open the Modeling page.

2. Click **Composite Events** to open the Composite Events and Groups page.

3. In the Composite Event and Groups page, click the name of a composite event group to display the list of composite events contained in that group.

4. Click the name of a composite event to view details about the composite event.

Viewing Details About a Composite Event

To view details about a composite event:

1. In the Admin Console, click **Modeling** to open the Modeling page.

2. Click **Composite Events** to open the Composite Events and Groups page.

3. In the Composite Event and Groups page, click the name of the composite event to view these composite event details:

   - The name of each event included in the composite event

   - Which event or events are the end events for the composite event

   - The composite event group to which the composite event belongs, if applicable

   - The attribute used to correlate the events

   - Each dimension associated with the composite event and the event attribute to which it is mapped

4. Select an event to view these details about that event's attributes:

   - The administrator-specified alias for each event attribute

   - The administrator-specified type for each event attribute

   - The XPath to each event attribute

Creating Composite Events

You must create one or more composite events before you can create KPIs or metrics.

To create a composite event:

1. In the Admin Console, click **Modeling.**

2. Click **Composite Events** to open the Composite Events and Groups page.

3. Click **Add Composite Event**. The Modeling – Create Composite Event page opens.

4. Select the row containing the event source name and event source type for which you want to create a composite event, then click **Create**. The Create Composite Event Wizard opens.

5. Follow the steps through the wizard. If you need help using a wizard page, click the **Help** link on that page.

Be aware that he first time you create a composite event using a generic event source, an exception such as the following is logged to the `monitor.log` file. This exception is harmless and can be safely ignored; the composite event will be created as expected.

Note that in the example, the time zone is Eastern Daylight Time (EDT). The time zone included in the log file is the time zone where the system running Oracle BPEL Process Analytics is running.

```
2005/06/06 16:01:32.781 EDT ERROR:1 MF/MF: [Generic Monitor] Error loading generic event
definitions java.io.FileNotFoundException:
```

```
C:\OraAs10g\integration\bam\config\GenericEventDefinitions.xml(The system cannot find the file
specified)
at java.io.FileInputStream.open(Native Method)
at java.io.FileInputStream.<init>(FileInputStream.java:106)
at oracle.tip.bam.monitor.generic.GenericEventDefinition.loadEventDefinitions(
GenericEventDefinition.java:137)
at oracle.tip.bam.monitor.generic.GenericEventDefinition.<init>(GenericEventDefin ition.java:92)
at oracle.tip.bam.monitor.generic.GenericMetadata.<init>(GenericMetadata.java:57).
at oracle.tip.bam.monitor.generic.GenericMonitor.getESMetadata(GenericMonitor.java:351)
at oracle.tip.bam.monitor.fw.MonitorFramework.startMonitors(MonitorFramework.java:346)
at oracle.tip.bam.monitor.fw.MetadataListenerImpl.metadataUpdated(MetadataListenerImpl.java:73)
at oracle.tip.bam.metadata.repos.GenerateReposEvent.run(GenerateReposEvent.java:1 43)
```

See "Locating Oracle BPEL Process Analytics Log Files" on page 5-12 for more information about the `monitor.log` file.

Creating Composite Event Groups

To correlate events that span event sources, you must first create at least one composite event for each event source, and then create a composite event group to correlate these individual composite events.

To create a composite event group:

1. In the Admin Console, click **Modeling.**

2. Click **Composite Events** to open the Composite Events and Groups page.

3. In the Composite Events and Groups page, click **Add Group** to open the Create Composite Events Group Wizard.

4. Follow the steps through the wizard. If you need help using a wizard page, then click the **Help** link on that page.

Note: When you create a composite event group, the composite events included in that group are no longer presented individually on the Composite Events and Groups page. You must click a group name to view the composite events within that group.

A given composite event cannot be included in two or more composite event groups; it can belong to either one composite event group or no composite event group.

Modifying Composite Events

If you find that an existing composite event is not configured as you need it, then you can modify it by following these steps:

1. In the Admin Console, click **Modeling.**

2. Click **Composite Events** to open the Composite Events and Groups page.

3. Click the **pencil** icon corresponding to the composite event that you want to modify.

4. A page opens to confirm that you want to modify the composite event. Click **Yes** to proceed; click **No** to cancel the operation. If you click **Yes**, then the Update Composite Event Wizard opens; if you click **No,** then the Composite Events and Group page opens with no changes made.

5. If you clicked **Yes** in the previous step, then follow the steps through the wizard. If you need help using a wizard page, then click the **Help** link on that page.

Modifying Composite Event Groups

If you find that an existing composite event or composite event group is not configured as you need it, then you can modify it.

Be aware that you can remove a composite event from a composite event group, but doing so will result in the entire composite event group being deleted. (You will be asked to confirm the operation before the composite event group is deleted.)

To modify a composite event group:

1. In the Admin Console, click **Modeling.**

2. Click **Composite Events** to open the Composite Events and Groups page.

3. Click the **pencil** icon corresponding to the composite event group that you want to modify. A confirmation page opens.

4. Click **Yes** if you want to continue; otherwise click **No.** If you click **Yes**, then the Update Composite Events Group Wizard opens; if you click **No,** then the Composite Events and Group page opens with no changes made.

5. If you clicked **Yes** in the previous step, then follow the steps through the wizard. If you need help using a wizard page, then click the **Help** link on that page.

Deleting Composite Events and Composite Event Groups

If you decide that a previously created composite event or composite event group is no longer needed, then you can delete it. When you request that an object be deleted, Oracle BPEL Process Analytics presents a list of other objects that will also be deleted before proceeding with the operation. The delete operation will not take effect until you confirm it.

To delete a composite event or composite event group:

1. In the Admin Console, click **Modeling.**

2. Click **Composite Events** to open the Composite Events and Groups page.

3. In the Composite Events and Groups page, click the **trash can** icon corresponding to the composite event or composite event group that you want to delete. A confirmation page opens.

4. In the confirmation page, click **Yes,** if you are sure you want to continue with the delete operation; otherwise, click **No.**

To delete a composite event that is one of multiple composite events in a composite event group, you must first modify the group to remove the composite event. See "Modifying Composite Event Groups" on page 2-25 for information about modifying a composite event group.

Modeling Metrics

A metric is the simplest kind of modeled data. You specify a metric by selecting attributes from the events contained within a composite event and, optionally, applying a calculation to those attributes.

The value returned by the expression that defines a metric must be one of the following types, referred to as a **metric value type:**

- Number

  An attribute value, or a calculation on attribute values, that can be represented as a number, for example, a credit rating.

- Duration

  An attribute value, or a calculation on attribute values, that represents the passage of time, for example, the time at which a loan was accepted minus the time at which a loan was requested.

- Currency

  An attribute value, or a calculation on attributes values, that represents a monetary value, such as the amount of a loan.

- Percent

  An attribute value, or a calculation on attributes values, such as the annual percentage rate (APR) of a loan.

- Text

  An attribute value that represents an alphanumeric string, such as a street address.

You can present metrics in Dashboard charts (clustered or stacked bar charts, line graphs, or area graphs) as a series of data points over time, or as a Dashboard table.

Oracle BPEL Process Analytics provides pages and wizards that enable you to view currently defined metrics, create new metrics, and update or delete metrics.

See the following topics for step-by-step instructions:

- Viewing Metrics on page 2-26
- Creating Metrics on page 2-26
- Modifying Metrics on page 2-27
- Deleting Metrics on page 2-27

Viewing Metrics

You can view a list of previously created metrics, as follows:

1. In the Admin Console, click **Modeling.**

2. Click **Metrics.** The Metrics page opens and displays all previously created metrics.

3. If desired, click a metric name. The Metrics Detail page opens and presents these details about the selected metric:

   - The metric name
   - The administrator-defined description of the metric
   - The metric value type
   - The composite event on which the metric is based
   - The expression that defines the metric
   - The constraints, if any, defined on the metric

   When you are done viewing metric details, click **Return to List** to return to the Metrics page.

Creating Metrics

Before you create a metric, you must first create one or more composite events or composite event groups (as described in "Modeling Composite Events and Composite Event Groups" on page 2-20).

To create a metric:

1. In the Admin Console, click **Modeling.**

2. Click **Metrics** to open the Metrics page.

3. In the Metrics page, click **Create** to open the Create Metric Wizard.

4. Follow the steps through the wizard. If you need help using a wizard page, then click the **Help** link on that page.

Modifying Metrics

After you define a metric, you may find you need to adjust it. You can change all characteristics of a metric, except the name and the composite event on which it is based.

To modify a metric:

1. In the Admin Console, click **Modeling.**

2. Click **Metrics** to open the Metrics page.

3. Click the **pencil** icon corresponding to the metric that you want to update.

4. In the confirmation page, click **Yes** if you want to continue; click **No** if you do not. If you click **Yes,** then the Update Metric Wizard opens. If you click **No,** then the Metrics page opens.

5. Follow the steps through the wizard. If you need help using a wizard page, then click the **Help** link on that page.

Deleting Metrics

If you decide that a previously created composite event is no longer needed, then you can delete it.

To delete a metric:

1. In the Admin Console, click **Modeling.**

2. Click **Metrics** to open the Metrics page.

3. Click the **trash can** icon corresponding to the metric you want to delete.

4. In the confirmation page, click **Yes** if you want to delete the metric; click **No** if you do not. If you click **Yes,** then the metric is deleted and the Metrics page opens. If you click **No,** then the metric is not deleted, and the Metrics page opens.

> **Note:** If you delete a metric and then reuse that metric's name for a new metric, values corresponding to the deleted metric are displayed in the Dashboard. You can avoid this situation by not reusing metric names.

Modeling KPIs and Related Objects

KPIs are the basis for providing the business analyst with the ability to perform root cause analysis and anticipate problems before they occur.

You specify a KPI by applying an expression to an event attribute (or set of attributes) within a composite event that is optionally constrained by one or more dimension values and then aggregating all of the values for that composite event over a given time period.

The expression you specify, may contain references to one or more composite event attributes, constant values, and mathematical operators

You can use the following mathematical functions to aggregate the composite event instances for presentation in the Dashboard:

- Average – The average of the values for a given time range. For example, you can use the average function to return the average monetary value of all instances of loan offers made during a given time range.

- Count – The number of times the composite event value occurs for a given time range. For example, you can use the count function to return the total number of loan offers made during a specified time range.

- Maximum – The maximum composite event value for a given time range. For example, you can use the maximum function to determine which value represents the loan offer with the highest monetary value during a specified time range.

- Minimum – The minimum composite event value for a given time range. For example, you can use the minimum function to determine which value represents the loan offer with the lowest monetary value during a specified time range.

- Sum – The sum of the values for a given time range. For example, you can use the sum function to return the monetary value of all instances of loan offers made during a given time range.

Similar to a metric, the value returned by the expression that defines a KPI must be one of the following types, referred to as a **KPI value type:** Number, Duration, Currency, or Percent. Note that unlike metrics, KPIs cannot evaluate to a type of text. However, you can use the count function to count attribute instances of type text.

Oracle BPEL Process Analytics provides pages and wizards that enable you to view the currently defined KPIs, create new KPIs, and update or delete KPIs. In addition, you can define KPI performance bands and context documents to provide more data in the Dashboard. Before you create a KPI, the composite event on which it will be based must be created.

See the following topics for step-by-step instructions:

- Defining KPI Context Documents on page 2-17
- Viewing Key Performance Indicators on page 2-28
- Creating Key Performance Indicators on page 2-29
- Modifying Key Performance Indicators on page 2-29
- Deleting Key Performance Indicators on page 2-29
- Defining Key Performance Indicator Target Values on page 2-30
- Viewing Key Performance Indicator Target Values on page 2-31

You can enhance the level of detail by defining dimensions. See "Modeling Dimensions" on page 2-32 for information.

For information about presenting existing KPIs in the Dashboard, see "Designing Customized Key Performance Indicator View Panes" on page 3-10.

Viewing Key Performance Indicators

To view previously created KPIs:

1. In the Admin Console, click **Modeling.**

2. Click **KPIs** to open the KPI page. The following properties are displayed:

- The KPI name

- The name of the composite event on which the KPI is based

- The administrator-specified description of the KPI

3. Click the name of a KPI to view these additional details about that KPI:

   - The KPI value type, which is the type to which the KPI evaluates

   - The aggregation type of the KPI, which is the type of aggregation applied to the event attributes to arrive at the KPI value

   - The expression, which defines the KPI value

4. Click **View Target Data** to open the KPI Target Detail page.

   See "Viewing Key Performance Indicator Target Values" on page 2-31 for information about using the KPI Target Detail page.

Creating Key Performance Indicators

To create a KPI, you must first create one or more composite events, as described in "Modeling Composite Events and Composite Event Groups" on page 2-20, then to create a KPI, follow these steps:

1. In the Admin Console, click **Modeling.**

2. Click **KPIs** to open the Key Performance Indicators (KPI) page.

3. Click **Create** to open the Create Key Performance Indicator Wizard.

4. Follow the steps through the wizard. If you need help using a wizard page, then click the **Help** link on that page.

Modifying Key Performance Indicators

To modify a KPI:

1. In the Admin Console, click **Modeling.**

2. Click **KPIs** to open the KPI page.

3. Click the **pencil** icon corresponding to the KPI you want to modify. A confirmation page opens.

4. In the confirmation page, click **Yes** if you want to continue; click **No** if you do not. If you click **Yes,** then the Update Key Performance Indicator Wizard opens. If you click **No,** then the KPI page opens.

5. Follow the steps through the wizard. If you need help using a wizard page, then click the **Help** link on that page.

Deleting Key Performance Indicators

If you decide that a previously created KPI is no longer needed, then you can delete it.

To delete a KPI:

1. In the Admin Console, click **Modeling.**

2. Click **KPIs** to open the KPI page.

3. Click the **trash can** icon corresponding to the KPI that you want to delete.

4. In the confirmation page, click **Yes** if you want to continue; click **No** if you do not. If you click **Yes,** then the Update Key Performance Indicator Wizard opens. If you click **No,** then the KPI page opens.

Defining Key Performance Indicator Target Values

A **KPI target value** is the goal value for a KPI that you specify. Target values are sometimes referred to as **budget values**.

If you specify target values for a KPI, then you can configure the actual and target values into the same Dashboard chart. This allows a Dashboard user to see, at a glance, how aspects of the business are performing relative to goals.

In addition, specifying target values is a prerequisite, along with specifying KPI performance bands, for presenting data in a radar chart and cause/effect tables. See "Understanding Key Performance Indicator Performance Bands" on page 2-38 for more information.

To specify target values for a KPI:

1. In the Admin Console, click **Modeling.**

2. Click **KPIs** to open the Key Performance Indicators (KPI) page.

3. Click the name of a KPI for which you want to specify target values. The KPI Detail page opens.

4. Click **Prepare New Target Data.** The KPI Target Data Wizard opens.

5. Follow the steps through the KPI Target Data Wizard. If you need help using a wizard page, then click the **Help** link on that page.

6. After you complete the KPI Target Data Wizard, click **Process Prepared Data** to generate a SQL*Loader file from the target data you prepared. The KPI Target Data page opens to display a list of files for use with SQL*Loader files.

7. At the command line (outside of the Oracle BPEL Process Analytics Console), run SQL*Loader to load the target data into the BAM schema.

Suppose, for example, the KPI Target Data page appears as shown in Figure 2–8. If your current directory is `C:\OraHome3\integration\bam\config\data`, and the database in which the BAM schema is located is db1, then the SQL*Loader command to load the data is:

```
sqlldr control=target_loanFlowHDim.ctl, DATA=target_loanFlowHDim.dat
USERID=bam/bam@db1
```

SQL*Loader returns messages, such as the following:

```
SQL*Loader: Release 10.1.0.3.0 - Production on Wed Mar 23 09:31:12 2005

Copyright (c) 1982, 2004, Oracle.  All rights reserved.

Commit point reached - logical record count 64
Commit point reached - logical record count 128
Commit point reached - logical record count 192
Commit point reached - logical record count 256
Commit point reached - logical record count 320
Commit point reached - logical record count 384
Commit point reached - logical record count 448
Commit point reached - logical record count 512
Commit point reached - logical record count 576
Commit point reached - logical record count 640
Commit point reached - logical record count 704
Commit point reached - logical record count 744
```

See *Oracle Database Utilities* for more information about using SQL*Loader.

Figure 2–8   Sample KPI Target Data Page
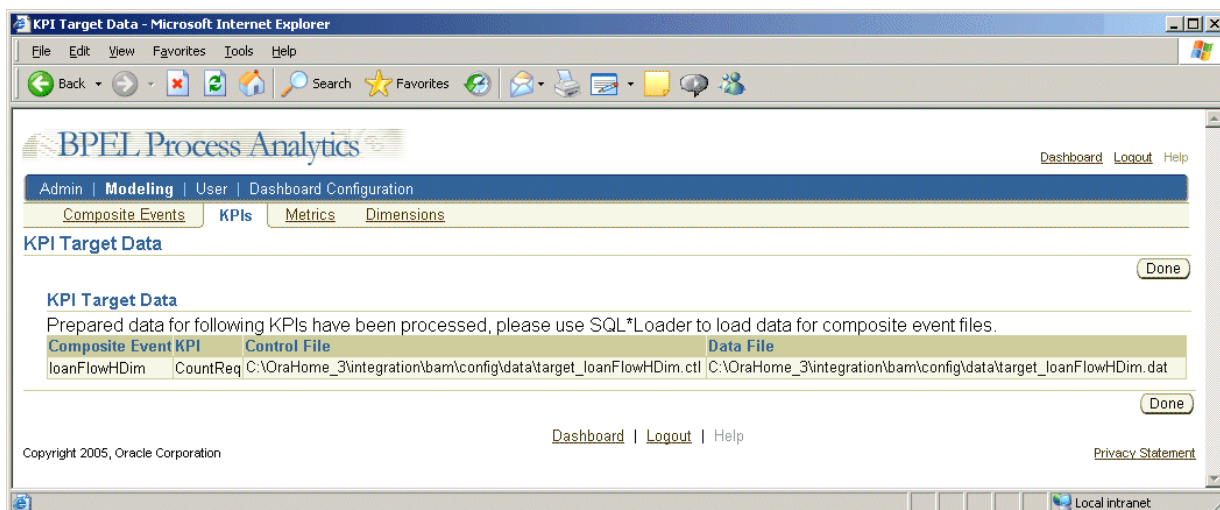
Viewing Key Performance Indicator Target Values

KPI target values are the goal values for a KPI that you specify, as described in "Defining Key Performance Indicator Target Values" on page 2-30. To view previously defined KPI target values:

1. In the Admin Console, click **Modeling.**

2. Click **KPIs** to open the Key Performance Indicators (KPI) page.

3. Click the name of the KPI for which you want to view target values. The KPI Detail page opens.

4. Click **View Target Data.** The KPI Target Data page opens.

5. If the amount of target data is large, then you can specify a subset of the target data you want to view, as described in the following list, and then click **View Target Data** again.

   ■ Time Grain – Select the time grain for which you want to view KPI target values. If you specify a time grain of Day, for example, then daily target values are displayed.

   ■ Time Period Start – Optionally, enter the start time for the time period using one of the following forms, depending on the time grain you specified:

      – yyyy, for example 2004, if the time grain is **Year**

      – yyyymm, for example 200401 for January 2004, if the time gain is **Month**

      – yyyymmdd, for example 20040121 for January 21, 2004, if the time grain is **Day**

      The values are inclusive, if specified.

   ■ Time Period End – Optionally, enter the end of the time period, using one of the forms as described for Time Period Start.

   ■ Dimension Columns – Select the dimension columns for which you want to view target values.

   ■ Dimension Value – Select the dimension values for which you want to view target values.

Modeling Dimensions

Dimensions are optional constructs you can build with the Admin Console so that composite event and KPI values can be presented in the Dashboard broken down by category. If a dimension is defined and mapped to one or more attributes in a composite event, then you can specify a value for that dimension when you specify the expression that defines a KPI. For example, suppose a KPI is defined to return the number of loans accepted in a day. That value alone is interesting. However, if you define a LoanType dimension, then the KPI can be presented in the Dashboard to show how many of that total were car loans and how many were home loans.

The modeling of KPIs and dimensions is based on the concept of a star join schema. A **star join schema**, commonly used for dimensional data warehouses, is composed of a fact table that is joined by primary keys to a number of dimension tables. A **fact table** typically contains fields that are additive and represent measurements whose values change each time they are taken. A **dimension table** typically contains a single primary key and, optionally, additional columns whose values are stable over time.

Consider Figure 2–9. In this figure, some of the event attributes from the loan procurement process flow presented in Figure 1–1 are presented in the form of a star join schema. The dimension tables present the stable data: loan service details, loan products, and time quantities. The fact table is in the center, which joins to each of the dimension tables by a primary key. The fact table includes fields that tend to vary with each measurement taken: number of loans offered, number of loans rejected, and loan amount.

Figure 2–9   Star Join Schema
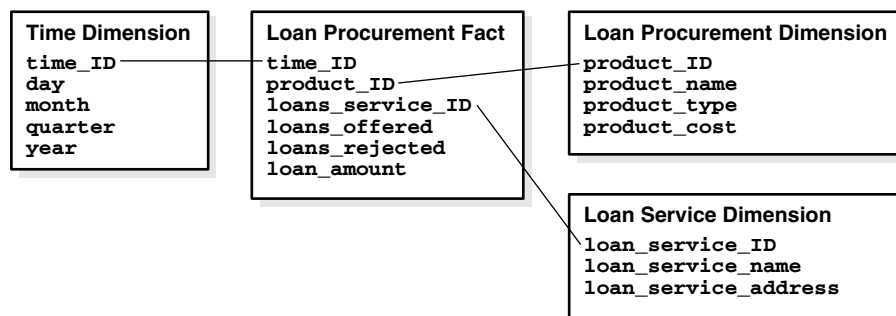
Composite events, KPIs, and dimensions correspond to the tables presented in the star join schema, as follows:

- A composite event corresponds to a fact table.

- A dimension corresponds to a (nonprimary key) column in a dimension table.

- A key performance indicator (KPI) corresponds to a column in the fact table aggregated over time.

Using the data tables presented in Figure 2–9, a composite event, LoanProcurement, is defined to include the time at which a loan is procured, the type of loan procured, and the service offering the loan, whether the loan was accepted or rejected and, if accepted, the amount of the loan. A KPI is defined to determine the total number of loans offered over a given time span. The loan service dimension allows the number of loans (as determined by the KPI) to be presented in the Dashboard according to which loan service offered the loan.

When you create dimensions, you can present Dashboard charts such as the charts shown in Figure 4–2.

The definition and application of a dimension is a multiple step process, as follows:

1. Create a dimension, as described in "Creating Dimensions" on page 2-34.

2. Specify dimension values, as described in "Specifying and Loading Dimension Values" on page 2-35.

3. Include the dimension in a composite event definition, as described in "Creating Composite Events" on page 2-23.

4. Create a KPI on the composite event defined in step 3, as described in "Creating Key Performance Indicators" on page 2-29.

5. Assign the KPI to a user, as described in "Creating and Managing Oracle BPEL Process Analytics Accounts" on page 2-36.

   When you assign a KPI to a user, you can specify a dimensional constraint for the KPIs. This enables the KPI to be returned along the specified dimension value. For example, if the KPI returns values for auto loans, you might specify a dimensional constraint of CarMake=Ford, if data only on Ford automobiles is of interest for the user to whom you are assigning the KPI.

Oracle BPEL Process Analytics provides pages and wizards that enable you to view currently defined dimensions, create new dimensions, and specify dimension values. See the following topics for step-by-step instructions:

- Viewing Dimensions on page 2-33

- Viewing Dimension Values on page 2-33

- Creating Dimensions on page 2-34

- Adding Dimension Values on page 2-34

Viewing Dimensions

To view dimensions that have been previously specified:

1. In the Admin Console, click **Modeling.**

2. Click **Dimensions.** The Dimensions page opens and displays all previously created dimension names.

3. If desired, click a dimension name to view these dimension properties:

   - Names of columns in the dimension

   - Primary key for the dimension

   - Data type of each column in the dimension

   - Hierarchy rank of each column in the dimension

     A **hierarchical dimension** is one that uses ordered levels as a means of organizing data. For example, in a time dimension, the hierarchy organizes data at the year level (hierarchy 1), the month level (hierarchy 2), or the day level (hierarchy 3).

Viewing Dimension Values

To view the values for a dimension:

1. In the Admin Console, click **Modeling.**

2. Click **Dimensions.** The Dimensions page opens.

3. Click the name of the dimension for which you want to view dimension values. The Properties page opens.

4. Click **View Data.** The Dimension Data page opens and displays the current dimension values.

Creating Dimensions

To specify a dimension:

1. In the Admin Console, click **Modeling** to open the Modeling page.

2. Click **Dimensions** to open the Dimensions page.

3. In the Dimensions page, click **Create** to open the Create Dimension Wizard.

4. Follow the steps through the Create Dimension Wizard. If you need help using a wizard page, then click the **Help** link on that page.

Note that once a dimension is created, it can neither be modified nor deleted.

Adding Dimension Values

Oracle BPEL Process Analytics provides two methods for adding dimension values. One method enables you to add dimension values one set at a time, the other method enables you to bulk load dimension values.

- To add dimension values one set at a time:

  1. In the Admin Console, click **Modeling.**

  2. Click **Dimensions.** The Dimensions page opens.

  3. Click the name of the dimension for which you want to add dimension values. The Properties page opens.

  4. Click **View Data.** The Dimension Data page opens and displays the current dimension values.

  5. Click **Add.** The Dimension Data page reopens and displays a table of columns and values.

  6. Enter the desired column values, then click **Submit.**

- To bulk load dimension values:

  1. In the Admin Console, click **Modeling.**

  2. Click **Dimensions** to open the Dimensions page.

  3. Click the name of a dimension for which you want to specify values. The Dimension Properties page opens.

  4. Click **Prepare New Data.** The Dimension Data page opens.

  5. Enter the new dimensional data, and then click **Submit.** An informational message is presented.

  6. Click **Done.**

  7. Click **Process Prepared Data** to generate a SQL*Loader file from the dimension data you prepared. The Dimension Data page opens to display a list of SQL *Loader files.

  8. At the command line (outside of the Oracle BPEL Process Analytics Console), run SQL*Loader to load the target data into the BAM schema.

     Suppose the following, for example:

- Dimension: CarMakeModel

- Control File: `C:\OraHome3\integration\bam\config\data\dim_CarMakeModel.ctl`

- Data File: `C:\OraHome3\integration\bam\config\data\dim_CarMakeModel.dat`

- The database in which the BAM schema is located is db1.

The SQL*Loader command to load the data is:

```
sqlldr CONTROL=dim_CarMakeModel.ctl, DATA=dim_CarMakeModel.dat
USERID=bam/bam@db1
```

Specifying and Loading Dimension Values

To specify and load values for a dimension:

See for an example of using SQL*Loader to load data into the BAM schema.

See *Oracle Database Utilities* for more information about using SQL*Loader.

Creating and Managing User Accounts

This section describes Oracle BPEL Process Analytics user accounts and how to create and manage them. It contains the following topics:

-

-

Understanding User Accounts and Console Access

Oracle BPEL Process Analytics uses Oracle Application Server Java Authentication and Authorization Service (JAAS) Provider in Oracle Application Server Containers for J2EE (OC4J) to restrict access to the Oracle BPEL Process Analytics Console based on the user who is logging in and what JAAS provider groups or roles were granted to that user.

Therefore, for individual users to access the Oracle BPEL Process Analytics Console, you:

1. Create an Oracle BPEL Process Analytics Console account for each *set* of users to whom you want to grant the same access in the Oracle BPEL Process Analytics Console.

2. Configure a JAAS Provider group or role (depending on whether you are using an XML or LDAP-based provider) for each of the Oracle BPEL Process Analytics user accounts created in step 1.

3. Create JAAS Provider users and grant them one of the JAAS provider groups or roles.

   An individual will log in to the Oracle BPEL Process Analytics Console using his or her JAAS Provider user name and password, not the Oracle BPEL Process Analytics Console user name and password.

By default, a nonprivileged account (where the user name and password are both `Manager`) is configured in the Oracle BPEL Process Analytics Console and as a JAAS

Provider role when Oracle BPEL Process Analytics is installed. To create additional accounts, you must follow the steps outlined in the preceding list.

For example, suppose you have two groups of users: one set you want to give full access to all of the KPIs, metrics, and composite event instances in the Dashboard; the other set you want to grant access to the KPI Dashboard multipane views only. The first set of users you call AnalystFull, and the second set you call AnalystLimited.

First, you create the Oracle BPEL Process Analytics user accounts AnalystFull and AnalystLimited, and specify their access to modeled data using the Oracle BPEL Process Analytics Console. Next, you set up access to these user accounts by defining a JAAS Provider group or role called AnalystFull and AnalystLimited. Last, you create JAAS Provider users, and grant them the desired group or role. An individual will log in to the Oracle BPEL Process Analytics Console using his or her JAAS Provider user name and password, not the Oracle BPEL Process Analytics Console user name and password.

> **Note:** This section contains information about how to set up the user accounts in the Oracle BPEL Process Analytics Console. The section "User Management for Oracle BPEL Process Analytics Console Access" on page 5-3 describes how to grant individuals access to these accounts by configuring them in JAAS Provider.

Creating and Managing Oracle BPEL Process Analytics Accounts

Oracle BPEL Process Analytics provides pages and wizards to assist you in creating and managing Oracle BPEL Process Analytics user accounts.

When you create an account, you not only specify a user name, password, and privileges, but you can also assign metrics, KPIs, and alerts to that user, as desired. By assigning these objects to the user, you make them accessible to that user through the Dashboard.

When you create a user, a Default KPI View and a Default Metric View are assigned to the account by default. However, you must assign the Real-Time Viewer to the account explicitly. See "Default Dashboard Pages" on page 3-8 for more information about the default pages.

This section includes information about, and instructions on, performing tasks associated with user accounts, as follows:

- Viewing User Accounts on page 2-37
- Creating User Accounts on page 2-37
- Modifying User Accounts on page 2-37
- Modifying a User's Account Password or Privileges on page 2-38
- Understanding Key Performance Indicator Performance Bands on page 2-38
- Modifying Key Performance Indicators Assigned to a User on page 2-38
- Modifying Metrics Assigned to a User on page 2-39
- Modifying Alerts Assigned to a User on page 2-39
- Modifying the Alert Delivery Channels Assigned to a User on page 2-40
- Deleting User Accounts on page 2-40

Viewing User Accounts

To view existing user accounts:

1. In the Admin Console, click **User** to open the User page.

2. Click the **pencil** icon corresponding to the desired user to view the user information and gain access to the other property pages you can view for the user:

   - Click **Properties** to view the privileges assigned to the user.

   - Click **KPIs** to view the KPIs assigned to the user.

   - Click **Metrics** to view the metrics assigned to the user.

   - Click **Alerts** to view the alerts created for the user.

   - Click **Delivery Channels** to view the methods by which alerts are currently designed to be delivered to the user.

3. Click **OK** when you are finished reviewing the properties for the selected user account.

Creating User Accounts

To create a user account:

1. In the Admin Console, click **User** to open the User page.

2. Click **Create User** to open the Create User Wizard.

3. Follow the steps through the wizard. If you need help using a wizard page, then click the **Help** link on that page.

Modifying User Accounts

To update an existing user account:

1. In the Admin Console, click **User.**

2. Click the pencil icon associated with the user account that you want to modify.

3. By clicking one of the following links, you can modify some of the attributes associated with the user:

   - Properties

     See "Modifying a User's Account Password or Privileges" on page 2-38 for information.

   - KPIs

     See "Modifying Key Performance Indicators Assigned to a User" on page 2-38 for information.

   - Metrics

     See "Modifying Metrics Assigned to a User" on page 2-39 for information.

   - Alerts

     See "Modifying Alerts Assigned to a User" on page 2-39 for information.

   - Alert Delivery Channels

     See "Modifying the Alert Delivery Channels Assigned to a User" on page 2-40 for information.

Modifying a User's Account Password or Privileges

To modify a user's account password or privileges:

1. In the Admin Console, click **User.**

2. Click the **pencil** icon corresponding to the user that you want to update. The Edit User: Properties page opens.

3. In the **New Password** and **Confirm Password** fields, enter the new password.

4. Select **Admin User** or **Normal User,** as desired to specify the privilege.

5. Click **Update.**

A user for whom you have modified the account password will not be able to log in using this password until you change the JAAS Provider password. See "User Management for Oracle BPEL Process Analytics Console Access" on page 5-3 for more information.

Understanding Key Performance Indicator Performance Bands

By default, when you create a KPI, its performance bands are defined, using the NormalBenefitsBand template, which is defined as follows:

■ Weak performer: 50% or more below the target value (exclusive)

■ Under performer: Between 50% to 20% below the target value (inclusive)

■ Normal: Within 20% of the target value (exclusive)

■ Over performer: Between 20% and 50% above the target value (inclusive)

■ Strong performer: 50% or more above the target value (exclusive)

Oracle BPEL Process Analytics provides templates for six performance bands, all of which you can modify, if desired.

You specify custom KPI performance bands when you create or modify a user account, rather than when the KPI is modeled, because different users may have different definitions of what constitutes a weak, under, normal, over, and strong performer for the same KPI.

Defining target values and KPI performance bands enables you to configure radar charts, gauges, and cause/effect tables in the Dashboard. See Figure 1–5 for an example of a radar chart. See Figure 4–22 for an example of a cause/effect table.

Modifying Key Performance Indicators Assigned to a User

To modify the KPIs assigned to a user:

1. In the Admin Console, click **User.**

2. Click the **pencil** icon corresponding to the user that you want to update. The Properties page opens.

3. Click **KPIs** to open the KPIs page.

4. On the KPIs page, you can either add a KPI to the user, or remove an existing KPI assigned to the user, as follows:

    ■ To add a KPI to the user:

    Click **Add KPI** to open the Add KPI for User Wizard, then follow the steps through the wizard. If you need help using a wizard page, then click the **Help** link on that page.

- To remove a KPI from the user:

  Click the **trash can** icon corresponding to the KPI that you want to remove, and then review the other objects that will be deleted if you delete this KPI. Click **Yes** in the confirmation page to commit the delete operation; click **No** to cancel it.

Modifying Metrics Assigned to a User

To modify the metrics assigned to a user:

1. In the Admin Console, click **User.**

2. Click the **pencil** icon corresponding to the user that you want to update. The Properties page opens.

3. Click **Metrics** to open the Edit User: Metrics page.

4. On the Metrics page, you can either add a metric to the user, or delete a metric currently assigned to the user, as described in the following list. You cannot modify the definition of a metric assigned to the user.

   - To add a metric to the user:

     Click **Add Metrics** to open the Metrics page, select the metrics that you want to associate with the user, click **Apply,** then click **OK.**

   - To remove a metric from the user:

     Click the **trash can** icon corresponding to with the metric that you want to remove, and then review the other objects that will be removed from the user if you remove this metric. Click **Yes** in the confirmation page to commit the operation; click **No** to cancel it.

Modifying Alerts Assigned to a User

To modify the alerts assigned to a user:

1. In the Admin Console, click **User.**

2. Click the **pencil** icon corresponding to the user that you want to update. The Properties page opens.

3. Click **Alerts** to open the Alerts page.

4. On the Alerts page, you can add an alert to the user, modify an existing alert assigned to the user, or remove an alert currently assigned to the user, as follows:

   - To add an alert to the user:

     Click **Add Alert,** and then follow the steps through the wizard. If you need help using a wizard page, then click the **Help** link on that page.

   - To modify an alert assigned to a user:

     Click the **pencil** icon corresponding to the alert you want to modify, and then follow the steps through the wizard. If you need help using a wizard page, then click the **Help** link on that page.

   - To remove an alert assigned to a user:

     Click the **trash can** icon corresponding to the alert that you want to remove. A confirmation page opens. Click **Yes** in the confirmation page to commit the remove operation; click **No** to cancel it.

Modifying the Alert Delivery Channels Assigned to a User

To modify the alert delivery channels assigned to a user:

1. In the Admin Console, click **User.**

2. Click the **pencil** icon corresponding to the user that you want to update. The Properties page opens.

3. Click **Delivery Channels** to open the Delivery Channels page.

4. Click the **pencil** icon corresponding to the delivery channel that you want to modify to open the Update Delivery Channel page.

5. Update the **Description** field, if desired.

6. Update the delivery channel parameters, if desired.

7. Click **Submit.**

There is no strict format for specifying delivery channels that include a number, such as a phone number. You can use any of the following characters:

- Integers between 0 and 9, inclusive

- Space

- Dash (-)

- Parentheses ( )

- Plus sign (+)

Deleting User Accounts

If a user account is no longer needed, you can delete it.

To delete a user account:

1. In the Admin Console, click **User.**

2. Click the **trash can** icon that corresponds to with the user account that you want to delete.

3. Review the additional objects, if any that will be deleted if you delete this user account.

4. Click **Yes** to confirm the delete operation, click **No** to cancel it.

5. Click **Submit.**

3

Using the Admin Console to Configure the Dashboard

The Oracle BPEL Process Analytics administrator configures the Dashboard using the Dashboard Configuration Console section of the Admin Console. The Dashboard Configuration Console allows the administrator to configure page layout and the content of Dashboard pages. If you are reading this chapter, it is assumed that you are an Oracle BPEL Process Analytics administrator.

This chapter includes the following topics:

- Dashboard Configuration Concepts on page 3-1
- View Pane Types on page 3-4
- Default Dashboard Pages on page 3-8
- Designing and Assigning Dashboard Pages to a User on page 3-9
- Refining Custom and Default Dashboard Pages on page 3-13

Dashboard Configuration Concepts

The Dashboard consists of a number pages that present modeled data. Each page presents one type of modeled data (composite events, KPIs, metrics, or alerts) in a variety of tables and charts.

Figure 3–1 shows a sample Dashboard page containing a table of alerts and two KPI charts. Metrics are presented in similar pages. A Dashboard page that presents KPIs or metrics can contain up to two columns of view panes and an unlimited number of rows. A **view pane** is the part of the page in which a table or chart is presented. Figure 3–1, for example, shows three view panes: Alert View, Loan Requests by Car Make Model Today, and Avg Approval time by Car Make Model Today.

Figure 3–2 shows a sample Dashboard page containing the Real-Time Viewer. The Real-Time Viewer provides a chart that displays composite event instances, in real-time and a table that summarizes the state of the composite events. (Complete information on reading the data presented in the Real-Time Viewer is presented in "Viewing Events and Composite Event Instances" on page 4-6.)

