Oracle® Communications Pricing Design Center

User's Guide

Release 11.2

E97177-01

July 2018

ORACLE®

Oracle Communications Pricing Design Center User's Guide, Release 11.2

E97177-01

Copyright © 2011, 2018, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Contents

3 Working with Profiles

4 About Configuring Setup Components

5 Synchronizing Setup Components

6 Importing and Exporting Pricing and Setup Components

7 About Enriching Event Definitions

8 About Creating Pricing Components by Using the PDC Web Service

9 Replicating Data Between PDC Systems

10 Pricing Design Center Utilities

11 BRM and PDC Terminology

Preface

This guide provides an overview of Oracle Communications Pricing Design Center (PDC).

Audience

This document is intended for all PDC users.

Downloading Oracle Communications Documentation

Product documentation is located on Oracle Help Center:

http://docs.oracle.com

Additional Oracle Communications documentation is available from the Oracle software delivery Web site:

https://edelivery.oracle.com

Documentation Accessibility

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

Access to Oracle Support

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

Related Documents

For step-by-step instructions for tasks that you perform in PDC, log in to PDC and see the PDC Help.

1

About Pricing Design Center

This chapter provides an overview of Oracle Communications Pricing Design Center (PDC).

Overview of PDC

PDC is a Web-based application that you use to create and manage your product offerings. A product offering represents the services you offer to your customers and the price of those services. A service is a commodity, such as a mobile phone line or an Internet connection, that your customers can purchase and use.

Product offerings comprise of pricing components that contain the criteria your billing system uses to calculate the price of your services and to determine how the services are offered to your customers. Pricing components include the following:

- **Charge Offers**: Specify how to charge your customers for your services. A charge offer consists of one or more charges that specify the cost of each billable event that affects a customer's account.
- **Discount Offers**: Specify when and how to reduce the cost of a billable event or to change the balance of an element such as free minutes or loyalty points. Discount offers consist of one or more discounts.
- **Chargeshare Offers**: Specify which events qualify for charge sharing and the conditions that must be met for charge sharing to apply. Additionally, they contain the pricing to determine the charge sharing amounts and balance impacts.
- **Bundles**: Contain one or more charge offers, discount offers, or both. You create offers as the building blocks of bundles and can reuse them in multiple bundles.
- **Packages**: Contain one or more bundles. You use packages to offer one or more services to customers. For example, you can offer one service with two different ways of charging for it, each defined in a separate bundle. When you create a package, you choose which of the bundles to include in the package.
- **Package List**: Contain a group of packages that is usually offered to a single type of customer. To make packages available for customers to purchase, you must include them in a package list.
- **Selectors**: Contain one or more rules based on the values of attributes in an event. Selectors are used in charge offers and discount offers to select between multiple charges, discounts, impact categories, or pricing instances.
- **Time Models**: Contain a set of time periods that consists of time segments that represent a particular time. Time models are used in charge offers to charge different prices for the same service depending on the day and time the service is used.

See "About Creating Product Offerings" for more information.

You also use PDC to configure setup components, which are the prerequisite data used to create pricing components. Setup components include the following:

- Extended pricing features, such as time models, that enables you to apply different prices based on the time the event occurred
- Ratable usage metrics (RUMs), which contain a set of information used by the rating engine to measure an event
- A service-event map, which links each service your system supports to the events that can have charges configured for that service
- Zone models, which are used to charge for calls based on their origin and destination

See "About Configuring Setup Components" for more information.

To help you efficiently manage product offerings, PDC provides changeset management for grouping and publishing setup and pricing components. See "About Changesets" for more information.

PDC is based on open application standards and is compliant with the TeleManagement Forum Shared Information/Data (TMF SID) model.