Figure 3–1   Sample Dashboard View
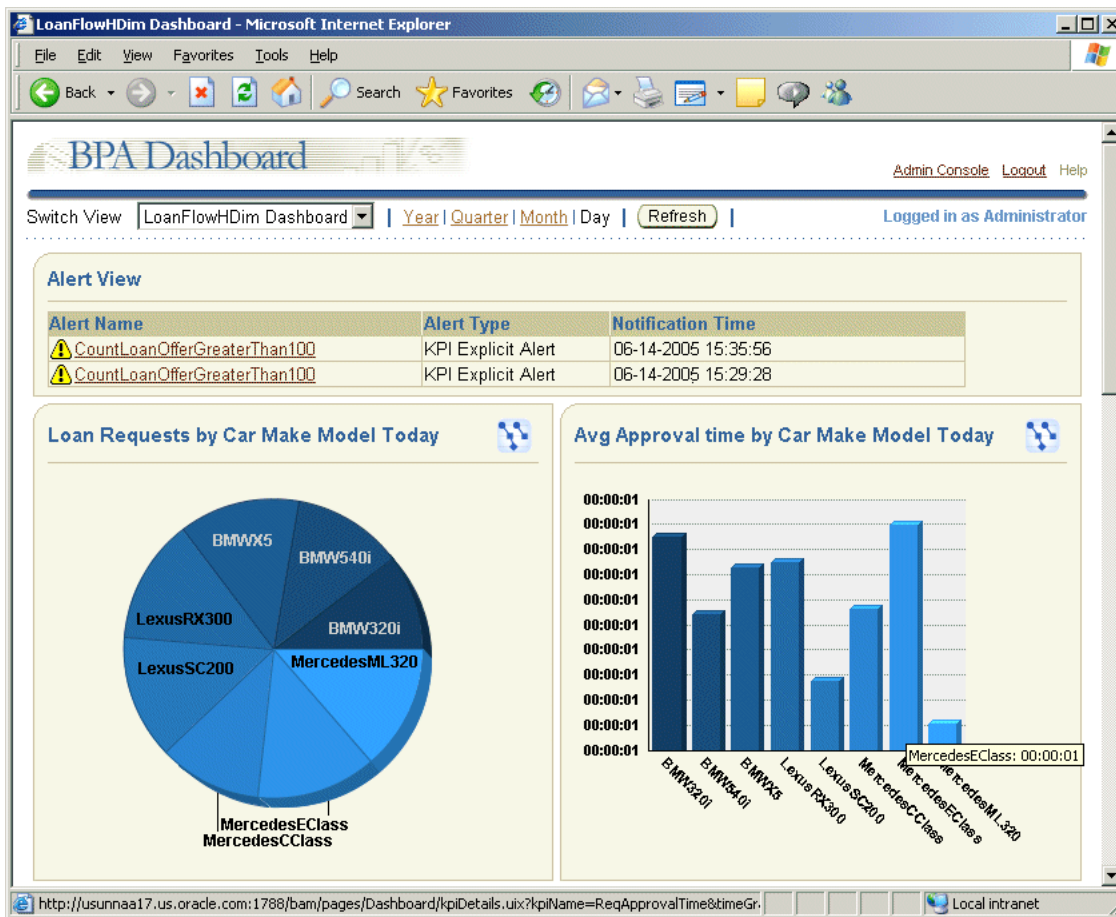
Figure 3–2   Sample Real-Time Viewer
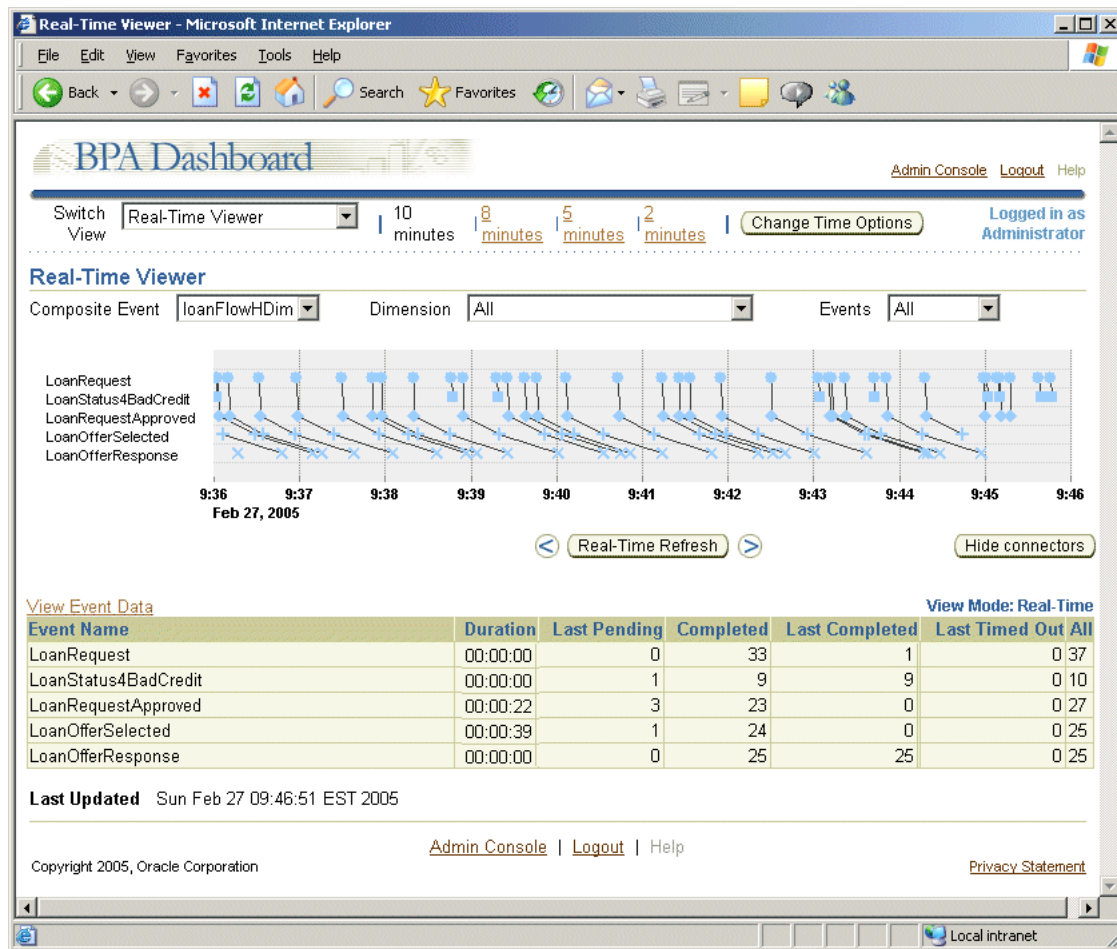
When you configure the Dashboard for a user, you can create custom pages, for which you specify the contents and layout, use the default pages provided by Oracle BPEL Process Analytics, or both.

The default pages, described in detail in "Default Dashboard Pages" on page 3-8 are:

- Real-Time Viewer

- Default KPI View

- Default Metric View

These default pages are intended to make it easy for you to get started with Dashboard configuration.

You might want to begin Dashboard configuration by using the default pages, and then create customized pages as you get more familiar with the data being returned and the data desired by Dashboard users.

After you have created customized Dashboard pages, or have decided to use the default pages, you assign the Dashboard pages to user accounts. A user will not be able to view a Dashboard page until it is assigned to that user's account. See "Assigning Default and Customized Dashboard Pages to a User Account" on page 3-12 for more information.

View Pane Types

Oracle BPEL Process Analytics provides four view pane types for displaying data in Dashboard pages, as described in the following topics:

- Key Performance Indicator View Panes on page 3-4
- Metric View Panes on page 3-6
- Alert View Panes on page 3-7
- Pending Activities View Panes on page 3-7

Key Performance Indicator View Panes

As shown in Figure 3–3, Oracle BPEL Process Analytics provides the following types of view panes for displaying KPI data in the dashboard:

- Snapshot

  This view type includes multiple KPIs over a single time period or time interval presented in a pie chart, vertical bar chart, horizontal bar chart, radar chart, gauge chart, or a table.

- Multi-snapshot

  This view type includes multiple KPIs over two or three time periods presented in a vertical bar chart, horizontal bar chart, radar chart, gauge chart, or a table.

- KPI Series

  This view type includes one KPI data series over time presented in a bar chart (clustered or stacked), line chart, or area chart.

- Multi-KPI Series

  This view type includes multiple KPI data series over time presented in a bar chart (clustered or stacked), line chart, or area chart.

- Actual/Target

  This view type includes a single KPI over time with both actual and target values, presented in a combination bar-line chart or a table of actual and target values. In the combination bar-line chart, actual data is presented as bars, and target data is presented as points connected by a line.

- Cause/Effect

  This view type lists cause and effect KPIs for the specified KPI in a table.

Figure 3–3   KPI View Types

Metric View Panes

As shown in Figure 3–4, Oracle BPEL Process Analytics provides the following types of view panes for displaying metric data in the Dashboard:

- Bar

    This view type includes multiple metrics over a single time period or time interval, presented in a clustered or stacked bar chart.

- Line

    This view type includes multiple metrics over a single time period or time interval, presented in a line chart.

- Area

    This view type includes multiple metrics over a single time period or time interval, presented in an area chart.

- Table

    This view type includes multiple metrics over a single time period or time interval, presented in a table.

- Point

    This view type includes multiple metrics over a single time period or a time interval, presented as points in a chart.

Figure 3–4   Metric View Types
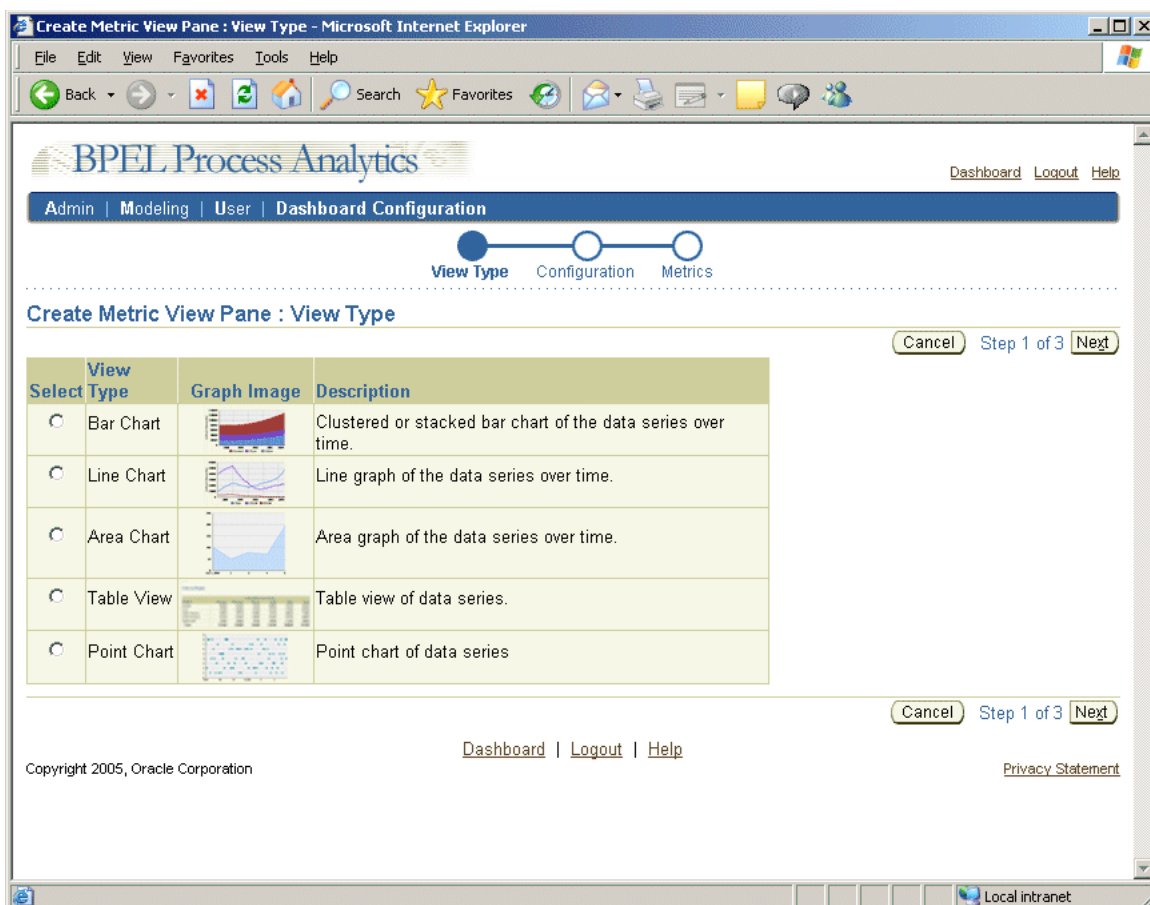
Alert View Panes

Alert view panes display KPI and Metric alerts in a table, which provides each alert name, alert type, and the time at which each alert was sent, as shown in Figure 4–28. Alert View Panes are configured by default. As the administrator, you can specify whether or not to include the Alert View pane in a Dashboard page, but you cannot control which alerts are included in the display.

Pending Activities View Panes

A **pending** composite event is one for which an end event has not yet been received and that has not timed out. In Figure 3–2, the last six composite events displayed in the chart are pending composite events (assuming they have not timed out.)

A pending activities view pane shows the following statistics on composite event instances that are pending:

- Last pending pie chart

  A pie chart is presented to show what percentage of each event instance contained in a specified composite event is considered to be the last pending. An event in a composite event instance is considered to be the **last pending** if it is the most recently received event within the composite event instance and the composite event has not timed out. For example, the LoanFlow composite event contains these events, which occur in the sequence shown, where LoanOfferResponse is the end event:

  1. LoanRequest

  2. LoanRequestApproved

  3. LoanOfferSelected

  4. LoanOfferResponse

  Assume that for the current time period, 10 composite event instances are pending, with the totals for the most recently received events (last pending) as follows:

  – LoanRequest - 5

  – LoanRequestApproved - 3

  – LoanOfferSelected - 2

  In other words, for the 10 incomplete composite event instances, five have just completed the LoanRequest event; three have completed the LoanRequest event and the LoanRequestApproved event; and two have completed the LoanRequest event, the LoanRequestApproved event, and the LoanOfferSelected event. The LoanOfferResponse event is not considered, because it is an end event and receipt of that event indicates that the composite event is no longer pending.

  Thus, the pie chart for this example would consist of three wedges. The LoanRequest event would be represented by 50% of the pie; the LoanRequestApproved event would be represented by 30% of the pie; and the LoanOfferSelected event would be represented by 20% of the pie.

- Duration bar chart

  A bar chart is presented to show the duration for each event in a pending composite event instance. (This corresponds to the Duration column in the table presented in Figure 3–2.) The **duration** for an event is defined as the average time between the specified event and the previous event in the sequence.

For example, assume the sequence shown for the last pending example, and assume the bar chart shows the following values:

– LoanRequest - 00:00:00

– LoanRequestApproved - 00:01:31

– LoanOfferSelected - 00:00:53

– LoanOfferResponse - 00:00:30

These values indicate that, on average, the time between a LoanRequest event occurring and a LoanRequestApproved event occurring is 1 minute, 31 seconds; the average time between a LoanRequestApproved event occurring and a LoanOfferSelected event occurring is 53 seconds; and the average time between a LoanOfferSelected event occurring and a LoanOfferResponse event occurring is 30 seconds.

Note: The duration bar chart is only presented when the layout of the Pending Activities view pane is specified as full-width. When the layout is specified as half-width, only the last pending pie chart is displayed. See "Specifying the Page Layout for Custom Dashboard Pages" on page 3-12 for more information on view pane widths.

Figure 3–5 provides an example of a full-width Pending Activities view pane.

Figure 3–5   Pending Activities View Pane
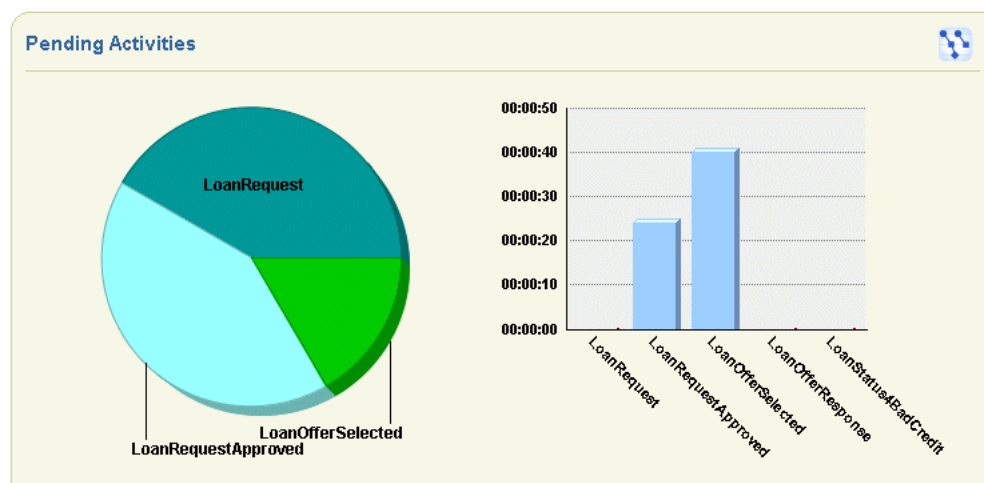
Default Dashboard Pages

By default, Oracle BPEL Process Analytics provides the following preconfigured pages:

■ Real-Time Viewer

This page contains a chart and table of composite event instances, as shown in Figure 3–2. All adjustments to the Real-Time Viewer are made by the Dashboard user when viewing the Real-Time Viewer page. Adjustments that the Dashboard user can make include the following:

– The composite event or group to display

- The dimension, if any, that should be used to separate the presentation of events

- The events to display, based on their current state

- The window of time over which events are to be displayed

- Whether or not lines connecting composite events should be included in the chart

> **Note:** If given access to the Real-Time Viewer, a Dashboard user can view all composite events underlying the modeled KPIs and metrics. If granting this level of access to a given Dashboard user is considered a security risk, you should not grant that Dashboard user access to the Real-Time Viewer.

- Default KPI View

This page displays all of the KPIs in various view panes. If possible, one pane of each view pane chart is displayed. In addition, one view pane is a multi-KPI chart (if more than one KPI exists that share a dimension), and another view pane is a full-width table containing multiple KPIs for a single composite event.

Adjustments that the Dashboard user can make to a Default KPI View differ depending on the view pane type, but in general, adjustments involve choosing KPIs and dimension columns for the view pane. Each view pane in the KPI view allows navigation to the KPI detail page. Charts that display KPIs across different dimension values provide links to view a lower level of the hierarchy (if the displayed dimension column is not the lowest level of the dimension hierarchy). Dimensional drill-downs are displayed in the KPI detail page.

- Default Metric View

This page contains one view pane for each event type on which a metric is defined. All metrics, except those that are of type TEXT, are graphed. A table presented as a full-width view pane includes all metrics.

Metrics are displayed in chronological order. Dashboard users can adjust a metric view pane by specifying the number of metrics to display in the chart or table, or by specifying a begin and end date for the time span to include.

Designing and Assigning Dashboard Pages to a User

Minimally, you must assign one or more default Dashboard pages to an account for a Dashboard user with access to that account to view those pages. See "Assigning Default and Customized Dashboard Pages to a User Account" on page 3-12 for more information.

If you want to design custom Dashboard pages for an account, follow these steps:

1. Design customized view panes and dashboard pages for the user, as described in "Designing Customized View Panes and Dashboard Pages" on page 3-10.

2. Assign the customized Dashboard pages to a user account, as described in "Assigning Default and Customized Dashboard Pages to a User Account" on page 3-12.

See "Refining Custom and Default Dashboard Pages" on page 3-13 for addition options that Oracle BPEL Process Analytics offers for configuring Dashboard pages.

Designing Customized View Panes and Dashboard Pages

To design customized dashboard pages, you must first specify each of the panes that will be presented in those pages. The following topics provided step-by-step instructions:

Designing Customized Key Performance Indicator View Panes

Designing KPI view panes for a user involves selecting the type of chart or table in which you want data presented, specifying whether you want the KPI view pane presented in a half-width view pane or a full-width view pane, selecting the KPIs you want to display in the table or chart, and selecting which dimensions (if any) you want to include in the table or chart. (See "Specifying the Page Layout for Custom Dashboard Pages" on page 3-12 for information about specifying half-width and full-width view panes.)

Before you an design KPI view panes, you must have created the user and assigned KPIs to that user. See "Creating and Managing User Accounts" on page 2-35 for more information.

To design a KPI view pane for a user:

1.   In the Admin Console, click **Dashboard Configuration.**

2.   In the **Users** table, click the **pencil** icon corresponding to the user for whom you want to configure a KPI view pane.

3.   Click the **Create** button associated with the KPI View Panes table to open the Create View Pane Wizard.

4.   Follow the steps through the wizard. If you need help using a wizard page, click the **Help** link on that page.

Designing Customized Metric View Panes

Designing Metric view panes for a user involves selecting the type of chart or table in which you want metrics presented; specifying a name for the view pane, a name for the chart or table that will be presented in the view pane, specifying whether you want the Metric view pane presented in a half-width view pane or a full-width view pane, and selecting the metrics you want to display in the table or chart. (See "Specifying the Page Layout for Custom Dashboard Pages" on page 3-12 for information about specifying half-width and full-width view panes.)

Before you can design Metric view panes, you must have created the user and assigned metrics to that user. See "Creating and Managing User Accounts" on page 2-35 for more information.

To design a Metric view pane for a user:

1.   In the Admin Console, click **Dashboard Configuration.**

2.   In the **Users** table, click the **pencil** icon corresponding to the user for whom you want to configure the Metric view pane.

3. Click the **Create** button associated with the Metric View Panes table. The Create Metric View Pane Wizard opens.

4. Follow the steps through the wizard. If you need help using a wizard page, click the **Help** link on that page.

Specifying Pending Activities View Panes

Specifying Pending Activities view panes for a user involves specifying a name for the view pane and a name for the chart that will be presented in the view pane, specifying whether you want the view pane presented in a half-width view or a full-width view, and selecting the composite event you want to display in the chart (See "Specifying the Page Layout for Custom Dashboard Pages" on page 3-12 for information about specifying half-width and full-width view panes.)

See "Pending Activities View Panes" on page 3-7 for a description of the values presented in a Pending Activities view pane.

To design a Pending Activities view pane for a user:

1. In the Admin Console, click **Dashboard Configuration.**

2. In the **Users** table, click the **pencil** icon corresponding to the user for whom you want to configure the Pending Activities view pane.

3. Click the **Create** button associated with the Pending Activities view panes table. The Create Pending Activities View page opens.

4. In the **Enter View Pane Name** field, enter a unique name for view pane.

5. In the **Enter Chart Title** field, enter a title for the chart.

6. From the **Choose Chart Width** box, select a width for the view pane.

7. From the **Select Composite Event** box, select the composite event for which you want to display statistics in the chart.

8. Click **Apply.** The Dashboard Configuration Console page opens.

Specifying Custom Dashboard Pages

Once view panes are defined, you can specify how you want them presented on Dashboard pages. Typically, several view panes are presented on a Dashboard page, as described in "Specifying the Page Layout for Custom Dashboard Pages" on page 3-12.

When you specify the Dashboard page layout, you can indicate whether or not the Alert View table and the Pending Activities view pane will be included in the Dashboard page. The Alert View table is presented as a full-width table by default.

To design a Dashboard page:

1. In the Admin Console, click **Dashboard Configuration.**

2. In the **Users** table, click the **pencil** icon corresponding to the user for whom you want to configure the Dashboard page.

3. Click the **Create** button associated with the Dashboard Pages table to open the Create Dashboard Page Wizard.

4. Follow the steps through the wizard. If you need help using a wizard page, click the **Help** link on that page.

Specifying the Page Layout for Custom Dashboard Pages

When you configure custom pages, you can set the view panes within it as either half width or full width. Half-width view panes are presented in two columns; full-width view panes span two columns. For example, you can construct a view to contain two rows of half-width panes, and one row with a full-width pane, as shown in Figure 3–6.

Figure 3–6   Sample Page Layout
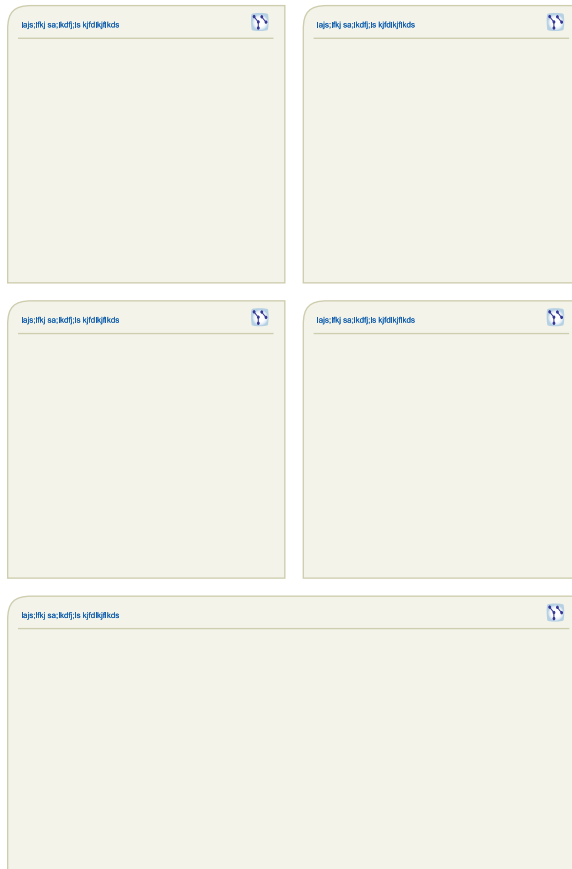
Assigning Default and Customized Dashboard Pages to a User Account

Dashboard pages must be assigned to a user account before they are visible to users who have been granted access to that account. Each Dashboard page can be assigned to one and only one Oracle BPEL Process Analytics user. However, you can specify a JAAS provider to specify users and classes of users to which Dashboard access is granted, so that you do not have to create Dashboard pages for each Dashboard user. See "User Management for Oracle BPEL Process Analytics Console Access" on page 5-3 for more information.

To assign Dashboard pages:

1. In the Admin Console, click **Dashboard Configuration.**

2. In the **Users** table, click the **pencil** icon corresponding to the user for whom you want to assign Dashboard views.

3. Click **Assign/Unassign.**

4. Move the Dashboard views from the **Available** box to the **Assigned** box, to specify the pages you want to assign to the user.

5. Click **Apply.**

> **Note:** If given access to the Real-Time Viewer, a Dashboard user can
> view all composite events underlying the modeled KPIs and metrics.
> If granting this level of access to a given Dashboard user is considered
> a security risk, you should not grant that Dashboard user access to the
> Real-Time Viewer.

Refining Custom and Default Dashboard Pages

You can make refinements to both custom and default dashboard pages to specify
colors used for presenting dimensions in the Dashboard, the Dashboard screen size,
and the Dashboard brand icon. Step-by-step instructions on making these refinements
are described in the following topics:

- Editing Dimension Colors on page 3-13
- Setting the Dashboard Screen Size on page 3-13
- Assigning a Dashboard Brand Icon on page 3-17
- Restoring a Dashboard Brand Icon on page 3-17

Setting the Dashboard Screen Size

The Dashboard Configuration Console offers a control to specify the Dashboard screen
size resolution, so that view panes can be optimized for presentation in that screen size
resolution. The Dashboard screen size resolution is set for all Dashboard users, not on
a per-user basis. Therefore, specify a screen size resolution setting that will work for all
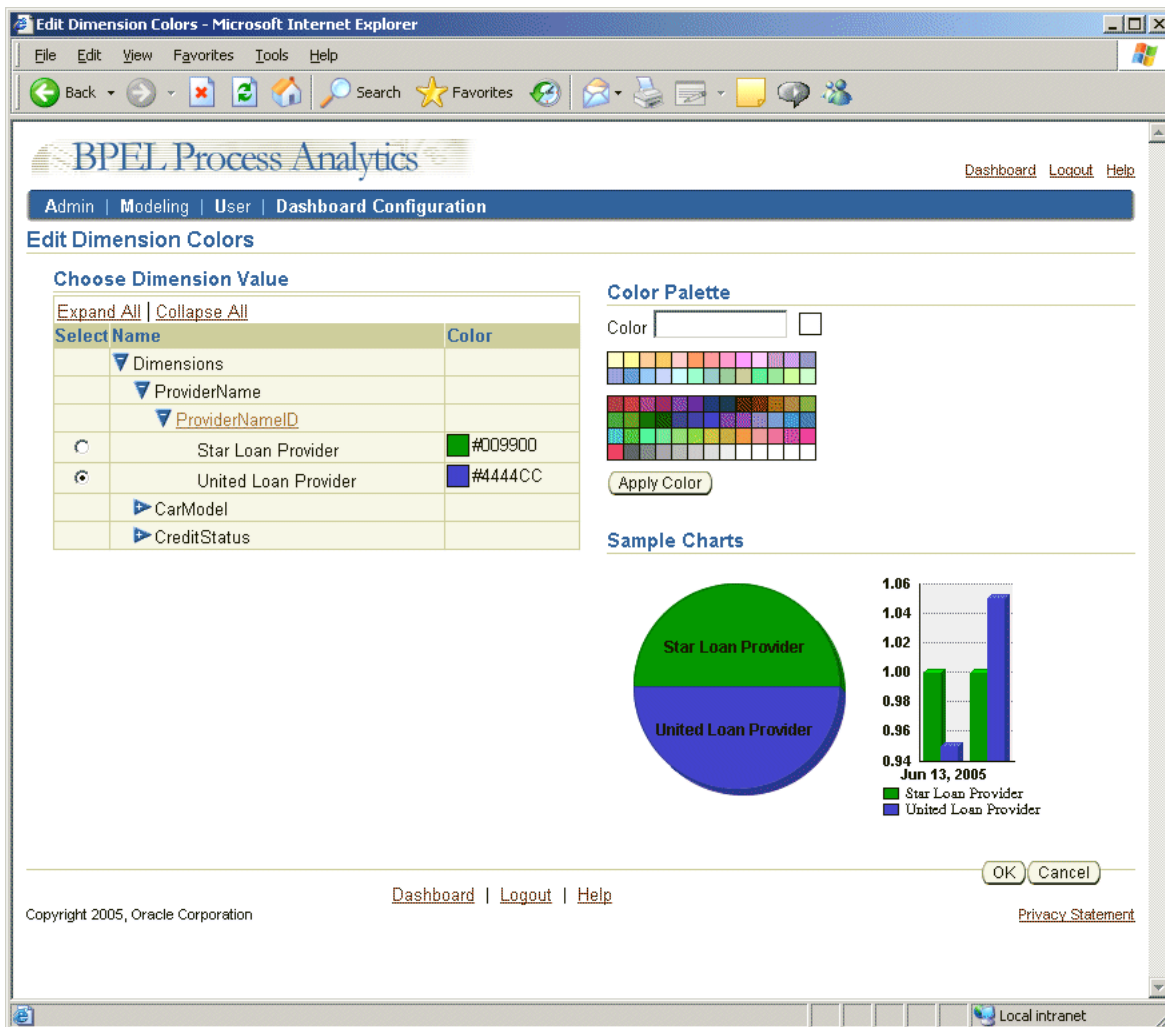your Dashboard users.

To set the Dashboard screen size resolution:

1. In the Admin Console, click **Dashboard Configuration** to open the Dashboard
    Configuration Console.

2. Click **Change Target Screen Size.**

3. Select a screen size resolution that best represents the screen size that your
    Dashboard users have. The choices are:

    - 800 x 600 pixels
    - 1024 x 768 pixels
    - 1280 x 1024 pixels

4. Optionally, scroll down the see how the charts will look using the various
    resolutions.

5. Click **Set Target Screen Size**.

Editing Dimension Colors

By default, when a dimension is created, Oracle BPEL Process Analytics assigns a color
to the dimension, and represents each value for that dimension with a different shade
of the assigned color.

> **Note:** If a new value for a dimension is returned after the dimension has been defined and colors have been specified for its values (either explicitly by you, or by default), that value will have a light blue color assigned to it automatically.
>
> For instance, assume you have defined a CarModel dimension and have loaded these dimension values: Toyota, Saturn, Nissan, and Ford. The dimension color map is initialized with this data, and either you or Oracle BPEL Process Analytics specify colors for those four car models. Now, if a new car model, Honda, is returned in an event, there is no color assigned to it, so Oracle BPEL Process Analytics uses a default color of light blue to represent Honda in charts that display the CarModel dimension.
>
> In such a situation, you can use the steps described in "Mapping Dimension Color Values" on page 3-16 to set the color for the new dimension, as desired.

If desired, you can override the default and specify the colors you want the Dashboard to use for each dimension that it presents in a chart in a view pane. (Colors used for KPI detail pages are determined by Oracle BPEL Process Analytics and cannot be altered.) You can perform this step before or after you assign Dashboard pages to a user.

Before you change the default, or currently assigned color values, you can preview them in a sample pie chart and bar graph.

The following topics describe how to perform these tasks:

- Previewing Dimension Color Values on page 3-14
- Mapping Dimension Color Values on page 3-16

Previewing Dimension Color Values

The following list provides the steps for previewing dimension color values:

1. In the Admin Console, click **Dashboard Configuration** to open the Dashboard Configuration Console.

2. Click **Edit Dimension Colors** to open the Edit Dimension Colors page. Note that the Sample Charts contain no data, as shown in Figure 3–7.

3. Click **Expand All.** The dimension columns are displayed.

4. Click the name of the column for which you want to preview dimension color values. The Sample Charts display the current colors assigned to that column.

Figure 3–8 provides an example of how the Sample Charts appear when the CarMakeModelID column in Figure 3–7 is clicked.

Figure 3–7   Edit Dimension Colors Page - No Data
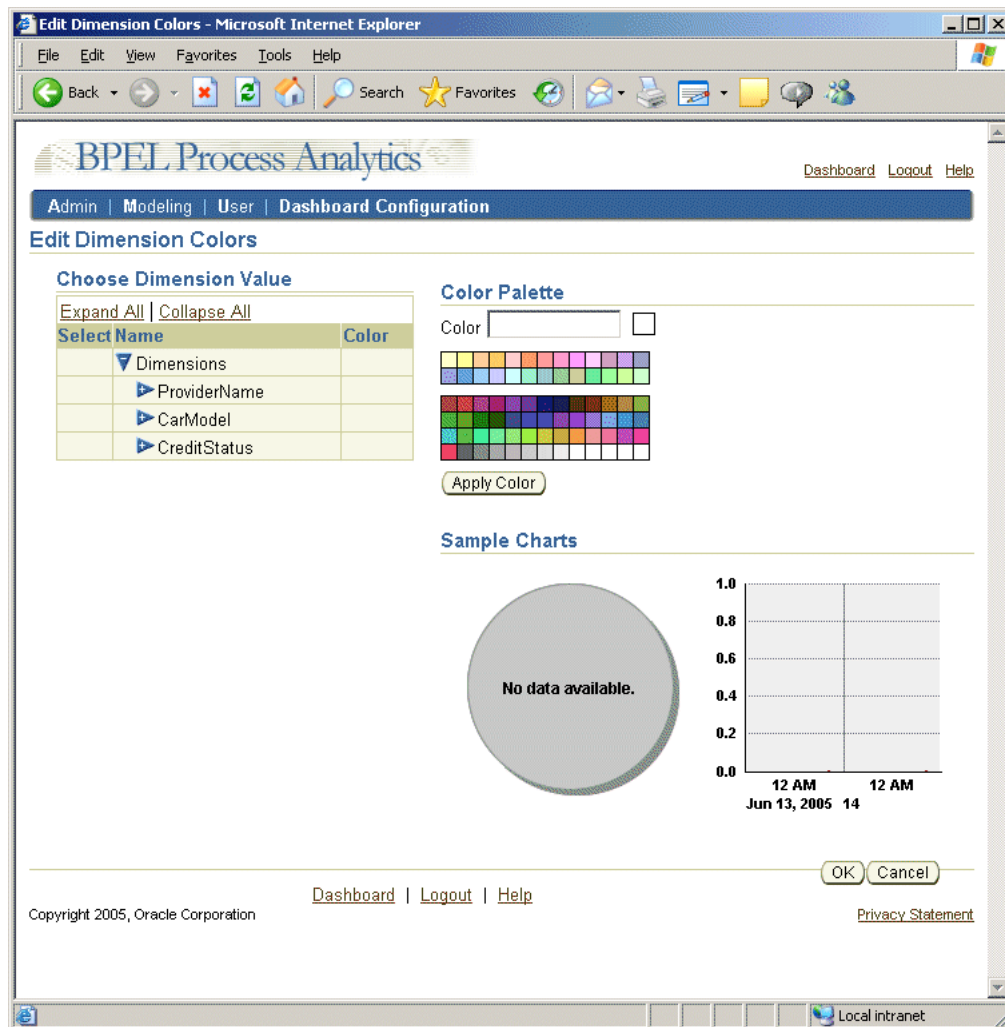
Figure 3–8   Edit Dimension Colors Page - Data

Mapping Dimension Color Values

The following list provides the steps for mapping dimension color values:

1. In the Admin Console, click **Dashboard Configuration** to open the Dashboard Configuration Console.

2. Click **Edit Dimension Colors** to open the Edit Dimension Colors page.

3. Expand the navigation tree to display the dimension for which you want to set color values.

4. Select a dimension value from the navigator tree, and then either click a color in the **Color Palette,** or enter the hexidecimal RGB value for the color in the **Color** box.

5. Click **Apply Color.**

    When you click **Apply Color,** the page refreshes and displays the color you selected in a pie chart and a bar chart to give you an opportunity to see how the selected color appears beside other dimension colors.

6. Repeat steps 4 and 5 for each dimension value in the dimension, or until the colors for each dimension suit your needs.

7. Click **OK.** The Dashboard Configuration Console page opens.

Assigning a Dashboard Brand Icon

By default, the brand icon that appears at the top of each Dashboard page is BPA Dashboard. If desired, you can change the brand icon to an icon of your choice.

To assign a Dashboard brand icon:

1. In the Admin Console, click **Dashboard Configuration.**

2. Click **Change Brand Icon.** The Dashboard Configuration: Brand Icon page opens.

3. In the **File Name** field, specify the path and file name of the image that you want to replace the default icon.

4. Click **Apply.**

Restoring a Dashboard Brand Icon

If you have previously replaced the default Dashboard Brand Icon (which is a graphic with the text BPA Dashboard), and you want to restore the default icon, follow these steps:

1. In the Admin Console, click **Dashboard Configuration.**

2. Click **Change Brand Icon.** The Dashboard Configuration: Brand Icon page opens.

3. Click **Restore Default Brand Icon.**

4. Click **Apply.**

4

Using the Dashboard

After an administrator configures the Dashboard and assigns pages to Oracle BPEL Process Analytics user accounts, an end user, such as a business analyst, can use the Oracle BPEL Process Analytics Dashboard to view and analyze the metrics, KPIs, and events to which the administrator has given that user access.

Because the Oracle BPEL Process Analytics administrator can customize the Dashboard for each user, this chapter cannot provide step-by-step instructions for using the Dashboard. Instead, it provides details about using each of the components the administrator can make available through the Dashboard. Your particular Dashboard configuration may or may not make all components available to you. If you find a desired page, chart, or table described in this chapter that you do not have in your Dashboard, discuss it with your Oracle BPEL Process Analytics administrator.

This chapter includes the following topics:

- Logging In to the Dashboard on page 4-1
- Introduction to the Dashboard on page 4-2
- Viewing Events and Composite Event Instances on page 4-6
- Viewing Metrics and KPIs in Multiple Pane Views on page 4-16

Logging In to the Dashboard

The method for logging in to the Dashboard varies, depending on whether you are using a Microsoft Windows or Unix system, as follows:

- On Microsoft Windows Systems:

  1. On the desktop, click **Start,** then choose **All Programs,** then *Oracle – OAS_Home_Name,* then **Oracle BPEL Process Analytics,** and then click **BPA Console.** The default Web browser opens and presents the Login to Oracle BPEL Process Analytics page.

  2. Enter the username and password given to you by your Oracle BPEL Process Analytics administrator, and then click **Login.**

  3. When you are done using the Dashboard, click **Logout** (in the upper-right side of the browser window).

- On Unix Systems:

  1. Open a supported Web browser and enter the following URL, where *host* is the host system where Oracle BPEL Process Analytics is running and *port* is the port on which it is running:

     ```
     http://host:port/bam/pages/login.uix
     ```

Ask your Oracle BPEL Process Analytics administrator for the *host* and `port` values (which were presented to him or her at the end of the Oracle BPEL Process Analytics installation).

2. Enter the username and password given to you by your Oracle BPEL Process Analytics administrator, and then click **Login.**

3. When you are done using the Dashboard, click Logout (in the upper-right side of the browser window).

After you log in, the interface opens and presents the Dashboard, unless the account you are using has Oracle BPEL Process Analytics administrator privileges. If the account has such administrator privileges, the Admin Console opens; click **Dashboard** to view the Dashboard.

> **Note:** Be sure to close your Web browser after you log out of the Oracle BPEL Process Analytics Console, to ensure that others cannot view Console pages using the Web browser's Back button.

Introduction to the Dashboard

The Dashboard is a collection of pages, called **views**, that present data in a variety of charts and tables. You select which one to display by making a selection from the **Switch View** box. The names and contents of the available views are determined by the Oracle BPEL Process Analytics administrator who configured your Dashboard.

In addition to a variety of views available from the **Switch View** box, details about the data within the views are presented in detail pages when you click a data point within a view. (In the case of radar charts, you click a KPI label.) This process is referred to as **drilling down.**

The following topics introduce the views and detail pages:

- Types of Views on page 4-2
- Types of KPI Detail Pages on page 4-5

Types of Views

Depending on how the administrator configured the Dashboard, the first view you see when you log in to the Dashboard is one of the following:

- The Real-Time Viewer

  This view presents composite events in real time. By default, all dimensions of the composite event are displayed. You can adjust which composite events are displayed, the dimensions displayed, the time window over which the data is presented, and so on. Data in the Real-Time Viewer is refreshed every minute, by default.

  In addition, this page presents a table of data. The table displays aggregate values of the data shown in the chart.

  Figure 4–1 shows a sample Real-Time Viewer.

- A mulitpane view

  A **multiple pane view** is a default, or administrator-configured, view that consists of one or more charts, tables, or both. Each table or chart is presented within a

pane in the view. As the name implies, a multiple pane view typically consists of more than one view pane.

The charts and tables in a multiple pane view can present KPIs, metrics, alerts, cause and effect data, or information about pending composite events. The content of each Dashboard view is specified by the administrator when configuring the Dashboard. Data in a multiple pane view is refreshed every two minutes, by default.

You can adjust the time frame over which the data is presented (year, quarter, month, or day).

Figure 4–2 shows a sample multiple pane view.

Figure 4–1   Sample Real-Time Viewer
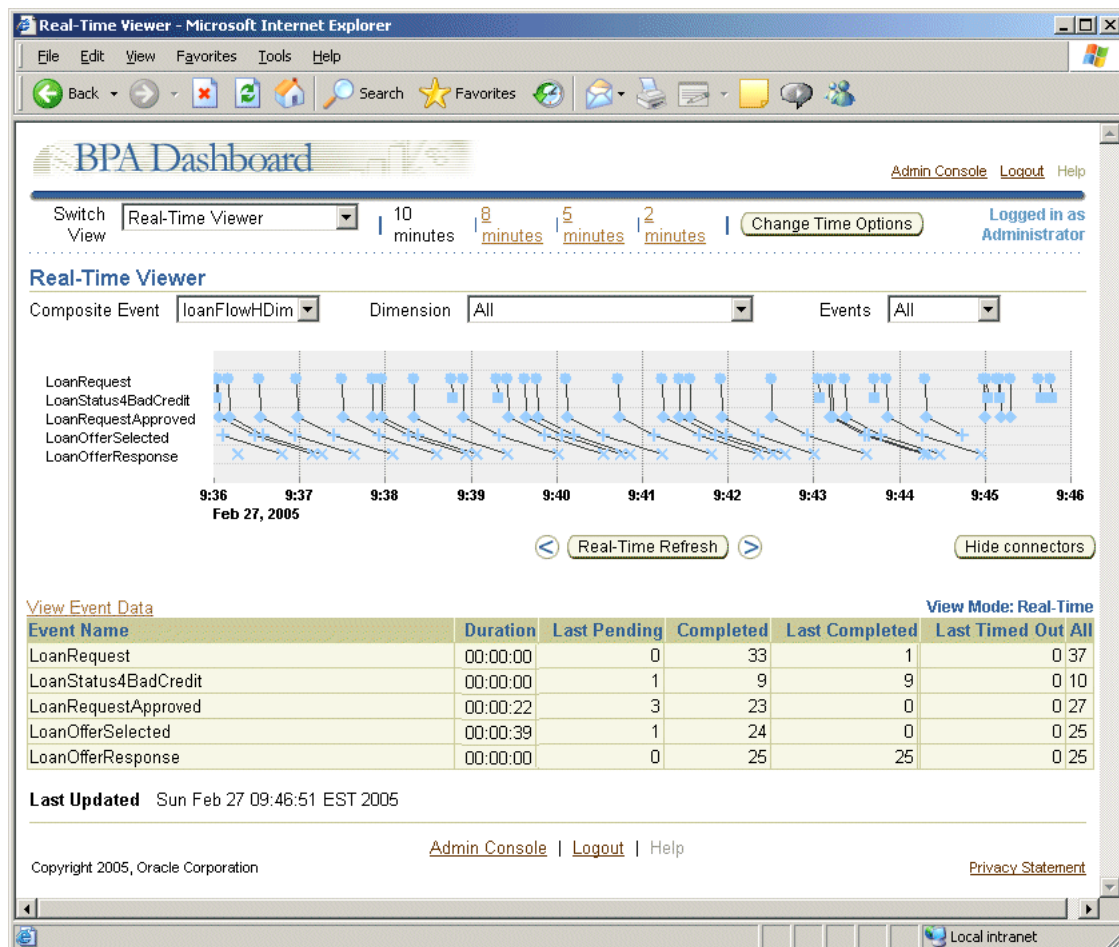
Figure 4–2   Sample Multiple Pane View
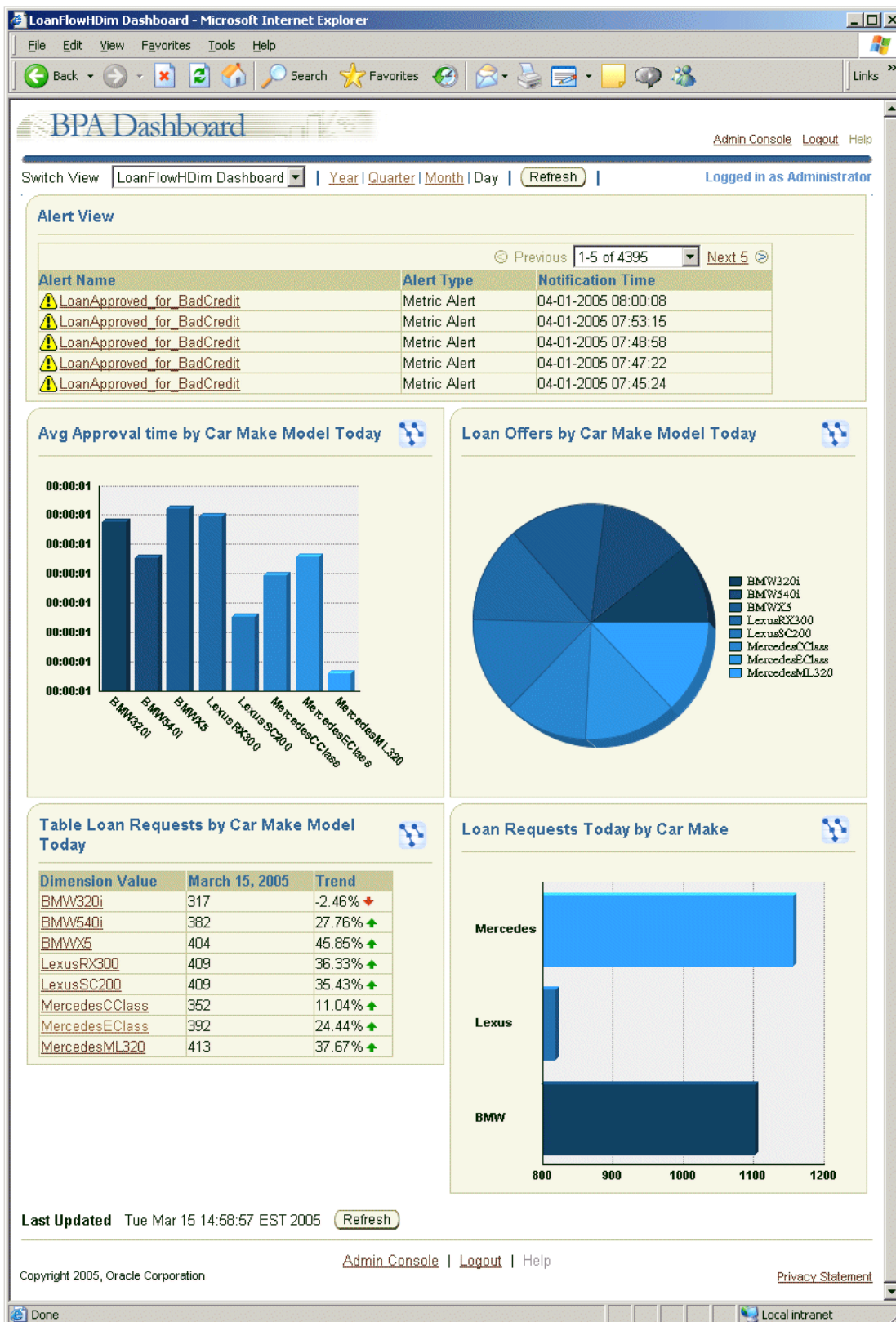
Types of KPI Detail Pages

In addition to the page presented when you open the Dashboard, you can access more pages by making a selection from the **Switch View** box, or, by clicking data within a chart in a KPI multiple pane view, you can access **detail pages** for any KPI in that chart.   Figure 4–3 shows the Performance detail page that is presented as a result of clicking a data point (a portion of the pie) in the Approvals by Provider chart presented in Figure 4–2, and then, in the Performance detail page, selecting a **Value** of **None.**

Figure 4–3   Sample Performance Detail Page
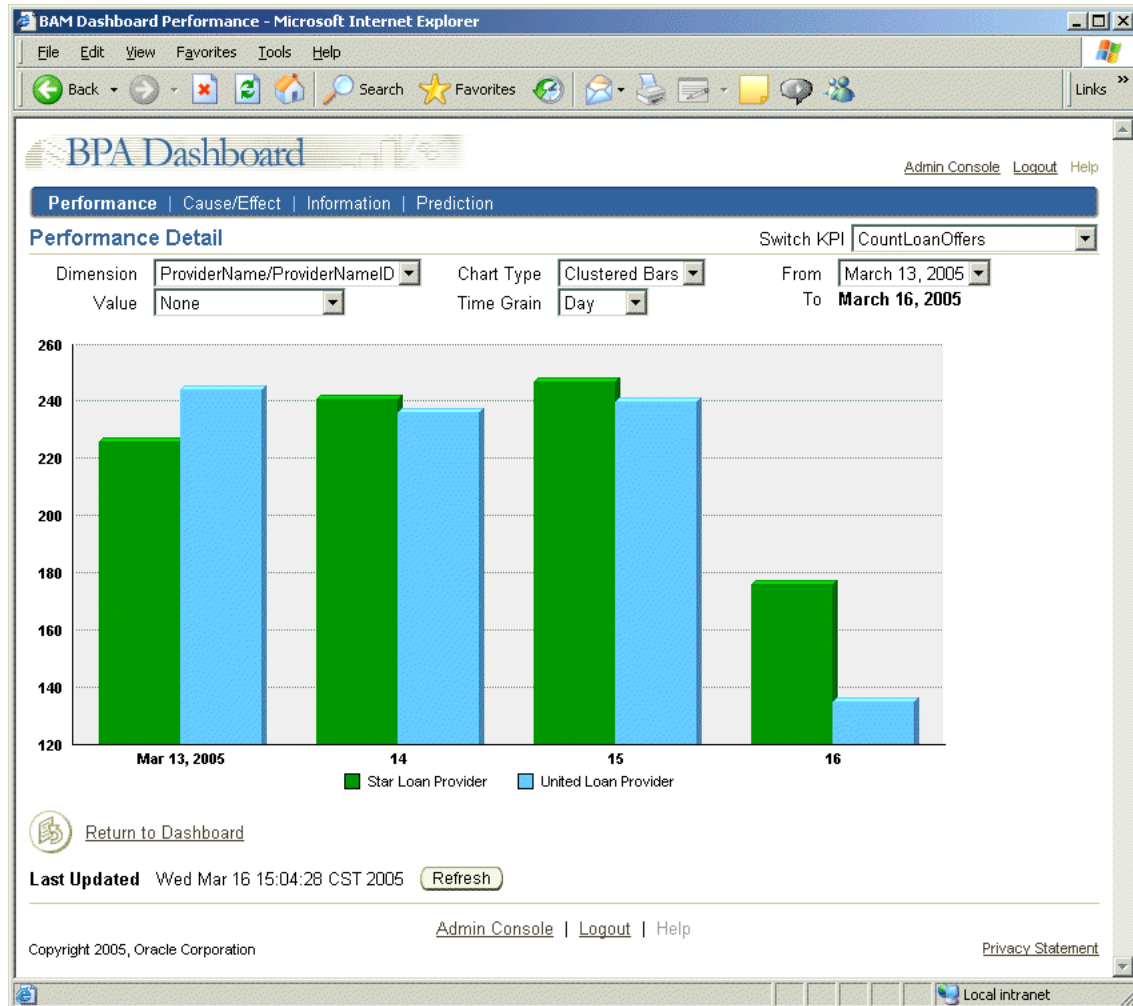
As shown on tabs within Figure 4–3, there are four detail pages available for each KPI:

■   Performance

This detail page shows how the selected KPI has performed for the selected time grain. If the administrator specified target values for the KPI, they can be included in the chart. See "Viewing and Understanding Performance Details" on page 4-19 for more information.

■   Cause/Effect

This detail page shows the KPIs that influenced the value of the selected KPI and the KPIs that the selected KPI influences (for a given time grain). See "Viewing and

Understanding Cause/Effect Tables – Performing Root-Cause Analysis" on page 4-26 for more information.

- Information

  This detail page lists KPI context documents, if available. KPI context documents are documents that provide some textual information about the selected KPI. See "Viewing and Understanding KPI Context Documents" on page 4-29 for more information.

- Prediction

  This detail page provides a prediction of the next KPI value for the current time grain. See "Viewing and Understanding KPI Predictions" on page 4-30 for more information.

Viewing Events and Composite Event Instances

The Real-Time Viewer displays instances of events within a composite event as they occur. For this reason, it is typically the most meaningful page during initial use of Oracle BPEL Process Analytics.

To access the Real-Time Viewer, choose one of these methods:

- In the **Switch View** box, select **Real-Time Viewer**.

- In a multiple pane view pane, click the **Real-Time Viewer** icon in the upper- right corner of any view pane (except one containing an Alert View). The **Real-Time Viewer** icon is shown in Figure 4–4.

  If a multiple pane view contains data from multiple composite events, then multiple Real-Time Viewer icons will be presented. Moving the cursor over any one of the Real-Time Viewer icons causes a pop-up window to open and display the composite event that will be displayed in the Real-Time Viewer if you click that icon.

Figure 4–4   Real-Time Viewer Icon

The Real-Time Viewer page opens and presents composite event instances in a chart and statistics about the event instances in a table. Figure 4–1 shows a sample Real-Time Viewer.

> **Note:**   Neither the Real-Time Viewer, nor the **Real-Time Viewer** icon will be available to you unless the Oracle BPEL Process Analytics administrator specified access to the Real-Time Viewer for your account when configuring the Dashboard. If **Real-Time Viewer** is not an option in your **Switch View** box (and the **Real-Time Viewer** icon is not present on view panes), it means the administrator did not grant access to your account.

Adjusting the Real-Time Viewer

You can make a variety of adjustments, as described in the following topics, to specify what is presented in the Real-Time Viewer:

- Specifying the Window of Time to Display on page 4-7

Specifying the Window of Time to Display

To specify the window of time that you want the event window to present, click one of the four time values at the top of the page. By default, you are presented with increments of 60 minutes, 45 minutes, 30 minutes, or 15 minutes. You can adjust the increments displayed on the page by clicking the **Change Time Options** button and then selecting a new set of time values. Available time windows from which you can select are the following:

- 24 hours, 18 hours, 12 hours, 6 hours

- 4 hours, 3 hours, 2 hours, 1 hour

- 60 minutes, 45 minutes, 30 minutes, 15 minutes

- 10 minutes, 8 minutes, 5 minutes, 2 minutes

Specifying the Composite Event to Display

To change the composite event displayed in the Real-Time Viewer:

1. In the **Composite Event** box, select the composite event you want to display. Note that you can select standalone composite events (those not part of a composite event group) or composite event groups. However, composite events that are part of a composite event group are not presented in this box.

    Standalone composite events and composite event groups are presented in the Real-Time Viewer as composite events. Composite events that are part of a composite event group are not displayed separately from the group.

2. Click **Apply.**

The Real-Time Viewer refreshes to display data for the selected composite event.

Specifying the Dimensions to Display

Using the **Dimension** control, you can specify that composite event instances be displayed along a dimension.

To specify the dimension to display:

1. In the **Dimension** box, select the dimension you want to display.

2. Click **Apply.**

The chart refreshes to present instances of the selected composite event along the selected dimension.

For example, using the loan flow example presented in Figure 1–1, suppose the administrator defined these two dimensions: CarMake and CarModel for auto loans. You can specify one of these dimensions as the category by which to present composite

events in the Real-Time Viewer. Figure 4–5 shows the Real-Time Viewer when CarMake is selected as the dimension by which to categorize composite events; Figure 4–6 shows the Real-Time Viewer when CarModel is selected as the dimension by which to categorize composite events.

> **Note:** If you select a dimension that was not included in the composite event when it was created, the refreshed chart displays data as though you selected **All** from the **Dimension** box, and displays a message that reads, "Warning: Dimension *dimension-name* is not defined for composite event *composite-event-name*", where *dimension-name* is the dimension you selected and *composite-event-name* is the composite event you selected.

Figure 4–5   Real-Time Viewer with CarMake Dimension Selected
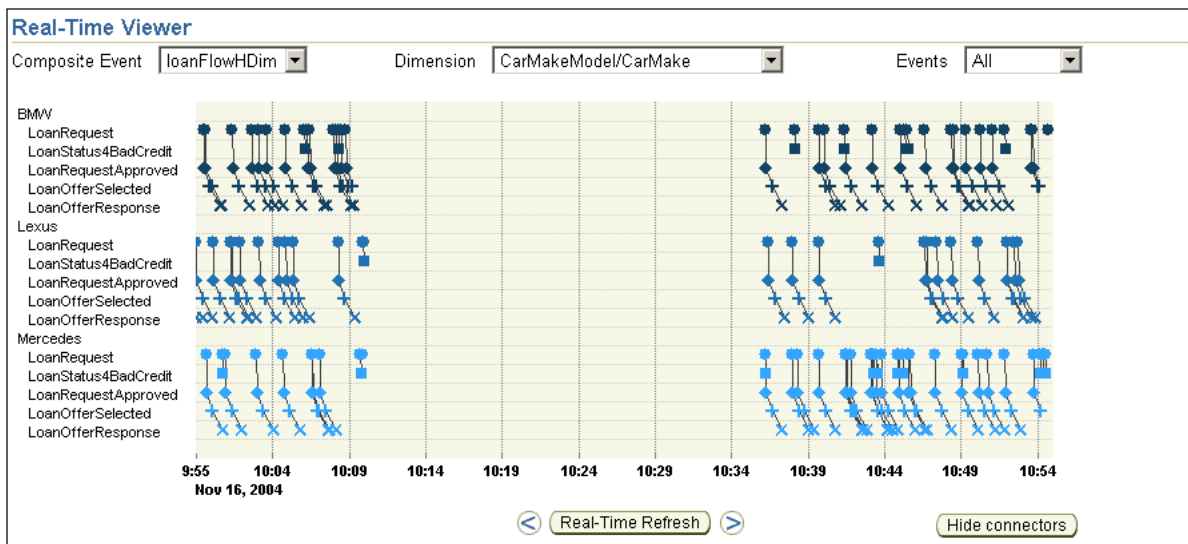
Figure 4–6   Real-Time Viewer with CarModel Dimension Selected
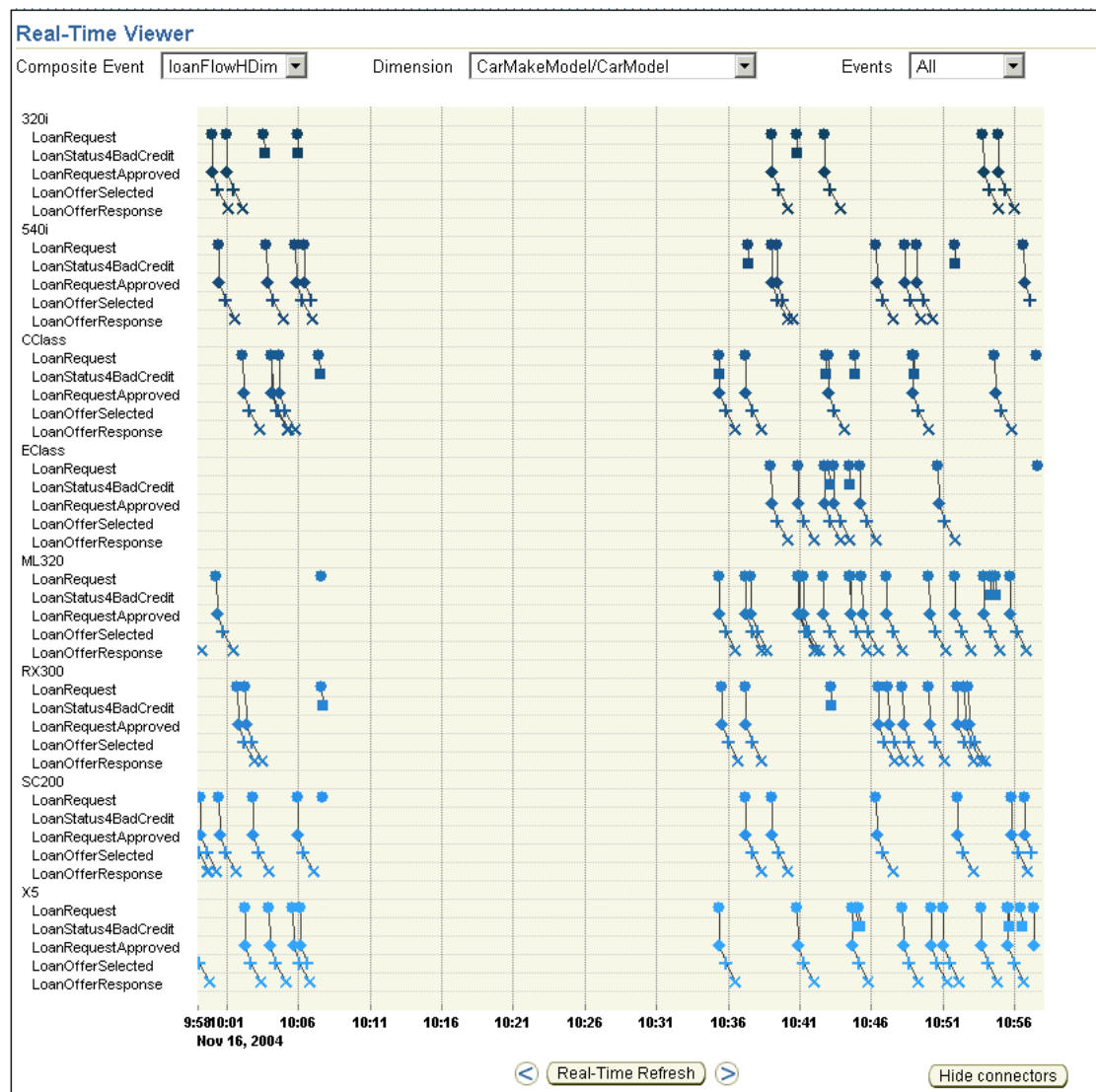
Specifying the Type of Composite Events to Display

To specify the type of composite events that are included in the Real-Time Viewer:

1.   In the **Events** box, select the type of composite event you want to display.

     The text following this list describes the types from which you can choose.

2.   Click **Apply.**

The Real-Time Viewer refreshes to display the selected type of composite events.

The types of composite events from which you can choose are as follows:

■   All

     When you select **All** from the **Events** box, all event instances captured by Oracle BPEL Process Analytics are presented in the Real-Time Viewer (chart and table).

     Figure 4–7 shows an example of a chart that contains all returned event instances, including timed-out events. Each marker (circle, diamond, square, plus sign, and x) represents a different event instance within each composite event instance. Line

segments connect event instances that are part of the same composite event instance.

- Complete

    When you select **Complete** from the **Events** box, only composite event instances for which end events have been returned are presented in the Real-Time Viewer. Figure 4–8 shows an example of an chart that contains only complete composite events.

- Pending

    When you select **Pending** from the **Events** box, only composite event instances for which an end event has not yet been returned (and which are not timed out) are presented in the Real-Time Viewer. Figure 4–9 shows an example of an Real-Time Viewer that contains only pending composite events.

- Timed Out

    When you select **Timed Out** from the **Events** box, only timed out composite event instances are presented in the Real-Time Viewer. A composite event instance is considered to be timed out if the next event in the process flow represented by the composite event has not been received within the time period specified by the time-out value. The time out value is specified by the administrator when the composite event is defined.

    A chart containing only timed-out events looks similar to an chart containing only pending events.

Figure 4–7   Real-Time Viewer Displaying All Composite Event Instances
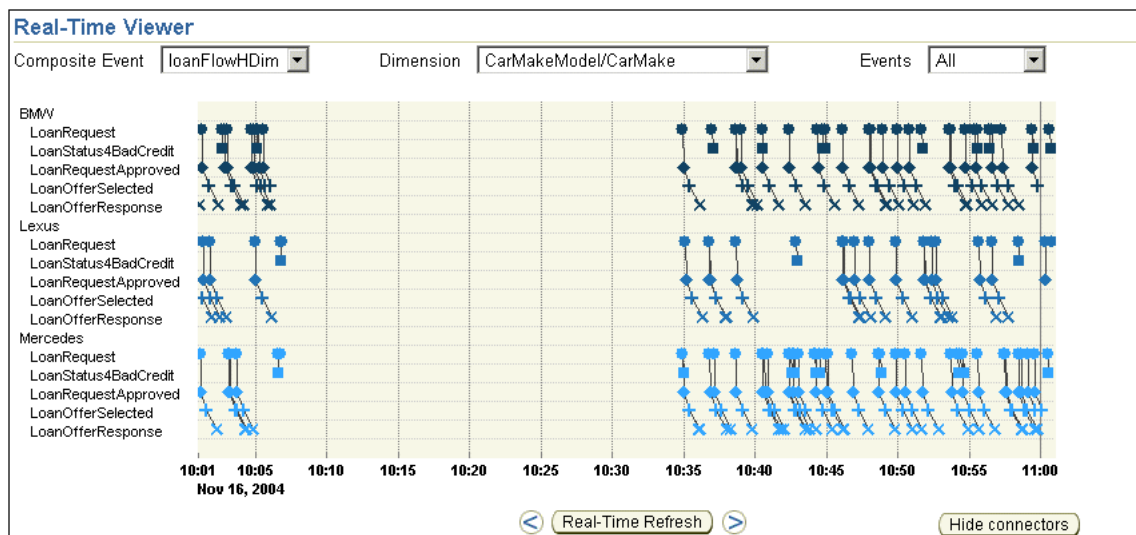
Figure 4–8   Real-Time Viewer Displaying Complete Composite Events Only
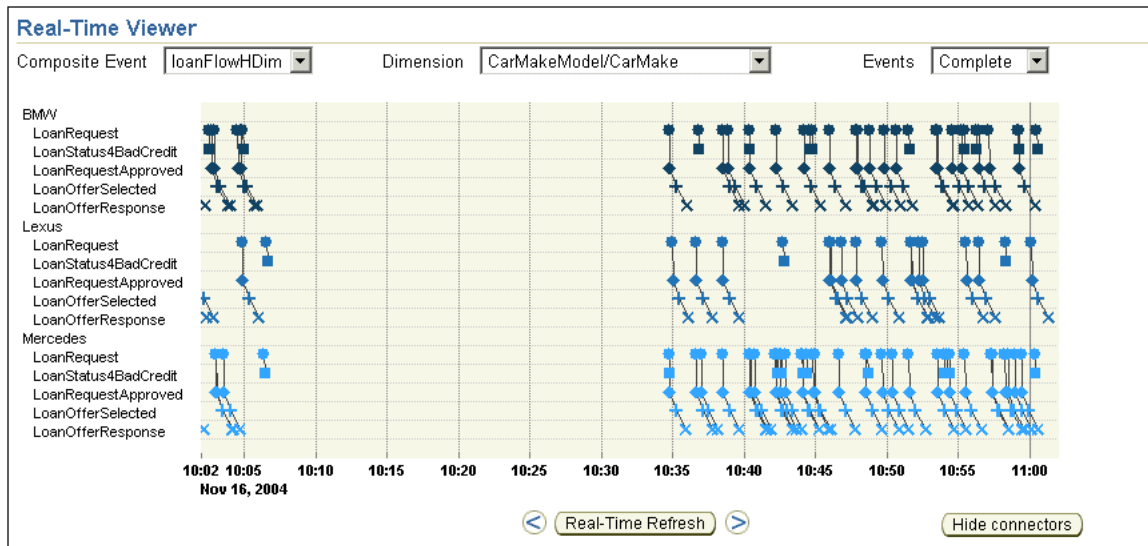
Figure 4–9   Real-Time Viewer Displaying Pending Composite Events Only
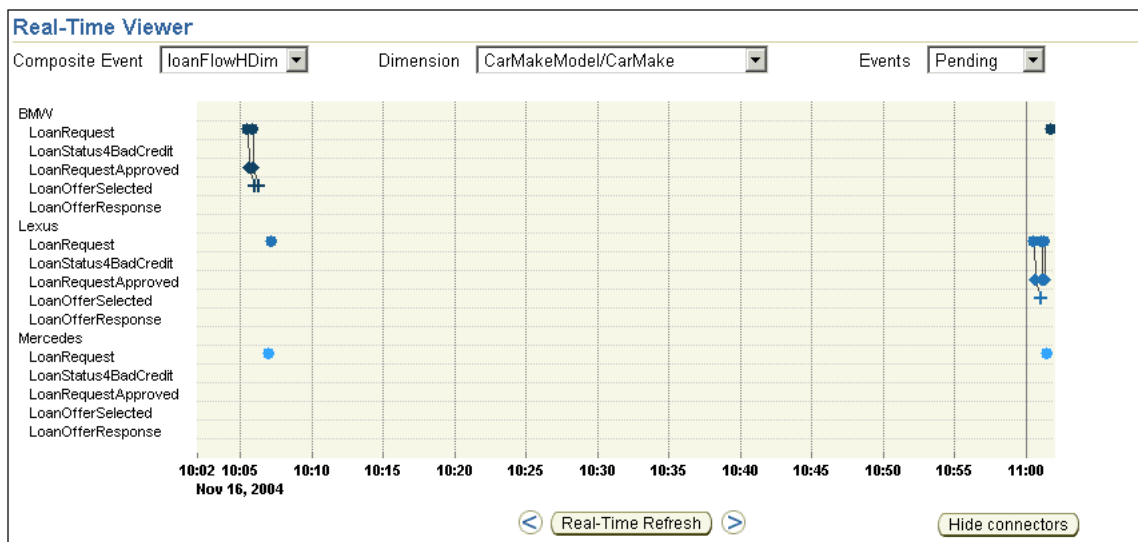
Scrolling and Refreshing the Real-Time Viewer Chart Data

The **left arrow** and **right arrow** under the chart allow you to scroll back and forth through the chart data. Click the **left arrow,** to scroll back one increment of time. Click the **left arrow** several times to scroll back additional increments of time. The time increment is one-half of the time unit you selected at the top of the page, as described in "Specifying the Window of Time to Display" on page 4-7. Similarly, click the **right arrow,** to scroll forward a unit of time.

To refresh the chart data and resume the presentation of data for the current time window, click **Real-Time Refresh**. By default, the chart is refreshed automatically every minute.

The View Mode is specified as Real-Time when you are viewing the current time window; it is specified as historical when you are viewing any other time window.

Showing and Hiding Connectors

In addition to specifying which composite events are included, you can specify whether or not connectors between related events within a composite event instance are connected. Click **Hide connectors** to hide the connectors; click **Show connectors** to show the connectors.

Figure 4–10 shows the Real-Time Viewer with the connectors hidden. Figure 4–8 shows the Real-Time Viewer with the connectors displayed.

When the events within a composite event occur within a relatively short time period, it can be useful to include the connectors between events. When the events within a composite event occur over a long period of time, days for instance, then the connectors will probably be of little value, and you might want to hide them.

Figure 4–10   Composite Event Instances with Hide Connectors Selected
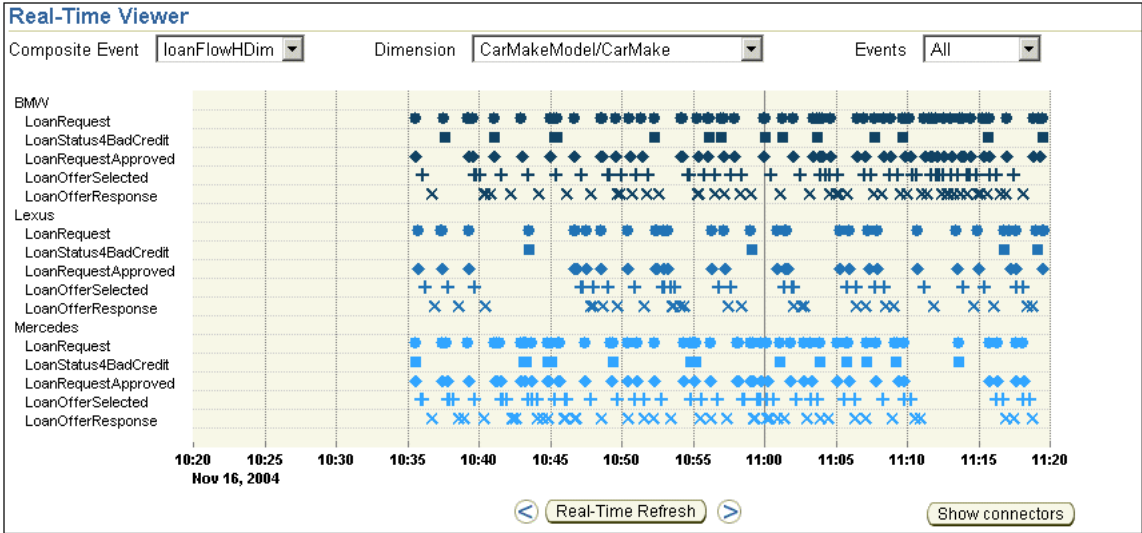
Viewing Event Data in a Table

To view the chart data in a tabular format, click **View Event Data**. A page opens with a table such as that shown in Figure 4–11, which presents the dimension name, if applicable, the time at which the event occurred, the type of event, and the **eyeglass** icon, which is a link to access composite event details.

See "Accessing Composite Event Details from the Chart or Event Data Table" on page 4-13 for information about displaying composite event details.

Figure 4–11   Sample Event Data

Accessing Composite Event Details from the Chart or Event Data Table

You can access composite event details from both the chart and the Event Data table, as follows:

- In the chart, click a data point within the chart.

- In the **Event Data** table, click an **Event Type** name or the **eyeglass** icon.

In all cases, the Composite Event Details page opens. However, when you click a data point in the chart or an Event Type name in the Event Data table, the Composite Event Details page opens with the navigation tree expanded to display the attributes of the selected event type, as shown in Figure 4–12. When you click the **eyeglass** icon, the Composite Event Details page opens with the navigation tree collapsed, as shown in Figure 4–13.

The Composite Event Details page presents the composite event in a navigation tree. You can expand the navigation tree to view the events and the attributes of the events within the named composite event. For each event, the time at which the event occurred is presented. For each event attribute, the attribute type and value are presented.

Figure 4–12   Sample Composite Event Details Page - Navigation Tree Expanded

Figure 4–13   Composite Event Details - Navigation Tree Collapsed

Understanding the Real-Time Viewer Table

The table that appears at the bottom of the Real-Time Viewer page displays statistics about the composite event instances in aggregate. Figure 4–14 shows a sample of this table.

Figure 4–14   Sample Table of Composite Event Instance Aggregate Values

Event Name	Duration	Last Pending	Completed	Last Completed	Last Timed Out	All
LoanRequest	00:00:00	0	201	0	0	205
LoanStatus4BadCredit	00:00:00	2	42	42	0	44
LoanRequestApproved	00:00:24	2	157	0	0	161
LoanOfferSelected	00:00:40	2	159	0	0	161
LoanOfferResponse	00:00:00	0	160	160	0	160

This table presents seven columns of data for each event in a composite event. The following list describes the meaning of each column:

- Event Name – Specifies the name of an event within the composite event.

- Duration – Specifies the average time between the named event and the previous event in the process captured by the composite event. In the example shown in Figure 4–14, the time between a LoanOfferSelected event and a LoanOfferResponse event is the longest, on average.

- Last Pending – Specifies the last event received for all pending composite event instances. A **pending composite event** is defined as one for which an end event has not yet been received, but which has not timed out. End events will never have a last pending value. In the example shown in Figure 4–14:

  - Two composite event instances have returned the LoanStatus4BadCredit event and are awaiting the next event in the process.

  - Two composite event instances have returned the LoanRequestApproved event and are awaiting the next event in the process.

  - Two composite event instances have returned the LoanOfferSelected event and are awaiting the next event in the process.

  - No events are considered last pending for the LoanOfferResponse event, as expected, because it is an end event.

- Completed – Specifies the number of instances of the specified event that have occurred for all completed composite event instances, where a **completed composite event instance** is defined as one for which an end event has been received. In the example shown in Figure 4–14, of all completed composite event instances:

  - 201 included a LoanRequest event instance

  - 42 included a LoanStatus4BadCredit event instance

  - 157 included a LoanRequestApproved event instance

  - 159 included a LoanOfferSelected event instance

  - 160 included a LoanOfferResponse event instance

- Last Completed – Indicates which events were end events for all of the completed composite events. In the example shown in Figure 4–14, 42 composite event instances had an end event of LoanStatus4BadCredit, and 160 composite event instances had an end event of LoanOfferResponse.

- Last Timed Out – Specifies the count of event instances that were the last received before a composite event instance timed out. In the example shown in Figure 4–14, no composite events had timed out during the time window represented in the table. This column can be useful with helping to determine where a problem is occurring. If one event has a high last timed-out value, it indicates that there is a problem getting from this event in the process to the next event and probably warrants further investigation.

- All – Indicates the total number of instances of the specified event received, including completed and pending composite events. In the example shown in Figure 4–14, for all complete and pending composite event instances:

  - 205 LoanRequest event instances have been received.

  - 44 LoanStatus4BadCredit event instances have been received.

  - 161 LoanRequestApproved event instances have been received.

  - 161 LoanOfferSelected event instances have been received.

  - 160 LoanOfferResponse event instances have been received.

Viewing Metrics and KPIs in Multiple Pane Views

Charts and tables representing values for metrics and KPIs are presented in multiple pane views. The following topics describe how to access multiple pane views and understand the charts and tables presented in them:

- Viewing Metrics in Multiple Pane Views on page 4-16

- Viewing KPIs in Multiple Pane Views on page 4-17

- Determining the Exact Value for a Chart Element on page 4-32

- Viewing a Metric Table View of a Data Series on page 4-32

- Viewing Alerts in Multiple Pane Views on page 4-33

Viewing Metrics in Multiple Pane Views

Metric multiple pane views, designed by the Oracle BPEL Process Analytics administrator when configuring the Dashboard (or as configured by default, in the case of the Default Metric View Pane), can contain any number and any combination of metric tables, metric charts, and an Alert View table.

To access a metric multiple pane view:

1. In the **Switch View** box, select the desired metric multiple pane view.

    A page opens, typically with several charts, tables, or a combination of the two.

2. Specify the window of time that you want the charts in the multiple pane view to present by selecting a time from the **Switch Time** box. Choices range from 5 minutes to 24 hours.

The **Refresh** button enables you to refresh the multiple pane view on demand; by default, a multiple pane view refreshes automatically, every 2 minutes.

Other than specifying the time window to include in the charts and tables, you cannot change the content of multiple pane views – the type of chart or table, the colors used in the chart, and so on are determined by the administrator when configuring the Dashboard.

The following topics provide more information about using and understanding the tables and charts that can be presented in a metric multiple pane view:

- Determining the Exact Value for a Chart Element on page 4-32
- Viewing Alerts in Multiple Pane Views on page 4-33
- Viewing a Metric Table View of a Data Series on page 4-32

Viewing KPIs in Multiple Pane Views

KPI multiple pane views, designed by the Oracle BPEL Process Analytics administrator when configuring the Dashboard (or configured by default, in the case of the Default KPI View Pane), can contain any number and any combination of KPI tables, KPI charts, and an Alert View table. Figure 4–2 shows a sample KPI multiple pane view.

To access a KPI multiple pane view:

1. In the **Switch View** box, select the desired KPI multiple pane view.

   A page opens, typically with several charts, tables, or a combination of the two.

2. Specify the window of time that you want the charts in the multiple pane view to present by clicking one of the four time values at the top of the page. Choices are: **Year, Quarter, Month,** and **Day.**

The **Refresh** button enables you to refresh the multiple pane view on demand; by default, a multiple pane view refreshes automatically, every 2 minutes.

Other than specifying the time window to include in the charts and tables, you cannot change the content of multiple pane views – the type of chart or table, the colors used in the chart, and so on are determined by the administrator when configuring the Dashboard.

The following topics provide more information about using and understanding the tables and charts that can be presented in a KPI multiple pane view:

- Viewing and Understanding Radar Charts on page 4-17
- Accessing the Real-Time Viewer for a KPI Presented in a View Pane on page 4-18
- Viewing and Understanding Cause/Effect Tables – Performing Root-Cause Analysis on page 4-26
- Determining the Exact Value for a Chart Element on page 4-32
- Viewing Alerts in Multiple Pane Views on page 4-33

Viewing and Understanding Radar Charts

A radar chart displays KPI values in relation to their performance bands. Performance bands are defined by the Oracle BPEL Process Analytics administrator. A radar chart can display either multiple KPIs or multiple dimensions of one KPI.

When performance bands are defined, they allow a KPI to be divided into discrete bands to categorize performance. Oracle BPEL Process Analytics supports five performance bands, based on five performance categories relative to a target value. (For example, the default categories for performance bands are 0 - 20%, 20% - 40%, 40% - 60%, 60% - 80%, and 80% - 100%). These bands are also referred to as weak, under, normal, over, and strong performers, respectively.

Reading a radar chart is best explained using an example. Figure 4–15 shows a sample radar char. This figure presents the sales performance of five KPIs over 2 days. A blue

polygon represents KPI performance relative to target values on March 16th and a green polygon represents KPI performance relative to target values on March 17th. The radar chart shows:

- AvgLoanAPR was a normal performer on both days.

- CountLoanOffers was a strong performer on the 16th, but an under performer on the 17th.

- CountLoanOffers4BadCredit was a strong performer on the 16th, but an under performer on the 17th.

- CountReq was a strong performer on the 16th, but an under performer on the 17th

- CountReqReject was a strong performer on the 16th, but a normal performer on the 17th.