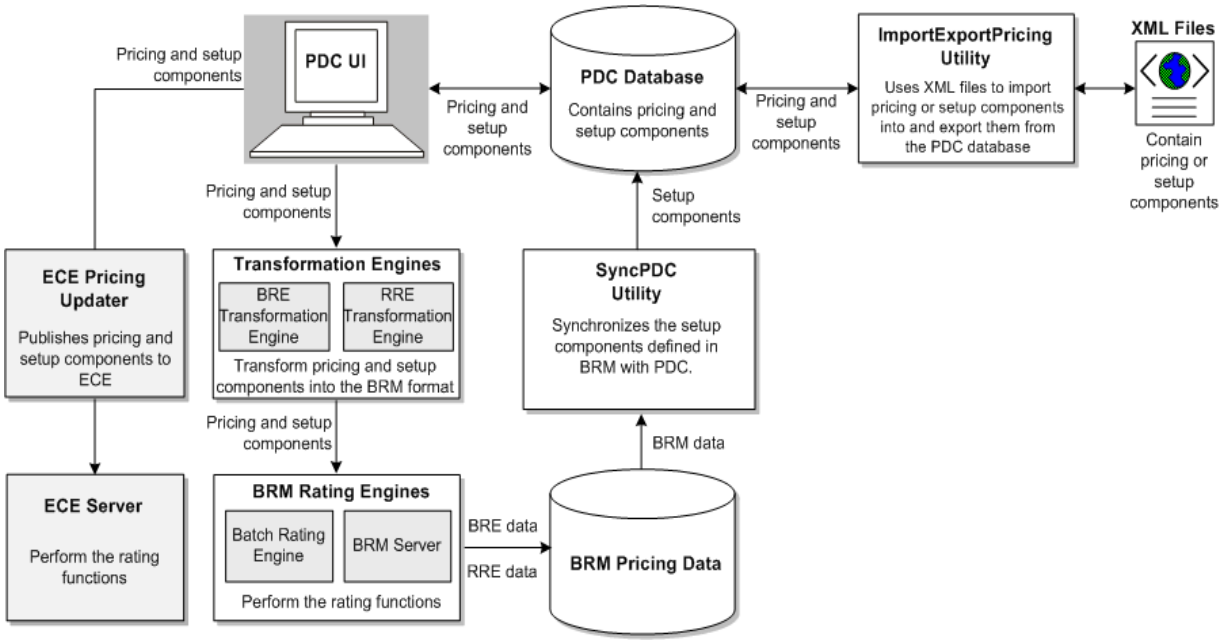
About the PDC Application Architecture

The PDC application architecture consists of the following components:

- **PDC UI**. The PDC UI is a Web-based user interface that you use to create and manage your product offerings.
- **PDC database**. The PDC database stores the pricing components and setup components that you configure in PDC, as well as the pricing data defined in Oracle Communications Billing and Revenue Management (BRM) and rating systems that are synchronized with PDC.
- **BRM Integration Pack**. The BRM Integration Pack provides the interface between PDC and BRM.
- **ImportExportPricing utility**. The **ImportExportPricing** utility provides an XML-based command-line interface for importing pricing and setup components to and exporting pricing and setup components from PDC.

Figure 1–1 shows the PDC application architecture and the process flow when used with BRM and ECE:

Figure 1–1 PDC, BRM, and ECE Integration Architecture



About the PDC and BRM Integration Process Flow

The BRM Integration Pack provides the interface between PDC and BRM to transform and load the pricing data configured in PDC into BRM and to synchronize the pricing data defined in BRM and the rating system with PDC.

Note: PDC supports BRM 7.5.

The BRM Integration Pack contains:

- **Transformation engines**: When you use real-time and batch rating engines for usage charging, the transformation engines convert pricing components and setup components configured in PDC into the format required by BRM and then loads the components into the BRM database.
- **SyncPDC utility**: The **SyncPDC** utility provides the capability to synchronize setup components defined in BRM with PDC. It also synchronizes event data defined in the rating system with PDC. See "Synchronizing Setup Components" for more information.

BRM uses the pricing and other data to measure customer activity and determine how much to charge the customer for the use of the services. See BRM documentation for information on how the rating is performed.

About the PDC and ECE Integration Process Flow

When you use ECE for usage charging, the ECE Pricing Updater automatically converts pricing components and setup components configured in PDC into the format required by ECE and then loads the components into ECE. The ECE Pricing Updater is packaged with ECE. See ECE documentation for more information.

ECE uses the pricing data and the pricing-related configuration data defined in PDC to rate the usage requests it receives. See ECE documentation for information on how ECE rates the usage requests.

2

About Creating Product Offerings

This chapter describes concepts that you should understand before using Oracle Communications Pricing Design Center (PDC) to create product offerings.

Before reading this chapter, you should have a basic understanding of PDC. See "About Pricing Design Center" for more information.

About Product Offerings

You create product offerings to define the services you offer and how much to charge for those services. Your billing system evaluates billable actions (called events) based on the product offerings to determine how much to charge for the use of your services.

About Events

An event is an action that is recognized by your billing system and recorded in the system's database. For example, when a customer makes a mobile phone call, an event is generated that contains information about that phone call, such as the time the call was made, the origin and destination of the call, and the duration of the call. An event can also be system-generated, such as monthly subscription fees that are applied to accounts. The billing system uses the data it collects about events to calculate how much to charge customers for them.

To determine the cost of an event, your billing system performs the following actions:

1. Measures the event based on the information in the applicable charge offer (see "About Measuring Events"). For example, it might measure the duration of a phone call in minutes.
2. Applies a charge to the resulting measurement based on specifications in the applicable charge offer or discount offer (see "About Applying Charges to Events").
3. Adds the charge to the customer's account balance (see "About Balance Elements").