As shown in Figure 4–15, due to limited space, KPI labels can be cropped. To view the full KPI name, place the mouse cursor over the KPI label; a pop-up window displays the full KPI name. Also note that radar charts are optimally displayed when the target screen size is set to 800 x 600 pixels. At other sizes, the radar chart appears off-center. The Oracle BPEL Process Analytics administrator can reset the target screen size, as described in "Setting the Dashboard Screen Size" on page 3-13.

To drill down in the radar chart and view KPI detail pages, click a KPI (or KPI dimension) label within the chart.

Figure 4–15   Sample Radar Chart
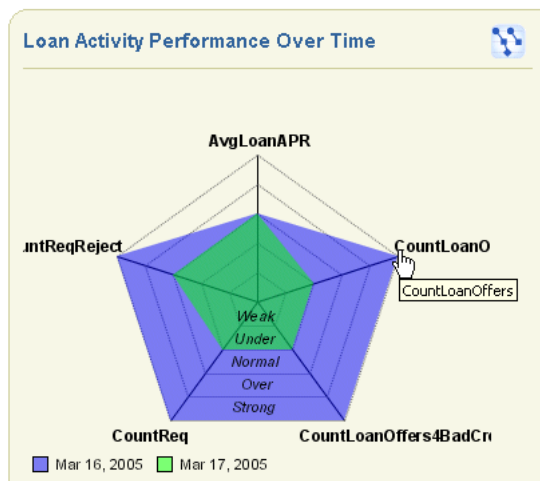
Accessing the Real-Time Viewer for a KPI Presented in a View Pane

After viewing a KPI in a view pane, you may decide you want to see the real-time delivery of the event instances within the composite event on which the KPI is defined.

To access the Real-Time Viewer from a multiple pane view, click the **Real-Time Viewer** icon (shown in Figure 4–4) that appears in the upper-right corner of the KPI view pane for which you want to view events.

If a multiple pane view contains data from multiple composite events, then multiple Real-Time Viewer icons will be presented. Moving the cursor over any one of the Real-Time Viewer icons causes a pop-up window to open and display the composite event that will be displayed in the Real-Time Viewer if you click that icon.

Viewing and Understanding KPI Details

KPI detail pages enable you to analyze, in depth, a KPI presented in a multiple pane view. Four KPI detail pages are available, as described in the Table 4–1.

Table 4–1   Summary of KPI Detail Pages

Page Name	Description	Detailed Information
Performance	Presents KPI details over a selected time period. If target values for the selected KPI have been specified, they are included in the presentation.	Viewing and Understanding Performance Details on page 4-19
Cause/Effect	Presents the KPIs that influence the selected KPI and the KPIs that the selected KPI influences. Useful for root-cause analysis.	Viewing and Understanding Cause/Effect Tables – Performing Root-Cause Analysis on page 4-26
Information	Presents a table of links to documents that provide information relevant to the selected KPI.	Viewing and Understanding KPI Context Documents on page 4-29
Prediction	Presents a prediction for the selected KPI value.	Viewing and Understanding KPI Predictions on page 4-30

Viewing and Understanding Performance Details

The Performance Detail page presents KPI performance data over time. If the administrator has specified target values for a KPI, the target values are also presented, so that you can compare the actual performance values to the target performance values.

To access performance details:

1. In the Dashboard, select a multiple pane view from the **View** box.

2. In the multiple pane view, click a data point in any chart or table. A Performance Detail page opens.

3. In the **Switch KPI** box, select the KPI for which you want to view performance. The Performance Detail page opens for the selected KPI.

Controls in the Performance Detail page allow you to specify the dimensions by which KPIs are filtered, the specific dimension values to present, the type of chart in which to present the data, and how far back from the current time you want the chart to include data. As with charts in multiple pane views, you can determine the exact value of an element in a chart presented in the Performance Detail page by placing the mouse cursor over the chart element of interest.

Table 4–2 describes the available controls.

Table 4–2   Performance Detail Page Controls

Control	Description
Switch KPI	Select the KPI for which you want to view details.
Dimension	Select the dimension by which to present the KPI data. If you select **All,** the data is not broken down by dimension.

Table 4–2   (Cont.) Performance Detail Page Controls

Control	Description
Value	Select the dimension value by which to present the KPI data. If you select **None,** the data is not limited to a dimension value.
Chart Type	Select the type of chart in which you want the KPI data presented. If you select **ALL** in the **Dimension** box, then the choices are Forecast, Line, or Bar chart; otherwise, the choices are Clustered Bar, Stacked Bar, or Line.
	A **forecast chart** presents actual data in a bar chart with the target data presented in a line graph superimposed on the bar chart.
Time Grain	Select the time units in which to present data in the KPI chart.
From	Select the start date on which you want the KPI data presented. Options for this control vary, depending on the unit of time you selected at the top of the page.
	For each unit of time, you can specify that data be presented as early as 8 time units before the current time (inclusive). For example, if the unit of time is Year, then you can specify that data be presented as far back as 8 years before the current year. Similarly, if the time unit is Quarter, then you can specify that data be presented as far back as 8 quarters before the current quarter.

The following figures provide some examples of the different choices you can make with controls in the Performance Detail page:

- Figure 4–16 displays CountLoanOffers KPIs in a line chart over a period of days. KPIs are not broken down by dimension. Figure 4–17 displays the same data, along the ProviderName dimension

- Figure 4–18 displays the CountLoanOffers KPIs for the ProviderName dimension and both dimension values (Star Loan Provider and United Loan Provider) in a clustered bar chart over a period of days. Figure 4–19 displays the same data, in a stacked bar chart.

Figure 4–16   Sample Performance Detail Page: Line Chart, All Dimensions
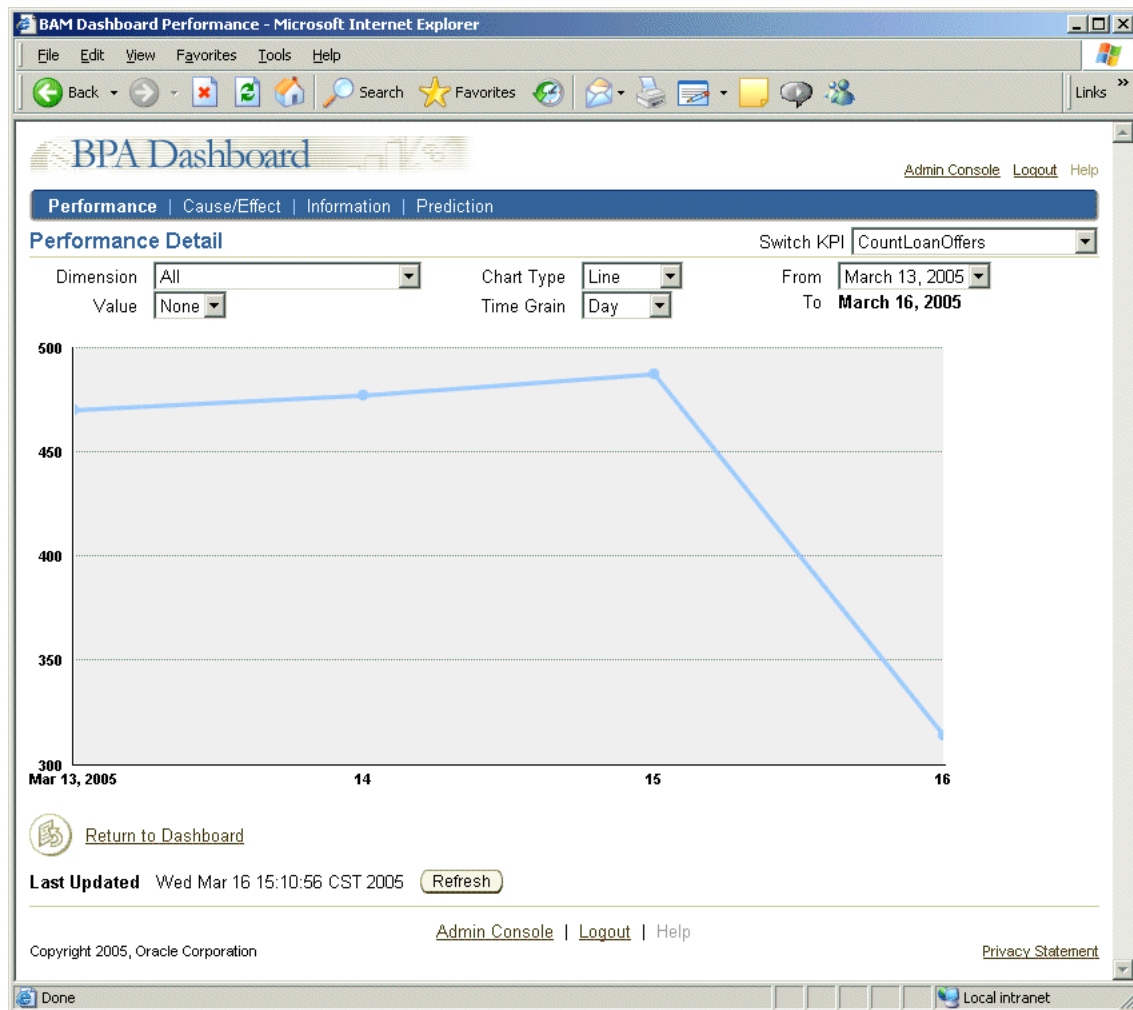
Figure 4–17   Sample Performance Detail Page: Line Chart, Selected Dimension
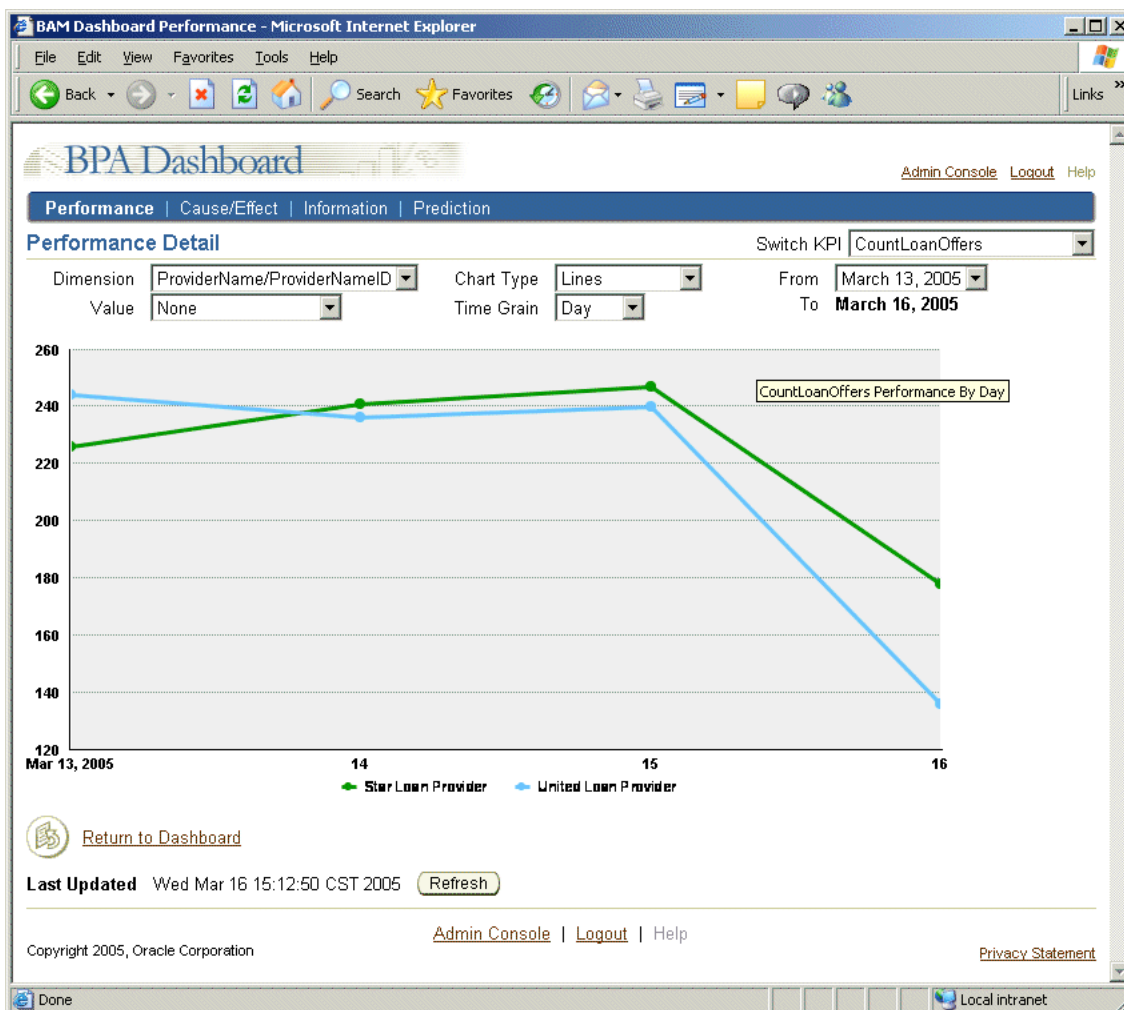
Figure 4–18   Sample Performance Detail Page: Clustered Bar Chart, Selected Dimension
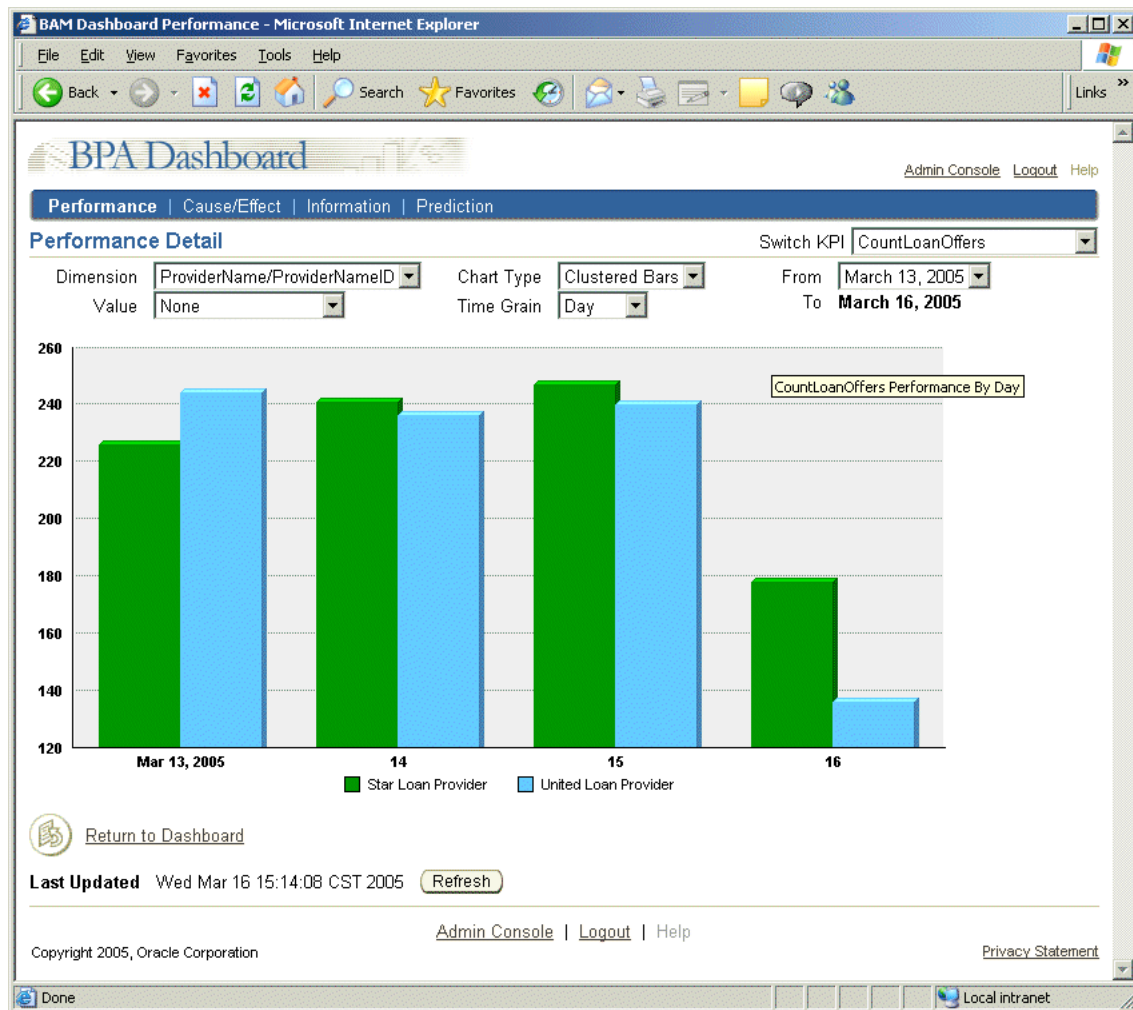
Figure 4–19   Sample Performance Detail Page: Stacked Bar Chart, Selected Dimension
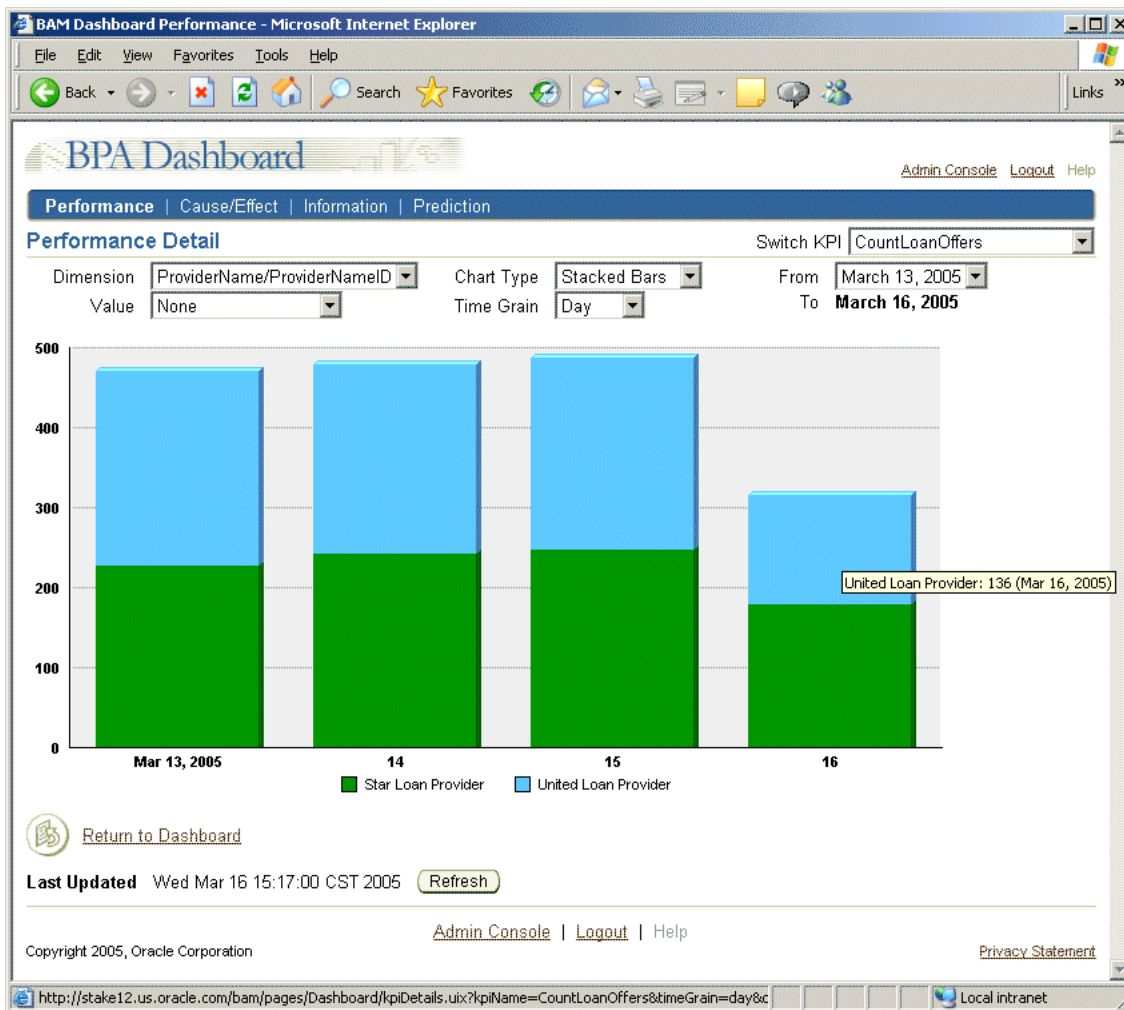
In addition to using the controls to view data along different dimensions, when
hierarchical dimensions are defined, you can drill down in the dimension by clicking
data points in the chart. A **hierarchical dimension** is one that uses ordered levels as a
means of organizing data. For example, in a time dimension, the hierarchy organizes
data at the year level (hierarchy 1), the month level (hierarchy 2), or the day level
(hierarchy 3).

Figure 4–20 shows the data presented along the CarMake dimension – a hierarchy 1
dimension. Figure 4–21 shows the result when you click a bar representing the BMW
make; all of the BMW models are presented.

Figure 4–20   Data Presented Along Hierarchy One Dimension

Figure 4–21   Data Presented Along Hierarchy Two Dimension

Viewing and Understanding Cause/Effect Tables – Performing Root-Cause Analysis

A Cause/Effect table is one of a number of tables and charts that can be configured by the administrator to include in the Dashboard. A Cause/Effect table is useful for performing root-cause analysis of a KPI value.

There are two forms of the Cause/Effect table: a full form of the table, presented in a KPI detail page; and an abbreviated form, which is presented when a Cause/Effect table is included in a multiple pane view.

To access a full form Cause/Effect table:

1. In the Dashboard, select a multiple pane view from the **Switch View** box.

2. In the multiple pane view, click a data point in any chart or table. A Performance Detail page opens.

3. Click **Cause/Effect.**

4. In the **Switch KPI** box, select the KPI for which you want to view cause and effect details.

5. In the **Analysis Time Grain** box, select the unit of time (day, month, quarter, year) over which you want Cause/Effect details presented.

Figure 4–22 shows an example of the full form of a Cause/Effect table. (The abbreviated form excludes the Performance, Cause/Effect, and Compare columns.)

A Cause/Effect table shows the following, in relation to a selected KPI:

- Cause KPIs – Each of the KPIs that influence the selected KPI's value, the amount of influence each Cause KPI has on that selected KPI, and the KPI performance band value for the Cause KPI are presented. In Figure 4–22, for example, the SumLoanAPR value has an influence of 30.89% on the CountLoanOffers4United KPI value. The SumLoanAPR KPI is under performing for the day (as indicated by the Band column).

- Effect KPIs – Each of the KPIs on which the selected KPI has an influence, the amount of influence the selected KPI has on each of the Effect KPIs, and the KPI performance band value for the Effect KPI are presented. In Figure 4–22, for example, the SumLoanAPR KPI's performance has an influence of 31.67% on the AvgLoanAPR KPI's performance. The AvgLoanAPR KPI is a normal performer for the day (as indicated by the Band column)

In addition, in the full form of the table, the Performance, Cause/Effect, and Compare columns provide links to detail pages about each of the KPIs presented in the table, as follows:

- Performance

  Provides a link to the KPI Performance detail page for the associated Cause or Effect KPI. If the Band column indicates that the KPI performance is other than expected, you might click this link to see how that particular KPI has been performing, particularly if it has a strong influence on the selected KPI or is strongly affected by the selected KPI.

  See "Viewing and Understanding Performance Details" on page 4-19 for more information.

- Cause/Effect

  Provides a link to the KPI Cause/Effect detail page for the associated Cause or Effect KPI. If the Band column indicates that the KPI performance is other than expected, you might click this link to see what is influencing this Cause or Effect KPI and what other KPIs this Cause or Effect KPI is affecting.

- Compare

  Provides a link to a page that presents the selected KPI and the Cause KPI in bar charts for easy visual comparison. Clicking the **Compare All** button displays all of the KPIs accessible to the user in bar charts.

  Figure 4–23 shows an example of the page that opens when you click a **Compare** icon. The page that is presented when you click the **Compare All** button is similar, but includes a chart for each of the Cause or Effect KPIs.

Figure 4–22   Cause/Effect Table in KPI Detail Page
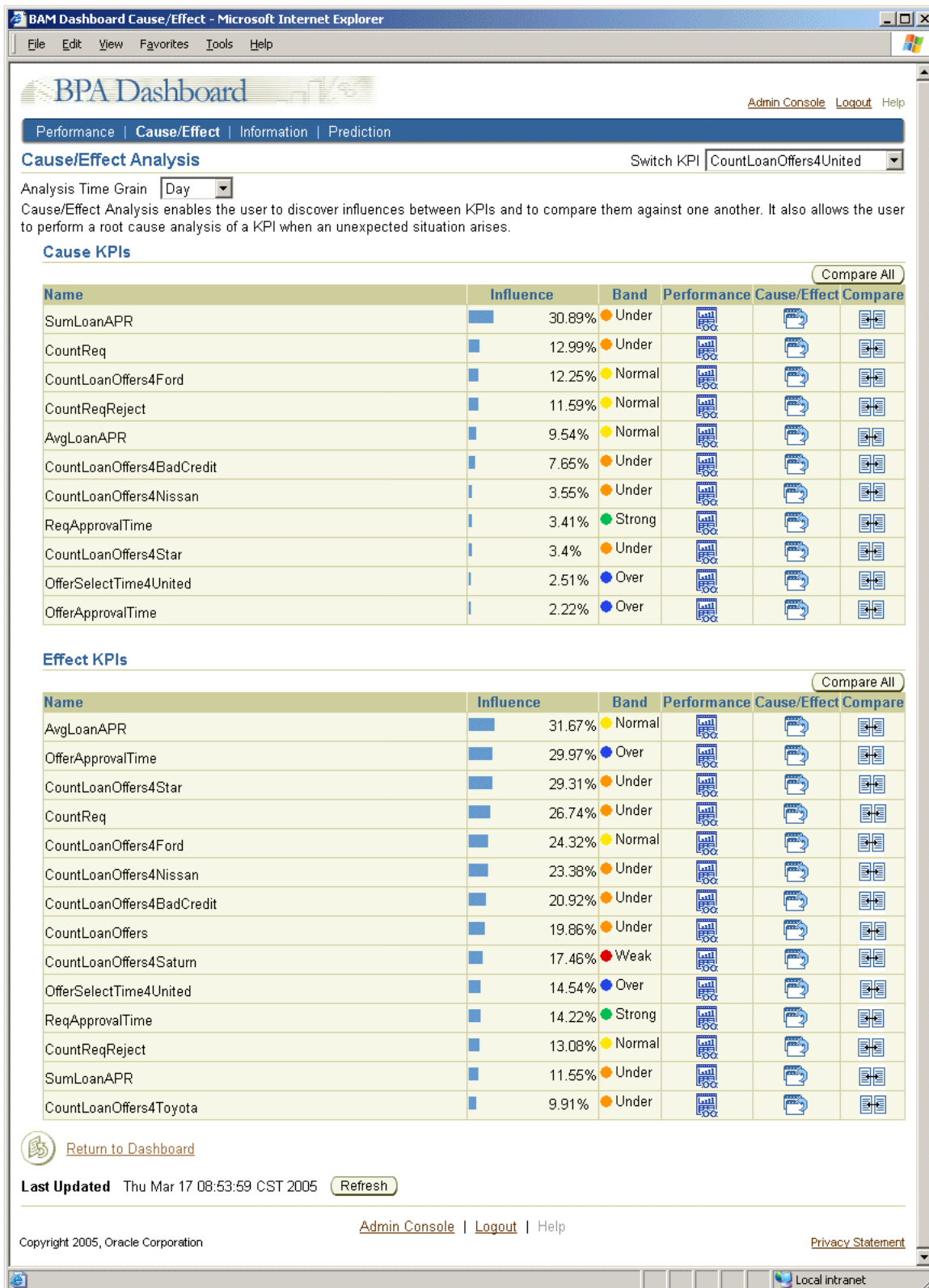
Figure 4–23   KPI Comparison Page
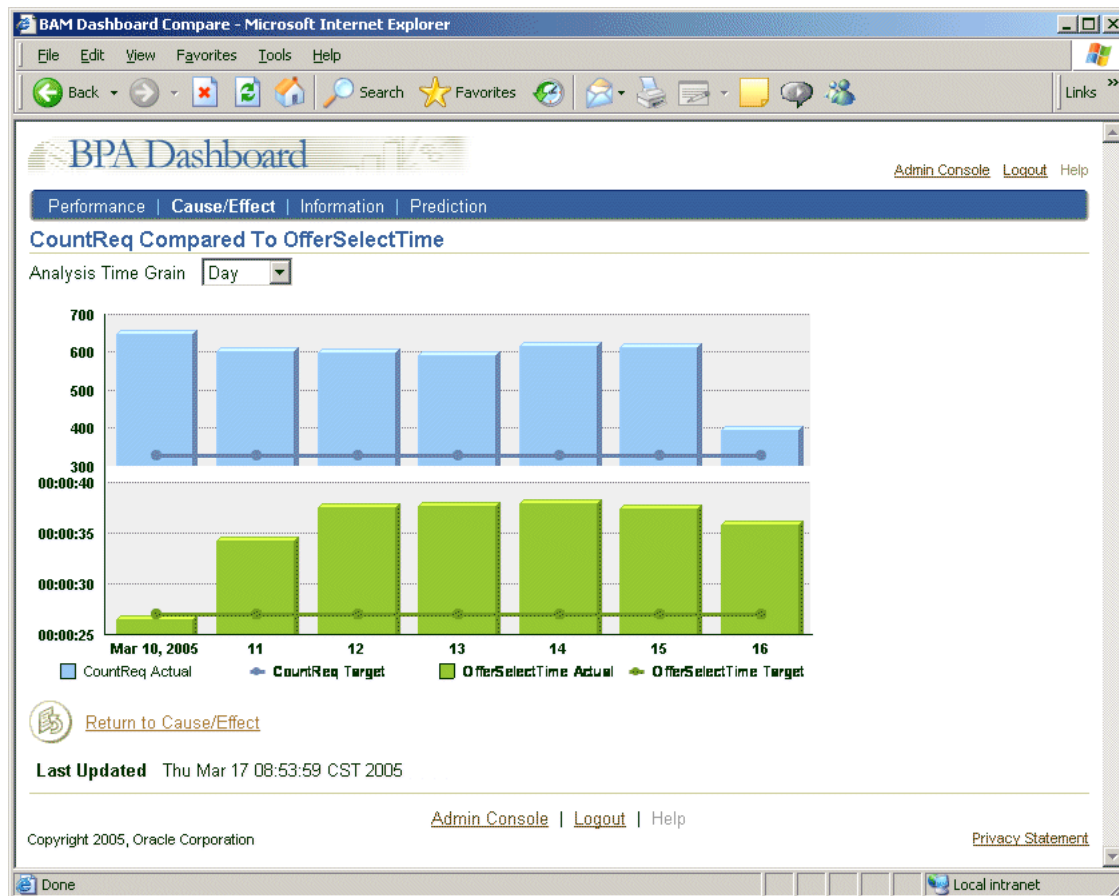
Viewing and Understanding KPI Context Documents

You can view a list of documents related to the KPI you select, assuming the administrator configured them, in the Information Documents page. Documents are listed in order of their weighted value. The weighted value indicates how likely the document will contain information of interest to you.

To access KPI context documents:

1. In the Dashboard, from the **Switch View** box, select a multiple pane view.

2. In the multiple pane view, click a data point in any chart or table. A Performance Detail page opens.

3. In the **Switch KPI** box, select the KPI for which you want to view KPI context documents. The Performance Detail page opens for the selected KPI.

4. Click **Information.** The Information Documents page opens and displays the list of information documents and their weighted values, as shown in Figure 4–24.

Figure 4–24   Sample Information Documents Page
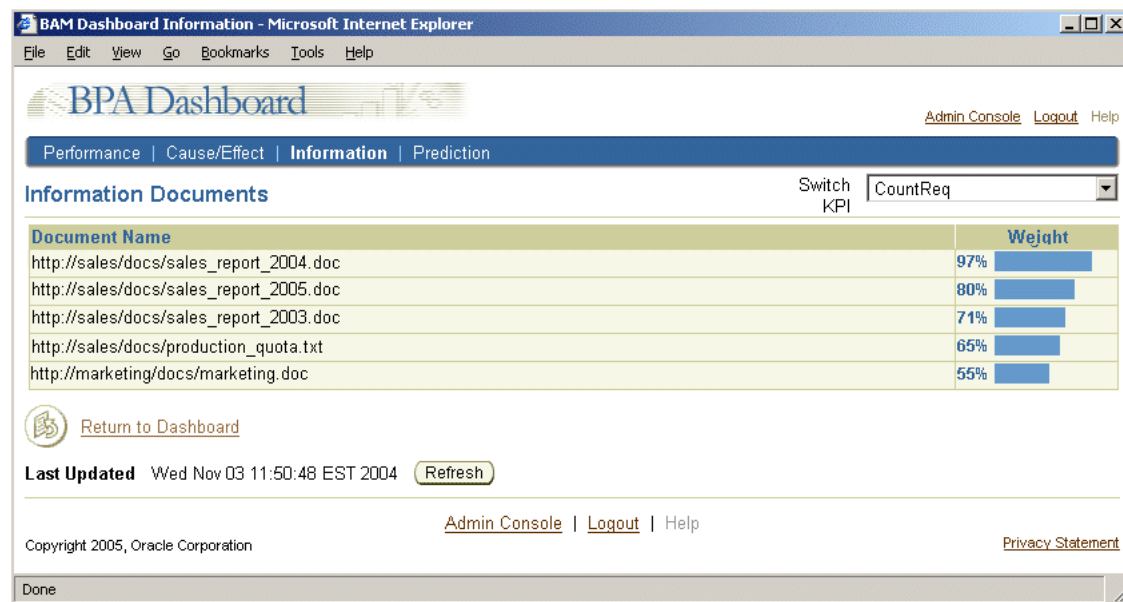
Viewing and Understanding KPI Predictions

The Dashboard Prediction Details page provides a prediction for a selected KPI value, based on related KPI values.

To view KPI predictions:

1. In the Dashboard, select a multiple pane view from the **Switch View** box.

2. In the multiple pane view, click a data point in any chart or table. A Performance Detail page opens.

3. Click **Prediction.**

4. In the **Switch KPI** box, select the KPI for which you want to view a prediction. The Prediction Detail page opens for the selected KPI.

5. In the **Analysis Time Grain** box, select the unit of time (day, month, quarter, year) over which you want Prediction details presented.

Figure 4–25 shows a sample Prediction Details page.

Figure 4–25   Sample Prediction Detail Page
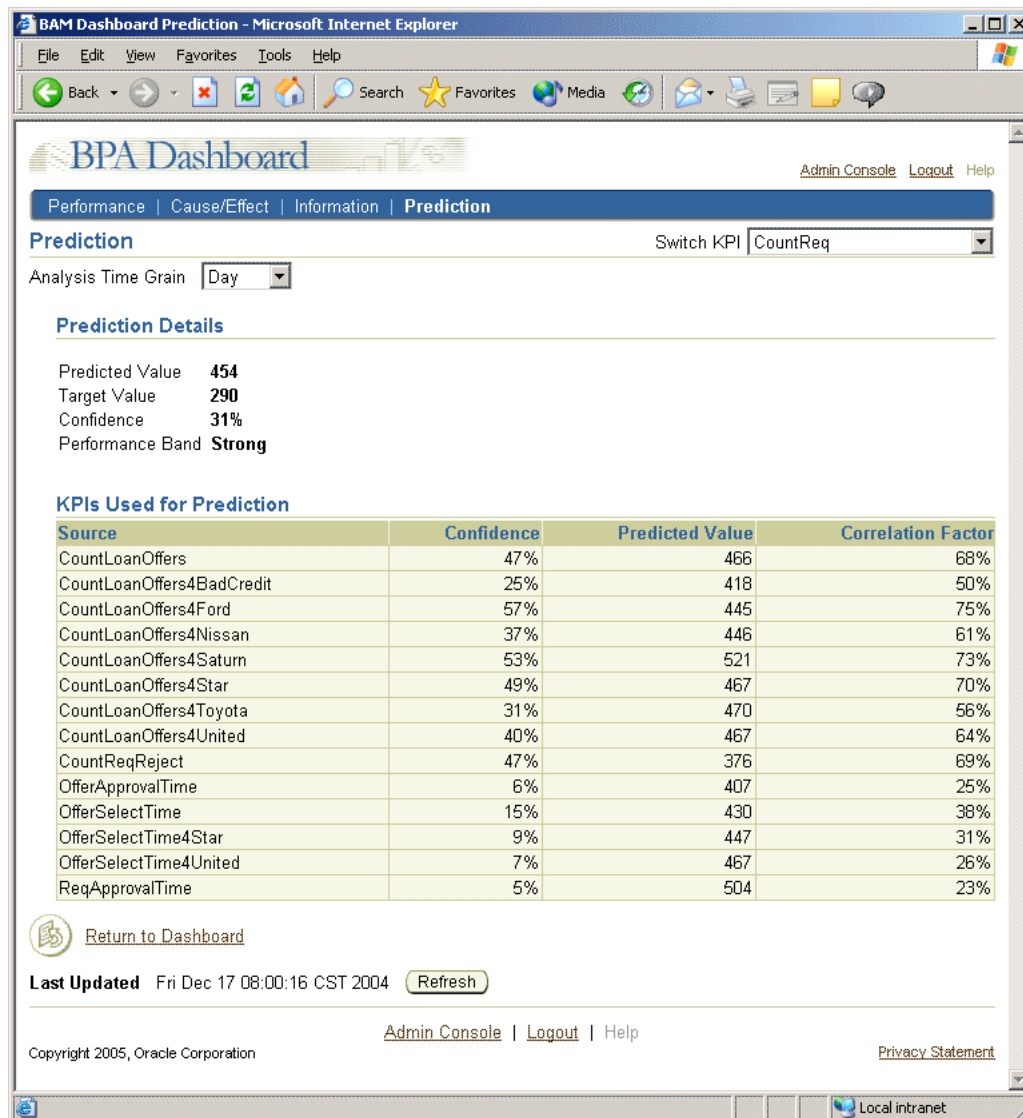
The Prediction Detail page identifies the selected KPI for which prediction details are being presented (in Figure 4–25, CountReq), and a table of KPIs that Oracle BPEL Process Analytics used in calculating the prediction. These KPIs are referred to as source KPIs.

The prediction details for the selected KPI, include the following information:

- The predicted value for the current time grain. In Figure 4–25, the time grain is the current day.

  Oracle BPEL Process Analytics arrives at the predicted value for the selected KPI by determining the sum of all of the predicting KPIs' predicted values, each weighted by its confidence value. Using the values in this example, the equation would be:

  ```
  (4715 x (9/(9+35))) + (4680 x (35/(9+35)))= 4687
  ```

- The target value, if a target value was specified. For the KPI in Figure 4–25, a target value of 290 was specified.

The target value is set by the Oracle BPEL Process Analytics administrator.

■ The confidence level in the predicted KPI value, if a target value was specified. For the KPI in Figure 4–25, the confidence level is 31%.

The confidence level is the average of the source KPIs' confidence values.

■ The performance band in which the predicted value falls.

Oracle BPEL Process Analytics determines the set of source KPIs to use to form the prediction based on how closely they are correlated with the KPI for which a prediction is being made. If the absolute value of the correlation of the KPI being predicted and another KPI is .8, then that KPI is included in the set of **source KPIs**.

The values for each of the source KPIs are determined, as follows:

■ The predicted value for each source KPI is determined by simple linear regression of its previous values.

■ The correlation factor for each source KPI is a number between -1 and 1, as determined by standard statistical equations. It indicates how a source KPI affects the selected KPI.

■ The confidence level for each source KPI is its correlation factor squared.

Determining the Exact Value for a Chart Element

You can determine the exact value for an element in a chart by placing the mouse cursor over the data of interest. In Figure 4–26, for example, a pop-up window displays the exact value, BMWX5: 89 (11.77%), represented by the portion of the pie chart that is presenting loan requests for the BMWX5.

Figure 4–26   Pop-up Window Displaying Chart Values
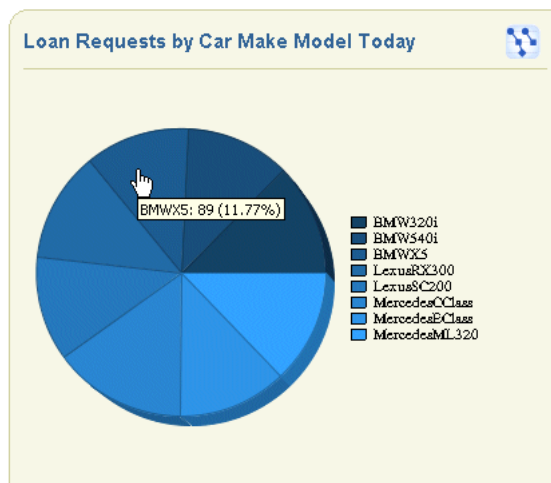
Viewing a Metric Table View of a Data Series

A metric table view of a data series, such as shown in Figure 4–27, presents metric values as they are generated, in real-time. Over time, this table can contain too many rows to present on a single page. Therefore, the table is limited to 10 rows, with **Next** and **Previous** buttons provided to allow you to scroll through the additional data. When you click the **Next** button, updates to the table are suspended until you click **Resume Updates.**

Data is presented in this table in real-time, over the time window you select from the **Switch Time** box. Suspending updates allows you to scroll back and forth among a given set of data until you are ready to see the next set of data. Otherwise, for example, suppose you were scrolling through the data. The first set of rows you viewed could be cleared from the table by the time you clicked the **Previous** button to return to that set of data. By suspending updates, Oracle BPEL Process Analytics prevents this from happening.

Figure 4–27   Metric Data Series

Figure 4–28   Sample Alert View Table

Viewing Alerts in Multiple Pane Views

Both KPI and Metric multiple pane views can include an Alert View table if the administrator chose to include it. The Alert View table specifies the alert name, alert type, and the time at which the alert was sent, as shown in Figure 4–28.

Figure 4–28   Sample Alert View Table

Click a value in the **Alert Name** column to view details about that alert. Figure 4–29 shows a sample Alert Detail page. Note that the list of values presented in the Alert Detail page may vary, depending on how the alert was defined and whether it is a KPI or metric alert.

Figure 4–29   Sample Alert Detail Page

The information in this page is described in the following table:

Page Element	Description
Owner	The account name of the user to whom the alert belongs.
Expression	The expression that describes the condition under which the alert is triggered.
Description	If specified, the description entered by the administrator when the trigger was defined.
Trigger Time	The time at which the alert was triggered.
Notification Time	The time at which the notification was sent, if a notification channel (also referred to as a delivery channel) was specified when the alert was defined.
Delivery Channel	The method (such as e-mail, phone, fax, and so on) by which a notification was sent, if a delivery channel was specified.
Time Grain	The frequency at which the alert will be fired. For example, if the time grain is Day, then it will fire once per day. This property applies to KPI alerts only.
Next Notification Time	The earliest time at which the next notification will be sent. This property applies to KPI alerts only.
Action	The URL of the Web service that is invoked when the alert is triggered.
Action Request	The URL of the Web service payload. (If the Web service takes an argument, this would be the URL for the `.xml` file containing the argument value.)

Page Element	Description
Action Response	The response from the Web service. A long Web service response may be truncated. If it is truncated, then the truncated text appears as a link. Click the link to view the full text.
Associated KPI or Metric	For a KPI alert, the name and value of the KPI that triggered the alert are presented as well as an icon that provides a link to the KPI Performance Detail page, such as shown in Figure 4–16.
	For a metric alert, the name of the metric that triggered the alert are presented, as well as the underlying composite event. The composite event is presented as a link. When you click the composite event name, the Composite Event Details page opens, such as shown in Figure 4–12.

5

Administering Oracle BPEL Process Analytics

This chapter provides information about administering and monitoring Oracle BPEL Process Analytics itself. It includes the following topics:

> **Note:** The directory paths shown in this chapter follow Microsoft Windows conventions (using backslashes (\)). If you are using Unix, modify the directory paths as required (using slashes (/)).

Accessing Oracle BPEL Process Analytics Documentation

Oracle BPEL Process Analytics documentation and help is included in the software installation. Accessing the documentation varies, depending on the operating system, as follows:

- On Microsoft Windows systems, follow these steps to view the documentation, where *OAS_Home_Name* is the name of the Oracle Application Server home in which you installed Oracle BPEL Process Analytics:

  From the desktop, click **Start**, then choose **Oracle – OAS_Home_Name,** then **Oracle BPEL Process Analytics,** and then click the name of the book that you want to open.

- On Unix systems, you can find the documentation in the following directory, where *OAS_HOME* is the directory specification for the Oracle Application Server home in which you installed Oracle BPEL Process Analytics:

  `OAS_HOME\integration\bam\doc`

The file containing the *Oracle BPEL Process Analytics User's Guide* is `bam_userguide.pdf`; the file containing the *Oracle BPEL Process Analytics Quick Start Guide* is `bam_quickstart.pdf`.

To view the help, open the Oracle BPEL Process Analytics Console and click the **Help** link on any Oracle BPEL Process Analytics console page.

> **Note:** The directory paths shown in this chapter follow Microsoft Windows conventions (using backslashes (\)). If you are using Unix, modify the directory paths as required (using slashes (/)).

Setting Up Oracle BPEL Process Analytics Samples

Oracle BPEL Process Analytics samples are included in the software installation. The *Oracle BPEL Process Analytics Quick Start Guide* has step-by-step instructions on using the LoanFlow demo.

The method for setting up the samples varies, depending on the operating system, as follows:

- On Microsoft Windows systems, follow these steps, where *OAS_Home_Name* is the name of the Oracle Application Server home in which you installed Oracle BPEL Process Analytics:

  1. If it is currently running, stop the Oracle BPEL Process Analytics Server.

     From the desktop **Start** menu, choose **All Programs,** then **Oracle - *OAS_Home_Name,*** then **Oracle BPEL Process Analytics,** and then click **Stop BPA Server.** A command window opens to show the progress of the operation.

  2. When the server stops, from the desktop **Start** menu, choose **All Programs**, then **Oracle - *OAS_Home_Name,*** then **Oracle BPEL Process Analytics,** and then click **Setup BPA Samples.** A command window opens and presents a choice of samples and tutorials to install.

  3. At the **Enter your choice** prompt, enter your choice, then press Return.

  4. You are prompted to confirm that you want to set up the samples (because doing so overwrites the existing Oracle BPEL Process Analytics configuration). Enter yes to proceed; enter no to stop.

  5. You are prompted to enter the database schema password that you specified when you installed Oracle BPEL Process Analytics. Enter the password and press Return.

  6. When prompted, start Oracle BPEL Process Analytics, as follows:

     From the desktop **Start** menu, select **All Programs**, then **Oracle - *OAS_Home_Name,*** then **Oracle BPEL Process Analytics,** and then click **Start BPA Server.**

- On Unix systems, follow these steps, where *OAS_HOME* is the directory specification for the Oracle Application Server home in which you installed Oracle BPEL Process Analytics:

  1. If it is currently running, stop Oracle BPEL Process Analytics, by setting the current directory to the Oracle Process Manager and Notification Server (OPMN) `bin` directory in the Oracle home where you installed Oracle BPEL Process Analytics and issuing the `opmnctl stopall` command:

     ```
     > opmnctl stopall
     ```

2. When the server stops, run `setupSamples.sh` in the following directory, where *ORACLE_HOME* is the directory specification for Oracle home into which you installed Oracle BPEL Process Analytics.

   ORACLE_HOME\integration\bam\bin\setupSamples.sh

3. At the **Enter your choice** prompt, enter your choice, then press Return.

4. You are prompted to confirm that you want to set up the samples (because doing so overwrites the existing Oracle BPEL Process Analytics configuration). Enter yes to proceed; enter no to stop.

5. You are prompted to enter the database schema password that you specified when you installed Oracle BPEL Process Analytics. Enter the password and press Return.

6. When prompted, start Oracle BPEL Process Analytics, as follows:

   From the Oracle Process Manager and Notification Server (OPMN) `bin` directory in the Oracle home where you installed Oracle BPEL Process Analytics, issue the `opmnctl startall` command.

   ```
   > opmnctl startall
   ```

User Management for Oracle BPEL Process Analytics Console Access

Oracle BPEL Process Analytics uses Oracle Application Server Java Authentication and Authorization Service (JAAS) Provider in Oracle Application Server Containers for J2EE (OC4J) to restrict access to the Oracle BPEL Process Analytics Console based on the user who is logging in and what JAAS provider groups or roles have been granted to that user.

By default, Oracle BPEL Process Analytics defines the following JAAS roles, and users for the XML-based provider type:

- Roles (for the XML-based provider type)

  - BAMUsers role

  - Administrator role - the BAMUsers role is granted to this role

  - Manager - the BAMUsers role is granted to this role

- Users

  - Administrator, with the password Administrator

  - Manager, with the password Manager

In addition, the following Oracle BPEL Process Analytics Console user accounts are defined:

- Administrator

- Manager

No default groups or users are created for the LDAP-based provider type.

The following topics describe and provide an example of how to configure a JAAS provider for an LDAP-based provider type and an XML-based provider type, respectively:

-

-

Configuring a JAAS Provider for an XML-Based Provider Type

The XML-based provider type is used for lightweight storage of information in Extensible Markup Language (XML) files.

The following steps describe how to set up an Oracle BPEL Process Analytics Console user account and then, in general terms, how to configure JAAS Provider for an XML-based provider type to provide an individual user access to the Oracle BPEL Process Analytics Console. This example creates an Oracle BPEL Process Analytics Console user account, BusinessAnalyst. It then creates a JAAS Provider role, BusinessAnalyst, and grants user, dbradford, that role.

For information about performing the tasks in these steps that are specific to the JAAS Provider, see *Oracle Application Server Containers for J2EE Security Guide.*

Wherever it occurs in the following steps, *OAS_HOME* is the directory specification for the Oracle Application Server home into which you installed Oracle BPEL Process Analytics.

1. Create BusinessAnalyst as a user in the Oracle BPEL Process Analytics Console:

   a. In the Admin console, click **User.** The User Management page opens.

   b. Click **Create.** The Create New User Wizard opens.

   c. In the **User Name** field, enter BusinessAnalyst.

   d. In the **Password** and **Confirm Password** fields, enter BusinessAnalyst.

   e. Do not check the **Admin** box.

   f. Click **Done.**

      Later, you can modify this user to add KPIs and metrics, and then configure the Dashboard for this user.

2. Use the command line JAZN Admintool to do the following:

   a. Set the current directory to the following location:

      ```
      OAS_HOME\j2ee\oc4j_bam
      ```

   b. Create the desired role or roles, for example BusinessAnalyst:

      ```
      java -jar ..\home\jazn.jar -addrole jazn.com BusinessAnalyst
      ```

   c. Grant the BAMUsers role to the role created in step 5b:

      ```
      java -jar ..\home\jazn.jar -grantrole BAMUsers jazn.com -role
      BusinessAnalyst
      ```

   d. Create a user and grant the role created in step 5b to that user. This example creates a user dbradford with a password of dbradford1 and then grants the BusinessAnalyst role to her:

      ```
      java -jar ..\home\jazn.jar -adduser jazn.com dbradford dbradford1
      java -jar ..\home\jazn.jar -grantrole BusinessAnalyst jazn.com dbradford
      ```

3. Stop the Oracle BPEL Process Analytics Server as described in "Stopping the Server on Microsoft Windows" on page 2-4 or "Stopping the Server on Unix" on page 2-4, depending on your system.

4. Start the Oracle BPEL Process Analytics Server as described in "Starting the Server and Opening the Console on Microsoft Windows Systems" on page 2-3 or "Starting

the Server and Opening the Console on Unix Systems" on page 2-4, depending on your system.

The user `dbradford` can now open and log in to the Oracle BPEL Process Analytics Console. She will have access to any Dashboard pages you assigned to user `BusinessAnalyst`. Note that when `dbradford` logs in, the Oracle BPEL Process Analytics Dashboard will display the text: Logged in as BusinessAnalyst. Also note, that you cannot log in directly as user `BusinessAnalyst`. To do so, you would have to create `BusinessAnalyst` as a user using the JAZN Admintool.

Configuring a JAAS Provider for an LDAP-Based Provider Type

The LDAP-based provider type is based on the Lightweight Directory Access Protocol (LDAP) for centralized storage of information in a directory.

The following steps describe how to set up an Oracle BPEL Process Analytics Console user account and then, in general terms, how to configure JAAS Provider for an LDAP-based provider type to provide an individual user access to the Oracle BPEL Process Analytics Console. This example creates an Oracle BPEL Process Analytics Console user account, LoanAnalyst. It then creates a JAAS Provider group, LoanAnalyst, and creates user, `istuart`, as a member of that group.

For detailed instructions on performing the tasks in these steps that are specific to the JAAS Provider, see *Oracle Identity Management Guide to Delegated Administration*.

Wherever it occurs in the following steps, *OAS_HOME* is the directory specification for the Oracle Application Server home into which you installed Oracle BPEL Process Analytics.

1. Update the `jazn.xml` file by following these steps:

   a. Open the `jazn.xml` files located in the following directory:

      `OAS_HOME\j2ee\oc4j_bam\config`

   b. Add the following lines to the `jazn.xml` file, replacing:

      – *ldaphost* with the system where Oracle Internet Directory server is running

      – *ldapport* with the port number on which the Oracle Internet Directory server is running

      – *default-realm-name* with the LDAP default realm name

      The default realm name is `us` if you are using Oracle Internet Directory (OID) in an Oracle Infrastructure environment.

      – *adminusername* with the admin user name used to log in to the Oracle Internet Directory database

      – *adminpassword* with the password for the admin account used to login to the Oracle Internet Directory database

      ```
      <jazn provider="LDAP" location="ldap://ldaphost:ldapport/"
      default-realm="default-realm-name">
         <property name="ldap.user" value="adminusername"/>
         <property name="ldap.password" value="!adminpassword"/>
      </jazn>
      ```

2. Update the `orion-application.xml` file by following these steps:

   a. Open the `orion-application.xml` file located in the following directory:

```
OAS_HOME\j2ee\oc4j_bam\appplication-deploysments\bam
```

   b. After the line that specifies the library path, add the following line, replacing:

- *ldaphost* with the system where Oracle Internet Directory server is running

- *ldapport* with the port number on which Oracle Internet Directory server is listening

- *default-realm-name* with the LDAP default realm name

```
<jazn provider="LDAP" location="ldap://ldaphost:ldapport/"
default-realm="default-realm-name" />
```

3. Open the following URL in a Web browser to login to Oracle Internet Directory Delegated Administration Services. In the URL specification, *host* is the system on which the Oracle HTTP server associated with Oracle Internet Directory Delegated Administration Services is running, and *port* specifies the port on which the Oracle HTTP server associated with Oracle Internet Directory Delegated Administration Services is listening:

```
http://host:port/oiddas
```

4. Using Oracle Internet Directory Delegated Administration Services, do the following:

   a. Create a group named BAMUsers to represent all Oracle BPEL Process Analytics users.

   b. Create a group named Administrator to represent the Oracle BPEL Process Analytics administrator, and assign BAMUsers to the Administrator group.

   c. Create a group named LoanAnalyst to represent a subset of Oracle BPEL Process Analytics Dashboard users, and assign BAMUsers to the LoanAnalyst group.

   d. Create a user named Administrator.

   e. Edit the Administrator group to add user Administrator as a member.

   f. Create a user named istuart.

   g. Edit the LoanAnalyst group to add user istuart as a member.

   h. Repeat steps f and g for any additional users that you want to be able to log in to the Oracle BPEL Process Analytics Console under the LoanAnalyst account.

5. If it is running, stop the Oracle BPEL Process Analytics Server as described in "Stopping the Server on Microsoft Windows" on page 2-4 or "Stopping the Server on Unix" on page 2-4, depending on your system.

6. Start the Oracle BPEL Process Analytics Server as described in "Starting the Server and Opening the Console on Microsoft Windows Systems" on page 2-3 or "Starting the Server and Opening the Console on Unix Systems" on page 2-4, depending on your system.

7. Create LoanAnalyst as a user in the Oracle BPEL Process Analytics Console:

   a. Log in to the Oracle BPEL Process Analytics Console using the Administrator account. (The password is Administrator, by default.)

   b. In the Admin console, click **User.** The User Management page opens.

   c. Click **Create.** The Create New User Wizard opens.

d. In the **User Name** field, enter LoanAnalyst. Note that the user name created in this step must match the group name created in step 4c.

e. In the **Password** and **Confirm Password** fields, enter LoanAnalyst.

f. Do not check the **Admin** box.

g. Click **Done.**

Later, you can modify this user to change the password, add KPIs and metrics, and then configure the Dashboard for this user.

The Administrator and the user `istuart` (as well as any other users you created and added to the LoanAnalyst group) can now open and log in to the Oracle BPEL Process Analytics Console. Those users will have access to any Dashboard pages that you assigned to user `LoanAnalyst`. Note that when `istuart` logs in, the Oracle BPEL Process Analytics Dashboard will display the text: Logged in as LoanAnalyst. Also note, that you cannot log in directly as `LoanAnalyst`. To do so, you would have to create `LoanAnalyst` as a user using Oracle Internet Directory Delegated Administration Services.

Changing the Administrator and Other User Passwords

Immediately after logging into the Console for the first time, the Administrator should change the default password for the Administrator account to ensure the security of the Oracle BPEL Process Analytics system and the information in the event sources to which it connects.

By default, the password for the `Administrator` account is `Administrator`.

To change the login password for a user when using the default XML-based provider type:

1. Set the current directory to the following location:

```
OAS_HOME\j2ee\oc4j_bam
```

2. Use the JAZN Admintool to change the password as shown in the following example, where the first occurrence of `Administrator` is the username, the second occurrence is the current password, and *newpassword* is the new password.

java -jar ..\home\jazn.jar -setpasswd jazn.com Administrator Administrator *newpassword*

3. Stop and Restart the Oracle BPEL Process Analytics Server as described in "Starting the Server and Opening the Console on Microsoft Windows Systems" on page 2-3 or "Stopping the Server on Unix" on page 2-4, depending on your system.

The next time the administrator logs into the Oracle BPEL Process Analytics Console, the new password must be used.

Using Oracle Enterprise Manager 10*g*

Oracle BPEL Process Analytics is integrated with Oracle Enterprise Manager 10*g* to allow you to stop, start, or restart the following components associated with Oracle BPEL Process Analytics:

- The Oracle Application Server Containers for J2EE (OC4J)

- The Oracle BPEL Process Analytics Server

- The Oracle BPEL Process Analytics monitor

In addition, it provides access to the port numbers in use by the event sources supported by Oracle BPEL Process Analytics.

The following topics describe how to access and use Oracle Enterprise Manager 10*g* with Oracle BPEL Process Analytics:

■   Accessing Oracle Enterprise Manager 10g Application Server Control on page 5-8

■   Starting, Stopping, or Restarting Components on page 5-9

■   Determining Port Numbers for Event Source Specifications and URLs on page 5-10

Accessing Oracle Enterprise Manager 10*g* Application Server Control

Oracle Enterprise Manager 10*g* is installed when you install Oracle Application Server. To access Oracle Enterprise Manager 10*g* Application Server Control:

1.   In a Web browser, enter the URL for the Oracle Enterprise Manager 10*g* Application Server Control that was presented at the end of the Oracle Application Server installation.

     This URL is also available in the `setupinfo.txt` file in the following location, where *OAS_HOME* is the directory specification for the Oracle Application Server home into which you installed Oracle BPEL Process Analytics:

     `OAS_HOME\install`

2.   After you enter the URL in a Web browser, a login window opens. Enter the user name and password for Oracle Application Server.

After you log in, your browser should look similar to that shown in Figure 5–1:

Figure 5–1   Sample Oracle Enterprise Manager 10g Control
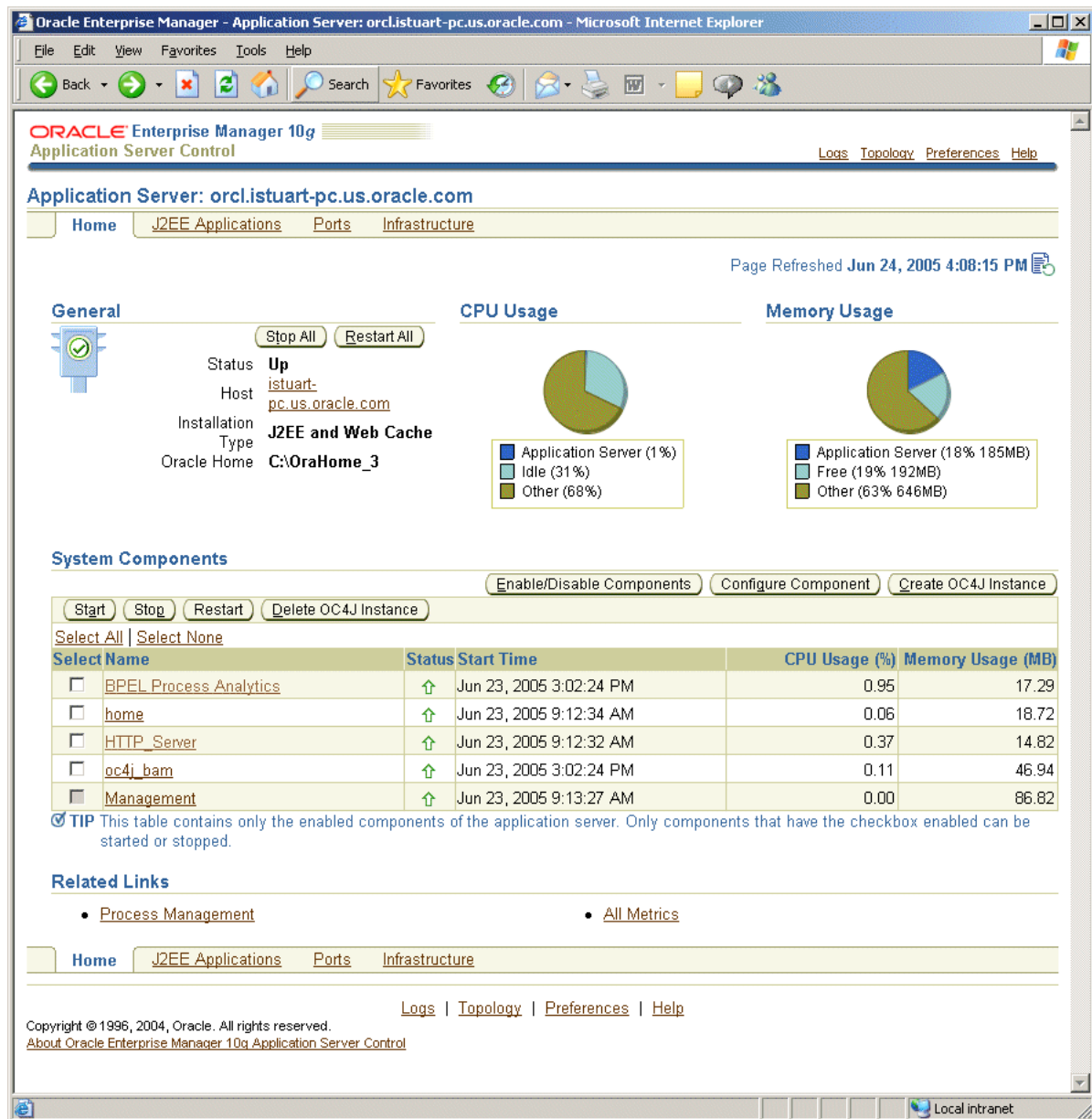
Starting, Stopping, or Restarting Components

To start, stop, or restart Oracle BPEL Process Analytics components:

1. Open Oracle Enterprise Manager 10*g* Application Server Control, as described in "Accessing Oracle Enterprise Manager 10g Application Server Control" on page 5-8.

2. Click **BPEL Process Analytics.** The browser refreshes and appears similar to Figure 5–2.

3. Click the **Start, Stop,** or **Restart** button associated with the desired component to perform the desired operation.

Alternatively, you can click the **Stop All, Restart All** or **Stop All** button to perform the operation on all of the Oracle BPEL Process Analytics components.

Figure 5–2   Sample Oracle Enterprise Manager - BPEL Process Analytics Page
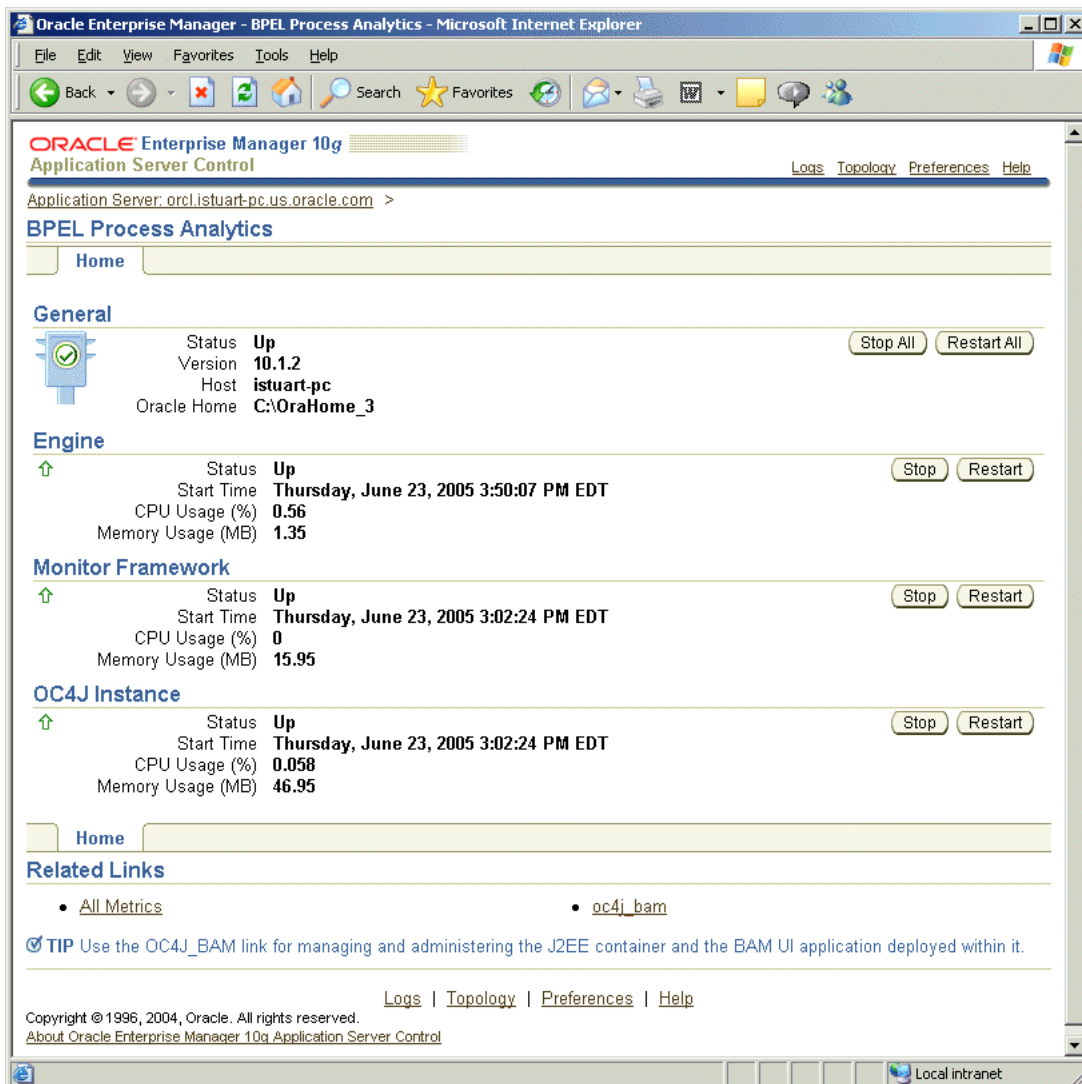
Determining Port Numbers for Event Source Specifications and URLs

When you specify event sources for Oracle BPEL Process Analytics or specify URLs related to Oracle BPEL Process Analytics, you are sometimes required to enter the port number of various event source components or the Oracle HTTP Server for Oracle BPEL Process Analytics. To determine what these values should be, follow these steps:

1. Open the Oracle Enterprise Manager 10*g* Application Server Control associated with the event source as described in "Accessing Oracle Enterprise Manager 10g Application Server Control" on page 5-8.

2. Click the **Ports** tab. The Ports page opens.

3. In the Ports page, search the Component and Type columns for the component and type of interest; the Port in Use column will specify the value of interest.

Figure 5–3 shows a sample Ports page.

Figure 5–3   Sample Ports Page

For example, suppose you are interested in using Oracle BPEL Process Analytics with an Oracle BPEL Process Manager event source that was installed in the Oracle Application Server middle tier. One of the values you must specify is the ORMI port. If you open Oracle Enterprise Manager 10*g* Application Server Control and click the Ports tab, you will find the following columns and values that correspond to Oracle BPEL Process Manager and the ORMI port, respectively:

- Component: OraBPEL

- Type: rmi

If you view the Port in Use column for this component and type, you see the required number for the ORMI port.

See "Oracle BPEL Process Manager Event Source Configuration Values" on page 2-11 for all of the values required to specify an Oracle BPEL Process Manager event source.

Locating Oracle BPEL Process Analytics Log Files

Oracle BPEL Process Analytics generates a number of log files that can be useful in troubleshooting problems that may arise. The format of each of these files is text. Table 5–1 provides the file name and description of the log files generated by Oracle BPEL Process Analytics. All of the log files are located in the following directory, where *OAS_HOME* is the directory specification for the Oracle Application Server home into which you installed Oracle BPEL Process Analytics:

OAS_HOME\integration\bam\log

Table 5–1   Oracle BPEL Process Analytics Log Files

File Name	Description
event.log	Logs output from the Oracle BPEL Process Analytics engine
monitor.log	Logs output from the Oracle BPEL Process Analytics monitor framework and monitors
ui.log	Logs output from the Oracle BPEL Process Analytics Console

Setting Log File, Dashboard, and Other Properties

Oracle BPEL Process Analytics allows you to set some properties for the Oracle BPEL Process Analytics engine, monitor, and console log files, as well as for the Oracle BPEL Process Analytics Dashboard.

The bam.properties file is a text file located in the following directory where *OAS_HOME* is directory specification for the Oracle Application Server home into which you installed Oracle BPEL Process Analytics:

OAS_HOME\integration\bam\config

The following tables describe properties that you can set by editing the file in any text editor. Oracle recommends that you do not adjust values for properties included in the file which are not described in one of the following tables:

- Log file and related properties are described in Table 5–2.

- Oracle BPEL Process Analytics Dashboard properties and properties that affect the speed at which the event simulator generates events are described in Table 5–3.

See "Starting, Stopping, or Restarting Components" on page 5-9 for information about how to restart the various components that must be restarted for a property value change to take effect.

Table 5–2   Log File and Related Properties

Property Name	Description	Component That Must Restart for Changed Values to Take Effect
LogLevel	Specifies the log level for the event.log file. Valid values are ERROR, WARNING, NOTIFICATION, TRACE, and DEBUG. The default value is NOTIFICATION.	Oracle BPEL Process Analytics Server
oracle.tip.bam.monitor.logLevel	Specifies the log level for the monitor.log file. Valid values are ERROR, WARNING, NOTIFICATION, TRACE, and DEBUG. The default value is NOTIFICATION.	Oracle BPEL Process Analytics monitor framework
oracle.tip.bam.ui.logLevel	Specifies the log level for the ui.log file. Valid values are ERROR, WARNING, NOTIFICATION, TRACE, and DEBUG. The default value is NOTIFICATION.	Oracle BPEL Process Analytics monitor framework
oracle.tip.bam.log.segmentSize	Specifies the size of the monitor.log file and the ui.log file, in bytes. By default, each is 100,000 bytes (100 KB).	Oracle BPEL Process Analytics OC4J and monitor framework
oracle.tip.bam.log.maximumSize	Specifies the maximum size of archive files (for the monitor.log and ui.log files) to keep on disk. By default, 500,000 bytes (500KB) are kept on disk.	Oracle BPEL Process Analytics OC4J and monitor framework
oracle.tip.bam.mf.eventListUpperLimit	The maximum number of events that can be buffered before Oracle BPEL Process Analytics processes them. A valid value is 1 or greater. By default, this value is 1k.	Oracle BPEL Process Analytics monitor framework

Table 5–3   Dashboard and Other Properties

Property Name	Description	Component that Must Restart for Changed Values to Take Effect
oracle.tip.bam.CETimeoutInterval	Specifies the interval at which composite event instances are evaluated to determine if any have timed out. A valid value is 5 or greater. By default, this value is 5 minutes.	Oracle BPEL Process Analytics monitor framework
oracle.tip.bam.ui.dashboard.chartBackground	Specifies the background color for the chart in the Real-Time Viewer. By default, this value is FFFFFF (white).	Oracle BPEL Process Analytics OC4J
oracle.tip.bam.ui.dashboard.chartHeight	Specifies the height of the charts displayed within the Dashboard multiple pane views. By default, this value is 300 pixels.	Oracle BPEL Process Analytics OC4J
oracle.tip.bam.ui.dashboard.refresh	The frequency at which the multiple pane views in the Dashboard are automatically refreshed. By default, this value is 120,000 milliseconds.	Oracle BPEL Process Analytics OC4J

Table 5–3 (Cont.) Dashboard and Other Properties

Property Name	Description	Component that Must Restart for Changed Values to Take Effect
oracle.tip.bam.ui.eventView.maxEvents	Specifies the maximum number of events to display in the Real-Time Viewer.By default, this value is 15000. If there are more events to display than are specified by this parameter, then an error message is returned. Reduce the time option in the Real-Time Viewer to a smaller value so that fewer events are included in the display. Reduce the value of this property if the Real-Time Viewer runs out of memory.	Oracle BPEL Process Analytics OC4J
oracle.tip.bam.ui.eventView.timeout	The frequency at which the Real-Time Viewer (in the Dashboard) is automatically refreshed. By default, this value is 60,000 milliseconds.	Oracle BPEL Process Analytics OC4J
oracle.tip.bam.monitor.test.LoadSeed	Specifies the time interval between generated events, in milliseconds. Increasing the value for this property decreases the speed at which the event simulator generates events.	Oracle BPEL Process Analytics Server
oracle.tip.bam.monitor.test.LoadThreads	Specifies the number of threads used to generate events for the event simulator. Increasing the value for this property increases the speed at which the vent simulator generates events.	Oracle BPEL Process Analytics Server

Running Multiple Instances of Oracle BPEL Process Analytics

It may be desirable to install and run multiple instances of Oracle BPEL Process Analytics. You might do this, for example, to enable multiple administrators to configure and manage event sources for a subset of Dashboard users within a single business. However, if you choose to do so, be aware that two or more instances of Oracle BPEL Process Analytics should not be configured to monitor the same event source.

If two or more instances of Oracle BPEL Process Analytics are monitoring the same event source, it is possible any given instance will capture all, some, or none of the events being published by that event source. This results in unpredictable behavior, including problems correlating the events and missing or incorrectly correlated events in the Real-Time Viewer.

Note however, that this restriction does not prevent multiple Dashboard users from viewing data from an event source monitored by a single instance of Oracle BPEL Process Analytics. Multiple Dashboard users can connect to the same instance of Oracle BPEL Process Analytics and view the events presented in the Real-Time Viewer without a problem (assuming they have privileges to access the Real-Time Viewer).

Archiving an Oracle BPEL Process Analytics Scenario

This topic describes how to archive an Oracle BPEL Process Analytics scenario (using Oracle BPEL Process Manager as the event source) running on one site so that you can

quickly set it up on another site. You may find this useful if you need to demonstrate at remote sites how Oracle BPEL Process Analytics can be used in your organization.

The following list provides step-by-step instructions for creating an archive on the local system. These instructions assume that Oracle Application Server middle tier, Oracle BPEL Process Manager, and Oracle BPEL Process Analytics are installed on both the local and remote systems.

1. Create a main directory, for example `MyScenario`, to serve as the main directory for the archive. (The remainder of these steps assume you have used the name `MyScenario`.)

2. Within the `MyScenario` directory, create the following two subdirectories: `config` and `sql`.

   The directory structure you create will appear similar to the structure of the following directory, which is created when you install Oracle BPEL Process Analytics (where *OAS_HOME* is the specification for the Oracle Application Server middle tier where Oracle BPEL Process Analytics is installed):

   ```
   OAS_HOME\integration\bam\test\tutorials\bpel
   ```

3. Copy the following files from the *OAS_HOME*`\integration\bam\config` directory to the `MyScenario\config` directory, where *OAS_HOME* is the specification for the Oracle Application Server middle tier where Oracle BPEL Process Analytics is installed:

   - Alerts.xml

   - BusinessProcessMapService.xml

   - CSFs.xml

   - DimensionColorMap.xml

   - eventSources.xml

   - PCBPELEventSources.xml

   - insite_insightdirectdefault.xml

   - UserManagerService.xml

   - UserPages.xml

4. Copy the `insightdirectdefault.xml` file from the *OAS_HOME*`\integration\bam\data\metadataServiceData` directory to the `MyScenario\config` directory, where *OAS_HOME* is the specification for the Oracle Application Server middle tier where Oracle BPEL Process Analytics is installed.

5. Write a script named `CreateTables.sql` that creates the fact and budget tables that are associated with each composite event currently defined in your scenario, and that creates the dimension tables for the dimensions that are associated with these composite events, if any.

   You can determine the names of the currently defined fact, budget, and dimension tables by connecting to the Oracle BPEL Process Analytics repository as the bam user and running SQL queries to find all tables that include the string `BAM_FACT`, `BAM_BUDGET`, or `BAM_DIM` as shown in the following example, where db1 is the server ID for the Oracle BPEL Process Analytics repository:

   ```
   SQL> connect bam/bam@db1
   SQL> SELECT TABLE_NAME FROM ALL_TABLES WHERE TABLE_NAME LIKE 'BAM_FACT%';
   TABLE_NAME
   ```

```
------------------------------
BAM_FACT_LOANFLOWDIM_T

SQL> SELECT TABLE_NAME FROM ALL_TABLES
     WHERE TABLE_NAME LIKE 'BAM_BUDGET%';
TABLE_NAME
------------------------------
BAM_BUDGET_LOANFLOWHDIM_T

SQL> SELECT TABLE_NAME FROM ALL_TABLES WHERE TABLE_NAME LIKE 'BAM_DIM%';
TABLE_NAME
------------------------------
BAM_DIM_CARMAKEMODEL_T
```

You can then determine the definition for each table using the SQL*Plus `DESCRIBE` statement with each table name, such as shown in the following examples:

```
SQL> describe BAM_FACT_COMPEVENTNAME_T;
Name                    Null?     Type
--------------------  --------  ---------------------------
TIMEID                NOT NULL   NUMBER
CARMAKEMODELID                   VARCHAR2(40)
COUNTREQ                         NUMBER
REQAPPROVALTIME                  NUMBER
COUNTLOANOFFERS                  NUMBER

SQL> describe BAM_DIM_DIMNAME_T;
 Name                   Null?     Type
 --------------------- --------  ---------------------------
 CARMAKEMODELID        NOT NULL VARCHAR2(40)
 CARMAKE                        VARCHAR2(20)
 CARMODEL                       VARCHAR2(20)
```

6. For each dimension in your current scenario, create a data (`.dat`) file that contains SQL `INSERT` statements to load data into the dimension table.

   For example, to load data into the `BAM_DIM_DIMNAME_T` table shown in the previous step, several SQL `INSERT` statements similar to the following would be included in the `BAM_DIM_DIMNAME_T.dat` file for that table:

   ```
   INSERT INTO BAM_DIM_CARMODEL_T (CARMODELID, CARMAKE, CARMODEL)VALUES ('BMW
   320i', 'BMW', '320i');
   ```

7. Create a file named `SetupSample.sql` that runs the `CreateTables.sql` script and loads all of the `DIMENSION_TABLE.dat` files.

8. Copy `CreateTables.sql`, `SetupSamples.sql`, and all the `DIMENSION_TABLE.dat` files to the `MyScenario\sql` directory.

9. Create a `.jar` or `.zip` file to hold the contents (and preserve the directory structure) of the `MyScenario` directory and its subdirectories.

   > **Note:** The directory structure you create will be similar to the

You have now completed archiving your scenario. To install the archive in another Oracle BPEL Process Analytics instance on a remote system, follow these steps:

1. Open the archive (`.jar` or `.zip` file) and copy all of the files in the `MyScenario\config` directory, except `insightdirectdefault.xml`, to the

OAS_HOME\integration\bam\config directory on the new system, where *OAS_HOME* is the specification for the Oracle Application Server middle tier where Oracle BPEL Process Analytics is installed

2. Copy insightdirectdefault.xml from MyScenario\config to the *OAS_HOME*\integration\bam\data\metadataServiceData directory on the new system.

3. Copy all the.sql files from the MyScenario\sql directory to a temporary directory on the new system.

4. Log in to SQL*Plus and connect to the Oracle BPEL Process Analytics repository as user bam with password bam.

5. Run SetupSamples.sql.

6

Using the Oracle BPEL Process Analytics Web Service

This chapter describes the Oracle BPEL Process Analytics Web service that enables applications, such as Oracle BPEL Process Manager, Microsoft Excel spreadsheets, portals, and any application that can invoke a Web service, to access Oracle BPEL Process Analytics KPIs and metrics. For example, an Oracle BPEL Process Manager process, can invoke the Oracle BPEL Process Analytics Web service to get LoanFlow metric and KPI values, and then alter the loan annual percentage rate (APR) dynamically, based on these values.

In addition, this chapter describes the Oracle BPEL Process Manager Web service wrapper, that enables you (or an application) to invoke the Oracle BPEL Process Analytics Web service from the Oracle BPEL Process Manager Console.

This chapter includes the following topics:

- Web Service End Point Definition on page 6-1
- Web Service Features on page 6-2
- Web Service Operations – Reference on page 6-2
- Invoking the Web Service from Oracle BPEL Process Manager on page 6-14

> **Note:** The directory paths shown in this chapter follow Microsoft Windows conventions (using backslashes (\)). If you are using Unix, modify the directory paths as required (using slashes (/)).

Web Service End Point Definition

When you install Oracle BPEL Process Analytics in the Oracle Application Server middle tier, the Web Service Design Language (WSDL) file, `BAMService.wsdl,` and the XML schema file, `bamws.xsd,` are installed. These files are placed in the following locations, respectively, where `system-name` is the system on which you installed Oracle Application Server, and `port` is the port number returned during the Oracle Application Server installation (at which you can access the Oracle Application Server main page).

WSDL and schema file location:

```
http://system-name:port/bam-ws/BAMService
http://system-name:port/bam-ws/bamws.xsd
```

The Web service endpoint:

```
http://mysystem.mycompany.com:1788/bam-ws/BAMService
```

Web Service Features

The Oracle BPEL Process Analytics Web service is a document/literal style Web service. As illustrated in Figure 6–1, it allows any Web Service client to access real-time metrics and KPIs computed by Oracle BPEL Process Analytics.

Figure 6–1   Web Service Request Response Model
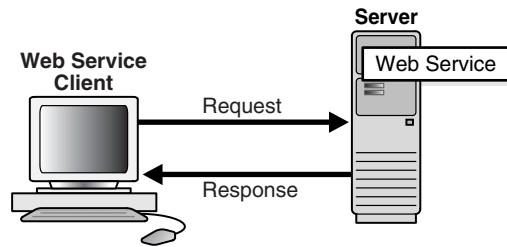
The Web Service supports the following operations:

- getEvents – Returns the events monitored by Oracle BPEL Process Analytics using a specified composite event within a given time window

- getKPIValue – Returns the value of a KPI that was defined through the Oracle BPEL Process Analytics Console

- getKPIDrilldownByDimensions – Returns the value of a KPI that was defined through the Oracle BPEL Process Analytics Console, filtered by a specified dimension value

- publishEvents – Publishes events to Oracle BPEL Process Analytics

These operations are described in detail in "Web Service Operations – Reference" on page 6-2.

Web Service Operations – Reference

The Oracle BPEL Process Analytics Web service supports four operations, as described in the following topics:

- getEvents Operation on page 6-2

- getKPIValue Operation on page 6-6

- getKPIDrilldownByDimensions Operation on page 6-9

- publishEvents Operation on page 6-12

getEvents Operation

Returns the events monitored by Oracle BPEL Process Analytics using a specified composite event within a given time window.

Before using the getEvents operation, you must create an event source and create a composite event that contains the events that you want returned by this operation.

See "Creating, Modifying, and Deleting Event Source Connections" on page 2-7 and "Modeling Composite Events and Composite Event Groups" on page 2-20 for more information about creating the prerequisite objects.

getEvents Request

/bam:GetEventsRequest/bam:UserName
A valid Oracle BPEL Process Analytics Administrator account name.

/bam:GetEventsRequest/bam:Password
A valid Oracle BPEL Process Analytics Administrator account password.

/bam:GetEventsRequest/bam:CompositeEvent/bam:Window
The time frame, *n*, in which events are to be returned, in seconds. The getEvents operation returns all of the events monitored by Oracle BPEL Process Analytics in last *n* seconds.

/bam:GetEventsRequest/bam:CompositeEvent/bam:Filter
Filters the type of events that are returned. Valid filter values are as follow:

- All

  Specifies that all of the events in the time frame be returned

- Pending

  Specifies that all of the events in the composite event that have not yet completed be returned

- Timeout

  Specifies that all of the events in the composite event that have timed out in the current time frame be returned

- Complete

  Specifies that all of the events in the composite event that have completed within the current time frame be returned

/bam:GetEventsRequest/bam:CompositeEvent/bam:Dimension
Specifies the dimension name by which to filter the returned events within a composite event.

/bam:GetEventsRequest/bam:CompositeEvent/@name
Specifies the name of the composite event for which you want events returned.

/bam:GetEventsRequest/bam:CompositeEvent/@details
Specifies whether or not event attributes are to be returned with the event. Set to this element to `true` to have event attributes returned with the event; otherwise set it to `false`.

getEvents Request Document Schema

```
<xsd:element name="GetEventsRequest"
             type="GetEventsRequestType"/>

  <xsd:complexType name="GetEventsRequestType">
    <xsd:sequence>
      <xsd:element username="UserName"
                      type="xsd:string"/>
      <xsd:element password="Password"
                      type="xsd:string"/>
     <xsd:element name="CompositeEvent"
                  type="EventRequestType"/>
    </xsd:sequence>
  </xsd:complexType>
```

```
<xsd:complexType name="EventRequestType">
  <xsd:sequence>
    <xsd:element name="Window" type="xsd:int"/>
    <xsd:element name="Filter" type="xsd:string"/>
    <xsd:element name="Dimension" type="xsd:string"/>
  </xsd:sequence>
  <xsd:attribute name="name" type="xsd:string" />
  <xsd:attribute name="details" type="xsd:boolean"/>
</xsd:complexType>
```

getEvents Response

/bam:GetEvents/bam:Event
Within the specified time frame, the array of events returned a composite event.

/bam:GetEventsResponse/bam:Event/Timestamp
The time stamp of the returned event in this format:

```
yyyy-MM-dd'T'HH:mm:ss[.SSS][Z | [+|-]hh:mm
```

The values for yyyy-MM-dd'T'HH:mm:ss and time zone are required. The milliseconds are optional. See the "Required Date Format" on page 2-19 for a description of the format values.

/bam:GetEventsResponse/bam:Event/@name
The name of the event within the returned composite event.

/bam:GetEventsResponse/bam:Event/@id
The ID of the event within the returned composite event.

/bam:GetEventsResponse/bam:Event/CompositeEvent/@name
The name of the composite event to which the returned event belongs.

/bam:GetEventsResponse/bam:Event/CompositeEvent/@id
The ID of the composite event to which the returned event belongs.

/bam:GetEventsResponse/bam:Event/bam:Attribute
The array of attributes returned for an event instance. The attributes are returned only if the details element is set to `true` in the input document.

/bam:GetEventsResponse/bam:Event/bam:Attribute/@Value
The value of an attribute within an event instance.

/bam:GetEventsResponse/bam:Event/bam:Attribute/@name
The name of an attribute within an event instance.

/bam:GetEventsResponse/bam:Event/bam:Attribute/@type
The data type of an attribute within an event instance. Oracle BPEL Process Analytics attributes can be any one of the following data types:

- Number

  Includes integers, doubles, floats, and so on

- Text

  Any string type

- Date

A date specification in the format:

```
yyyy-MM-dd'T'HH:mm:ss[.SSS] Z | [+|-]hh:mm
```

The values for yyyy-MM-dd'T'HH:mm:ss and Z are required. The milliseconds are optional. See the "Required Date Format" on page 2-19 for a description of the format values.

getEvents Response Document Schema

```
<xsd:element name="GetEventsResponse"
             type="GetEventsResponseType"/>

  <xsd:complexType name="GetEventsResponseType">
    <xsd:sequence>
      <xsd:element name="Event" type="EventDetailsType"
              maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="EventDetailsType">
    <xsd:sequence>
      <xsd:element name="Timestamp" type="xsd:dateTime"/>
      <xsd:element name="CompositeEvent"
                   type="CompositeEventType"/>
      <xsd:element name="Attribute"
                   type="AttributeType"
                   maxOccurs = "unbounded"
                   minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute name="name" type="xsd:string"/>
    <xsd:attribute name="id" type="xsd:string"/>
  </xsd:complexType>

  <xsd:complexType name="CompositeEventType">
    <xsd:attribute name="name" type="xsd:string"/>
    <xsd:attribute name="id" type="xsd:string"/>
  </xsd:complexType>


  <xsd:complexType name="AttributeType">
    <xsd:sequence>
      <xsd:element name="Value" type="xsd:string"/>
    </xsd:sequence>
      <xsd:attribute name="name" type="xsd:string"/>
      <xsd:attribute name="type" type="xsd:string"/>
  </xsd:complexType>
```

getEvents Operation Examples

The following are examples of request and response documents for the getEvents operation:

Example 1 getEvents Request Document

```
<GetEventsRequest xmlns="http://xmlns.oracle.com/integration/service/bam">
  <AuthenticateLogin>
    <UserName>Administrator</UserName>
    <Password>Administrator</Password>
  </AuthenticateLogin>
```

```
      <CompositeEvent name="LoanFlow" details="true">
        <Window>1000</Window>
        <Filter>All</Filter>
      </CompositeEvent>
    </GetEventsRequest>
```

Example 2 getEvents Response Document

```
<GetEventsResponse xmlns="http://xmlns.oracle.com/integration/service/bam">
 <Event name="LoanRequest" id="oc4j-1234">
   <CompositeEvent name="LoanFlow" id="LoanFlow-1234"/>
   <Timestamp>2004/10/16 16:31:43.453 PDT</Timestamp>
   <Attribute name="SSN" type="Text">123-456-7890</Attribute>
   <Attribute name="Amount" type="Number">20000.0</Attribute>
 </Event>
 <Event name="LoanOfferResponse" id="AD19A634C6399ED6167DF6CBE25351CF" >
    <CompositeEvent name="LoanFlow" id="LoanFlow-1234"/>
    <Timestamp>2004-10-25 13:04:16</Timestamp>
    <Attribute name="APR" type="number" >7</Attribute>
    <Attribute name="LoanApplicationId" type="number" >36168</Attribute>
    <Attribute name="LoanApproved" type="text" >true</Attribute>
    <Attribute name="ProviderName" type="text" >StarLoan Provider</Attribute>
 </Event>
</GetEventsResponse>
```

getKPIValue Operation

Returns the value of a key performance indicator (KPI) that was configured in the Oracle BPEL Process Analytics Console.

Before using the getKPIValue operation, you must create an event source, create a composite event, and create the KPI for which you want to get values.

 See "Creating, Modifying, and Deleting Event Source Connections" on page 2-7, and "Modeling Composite Events and Composite Event Groups" on page 2-20, and "Modeling KPIs and Related Objects" on page 2-27 for more information about creating the prerequisite objects.

getKPIValue Request

/bam:GetEventsRequest/bam:UserName
A valid Oracle BPEL Process Analytics Administrator account name.

/bam:GetEventsRequest/bam:Password
A valid Oracle BPEL Process Analytics Administrator account password.

/bam:GetKPIValueRequest/bam:KPI/bam:TimeGrain
The time grain over which the KPI value is computed. Valid values are:

- Day
- Month
- Quarter
- Year

/bam:GetKPIValueRequest/bam:KPI/bam:Periods
The number of time periods for which the KPI value is to be returned. The values are returned beginning with the current period and working backward. The length of time required to complete this operation corresponds to the number of time periods requested.

/bam:GetKPIValueRequest/bam:KPI/@name
The name of the KPI for which you want values to be returned.

/bam:GetKPIValueRequest/bam:KPI/@CompositeEvent
The composite event on which the KPI to be returned is defined.

/bam:GetKPIValueRequest/bam:KPI/bam:Dimension/@name
The name of the dimension with which to filter the KPI value. If no dimension name is specified, then the KPI value for all of the dimension values is returned.

/bam:GetKPIValueRequest/bam:KPI/bam:Dimension/Value
The value for the dimension with which to filter the KPI value. This element is optional.

getKPIValue Request

```
<xsd:element name="GetKPIValueRequest"
             type="GetKPIValueRequestType"/>
<xsd:complexType name="GetKPIValueRequestType">
   <xsd:sequence>
     <xsd:element username="UserName"
                      type="xsd:string"/>
     <xsd:element password="Password"
                      type="xsd:string"/>
     <xsd:element name="KPI"
                   type="KPIRequestType"/>
   </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="KPIRequestType">
  <xsd:sequence>
    <xsd:element name="TimeGrain" type="xsd:string" />
    <xsd:element name="Periods" type="xsd:int" />
    <xsd:element name="Dimension" type="DimensionType"/>
  </xsd:sequence>
  <xsd:attribute name="name" type="xsd:string" />
  <xsd:attribute name="CompositeEvent" type="xsd:string" />
</xsd:complexType>

<xsd:complexType name="DimensionType">
    <xsd:sequence>
      <xsd:element name="Value" type="xsd:string" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute name="name" type="xsd:string"/>
</xsd:complexType>
```

getKPIValue Response

/bam:GetKPIValueResponse/bam:KPI/@name
The name of the KPI for which you want values returned.

/bam:GetKPIValueResponse/bam:KPI/bam:KPIValue
An array of KPI values. There is one array element for each of the time periods
requested.

/bam:GetKPIValueResponse/bam:KPI/bam:KPIValue/bam:For
The duration over which the KPI value is computed. Valid duration values are:

- Day

- Month

- Quarter

- Year

/bam:GetKPIValueResponse/bam:KPI/bam:KPIValue/bam:Value
The value of the KPI for the time duration specified.

/bam:GetKPIValueResponse/bam:KPI/bam:KPIValue/bam:Target
If specified, the target value specified for the KPI.

/bam:GetKPIValueResponse/bam:KPI/bam:KPIValue/bam:PerformanceBand
The KPI status with respect to the performance band.

/bam:GetKPIValueResponse/bam:KPI/bam:KPIValue/bam:Variance
The variance of the KPI value, with respect to the target value (if a target value was
specified).

/bam:GetKPIValueResponse/bam:KPI/bam:KPIValue/bam:Trend
The upward or downward trend of the KPI, with respect to previous KPI values.

getKPIValue Response Document Schema

```
<xsd:element name="GetKPIValueResponse"
              type="GetKPIValueResponseType"/>

<xsd:complexType name="GetKPIValueResponseType">
  <xsd:sequence>
    <xsd:element name="KPI" type="KPIType"/>
  </xsd:sequence>
  <xsd:attribute name="name" type="xsd:string" />
</xsd:complexType>

<xsd:complexType name="KPIType">
  <xsd:sequence>
    <xsd:element name="KPIValue"
                 type="KPIValueType"
           maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="KPIValueType">
  <xsd:sequence>
    <xsd:element name="For" type="xsd:string"/>
    <xsd:element name="Value" type="xsd:int"/>
    <xsd:element name="Target" type="xsd:int"/>
    <xsd:element name="PerformanceBand" type="xsd:string" />
    <xsd:element name="Variance" type="xsd:int" />
    <xsd:element name="Trend" type="xsd:string" />
  </xsd:sequence>
</xsd:complexType>
```

getKPIValue Operation Examples

The following are examples of request and response documents for the getKPIValue operation:

Example 1 getKPIValue Request Document

```
<GetKPIValueRequest xmlns="http://xmlns.oracle.com/integration/service/bam >
  <AuthenticateLogin>
    <UserName>Administrator</UserName>
    <Password>Administrator</Password>
  </AuthenticateLogin>
  <KPI name="POCount" CompositeEvent="PurchaseOrders">
    <TimeGrain>Day</TimeGrain>
    <Periods>2</Periods>
    <Dimension name="Product">Laptop</Dimension>
  </KPI>
</GetKPIValueRequest>
```

Example 2 getKPIValue Response Document

```
<GetKPIValueResponse>
  <KPI name="POCount">
    <KPIValue>
      <For TimeGrain="Day">2004-08-11</For>
      <Value>25</Value>
      <Target>30</Target>
      <PerformanceBand>normal_performer</PerformanceBand>
      <Variance>30</Variance>
      <Trend>Up</Trend>
    </KPIValue>
    <KPIValue>
      <For TimeGrain="Day">2004-08-10</For>
      <Value>30</Value>
      <Target>50</Target>
      <PerformanceBand>under_performer</PerformanceBand>
      <Variance>30</Variance>
      <Trend>Down</Trend>
    </KPIValue>
  </KPI>
</GetKPIValueResponse>
```

getKPIDrilldownByDimensions Operation

Returns the value of a key performance indicator (KPI) that was configured in the Oracle BPEL Process Analytics Console, filtered by a specified dimension name and value.

Before using the getKPIDrilldownByDimensions operation, you must create an event source, create a composite event, and create the KPI for which you want to get values.

See "Creating, Modifying, and Deleting Event Source Connections" on page 2-7, and "Modeling Composite Events and Composite Event Groups" on page 2-20, and "Modeling KPIs and Related Objects" on page 2-27 for more information about creating the prerequisite objects.

getKPIDrilldownByDimensions Request

/bam:GetEventsRequest/bam:UserName
A valid Oracle BPEL Process Analytics Administrator account name.

/bam:GetEventsRequest/bam:Password
A valid Oracle BPEL Process Analytics Administrator account password.

/bam:GetKPIValueRequest/bam:KPI/bam:TimeGrain
The time grain over which the KPI value is computed. Valid values are:

- Day
- Month
- Quarter
- Year

/bam:GetKPIValueRequest/bam:KPI/bam:Periods
The number of time periods for which the KPI value is to be returned. The values are returned beginning with the current period and working backward. The length of time required to complete this operation corresponds to the number of time periods requested.

/bam:GetKPIValueRequest/bam:KPI/@name
The name of the KPI for which you want values to be returned.

/bam:GetKPIValueRequest/bam:KPI/@CompositeEvent
The composite event on which the KPI to be returned is defined.

/bam:GetKPIValueRequest/bam:KPI/bam:Dimension/Value
The value for the dimension by which to filter the KPI value.

/bam:GetKPIValueRequest/bam:KPI/bam:Dimension/@name
The name of the dimension by which to filter the KPI value. If no dimension name is specified, then the KPI value for all of the dimension values is returned.

getKPIDrilldownByDimensions Request Document Schema

```
<xsd:element name="GetKPIValueRequest"
             type="GetKPIValueRequestType"/>

<xsd:complexType name="GetKPIValueRequestType">
  <xsd:sequence>
    <xsd:element username="UserName"
                      type="xsd:string"/>
    <xsd:element password="Password"
                      type="xsd:string"/>
    <xsd:element name="KPI"
                 type="KPIRequestType"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="KPIRequestType">
  <xsd:sequence>
    <xsd:element name="TimeGrain" type="xsd:string" />
    <xsd:element name="Periods" type="xsd:int" />
    <xsd:element name="Dimension" type="DimensionType"/>
  </xsd:sequence>
  <xsd:attribute name="name" type="xsd:string" />
```

```
      <xsd:attribute name="CompositeEvent" type="xsd:string" />
</xsd:complexType>

<xsd:complexType name="DimensionType">
  <xsd:sequence>
    <xsd:element name="Value" type="xsd:string" minOccurs="0"/>
  </xsd:sequence>
  <xsd:attribute name="name" type="xsd:string"/>
</xsd:complexType>
```

getKPIDrilldownByDimensions Response

/bam:GetKPIDrilldownByDimensions/bam:KPIResponse/@KPI
The name of the KPI.

/bam:GetKPIDrilldownByDimensions/bam:KPIResponse@Dimension
The name of the dimension by which returned KPI values are filtered.

/bam:GetKPIDrilldownByDimensions/bam:KPIResponse/@Constraint
The value of the constraint by which returned KPI values are filtered.

/bam:GetKPIDrilldownByDimensions/bam:Period/bam:Count
The number of periods over which the KPI value is calculated.

/bam:GetKPIDrilldownByDimensions/bam:Period/bam:Performance
The performance band in which the KPI falls.

getKPIDrilldownByDimensions Response Document Schema

```
<xsd:element name="GetKPIDrilldownByDimensions"
              type="GetKPIDrilldownByDimensionsType"/>

<xsd:complexType name="GetKPIDrilldownByDimensionsType">
  <xsd:sequence>
    <xsd:element name="KPIResponse" type="KPIResponseType"
                                  maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="KPIResponseType">
  <xsd:sequence>
    <xsd:element name="Period" type ="PeriodType"
            maxOccurs="unbounded" />
  </xsd:sequence>
  <xsd:attribute name="KPI"  type="xsd:string"/>
  <xsd:attribute name="Dimension" type="xsd:string"/>
  <xsd:attribute name="Constraint"  type="xsd:string"/>
</xsd:complexType>

<xsd:complexType name = "PeriodType">
  <xsd:sequence>
    <xsd:element name="Count" type="xsd:int"/>
    <xsd:element name="Performance" type="xsd:string"/>
  </xsd:sequence>
</xsd:complexType>
```

getKPIDrilldownByDimensions Operation Examples

The following are examples of request and response documents for the getKPIDrilldownByDimensions operation:

Example 1 getKPIDrilldownByDimensions Request Document Schema

```
<GetKPIValueRequest>
   <AuthenticateLogin>
      <UserName>Administrator</UserName>
      <Password>Administrator</Password>
   </AuthenticateLogin>
   <KPI name="POCount" CompositeEvent="PurchaseOrders">
      <TimeGrain>Day</TimeGrain>
      <Periods>2</Periods>
      <Dimension name="CarModel">All</Dimension>
   </KPI>
</GetKPIValueRequest>
```

Example 2 getKPIDrilldownByDimensions Response Document Schema

```
<GetKPIDrillDownResponse>
  <KPIResponse KPI="LoanFlow.CountReq" Dimension="CarModel" Constraint="Ford">
    <Period Date="Mon Feb 07 00:00:00 PST 2005">
       <Count>34</Count>
       <Performance>strong_performer</Performance>
    </Period>
  </KPIResponse>
  <KPIResponse KPI="LoanFlow.CountReq" Dimension="CarModel"
 Constraint="Toyota">
    <Period Date="Mon Feb 07 00:00:00 PST 2005">
       <Count>40</Count>
       <Performance>strong_performer</Performance>
    </Period>
  </KPIResponse>
  <KPIResponse KPI="LoanFlow.CountReq" Dimension="CarModel" Constraint="Nissan">
    <Period Date="Mon Feb 07 00:00:00 PST 2005">
       <Count>38</Count>
       <Performance>strong_performer</Performance>
    </Period>
  </KPIResponse>
  <KPIResponse KPI="LoanFlow.CountReq" Dimension="CarModel" Constraint="Saturn">
    <Period Date="Mon Feb 07 00:00:00 PST 2005">
       <Count>45</Count>
       <Performance>strong_performer</Performance>
    </Period>
  </KPIResponse>
</GetKPIDrillDownResponse>
```

publishEvents Operation

Publishes events to Oracle BPEL Process Analytics. There is no response schema for this operation.

Before using the publishEvents operation, you must create a generic event source and create a composite event that contains the events that you to publish using this operation.

See "Generic Event Source Configuration Values" on page 2-13 and "Modeling Composite Events and Composite Event Groups" on page 2-20 for more information about creating the prerequisite objects.

publishEvents Request

/bam:publishEvents/bam:PublishEventsRequest/bam:UserName
A valid Oracle BPEL Process Analytics Administrator account name.

/bam:publishEvents/bam:PublishEventsRequest/bam:Password
A valid Oracle BPEL Process Analytics Administrator account password.

/bam:publishEvents/bam:bamEvent/bam:PublishEventsRequest/bam:payload
The published event data. The data must use the format of the event definition .xsd
file that Oracle BPEL Process Analytics loaded when the composite event was created.
The composite event must be created on a generic event source.

The .xsd files that Oracle BPEL Process Analytics loads are located in the following
directory, where *OAS_HOME* is the directory specification for the Oracle Application
Server home into which you installed Oracle BPEL Process Analytics:

OAS_HOME/integration/bam/config/generic

> **Note:** Do not alter any .xsd file in the *OAS_
> HOME*/integration/bam/config/generic directory. Doing
> so will result in Oracle BPEL Process Analytics being unable to
> monitor events defined in the altered .xsd file.

/bam:publishEvents/bam:PublishEventsRequest/bam:header/bam:timestamp
The time stamp at which the published event occurred.

See the "Required Date Format" on page 2-19 for more information.

/bam:publishEvents/bam:PublishEventsRequest/bam:header/bam:timezone
The time zone in which the published event occurred.

See the "Required Date Format" on page 2-19 for more information.

/bam:publishEvents/bam:PublishEventsRequest/bam:header/bam:type
The name of the event that you want to publish. This is one of the event names that
was specified when the composite event was created on the generic event source.

publishEvents Request Document Schema

```
<xsd:element name="PublishEventsRequest"
             type="PublishEventsRequestType" />

<xsd:complexType name="PublishEventsRequestType">
  <xsd:sequence>
    <xsd:element name="UserName" type="xsd:string"/>
    <xsd:element name="Password" type="xsd:string"/>
    <xsd:element name="bamEvent" type="bamEventType"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="bamEventType">
  <xsd:sequence>
    <xsd:element name="header" type="headerType"/>
    <xsd:element name="payload"  type="payloadType"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="headerType">
```

```
<xsd:sequence>
  <xsd:element name="id"  type="xsd:int" />
  <xsd:element name="timestamp" type="xsd:dateTime"/>
  <xsd:element name="timezone" type="xsd:string"/>
  <xsd:element name="type" type="xsd:string"/>
</xsd:sequence>
</xsd:complexType>

<xsd:complexType name="payloadType">
  <!--event-specific schema -->
</xsd:complexType>
```

publishEvents Example

The following is an example of a response document for the publishEvents operation:

```
<PublishEventsRequest xmlns=""http://xmlns.oracle.com/integration/services/bam">
   <AuthenticateLogin>
     <UserName>Administrator</UserName>
     <Password>Administrator</Password>
   </AuthenticateLogin>
     <bamEvent>
       <header>
        <timestamp>2004-10-16T16:31:43.453</timestamp>
        <timezone>-07:00</timezone>
        <type>Employee>
        <eventsourcename>MyEventSource</eventsourcename>
       </header>
       <payload>
         <ROWSET>
          <ROW num="1">
            <EMPNO>7369</EMPNO>
            <ENAME>SMITH</ENAME>
            <JOB>CLERK</JOB>
            <MGR>7902</MGR>
            <HIREDATE>1980-10-17T00:00:00</HIREDATE>
            <SAL>800</SAL>
            <COMM>0</COMM>
            <DEPTNO>20</DEPTNO>
          </ROW>
         </ROWSET>
       </paylload>
     </bamEvent>
   </PublishEventsRequest>
```

Invoking the Web Service from Oracle BPEL Process Manager

A sample BPEL process, BAMServiceWrapper, is provided that enables you to invoke the Oracle BPEL Process Analytics Web Service from Oracle BPEL Process Manager. The following topics provide step-by-step instructions:

- Setting Up and Deploying the BAMServiceWrapper Process on page 6-14
- Running the BAMServiceWrapper Process on page 6-16

Setting Up and Deploying the BAMServiceWrapper Process

The files (bpel.xml and BAMServiceWrapper.wsdl) required for invoking BAMServiceWrapper are located in the following directory, where *OAS_HOME* is the

directory specification for the Oracle Application Server home into which you installed Oracle BPEL Process Analytics:

OAS_HOME\integration\bam\test\tutorials\webservices\BAMServiceWrapper

The following steps provide an example that describes how to set up and deploy BAMServiceWrapper. These steps assume that you have specified Oracle BPEL Process Manager as an event source and have created some KPIs.

1. If Oracle BPEL Process Manager and Oracle BPEL Process Analytics are not installed on the same system, copy all of the files in the BAMServiceWrapper directory from the Oracle BPEL Process Analytics system to a directory on the Oracle BPEL Process Manager system.

2. On the Oracle BPEL Process Manager system, open the `bpel.xml` file in a text editor, and change all occurrences of the `wsdlLocation` value to the system where you have installed Oracle BPEL Process Analytics.

   For example, if your system is `mysys.myco.com:1788,` the edited file should appear as follows:

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<BPELSuitcase>
  <BPELProcess id="BAMServiceWrapper" src="BAMServiceWrapper.bpel">
    <partnerLinkBindings>
        <partnerLinkBinding name="client">
            <property name="wsdlLocation">BAMServiceWrapper.wsdl</property>
        </partnerLinkBinding>
        <partnerLinkBinding name="BAMAuthentication">
            <property
name="wsdlLocation">http://mysys.myco.com:1788/bam-ws/BAMService.wsdl</property
>
        </partnerLinkBinding>
        <partnerLinkBinding name="BAMGetKPI">
            <property
name="wsdlLocation">http://mysys.myco.com:1788/bam-ws/BAMService.wsdl</property
>
        </partnerLinkBinding>
        <partnerLinkBinding name="BAMGetEvents">
            <property
name="wsdlLocation">http://mysys.myco.com:1788/bam-ws/BAMService.wsdl</property
>
        </partnerLinkBinding>
        <partnerLinkBinding name="BAMPublishEvents">
            <property
name="wsdlLocation">http://mysys.myco.com:1788/bam-ws/BAMService.wsdl</property
>
        </partnerLinkBinding>
    </partnerLinkBindings>
  </BPELProcess>
</BPELSuitcase>
```

3. On the Oracle BPEL Process Manager system, open the `BAMServiceWrapper.wsdl` file in a text editor and specify the system where you installed Oracle BPEL Process Analytics as the schema location in the **Type Definitions** section of the file.

   For example, if your system is `mysystem.myco.com:1788,` the Type Definitions section should appear as follows:

```
<!-- ====================== Type Definitions ===========================-->
  <types>
```

```
        <schema xmlns="http://www.w3.org/2001/XMLSchema">
         <import namespace="http://xmlns.oracle.com/integration/services/bam"
          schemaLocation="http://mysys.myco.com:1788/bam-ws/bamws.xsd"/>
        </schema>
      </types>
```

4. If it is not already started, start the Oracle BPEL Process Analytics Server.

5. Start the Oracle BPEL Process Manager Server.

6. Deploy the Oracle BPEL Process Analytics Service Wrapper, by opening a command window on your system and running the obant command in the BAMServiceWrapper directory on the Oracle BPEL Process Manager system. Your path must be set to the following directory:

    BPEL_HOME\integration\orabpel\bin

> **Note:** The directory paths shown in this topic follow Microsoft Windows conventions (using backslashes (\)). If you are using Unix, modify the directory paths as required (using slashes (/)).

Running the BAMServiceWrapper Process

This section provides step-by-step instructions on running the BAMServiceWrapper process. It is assumed that you have already followed the steps described in "Setting Up and Deploying the BAMServiceWrapper Process" on page 6-14.

1. Open the Oracle BPEL Process Manager Console, and log in to the default domain.

2. On the **Dashboard** tab, click **BAMServiceWrapper.** The **BPEL Processes** tab displays.

3. Select **getKPIValue**, then click **select the operation.**

4. Enter values as described in the following table, and then click **Post XML Message.** Figure 6–2 shows the values you might enter if you previously defined a composite event named LoanFlow, for which you created a KPI named CountLoanOffers. Data is returned, similar to that shown in Figure 6–3.

Page Element	Description
Username	The Oracle BPEL Process Analytics Administrator account user name.
Password	The Oracle BPEL Process Analytics Administrator account password.
KPI name	The name of the KPI for which you want to view values.
CompositeEvent	The name of the composite event on which the named KPI is defined.
TimeGrain	The time grain over which you want the KPI value computed.Valid values are Day, Month, Quarter, and Year.
Periods	The number of time periods for which you want the KPI value to be returned.
Dimension name	The name of the dimension with which you want to filter the returned KPI values. This field is optional.

Page Element	Description
Dimension value	If you specify a dimension name, the dimension value with which you want to filter the returned KPI values.

5. Click **Audit Instance** at the bottom of the page.

The **Instances** tab displays, such as shown in Figure 6–4.

6. Click the **More** link to see details about the instance.

Figure 6–2   Getting KPI Values

Figure 6–3   Value Returned

Figure 6–4   Instances Tab

7

Monitoring Events from Custom Applications

This chapter describes the steps required to publish and monitor events from applications for which Oracle BPEL Process Analytics does not provide specified support.

Oracle BPEL Process Analytics supports publishing events from the following types of custom applications:

- Oracle database applications that use PL/SQL packages installed with Oracle BPEL Process Analytics to publish the events to the preconfigured OJMS queue, `bam_genericmonq`.

- Oracle database applications that publish events to a user-specified OJMS queue.

- Java 2 Platform, Enterprise Edition (J2EE) applications that use JMS application programming interfaces (APIs) to publish events to an OJMS or user-specified JMS queue.

- Events generated by the Oracle BPEL Process Analytics Web Service published to an OJMS or a user-specified JMS queue.

This chapter includes the following topics:

- Basic Steps for Publishing and Monitoring Events from Custom Applications on page 7-1
- Publishing Events for Oracle BPEL Process Analytics on page 7-2
- Example of Monitoring an Oracle Database as a Generic Event Source on page 7-5

> **Note:** The directory paths shown in this chapter follow Microsoft Windows conventions (using backslashes (\)). If you are using Unix, modify the directory paths as required (using slashes (/)).

Basic Steps for Publishing and Monitoring Events from Custom Applications

You monitor events from a custom application by configuring and monitoring them as a generic event source, and then publishing events from those applications to one of queues described in the previous section.

To do so you must define the events that you want to monitor using `.xsd` or `.dtd` files, and then ensure that those events are published to a queue to which Oracle BPEL Process Analytics has access.

The steps for monitoring events from a custom application are as follows:

1. Ensure that the XML document type definition (`.dtd` files) or W3C XML Schema (`.xsd` files) that define the events that you want to monitor are available.

    If an Oracle database is your generic event source, you can use the XML Schema Utility (XSU) to define the events, as described in "Using the XSU Command Line to Generate .xsd Files for Database Tables" on page 7-6.

2. Use the Admin Console to specify the connection parameters to the generic event source, as documented in "Creating, Modifying, and Deleting Event Source Connections" on page 2-7.

3. Use the Admin Console to create a composite event for the events you intend to publish, as described in "Creating Composite Events" on page 2-23.

4. Publish the events to the queue for Oracle BPEL Process Analytics to capture, as documented in "Publishing Events for Oracle BPEL Process Analytics" on page 7-2.

5. Open the Dashboard and view the published events in the Dashboard Real-Time Viewer, as documented in "Viewing Events and Composite Event Instances" on page 4-6.

Publishing Events for Oracle BPEL Process Analytics

Oracle BPEL Process Analytics can capture events that you format and publish to a queue as described in the following topics:

- Event Format Required by Oracle BPEL Process Analytics on page 7-2

- Publishing Database Events to the Preconfigured OJMS Queue Using PL/SQL Packages on page 7-2

- Publishing Events to a User-Specified JMS Queue on page 7-5

Event Format Required by Oracle BPEL Process Analytics

Oracle BPEL Process Analytics requires that events published by custom applications use the same format as that which is defined for the publishEvent request document schema, described in "publishEvents Operation" on page 6-12.

Publishing Database Events to the Preconfigured OJMS Queue Using PL/SQL Packages

The preconfigured OJMS queue, `bam_genericmonq`, is installed when you install Oracle BPEL Process Analytics. It enables Oracle BPEL Process Analytics to monitor events that are generated by the same database as that which serves as the Oracle BPEL Process Analytics repository.

The following topics describe the prerequisites for using this queue and how to publish events to it:

- Prerequisites for Publishing Events to the bam_genericmonq Queue on page 7-2

- PL/SQL Packages for Publishing to the bam_genericmonq Queue on page 7-3

Prerequisites for Publishing Events to the bam_genericmonq Queue

To publish database events to the `bam_genericmonq` queue, you must first connect to the database as the `BAM` user and grant privileges as follows, where *user-name* is the name of the user who will publish events to the queue:

```
GRANT EXECUTE ON bam_generic_monitor TO user-name;
EXECUTE DBMS_AQADM.GRANT_SYSTEM_PRIVILGE ('ENQUEUE_ANY', 'user-name', TRUE);
```

The specified user can then use the tools described in "PL/SQL Packages for Publishing to the bam_genericmonq Queue" on page 7-3 to publish events to the bam_genericmonq queue. No queue configuration is required.

PL/SQL Packages for Publishing to the bam_genericmonq Queue

Oracle BPEL Process Analytics provides PL/SQL packages to assist you with installing, deinstalling, and publishing to the bam_genericmonq. These packages are defined in the following SQL script, where OAS_HOME is the Oracle Application Server home in which you installed Oracle BPEL Process Analytics:

```
OAS_HOME\integration\bam\admin\sql\CreateGenericMonitorPkg.sql
```

By default, this SQL script is run when you install Oracle BPEL Process Analytics.

You can use the procedures and helper methods defined by this package to deinstall (and then install again, if desired) the generic monitor JMS queue and to publish events to the generic monitor JMS queue. Details about using these procedures are described in the following topics:

- Using the Uninstall Procedure on page 7-3
- Using the Install Procedure on page 7-3
- Using the Publish Helper Methods on page 7-3

Using the Install Procedure The generic monitor JMS queue (bam_genericmonq) is installed when you install Oracle BPEL Process Analytics. However, if you have deinstalled it and now want to reinstall it, follow these steps:

1. Connect to the database as SYSDBA.

2. Run the install procedure:

   ```
   SQL> exec bam.bam_generic_monitor.install;
   PL/SQL procedure successfully completed.
   ```

Using the Uninstall Procedure If you decide you want to deinstall the generic monitor JMS queue (bam_genericmonq), follow these steps:

1. Connect to the database as SYSDBA.

2. Run the uninstall procedure:

   ```
   SQL> exec bam.bam_generic_monitor.uninstall;
   PL/SQL procedure successfully completed.
   ```

Using the Publish Helper Methods The createGenericMonitorPkg.sql script defines two publisher helper methods to assist you in publishing events to the generic monitor JMS queue:

- A procedure to publish SQL SELECT statement output

  The following procedure converts data returned by a SQL SELECT statement to the event format expected by the generic monitor, and then publishes the data returned by the SQL statement to the queue:

  ```
  PROCEDURE  publish(inSQL    varchar2,
                     inEventType varchar2,
                     inTimeStamp VARCHAR2 DEFAULT NULL,
                     inTimeZone VARCHAR2 DEFAULT NULL);
  ```

Parameters for this procedure are as follows:

- – inSQL – The SQL SELECT statement that you want to convert to the required event format

- – inEventType – The event name to be assigned to the SQL SELECT statement output

- – inTimeStamp – The time stamp at which the event occurs

  See "Required Date Format" on page 2-19 for information about formatting the time stamp.

- – inTimeZone – The time zone in which the event occur

  See "Required Date Format" on page 2-19 for information about formatting the time zone.

The input to the inSQL procedure is converted to XML (using the DBMS_XMLGEN PL/SQL package) and then published to the bam_genericmonq queue.

Note: If this publish procedure is called from within a SQL trigger, you must either select from dual or use statement-level triggers (to avoid a mutating table exception with row-level triggers).

- A procedure to publish XML data to the bam_genericmonq

  Use the following procedure to convert XML data to the required event format and publish it to the queue. Setting the constructGenericEvent parameter to FALSE, will publish the XML data to the queue, as is (unconverted):

```
PROCEDURE  publish(inCLOB  IN OUT NOCOPY CLOB,
                   inEventType varchar2,
                   inTimeStamp VARCHAR2 DEFAULT NULL,
                   inTimeZone VARCHAR2 DEFAULT NULL,
                   constructGenericEvent BOOLEAN default TRUE);
```

  Parameters for this procedure are as follows:

- inCLOB – The XML data that you want to publish to the bam_genericmonq JMS queue.

- inEventType – The event name to be assigned to the XML data.

- inTimeStamp – The time stamp at which the event occurs.

  See "Required Date Format" on page 2-19 for information about formatting the time stamp.

- inTimeZone – The time zone in which the event occurs.

  See "Required Date Format" on page 2-19 for information about formatting the time zone.

- constructGenericEvent - A Boolean value that specifies whether you want the XML data converted to the required event format. Specify TRUE to have the conversion performed; specify FALSE if you have already formatted the XML data using the required event format.

Using the bamShowGenericEvent.sql Script to View Generic Events in the Queue

Oracle BPEL Process Analytics provides a script that displays the number of events in the queue and a portion of the payload from the most recent event. You can find this script in the following location, where OAS_HOME is the directory specification of the Oracle Application Server home where you installed Oracle BPEL Process Analytics:

OAS_HOME\integration\bam\admin\sql\bamShowGenericEvent.sql

When you connect to the database on which the queue is running and run the script, data such as the following is returned:

```
SQL> @bamShowGenericEvent.sql
Number of events in bam_genericmonq: 7
Displaying the most recent event in queue :
<?xml version="1.0"?> <BAMEvent><header>
<timestamp>2005-05-13T07:58:36.772</timestamp> <timezone>-07:00</timezone>
<type>deleteEmployee</type> </header>  <payload><ROWSET><ROW num="1">
<EMPNO>30528</EMPNO> <ENAME>Name_30528</ENAME> <JOB>job_30528</JOB> <
<MGR>30528</MGR> <HIREDATE>2005-04-22"T"16:00:00</HIREDATE> <SAL>3500</SAL>
<COMM>0</COMM> <DEPTNO>10</DEPTNO> </ROW></ROWSET> </payload>  </BAMEvent>
PL/SQL procedure successfully completed.
```

Publishing Events to a User-Specified JMS Queue

You can publish events to a user-specified OJMS or JMS queue to do either of the following:

- Publish events from a J2EE application

  See *Oracle Application Server Containers for J2EE Services Guide* for conceptual information, and see the demonstrations at the following URL for examples:

  http://www.oracle.com/technology/tech/java/oc4j/demos/904/index.html

- Publish events from an Oracle database

  See the scripts for implementing the BooksOnLine example in the *Streams Advanced Queuing User's Guide and Reference* for examples.

Example of Monitoring an Oracle Database as a Generic Event Source

This section provides step-by-step instructions for monitoring an Oracle Database as a generic event source. This example demonstrates the following:

1. Using the XML SQL Utility (XSU) to create an .xsd file for the EMP table in the Oracle Database SCOTT sample schema.

2. Defining triggers on the EMP table such that every time an insert or delete operation is performed on the table, the data added or removed from the table is published to the bam_genericmonq.

3. Specifying the connection parameters to the generic event source.

4. Creating a composite event that contains both the insert and delete events. The correlation attribute used for the composite event is EMPNO.

5. Generating an event and viewing it in the Dashboard.

Using the XSU Command Line to Generate .xsd Files for Database Tables

To monitor a generic event source, Oracle BPEL Process Analytics requires that events be defined using XSD or DTD. When the generic event source is an Oracle Database, you can use the XML Schema Utility to generate `.xsd` files from database tables, which then require minor adjustments for use with Oracle BPEL Process Analytics.

See *Oracle XML Developer's Kit Programmer's Guide* for complete details about the XML Schema Utility. Follow these steps to generate an `.xsd` file from an Oracle database table or view for use with Oracle BPEL Process Analytics:

1. Add the following libraries to the JAVA classpath:

   - `classes12.jar` (JDBC)

   - `xsu12.jar` (XML Schema Utility, which contains the Oracle XML class)

   - `xmlparserv2.jar` (Oracle XML parser)

2. Create an `.xsd` file for the database table data for which you want to capture events, by saving the output from the following command, where:

   - `username/password` is the user name and password for the account that you want to use to access the database

   - `db-machine-name` is the name of the machine on which the database is running

   - `db-port` is the port number on which the listener is listening, and `db-sid` is the database system identifier (SID)

   - `SQL-statement` is the SQL statement that specifies the table and table attributes to include in the schema definition file (`.xsd`)

   ```
   java -classpath $CLASSPATH OracleXML getXML -user "username/<password" -conn
   "jdbc:oracle:thin:@db-machine-name:db-port:db-sid" -withSchema SQL-statement"
   ```

3. Delete the <Document> and <ROWSET> elements from the `.xsd` file and save it. You do not need to save it to a particular directory, but you will be asked to specify the directory location when you create a composite event on the generic event source.

See "Generating an .xsd File for the EMP Table" on page 7-6 for an example of this process.

Generating an .xsd File for the EMP Table

Follow these steps to generate an `.xsd` file for the `EMP` table and all of the table attributes:

Step 1: Specify the Java classpath.

Add the following classes to the Java classpath:

- `classes12.jar` (JDBC)

- `xsu12.jar` (XML Schema Utility, which contains the Oracle XML class)

- `xmlparserv2.jar` (Oracle XML parser)

Step 2: Generate XML output for the database table of interest.

Use the XML Schema Utility to generate the required schema definition using the following command, which:

1. Connects user SCOTT with password TIGER to a database with SID db1 through port 1521 of host myhost, using the Thin driver.

2. Executes the query SELECT * FROM EMP WHERE ROWNUM < 1

   The WHERE ROWNUM < 1 clause is specified to exclude the EMP table data from the generated .xsd file.

3. Converts the result to XML.

4. Places the results into a file named Employee.xsd.

If you want to try this example on your system, you need to replace *myhost:1521:db1* with the system, port number, and database SID for your environment.

```
java -classpath $CLASSPATH OracleXML getXML -user "scott/tiger" -conn
"jdbc:oracle:thin:@myhost:1521:db1" -withSchema "select * from emp where rownum<1"
> Employee.xsd
```

The output should appear similar to that shown in Example 7–1. However, your output will not contain bold text; the bolding is added in Example 7–1 for clarity in describing the next step.

Example 7–1   Sample XSD Output from the XML Schema Utility

```
<?xml version = '1.0'?>
<DOCUMENT xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
 <xsd:element name="ROWSET">
    <xsd:complexType>
       <xsd:sequence>
          <xsd:element name="ROW" minOccurs="0" maxOccurs="unbounded">
             <xsd:complexType>
                <xsd:sequence>
                   <xsd:element name="EMPNO" type="xsd:integer" minOccurs="0"/>
                   <xsd:element name="ENAME" nillable="true" minOccurs="0">
                      <xsd:simpleType>
                         <xsd:restriction base="xsd:string">
                            <xsd:maxLength value="10"/>
                         </xsd:restriction>
                      </xsd:simpleType>
                   </xsd:element>
                   <xsd:element name="JOB" nillable="true" minOccurs="0">
                      <xsd:simpleType>
                         <xsd:restriction base="xsd:string">
                            <xsd:maxLength value="9"/>
                         </xsd:restriction>
                      </xsd:simpleType>
                   </xsd:element>
                   <xsd:element name="MGR" type="xsd:integer" nillable="true" minOccurs="0"/>
                   <xsd:element name="HIREDATE" type="xsd:string" nillable="true" minOccurs="0"/>
                   <xsd:element name="SAL" type="xsd:float" nillable="true" minOccurs="0"/>
                   <xsd:element name="COMM" type="xsd:float" nillable="true" minOccurs="0"/>
                   <xsd:element name="DEPTNO" type="xsd:integer" nillable="true" minOccurs="0"/>
                </xsd:sequence>
                <xsd:attribute name="num" type="xsd:integer"/>
             </xsd:complexType>
          </xsd:element>
       </xsd:sequence>
    </xsd:complexType>
 </xsd:element>
```

```
</xsd:schema>
<ROWSET xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xsi:noNamespaceSchemaLocation="#/DOCUMENT/xsd:schema[not(@targetNamespace)]"/>
</DOCUMENT>
```

Step 3: Edit and Save the Output.

The output generated in the previous step contains two extraneous elements, <DOCUMENT> and <ROWSET>, that you must delete. These elements are bold in the previous step.

1. Delete the <DOCUMENT> and <ROWSET> elements.

2. Save the edited output.

Publishing Events from the EMP Table to the bam_genericmonq Queue

Connect to the database as the BAM user and grant privileges to the user SCOTT, by running the addGrantsForScottSchema.sql file located in the following directory, where OAS_HOME is the directory specification for the Oracle Application Server home in which you installed Oracle BPEL Process Analytics:

OAS_HOME\integration\bam\admin\sql

For example:

```
> sqlplus BAM/BAM
SQL> @addGrantsForScottSchema.sql
Grant succeeded.
PL/SQL procedure successfully completed.
SQL> exit
```

To create triggers on the EMP table and publish the SQL INSERT and DELETE statements to the bam_genericmonq queue, connect to the database using the username SCOTT and the password TIGER and run the SQL script, createScottSchemaTriggers.sql.

```
> sqlplus SCOTT/TIGER
SQL> @createScottSchemaTriggers.sql
Trigger created.
No errors.
Trigger created.
No errors.
SQL> exit
```

Connecting to the Generic Event Source in the Admin Console

Follow these steps to specify Oracle Database as a generic event source:

1. Open the Oracle BPEL Process Analytics Console and log in, as described in "Getting Started with the Admin Console" on page 2-2.

2. Click **Admin.**

3. Click **Event Source.**

4. Click the **Create** button associated with the **BPA Generic Sources** table.

5. Specify ScottTiger for the event source name and specify the rest of the field values as described in Table 2–2 on page 2-13.

6. Click **Finish.**

Creating a Composite Event for an Oracle Database Generic Event Source

Follow these steps to create a composite event that includes insert and delete events, correlated by the EMPNO column:

1. If the Oracle BPEL Process Analytics Console is not open, open it and log in, as described in "Getting Started with the Admin Console" on page 2-2.

2. Click **Modeling.**

3. Click **Composite Events.**

4. Click the **Add Composite Event** button.

5. Select the **ScottTiger** event source, and then click the **Create** button. The first page of the Add Composite Event Wizard opens.

6. On the Properties page of the wizard, follow these steps:

   a. In the **Name** field, enter ScottTiger_CE.

   b. In the **Description** field, enter: Composite event for ScottTiger example.

   c. In the **Timeout** field, leave the values as 0 **Days** 8 **Hours** 0 **Minutes.**

   d. Click **Next.** The Schema Definitions page opens.

7. On the Schema Definitions page, follow these steps:

   a. In the **Schema File** field, enter the complete path to the Employee.xsd file you created in "Generating an .xsd File for the EMP Table" on page 7-6.

   b. Click the **Add** button.

   c. Leave the **Root Element for DTD** field empty.

   d. Click **Next.** The Events page opens.

8. On the Events page, follow these steps:

   a. In the **Event Name** field, enter Employee. (This string, including uppercase and lowercase characters used in the string, must match the second parameter passed to the first bam.bam_generic_monitor.publish procedure in the SQL script that generates the events (createScottTigerSchemaTriggers.sql, in this example).

   b. In the **Schema Name** field, keep Employee.xsd.

   c. In the **Root Element for DTD** field, keep ROWSET.

   d. Click the **Add** button.

   e. In the **Event Name** field, enter deleteEmployee. (This string, including uppercase and lowercase characters used in the string, must match the second parameter passed to the first bam.bam_generic_monitor.publish procedure in the SQL script that generates the events (createScottTigerSchemaTriggers.sql, in this example.)

   f. In the **Schema Name** field, keep Employee.xsd.

   g. In the **Root Element for DTD** field, keep ROWSET.

   h. Click the **Add** button.

   i. Select **deleteEmployee** as the **End Event.**

   j. Click **Next.** The Attributes page opens.

9. In the Attributes page, follow these steps:

    a. In the first table, select **Employee.**

    b. Expand the navigation tree.

    c. Select **EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM,** and **DEPTNO.**

    d. In the first table, select **deleteEmployee.**

    e. Expand the navigation tree.

    f. Select **EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM,** and **DEPTNO.**

    g. Click **Next.** The Attribute Properties page opens.

10. In the Attribute Properties page, follow these steps:

    a. In the first table, select **Employee.**

    b. In the second table, select **EMPNO** as the correlation attribute.

    c. In the first table, select **deleteEmployee.**

    d. In the second table, select **EMPNO** as the correlation attribute.

    e. Click **Finish.**

Generating Events and Viewing Them in the Dashboard

In a production system, events would be generated as a result of normal system operations. For the purposes of this example, connect to the database using the SCOTT/TIGER account, and issue the statements such as the following to generate events. Be sure to commit each statement and, for ease of viewing in the Real-Time Viewer, pause for a few minutes in between issuing each statement:

```
SQL> INSERT INTO EMP VALUES(1249,'istuart','Designer',10,sysdate,2000,0,20);
SQL> COMMIT;
SQL> INSERT INTO EMP VALUES(1300,'dBradford','ANALYST',10,sysdate,30000,0,30);
SQL> COMMIT;
SQL> DELETE FROM EMP WHERE EMPNO=1249;
SQL> COMMIT;
SQL> SQL> DELETE FROM EMP WHERE EMPNO=1300;
SQL> COMMIT;
```

Open the Oracle BPEL Process Analytics Dashboard and view the events:

1. Open the Oracle BPEL Process Analytics Console and log in, as described in "Getting Started with the Admin Console" on page 2-2.

2. Click **Dashboard.**

3. In the **Switch View** box, select **Real-Time Viewer.**

4. In the Composite Event box, select **ScottTiger_CE.** The Dashboard should appear similar to Figure 7–1.

Figure 7–1   Sample Dashboard for Generic Event Source
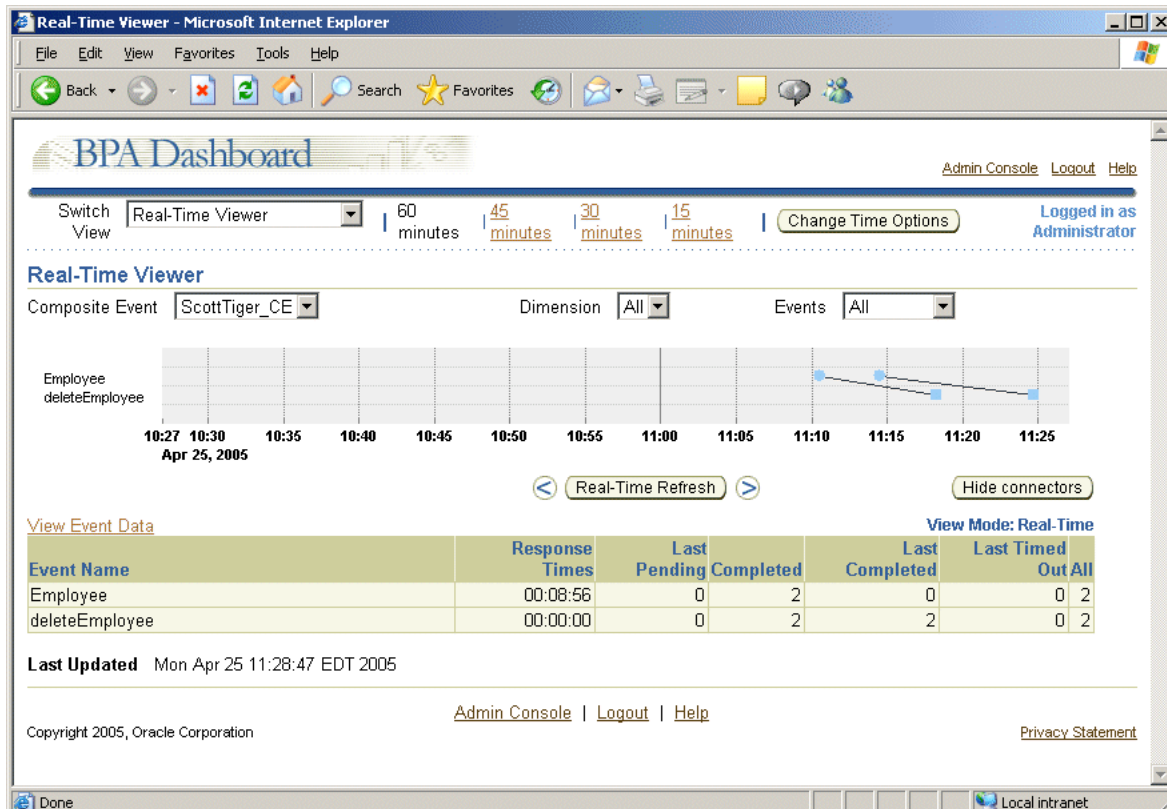
5. Click a data point in the Real-Time Viewer to see details about the event. For example, clicking the first data point in Figure 7–1, reveals the details shown in Figure 7–2.

Figure 7–2   Sample Composite Event Details for Generic Event Source

8

Simulating Events

The Oracle BPEL Process Analytics event simulator enables you to generate events without connecting to an actual event source. This can be useful in designing and testing your data modeling and Dashboard configuration before implementing it with a production event source.

To use the event simulator, you must be familiar with the concept of a composite event and how the events you intend to simulate can be grouped as composite events.

> **Note:** This chapter uses the files provided for Sample 2: LoanFlow with Hierarchical Dimension Demo to demonstrate how to use the event simulator. If you have previously set up this demonstration, remove the loanFlowHDim event source from your Oracle BPEL Process Analytics configuration, by following these steps:
>
> 1. Start Oracle BPEL Process Analytics, and open the Oracle BPEL Process Analytics Console as described in "Getting Started with the Admin Console" on page 2-2.
>
> 2. Click **Admin.**
>
> 3. Click **Event Source.**
>
> 4. Under **BPA Event Simulator Sources,** click the **trash can** icon associated with **loanFlowHDim.**
>
> 5. Click **Yes,** on the Are You Sure? page.

This chapter includes the following topics:

- Overview of Steps for Using the Event Simulator on page 8-1

- Files Required for Using the Event Simulator on page 8-2

- Using the Admin Console to Model the Simulated Events on page 8-8

- Viewing the Simulated Events in the Dashboard on page 8-11

- Adjusting the Speed at Which Events Are Generated on page 8-12

> **Note:** The directory paths shown in this chapter follow Microsoft Windows conventions (using backslashes (\)). If you are running Unix, modify the directory paths as required (using slashes (/)).

Overview of Steps for Using the Event Simulator

To use the event simulator, the basic process is as follows:

1. Create the files and directory structure required by the event simulator. For more information, see "Files Required for Using the Event Simulator" on page 8-2.

2. Open the Oracle BPEL Process Analytics Admin Console, and specify the event simulator as an Oracle BPEL Process Analytics Event Simulator event source.

3. Model the event (by creating composite events, KPIs, and metrics, as desired).

4. Open the Oracle BPEL Process Analytics Dashboard, and view the events, KPIs, and metrics in the Dashboard.

Files Required for Using the Event Simulator

To generate events, the event simulator requires the following files for each composite event for which you want to generate events. These files must reside in a single directory:

- A file named `events.xml`.

  This file is used by the event simulator to generate the events.

- An `.xsd` file (or files) to specify the event attribute metadata for the events you want to generate.

  You can specify a single `.xsd` file that contains the event attribute metadata definition for all of the events in the composite event, or you can create a separate `.xsd` file for each event. The example shown in this chapter uses a single `.xsd` file.

- An `.xml` file to specify an event instance for each event you want to generate.

This chapter uses the Loan Provider demonstration (with hierarchical dimensions) to demonstrate the structure and contents of each of these files. The files for the demo are located in the following directory (where *OAS_HOME* is the Oracle Application Server home into which you installed Oracle BPEL Process Analytics).

OAS_HOME\integration\bam\test\data\loadGen\loanFlowHDim\loanFlowHDim

The following topics provide details about these required files and their location in the directory structure:

- Setting Up the Directory Structure for the Required Files on page 8-2

- Required .xsd Files on page 8-3

- Required .xml Files on page 8-4

- Required events.xml File on page 8-5

Setting Up the Directory Structure for the Required Files

The Oracle BPEL Process Analytics event simulator requires a parent directory that contains one or more subdirectories. Each subdirectory corresponds to a single composite event. The name of a subdirectory is the name of the composite event.

In the directory for the Loan Provider demonstration (with hierarchical dimensions), there is a parent directory named `loanFlowHDim` and a single subdirectory, also named `loanFlowHDim`. This subdirectory holds the `.xml` and `.xsd` files required to generate events for the loanFlowHDim composite event. If you want to create additional composite events for the Loan Provider demonstration, then create additional subdirectories in the following directory, where *OAS_HOME* is the Oracle Application Server home in which you installed Oracle BPEL Process Analytics:

OAS_HOME\integration\bam\test\data\loadGen\loanFlowHDim\loanFlowHDim

Required .xsd Files

The event simulator requires an `.xsd` file to specify the event attribute metadata. You can specify one `.xsd` file for all of the events in the composite event, or you can specify an `.xsd` file for each event. The `.xsd` file for the loanFlowHDim composite event is named `LoanFlowPlus.xsd`. The contents of `LoanFlowPlus.xsd`, as shown in Example 8–1, shows that an `.xsd` file for the event simulator contains the following elements to define the schema:

■   A declaration of the XML version.

■   A schema declaration that includes attributes to describe the namespace.

■   For each event, a parent `element` declaration that names each event and contains child `element` declarations that specify the name and type of each of the event attributes. (Note that the parent `element` names in the `.xsd` file are used to specify the `rootElement` values in the `events.xml` file.)

Providing a full description of how an `.xsd` file should be formed is beyond the scope of this guide. See standard texts on XML schema for detailed information.

Example 8–1   Contents of LoanFlowPlus.xsd

```
<?xml version="1.0"?>
<schema attributeFormDefault="qualified"
        elementFormDefault="qualified"
targetNamespace="http://www.autoloan.com/ns/autoloan"
xmlns="http://www.w3.org/2001/XMLSchema">

 <element name="loanOffer">
  <complexType>
   <sequence>
    <element name="id" type="string"/>
    <element name="providerName" type="string"/>
    <element name="selected" type="boolean"/>
    <element name="approved" type="boolean"/>
    <element name="APR" type="double"/>
   </sequence>
  </complexType>
 </element>

 <element name="loanApplication">
  <complexType>
   <sequence>
    <element name="id" type="string"/>
    <element name="SSN" type="string"/>
    <element name="email" type="string"/>
    <element name="customerName" type="string"/>
    <element name="loanAmount" type="double"/>
    <element name="carModel" type="string"/>
    <element name="carYear" type="string"/>
    <element name="creditRating" type="int"/>
   </sequence>
  </complexType>
 </element>

 <element name="task">
   <complexType>
```

```
     <sequence>
      <element name="taskId" type="string" minOccurs="0"/>
      <element name="title" type="string" minOccurs="0"/>
      <element name="creationDate" type="dateTime" minOccurs="0"/>
      <element name="creator" type="string" minOccurs="0"/>
      <element name="modifyDate" type="dateTime" minOccurs="0"/>
      <element name="modifier" type="string" minOccurs="0"/>
      <element name="assignee" type="string" minOccurs="0"/>
      <element name="status" minOccurs="0">
       <simpleType>
        <restriction base="string">
          <enumeration value="active"/>
          <enumeration value="completed"/>
        </restriction>
       </simpleType>
      </element>
      <element name="expired" type="boolean" minOccurs="0"/>
      <element name="expirationDate" type="dateTime" minOccurs="0"/>
      <element name="duration" type="duration" minOccurs="0"/>
      <element name="priority" type="int" minOccurs="0"/>
      <element name="template" type="string" minOccurs="0"/>
      <element name="customKey" type="string" minOccurs="0"/>
      <element name="conclusion" type="string" minOccurs="0"/>
      <element name="attachment" type="anyType"/>
     </sequence>
   </complexType>
 </element>
</schema>
```

Required .xml Files

The event simulator requires an .xml file for each event it is to generate. See the
LoanOfferResponse.xml file specified for the loanFlowHDim composite event,
shown in Example 8–2, for an example.

The .xml file for a simulated event contains the following:

■ A declaration of the XML version.

■ An instance of the event – the element name (loanOffer, in this example) must
   match the name of the event in the .xsd file.

Be aware that the namespace attribute values must match the namespace attribute
values specified in the .xsd file.

Example 8–2   Contents of the LoanOfferResponse.xml File

```
<?xml version='1.0' ?>
<loanOffer xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns="http://www.autoloan.com/ns/autoloan">
        <id>123456</id>
        <providerName>Star Loan Provider</providerName>
        <selected>true</selected>
        <approved>true</approved>
        <APR>450</APR>
</loanOffer>
```

Required events.xml File

The event simulator requires an `events.xml` file that minimally provides references to the `.xsd` and event instance `.xml` files, the root element for the event, how attribute values should be correlated, and which values should be used in generating events. If desired, you can specify the frequency at which each event should be generated and the time delay between events.

The parent elements in the `events.xml` file (in addition to the XML version declaration) are the following:

- <EventMap>, described in "<EventMap> Element" on page 8-5

- <Paths>, described in "<Paths> Element" on page 8-5

- <Events>, described in "<Events> Element" on page 8-6

- <LoadGenerator>, described in "<Load Generator> Element" on page 8-7

Although code examples are presented with each element description, you may find it helpful to view the `events.xml` file for the sample LoanFlow with Hierarchical Dimension Demo as you read this section.

<EventMap> Element

The <EventMap> element contains all of the other elements contained within the `events.xml` document. This element is required. Specify one <EventMap> element in the `events.xml` file.

<Paths> Element

The <Paths> element specifies how often a given instance of a composite event is to be generated. Within the <Paths> element are two or more <Path> elements. Each <Path> element represents one full set of events in a process flow, from the start event to an end event.

The <Paths> element is optional. If used, specify one <Paths> element in the `events.xml` file. If the <Paths> element is not specified, the distribution of events generated for a composite event instance is uniform across all events specified.

The <Paths> element in the `events.xml` file for the Loan Provider demonstration (with hierarchical dimensions) appears as follows:

```
<Paths>
  <Path percent="20">
    <Event>LoanRequest</Event>
    <Event>LoanStatus4BadCredit</Event>
  </Path>
  <Path percent="80">
    <Event>LoanRequest</Event>
    <Event>LoanRequestApproved</Event>
    <Event>LoanOfferSelected</Event>
    <Event>LoanOfferResponse</Event>
  </Path>
</Paths>
```

This <Paths> declaration specifies that for each loan requested, the composite event ending with the LoanStatus4BadCredit event should be generated 20 percent of the time, whereas the composite event ending with the LoanOfferResponse event should be generated 80 percent of the time.

<Events> Element

The <Events> element specifies an <Event> element for each event to be generated. This element is required. Specify one <Events> element in the events.xml file. You can specify multiple <Event> elements within the <Events> element.

The <Event> element for the LoanStatus4BadCredit event in the events.xml file for the Loan Provider demonstration (with hierarchical dimensions) appears as follows:

```
<Event name="LoanStatus4BadCredit" endEvent="true">
  <schema>LoanFlowPlus.xsd</schema>
  <delay>6</delay>
  <rootElement>task</rootElement>
  <Attribute correlateAttr="true">
     <XPath>/task/taskiId</XPath>
     <MappedType>Number</MappedType>
     <Alias>LoanApplicationId</Alias>
  </Attribute>
  <Attribute>
     <XPath>/task/conclusion</XPath>
     <MappedType>Text</MappedType>
     <Alias>Status</Alias>
  </Attribute>
  <instance>
   <file>LoanStatus4BadCredit.xml</file>
   <correlate attr="LoanApplicationId">
     <with event="LoanRequest" attr="LoanApplicationId"/>
   </correlate>
   <randomize attr="Status">
     <values generator="loanStatus"/>
   </randomize>
  </instance>
 </Event>
```

For each event that you want to generate, specify an <Event> element, as follows. Unless otherwise noted, each element is required.

- <Event > – Specify a name for the event.

- <schema> – Specify the schema (.xsd) file for the event.

- <delay> – Specify the amount of time (in milliseconds) between the time that the previous event in the process flow is generated and this event is generated. This element is optional.

- <rootElement> – Specify the root element for the event. The root element for an event must be the same as the corresponding <element> name attribute in the .xsd file.

- <Attribute> – Specify an <Attribute> element for each event attribute to be included when the event is generated. For each event attribute, specify the following:

  - If the event attribute is the correlation attribute for the composite event, specify the correlateAttr=True attribute. By default, an event attribute is not a correlation attribute for the composite event.

  - <XPath> – Specify the XPath for the event attribute.

  - <MappedType> – Specify the data type of the event attribute.

  - <Alias> – Specify an alias to serve as an abbreviation for the XPath.

- <instance> – Specify the following details about the event instance to be generated:

  – <file> – Specify the `.xml` file that corresponds to the event instance.

  – <correlate> – Specify the event attribute that you want to use to correlate this event with another event by using the <Alias> value. Within the <correlate> element, specify the <with> element to indicate the event and event attribute to which this event will be correlated.

  In the <Event> element for the LoanStatus4BadCredit event in the `events.xml` file, the LoanApplicationID for the LoanApplication4BadCredit event is correlated with the LoanApplicationID for the LoanRequest event. This correlation enables you see which events are associated with a given loan request.

  – <randomize> – Specify the alias for the event attribute for which you want to generate random events. Within the <randomize> element, specify the <values> element with the generator attribute to indicate which of the load generators to use to generate the events. (The load generators are specified by the <LoadGenerator> elements at the end of the file.)

<Load Generator> Element

The <Load Generator> element allows you to replace event attribute values with randomly generated values. Multiple <Load Generator> elements can be specified in a single `events.xml` file. This element is optional, but only one instance of each event will be generated if you do not specify this element for each event you want to generate.

Two types of generator are available:

- String

  For a string generator, you specify the list of strings from which the event simulator can randomly choose for each event instance it simulates. One of the <Load Generator> elements in the `events.xml` file for the Loan Provider demonstration (with hierarchical dimensions) is as follows:

```
<LoadGenerator name="providerName" type="string">
 <value>Star Loan Provider</value>
 <value>United Loan Provider</value>
</LoadGenerator>
```

  This specifies that the event simulator can use Star Loan Provider or United Loan Provider as a value for the `providerName` event attribute.

- Integer

  For an integer generator, you specify a range of values from which the event simulator can randomly choose for each event instance it simulates. One of the <Load Generator> elements in the `events.xml` file for the Loan Provider demonstration (with hierarchical dimensions) is as follows:

```
<LoadGenerator name="crRating" type="integer">
 <value begin="200" end="800"/>
</LoadGenerator>
```

  This specifies that the event simulator can use any value between 200 and 800 (inclusive) as a value for the crRating (credit rating) event attribute.

Using the Admin Console to Model the Simulated Events

Once you have created the events.xml, .xsd file (or files), and the event instance .xml files for each composite event within a single directory, you are ready to use the Admin Console to specify the event simulator as the event source and model the events, as described in the following topics:

- Specifying the Event Simulator as the Event Source on page 8-8
- Creating a Composite Event and Dimensions for Simulated Events on page 8-9
- Creating Metrics and Key Performance Indicators for Simulated Events on page 8-10

Specifying the Event Simulator as the Event Source

You specify the event simulator as the event source using the Admin section of the Admin Console, as follows:

1. Open and log in to the Admin Console as the Administrator, as described in "Getting Started with the Admin Console" on page 2-2.

2. Click **Admin.**

3. Click **Event Source.**

4. Click the **Create** button associated with the **BPA Event Simulator Sources** table. The Add Event Source page opens.

5. Specify the values for the event simulator as described in the table that follows this list and as shown in Figure 8–1. If you want to follow the steps described in this chapter, specify the values listed in the Example Values column, where *OAS_HOME* is the Oracle Application Server home where you installed Oracle BPEL Process Analytics.

6. Click **Finish.**

Page Element	Description	Example Values
Name	Enter an alphanumeric name by which you want to refer to the event simulator.	LoanFlowSim
Description	Enter a description of the event simulator.	Event simulator for the Loan Provider demonstration
Event Simulator Event Source Data Directory	Enter the specification for the parent directory that holds the composite event directory (or directories) for your event simulator files. Do not specify a composite event directory name.	*OAS_HOME*\integration\bam\test\data\loadGen\loanFlowHDim The directory path shown follows Microsoft Windows conventions (using backslashes (\)). If you are using Unix, modify the directory path as required (using slashes (/))

Figure 8–1   Event Source Page for Sample Event Simulator

Creating a Composite Event and Dimensions for Simulated Events

Creating a composite event for simulated events is similar to creating a composite event for actual events. The main differences arise because the composite event details for a simulated event are specified in the `events.xml` document.

Creating a dimension for simulated events is the same as it is for actual events.

The following two topics step you through the process of defining a dimension and composite event using the Loan Provider demonstration (with hierarchical dimensions).

Creating the ProviderName Dimension

To create the ProviderName dimension:

1. In the Admin Console, click **Modeling.**

2. Click **Dimensions.**

3. Click the **Create** button.

4. In the **Name** field, enter ProviderName.

5. Click **Next.**

6. In the **Data Type** field, select varchar2.

7. In the **Column size** field, enter 25.

8. Click **Finish.**

Creating the loanFlowHDim Composite Event

To create the loanFlowHDim composite event:

1. In the Admin Console, click **Modeling.**

2. Click **Composite Events.**

3. Click the **Add Composite Event** button.

4. Click the **Create** button associated with the **LoanFlowSim** event source. The Properties page of the Add Composite Event Wizard opens.

5. In the **Description** field, enter a description for the composite event.

6. Leave the **Timeout** field values as they are.

7. Click **Next.**

8. Select the loanFlowHDim composite event, then click **Next.**

9. The values for this page are prefilled, based on the <Event> elements specified in the events.xml file. Click **Next.**

10. Move the **ProviderName** dimension from the **Available Dimensions** box to the **Selected Dimensions** box, and click **Next.**

11. Expand the navigation tree and select the **LoanOfferSelected/ProviderName** (as shown in Figure 8–2) and the **LoanOfferResponse/ProviderName** attributes.

Figure 8–2   Navigation Tree for LoanFlowHDim Dimension Mapping

12. Click **Finish.**

Creating Metrics and Key Performance Indicators for Simulated Events

You create metrics and KPIs for simulated events as described for actual events. See "Creating Metrics" on page 2-26 and "Creating Key Performance Indicators" on page 2-29 for more information.

Viewing the Simulated Events in the Dashboard

When you have created all the files required for the event simulator, have specified the event simulator as an event source, and have created a composite event, then you can use the Real-Time Viewer in the Dashboard to view the events as they are simulated.

To view simulated events in the Dashboard:

1. Open the Dashboard in the Oracle BPEL Process Analytics Console.

2. In the **Switch View** box, select **Real-Time Viewer.**

3. Select 15 minutes.

4. In the Composite Event field, select the composite event you want to view. If you are following this chapter using the Loan Provider demonstration, select **loanFlowHDim.** The Real-Time Viewer should appear similar to Figure 8–3.

5. Click a data point within the Real-Time Viewer to display details about that event instance. A Composite Event Details page opens, similar to Figure 8–4.

Figure 8–3   Simulated Events in Real-Time Viewer
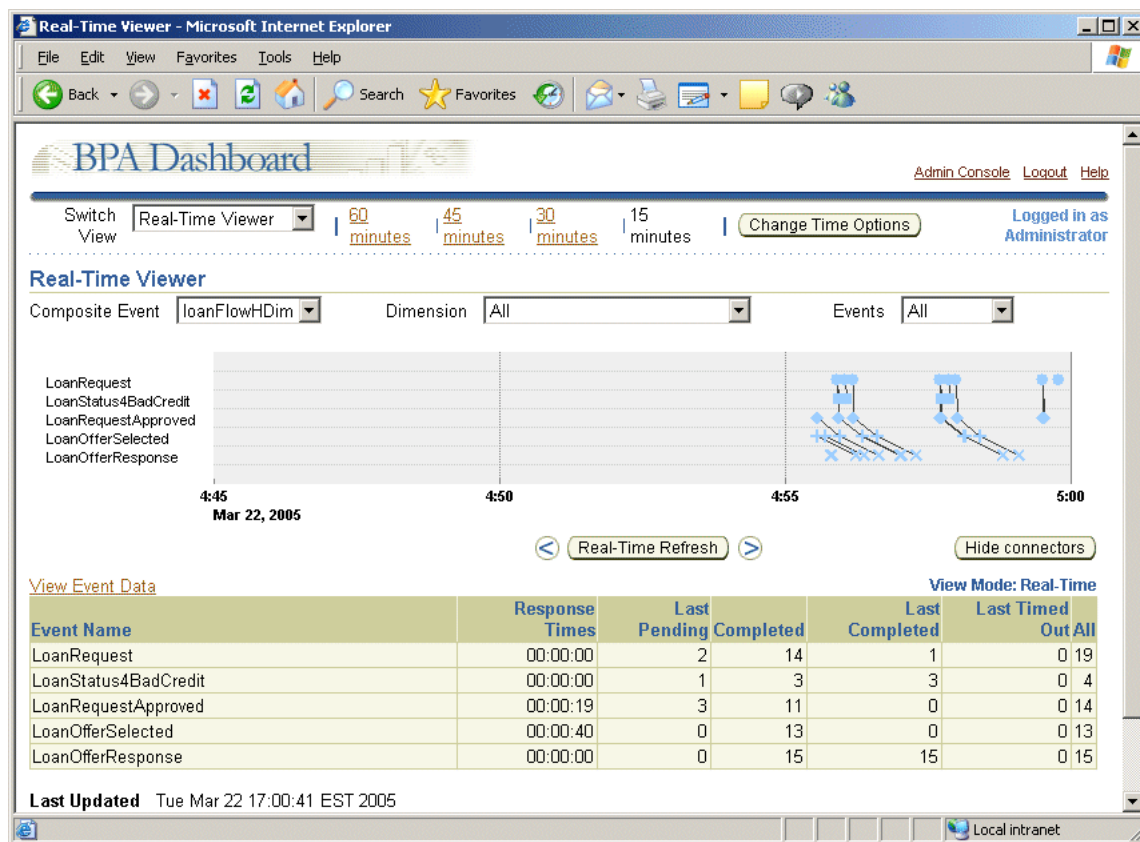
Figure 8–4 Sample Composite Event Details Page
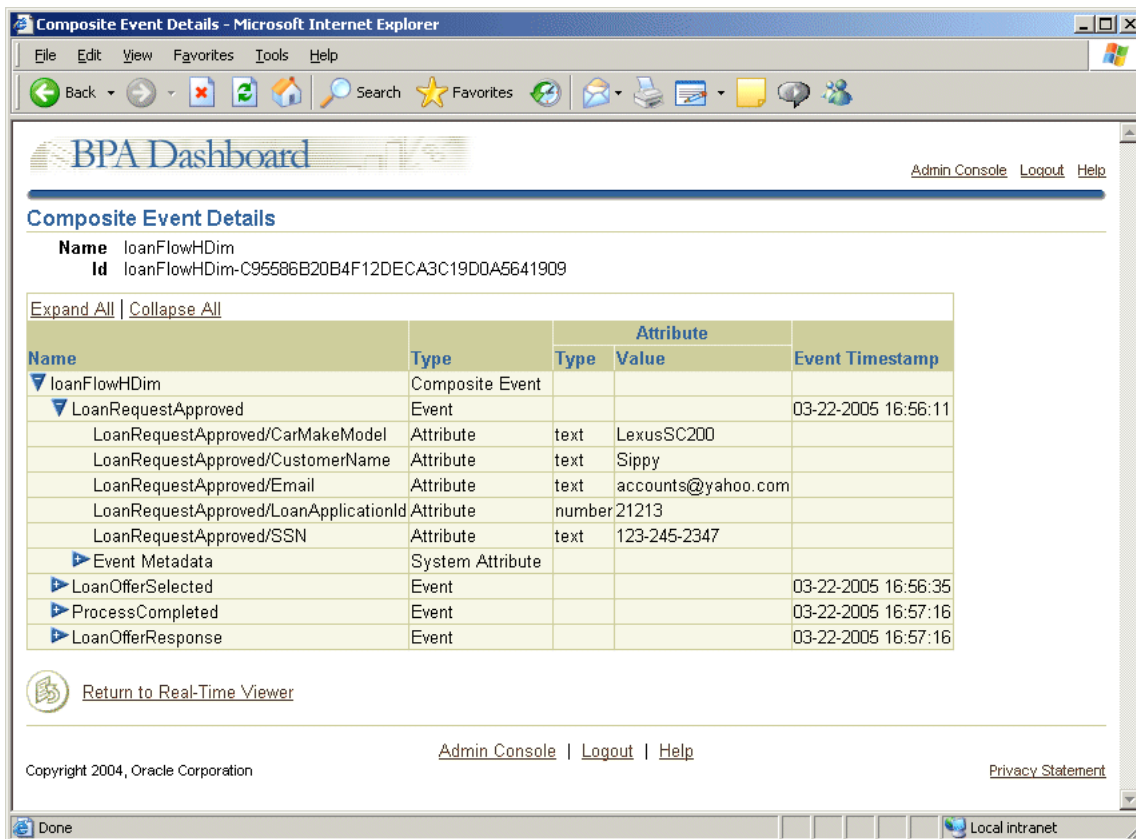
If you created KPIs or metrics, you can view them in the Dashboard by selecting **Default KPI View** or **Default Metric View** in the **Switch View** box.

Adjusting the Speed at Which Events Are Generated

You can adjust the speed at which events are generated by adjusting the values for the following entries in the `bam.properties` file and restarting the Oracle BPEL Process Analytics server:

- `oracle.tip.bam.monitor.test.LoadThreads`

- `oracle.tip.bam.monitor.test.LoadSeed`

See "Setting Log File, Dashboard, and Other Properties" on page 5-12 for information about the location of the `bam.properties` file and how to set the values for these properties.

9

Introduction to Analyzing Oracle BPEL Process Analytics Data with Discoverer

Oracle BPEL Process Analytics generates real-time data, but it also archives data as tables for future analysis. This analysis is performed with Oracle Business Intelligence Discoverer.

This chapter includes the following topics:

- Analyzing Archived Oracle BPEL Process Analytics Data with Discoverer on page 9-1
- LoanFlowPlus Example on page 9-4

Analyzing Archived Oracle BPEL Process Analytics Data with Discoverer

Oracle BPEL Process Analytics not only provides real-time analysis on incoming data, it also archives this data for later examination. This information is stored in a star join schema composed of fact and dimension tables.

Oracle Business Intelligence Discoverer locates the archived information and presents it through an end user layer (EUL), which simplifies working with the database for the user. This chapter assumes that the user is already familiar with Discoverer and is focused on using Discoverer to analyze archived BPA data.

Once you have created your EUL, define business areas, folders, and items in Discoverer to access only the data you want to analyze.

> **See Also:** "Creating and Maintaining the EUL" in *Oracle Business Intelligence Discoverer Administration Guide* for additional information on the EUL

Oracle BPEL Process Analytics uses a number of data structures to record the data in tables arranged in a star join schema. These components include composite events, Key Performance Indicators (KPIs), facts and dimensions, item classes, tables, and the star join schema itself. Most of these items have been defined previously in this guide, but are reviewed in the following sections. Item classes and business areas (along with the EUL) are discussed in detail in the Oracle Business Intelligence Discoverer Administration Guide.

Composite Events

A composite event can include one or more events, from one event source only. Events are correlated on the basis of a common event attribute (referred to as a correlation attribute).

> **See Also:** "Modeling Composite Events and Composite Event Groups" on page 2-20 for additional information on composite events

Key Performance Indicators

A key performance indicator (KPI) consists of instances of a composite event attribute (or attributes) aggregated over a period of time, to which a mathematical function is applied. While metrics can be used to study general patterns and trends, KPIs enable an analyst to perform in-depth analysis of the event data.

> **See Also:** "Modeling KPIs and Related Objects" on page 2-27 for additional information on KPIs

Facts and Dimensions

In online analytical processing (OLAP) metadata, facts are data that can be examined and analyzed in crosstabs and graphs.

Facts have dimensions that categorize the data in the fact. For example, a sales fact might have product, time, and geography as its dimensions. When a fact has a particular dimension, the fact is said to be dimensioned by that dimension. For example, sales is dimensioned by product. The group of dimensions for a fact constitute the dimensionality of that fact. For example, the dimensionality of Sales is product, time, and geography. Each element in a dimension is a dimension member. For example, January 2005, February 2005, March 2005, Quarter 1 2005, and the year 2005 are likely members of the time dimension.

Facts can have more than three dimensions.

> **See Also:** "Modeling Dimensions" on page 2-32 for additional information on facts and dimensions

Item Class

An item class is a list of values for dimensions, which associate Oracle BPEL Process Analytics data with Discoverer table entries and provide names.

> **See Also:** For more information about how to create custom folders and item classes, see *Oracle Business Intelligence Discoverer Administration Guide*

Tables

In a relational data source, data is organized in tables. A table is a data structure with columns and rows. The tables are created by Oracle BPEL Process Analytics. In the star join schema that BPA uses to store information, there are fact tables and dimension tables. Multiple dimension tables are joined with each fact table.

Fact Tables

A fact table typically contains fields that are additive and represent measurements whose values change each time they are taken. For example, the number of loans offered, number of loans rejected, and loan amounts.

Dimension Tables

A dimension table typically contains a single primary key and, optionally, additional columns whose values are stable over time, such as a car's make, model, and year.

Time Dimension The time dimension is a specific kind of dimension that is always created and associated with each fact table. The time dimension is not user defined.

The time dimension is defined as shown in Table 9–1.

Table 9–1   The Time Dimension Table Structure

Name	Null?	Type
TIMEID	NOT NULL	NUMBER
HOUR		CHAR(10)
DAY		CHAR(8)
MONTH		CHAR(6)
QUARTER		CHAR(5)
YEAR		CHAR(4)

The time dimension is joined with Oracle BPEL Process Analytics fact tables to calculate KPI values for pre-defined time periods. Table 9–2 is a sample row from the time dimension table bam_dim_time_t.

Table 9–2   bam_dim_time_t Sample Row

TIMEID	HOUR	DAY	MONTH	QUARTER	YEAR
2005010101	2005010101	20050101	200501	20051	2005

The year column represents the four-digit year. The quarter column represents the quarter of the year using a four-digit year and 1 to 4 representing the four quarters of the year. The month column contains the month of the year using the four-digit year and a two-digit, zero-filled month (01 - 12). The day column has the format of YYYYMMDD where day is the day of month (also zero-filled). The Hour column has the format YYYYMMDDHH where hour is zero-filled hour of day (00 - 23). The values in the TimeID column are numeric and are used to join with Oracle BPEL Process Analytics fact tables. The only requirement for the TimeID is that it be a unique value. However, a convenient unique key for each row in the time dimension is identical to the Hour column.

Star Join Schemas

The modeling of KPIs and dimensions is based on the concept of a star join schema. A star join schema, commonly used for dimensional data warehouses, is composed of a fact table that is joined by primary keys to a number of dimension tables. A fact table typically contains fields that are additive and represent measurements whose values change each time they are taken. A dimension table typically contains a single primary key and, optionally, additional columns whose values are stable over time.

The star join schema contains a fact value for each possible combination of the different dimensions. It is therefore very quick for applications such as Discoverer Plus OLAP to find the value for sales of a particular product in a particular city in a particular year.

For example, to find sales of Product C in 2005 in Dallas, Discoverer Plus OLAP simply uses the product, time, and city dimensions to identify the cell containing the required value.

Creating a Business Area

A business area is a collection of related information in the database. The Discoverer manager locates the information in the database and groups it into business areas. Within each business area, the Discoverer manager organizes information into folders.

For example, the key areas of a company's business might be sales, production, and human resources. Therefore, the Discoverer manager creates three corresponding business areas.

See Also: Creating and Maintaining Business Areas in the *Oracle Business Intelligence Discoverer Administration Guide* for additional information creating business areas in Discoverer.

LoanFlowPlus Example

The LoanFlowPlus example in the *Oracle BPEL Process Analytics Quick Start Guide* is a good example of how Oracle BPEL Process Analytics creates a star join schema that can be analyzed using Discoverer.

1. Log into the Oracle BPEL Process Analytics Admin Console. The following welcome page appears:
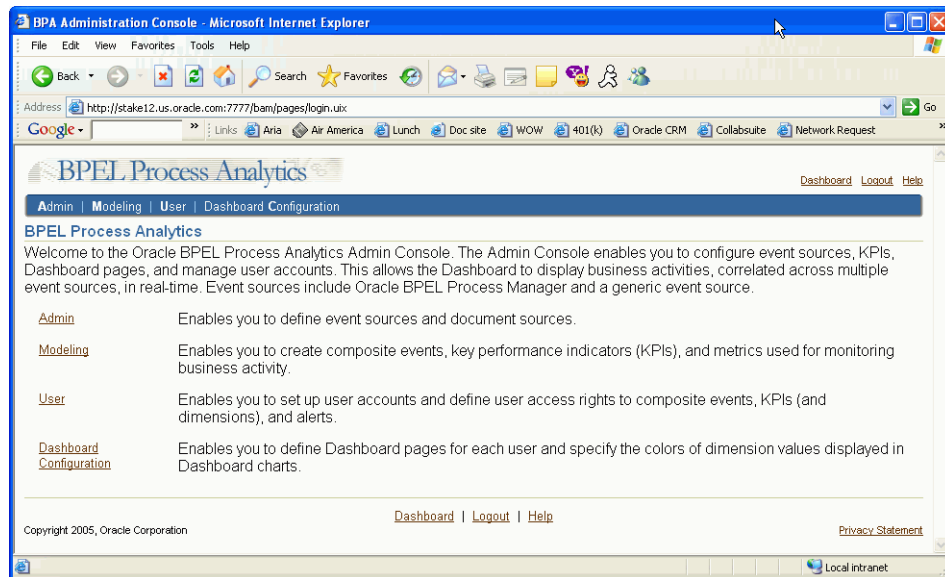
2. Select **Modeling**, either by clicking the link or by clicking **Modeling** on the menu bar.
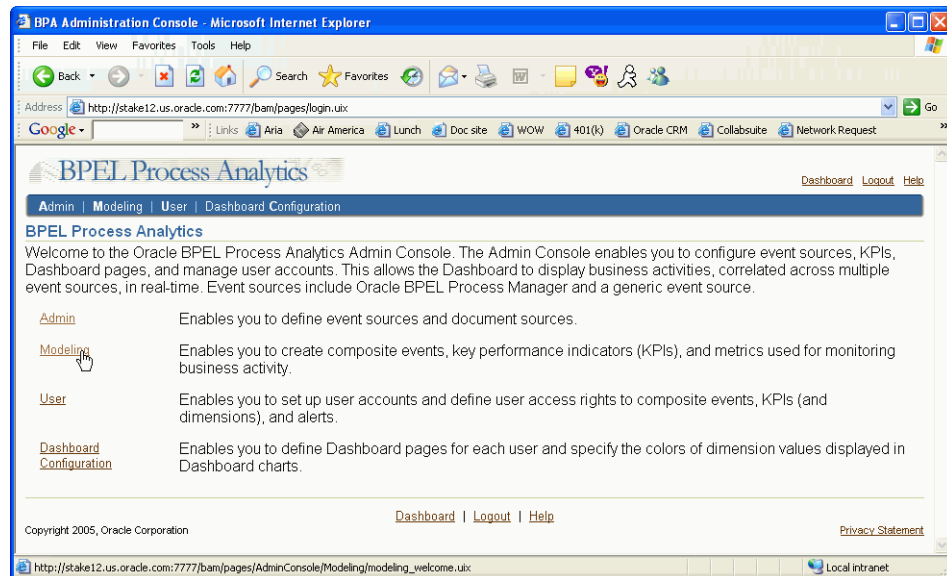
3. The Modeling page appears. The first task demonstrated is defining dimensions. Click the **Dimensions** link.
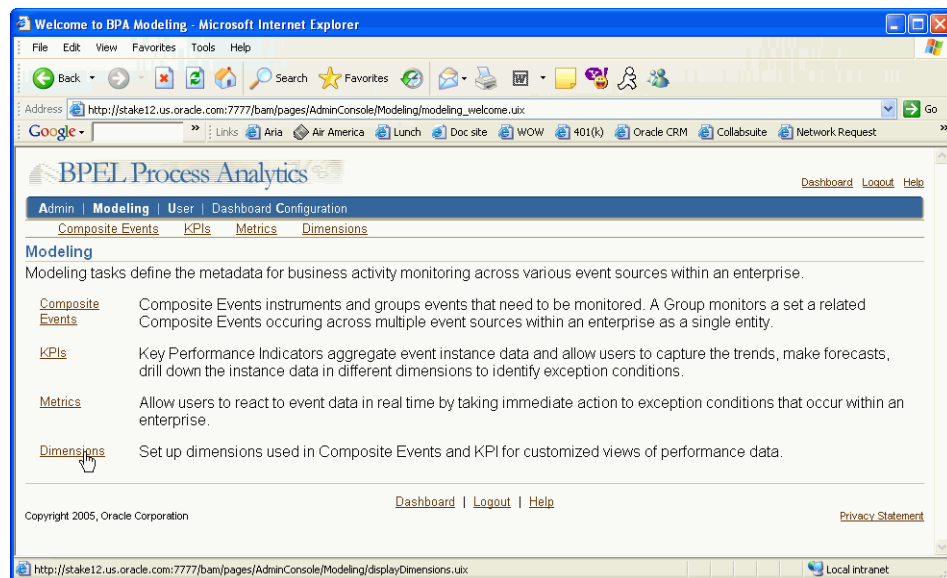
Defining Dimensions

The Dimensions page displays the current dimensions, and a **Create** button to define new dimensions. While the LoanFlowPlus tutorial already comes with a full set of dimensions and KPIs, the following pages demonstrate how to define a new dimension.

1. Click the **Create** button.
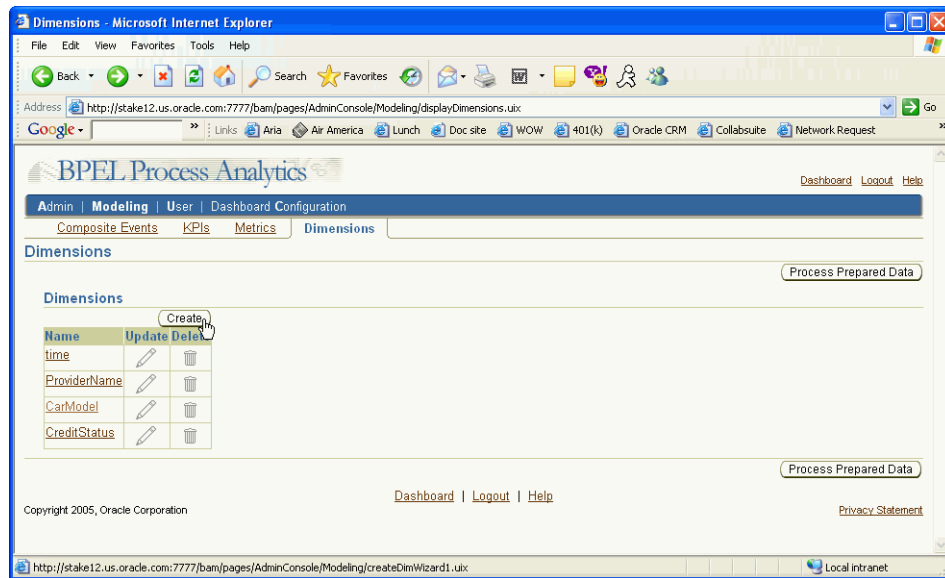
2. The following Dimension:Name page appears:
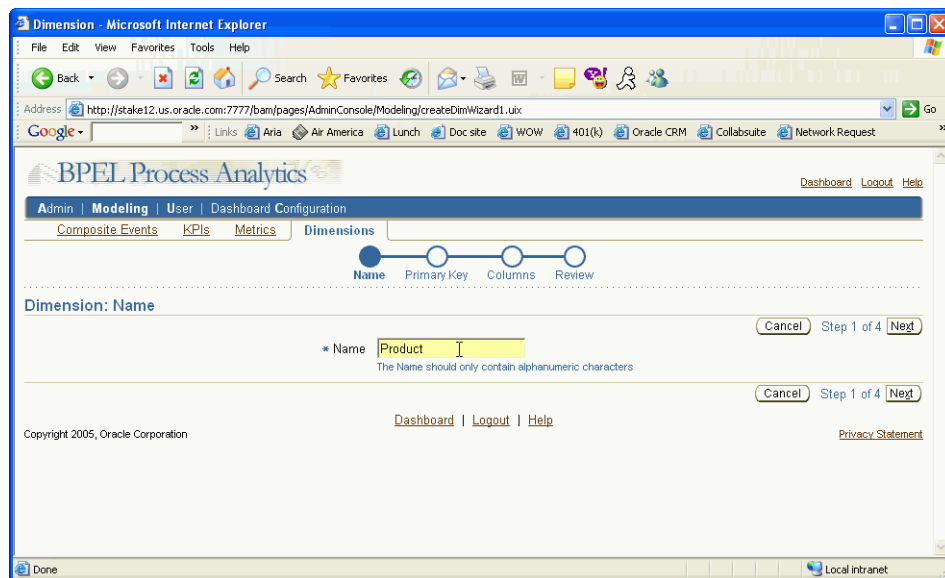
3. Enter the dimension name and click **Next**.

4. Select the data type for the dimension key column from the **data type** box. The possible choices are:

- **Integer**
- **Number**
- **Varchar2**

For Varchar2, you must also enter a column size.

5. Click **Next**.

Adding Columns to a Dimension

You may also add additional columns with different hierarchy ranks into the dimension. To add a new column, do the following:

1. Enter the **Column Name**, **Data Type**, and **Column Size** (if the data type is varchar2).

2. Click **Add**.

3. Repeat the previous steps to add addition columns.

4. Once you have added all the columns to the dimension, click **Next**.
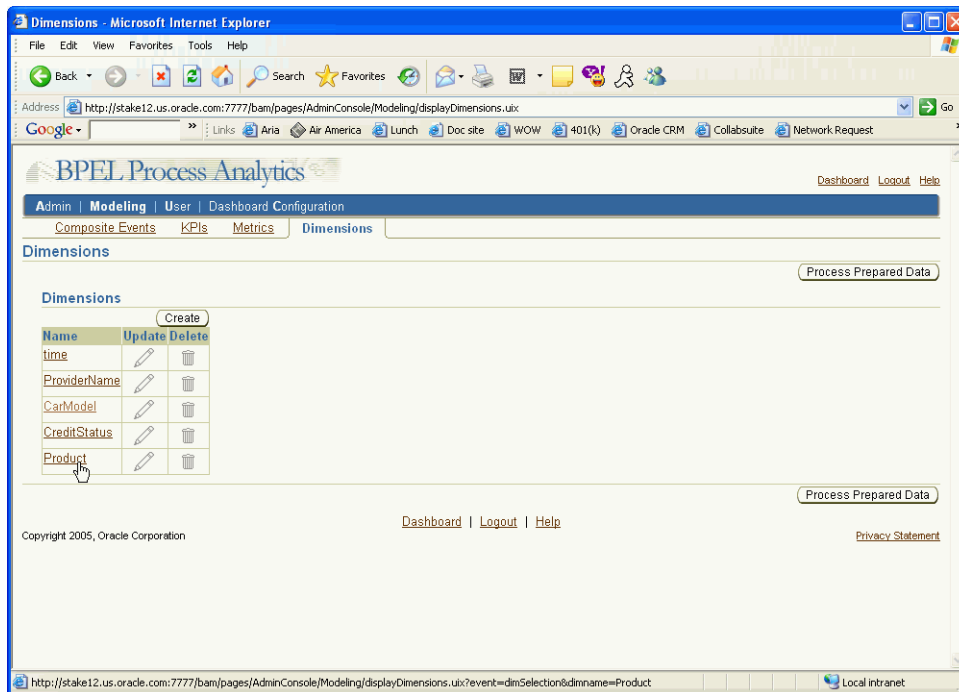
The Dimensions page appears again, now with the new dimension listed.

Defining KPIs

The next task is to define KPIs. Again, LoanFlowPlus comes with a full set of KPIs. The following demonstrates how to add additional KPIs.
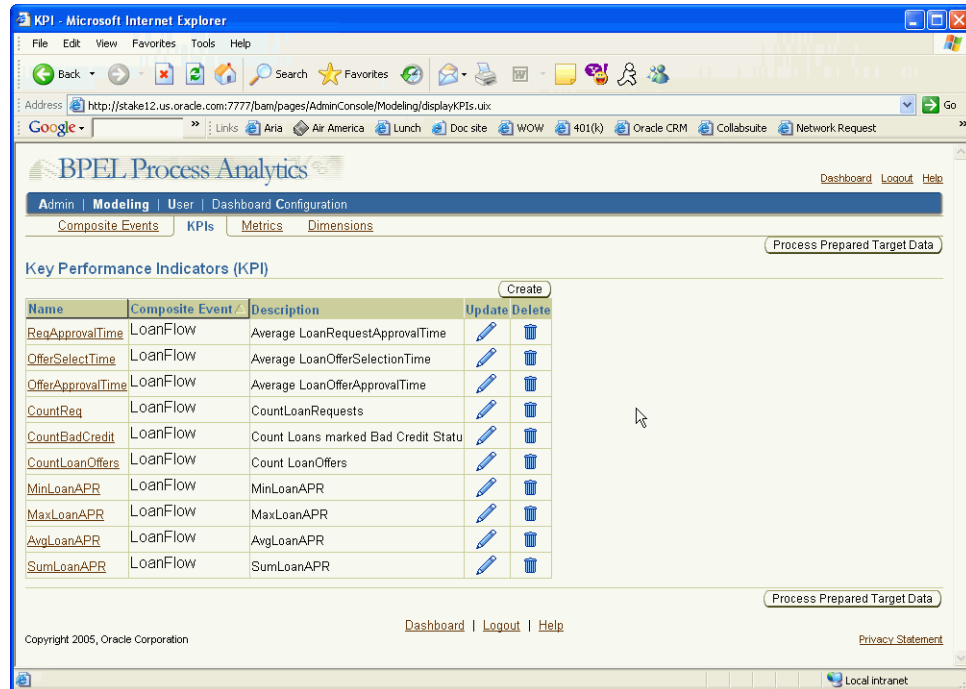
1. Click the **Create** button. The KPI:Name page appears:
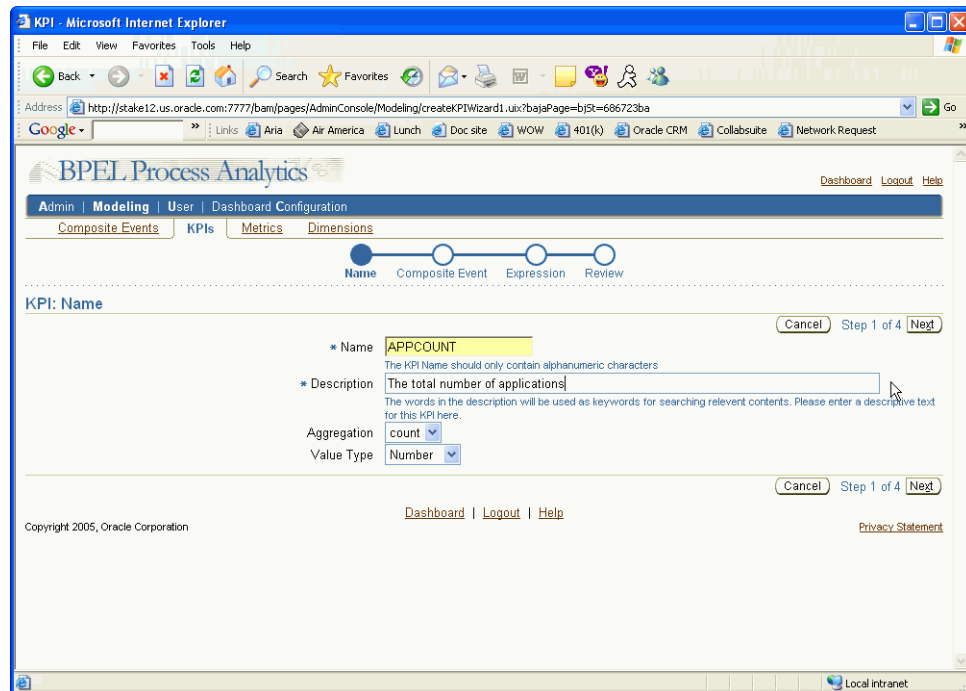
2. Enter the KPI name and description, and select the aggregation and the value type.

3. Click **Next**.
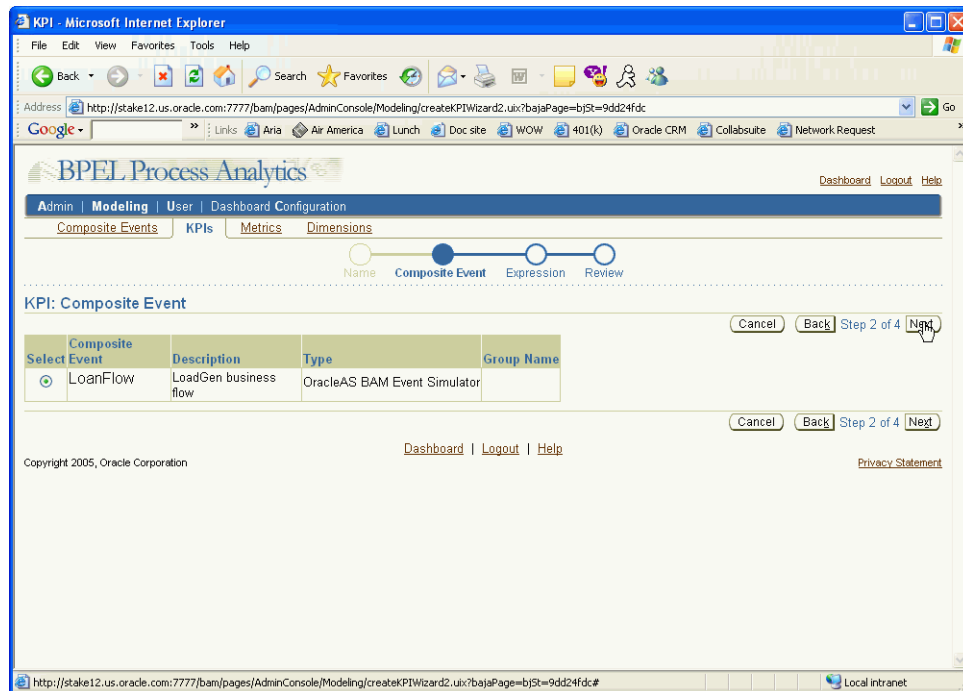
The Composite Events page lists the current composite events.

4. Map the event attributes to the KPI by selecting the composite event.

5. Click **Next**.
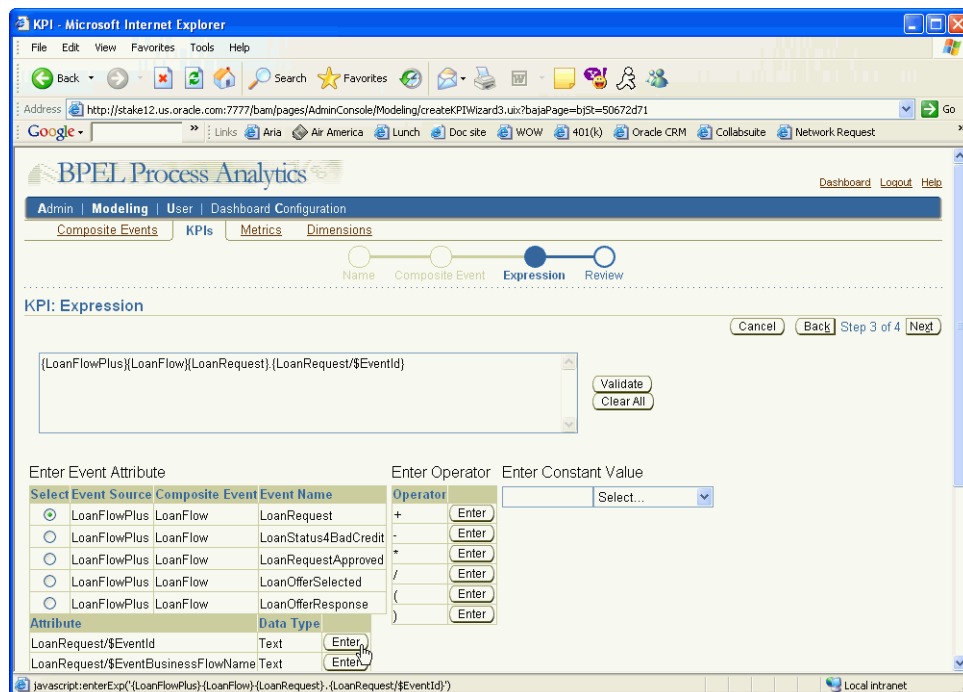
6. Compose the KPI expression by selecting the event name, attribute, operators, constants, and values.

7. Once the expression is complete, you may validate it by clicking the **Validate** button. A successful validation displays the following confirmation:

8. Click **Next**.

9. Review the KPI information. Click **Back** to make changes, or **Finish** if the information is complete.

The list of KPIs appears again, including the KPI you just defined:
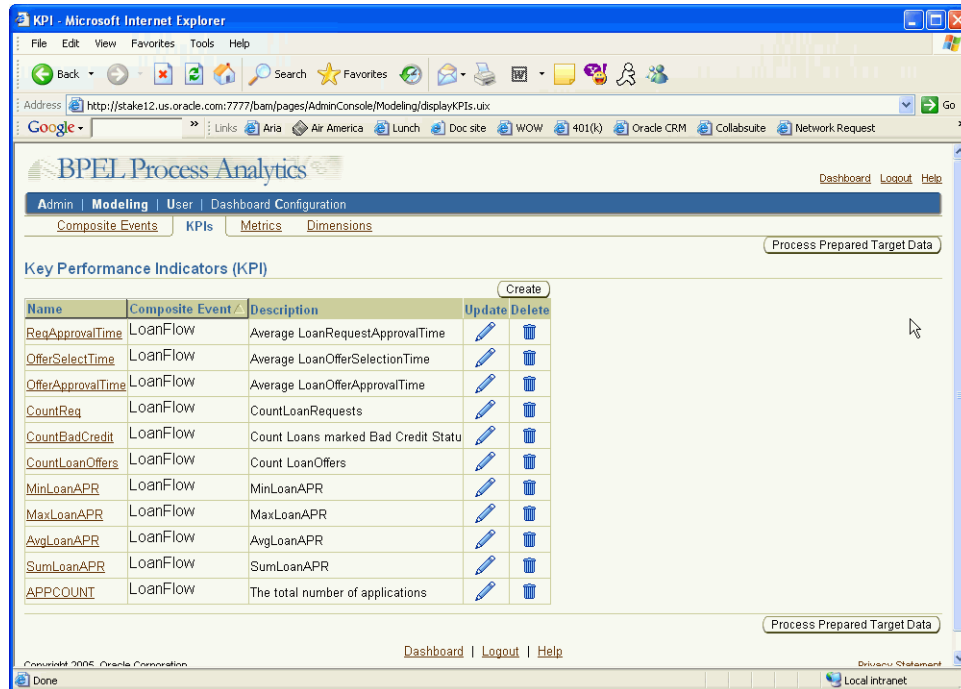
Assigning KPIs to a User

1. Click the **User** tab to begin assigning KPIs to a user.
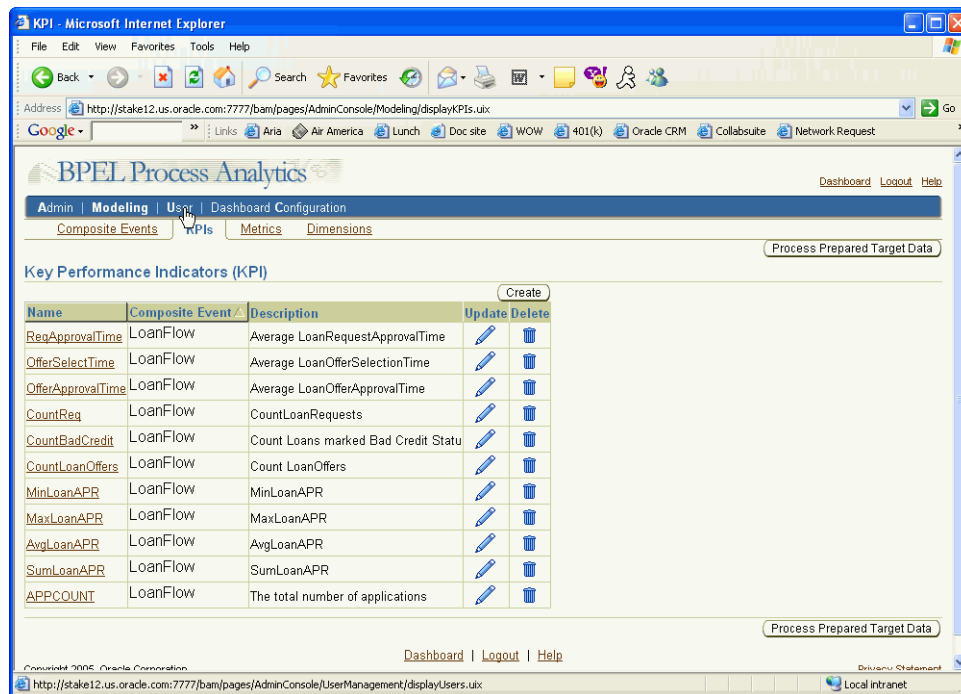
2. A list of current users appears. You may also add new users by clicking the **Create** button. The following example demonstrates how to add KPIs to an existing user.
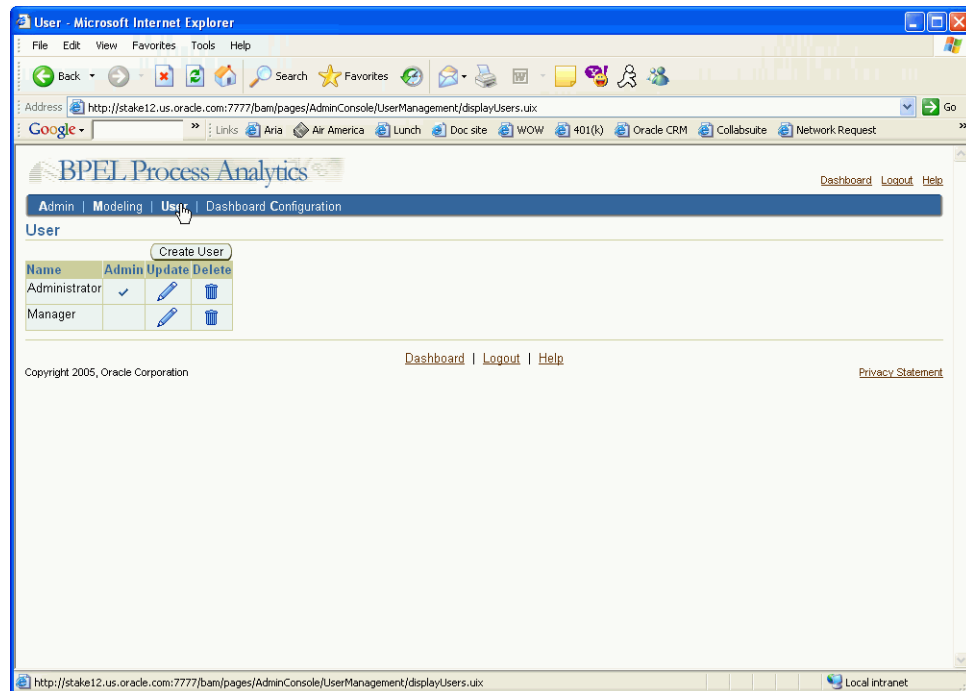
3. Select the **pencil** icon (update) in the **Administrator** row.

4. Select the **KPIs** tab.

A list of the current KPIs appears.

5. Add a new KPI by clicking on the **Create** button. The following example demonstrates how to add a KPI to that user's list.

6. Select the KPI from the KPI box. Click **Next**.

Here you may enter constrains on the KPI. In this case, there are no constraints.

7. Click **Next**.

8. Set performance bands by either selecting a template from the **Templates** box or by selecting **Custom** and defining your own.

9. Click **Next**.
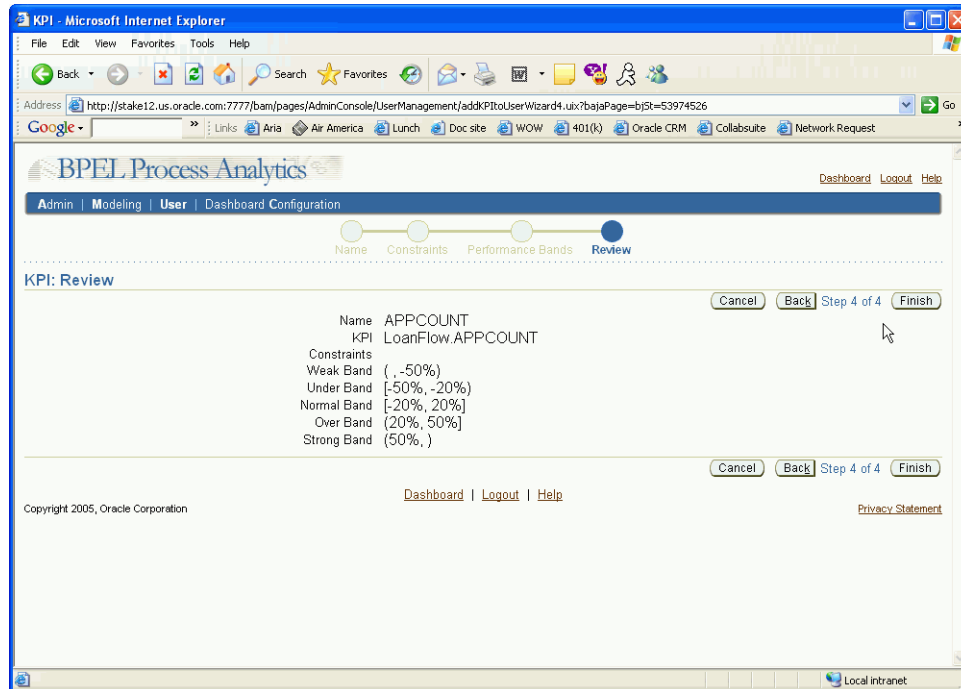
10. Review the KPI information. To make changes, click the **Back** button. To accept the KPI, click **Finish**.

Once you click **Finish**, the updated KPI list appears.
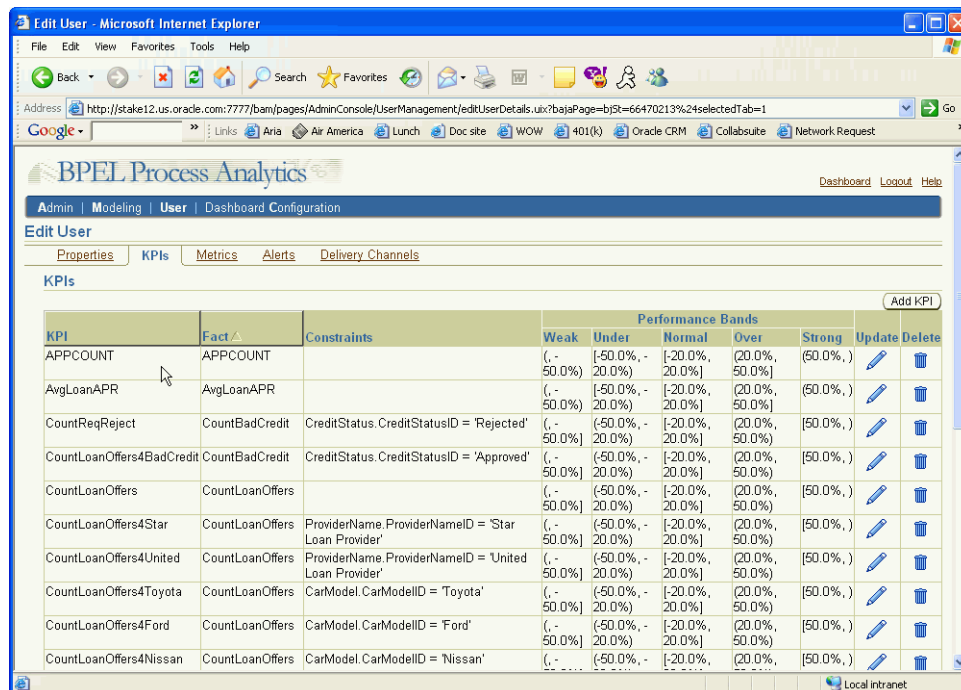
Loan Flow Example Star Join Schema

The star join schema for LoanFlowPlus consists of a fact table, BAM_FACT_ LOANFLOW1_T, and four dimension tables, BAM_DIM_PROVIDERNAME_T, BAM_DIM_ CARMODEL_T, and BAM_DIM_CREDITSTATUS_T, and BAM_DIM_TIME_ID.

There is another star join schema for the budget table, BAM_BUDGET_LOANFLOW1_T, with the same dimension tables. The entries in the budget table are the same as for the fact table.

The dimensions tables are joined to the fact and budget tables, forming the star join schema. Table 9–3 shows the BAM_FACT_LOANFLOW1_T fact table.

Table 9–3 BAM_FACT_LOANFLOW1_T

Name	Type	Null?
TIMEID	NUMBER	NOT NULL
CARMODELID	VARCHAR2 (25)	,
PROVIDERNAMEID	VARCHAR2 (25)	,
CREDITSTATUSID	VARCHAR2 (25)	,
REQAPPROVALTIME	NUMBER	
OFFERSELECTTIME	NUMBER	
OFFERAPPROVALTIME	NUMBER	
AVGLOANAPR	NUMBER	
MAXLOANAPR	NUMBER	
MINLOANAPR	NUMBER	
SUMLOANAPR	NUMBER	
COUNTREQ	NUMBER	
COUNTBADCREDIT	NUMBER	
COUNTLOANOFFERS	NUMBER	

Dimension Tables

The following dimension tables are associated with the fact and budget tables:

- BAM_DIM_PROVIDERNAME_T, as shown inTable 9–4.
- BAM_DIM_CARMODEL_T, as shown in Table 9–5
- BAM_DIM_CREDITSTATUS_T as shown in Table 9–6
- BAM_DIM_TIME_1 as shown in Table 9–7

Table 9–4 BAM_DIM_PROVIDERNAME_T

Name	Type	Null?
PROVIDERNAMEID	VARCHAR2 (25)	NOT NULL

Table 9–5 BAM_DIM_CARMODEL_T

Name	Type	Null?
CARMODELID	VARCHAR2 (25)	NOT NULL

Table 9–6 BAM_DIM_CREDITSTATUS_T

Name	Type	Null?
PROVIDERNAMEID	VARCHAR2 (25)	NOT NULL

Table 9–7   BAM_DIM_TIME_T

TIMEID
HOUR
DAY
MONTH
QUARTER
YEAR

Index

X

XML Schema Utility
    Java classpath and,   7-6
.xsd file
    creating for database table data,   7-6
    for event simulator,   8-2
XSU
    See XML Schema Utility