See "About Configuring Setup Components" for more information about configuring events in PDC.

About Measuring Events

Before your billing system can apply a charge to an event, it must measure the event. To enable your billing system to measure events, you configure ratable usage metrics (RUMs), which specify the units to measure and how to calculate the measurement.

You can base a measurement on any data captured in an event, such as how long a event lasted or the number of bytes downloaded during a event.

Common types of event measurements include the following:

- **Duration:** Length of an event in units such as seconds or minutes
- **Occurrence:** Number of events generated during a specified period
- **Volume:** Size of an event in units such as kilobytes or megabytes

Typically, a single RUM is used to measure an event. But you can also use multiple RUMs to measure an event.

About Applying Charges to Events

After measuring an event, your system must apply a charge to the measurement. To determine the charge to apply, you can use attributes or quantities (or both).

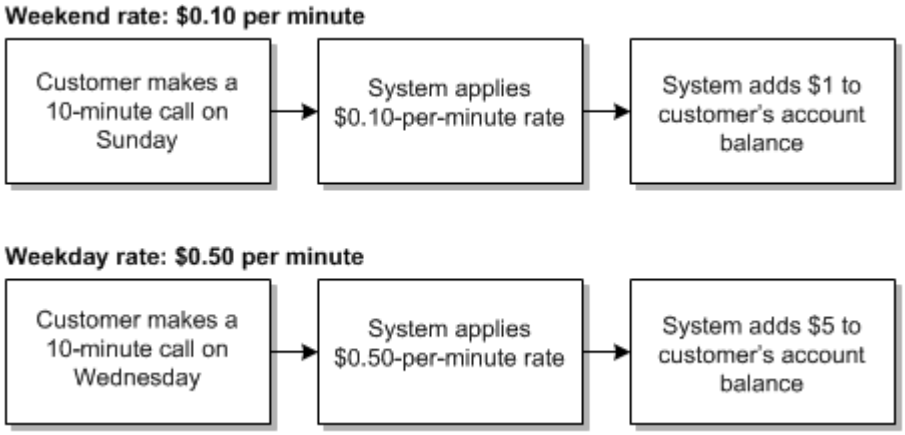
Charges Based on Attributes

An attribute is a characteristic of an event, such as:

- Date and time a customer connected to your Web site
- Type of movie downloaded, such as classic, action, or comedy
- Origin and destination of a call
- Quality of service, such as standard or premium

Different pricing can be associated with different values of the same attribute. For example, Figure 2–1 shows how two 10-minute calls can be priced differently based on the day they occurred.

Figure 2–1 Charges Based on the Day an Event Occurs



You can also base pricing on the attribute values.

For example, the values of the event attribute that stores the call destination might be associated with the following charges:

- If the call destination is France, use the Calls to Europe charge.
- If the call destination is Ghana, use the Calls to Africa charge.

To implement attribute-based charges, you can use zones, selectors, and time periods. See "About Zone Models", "About Value Maps", "About Selectors", and "About Time Models" for more information.

Charges Based on Quantities

You can vary pricing for events by basing your charges on the following quantities:

- *Usage,* such as the number of minutes in a phone call. For example:
  - 10 cents per minute for the first 30 minutes
  - 5 cents per minute for any time over 30 minutes
- *Balance*, such as the number of loyalty points accrued or the number of faxes sent during one billing cycle. For example:
  - $1 per fax for the first 10 faxes
  - 50 cents per fax for the next 90 faxes
  - No charge for any faxes after the 100th fax

You can also base discounts on quantities. For example, you could take 10% off a call's normal per-minute pricing for any time over 60 minutes and 20% off for any time over 120 minutes.

To implement quantity-based charges, you specify quantity ranges. See "About Quantity Ranges in Pricing" for more information.

About Balance Elements

A balance element represents one of the following:

- A currency or noncurrency asset of economic value, such as U.S. dollars or included minutes
- A counter that tracks items such as dollars spent or minutes talked

Balance elements are specified when you configure pricing in a charge and are debited, credited, increased, or decreased when the charge offer is used to rate an event. For example, a charge of one dollar per minute for a phone call affects the US Dollars balance element.

Before creating any charge offers, you should configure currency balance elements for the system currencies and account currencies that you support.

Typically, you create noncurrency balance elements to track noncurrency account balances (such as minutes, bytes, loyalty points, and frequent flyer miles) as the need arises for particular charge offers.

See "About Configuring Setup Components" for information about configuring balance elements in PDC.

About Product Offerings

A product offering represents the services available to your customers and the price of those services. A service is a commodity, such as a mobile phone line or an Internet connection, that your customers can purchase and use.

To determine how much to charge for your services, your billing system uses data collected about them (such as the length of a phone call or the number of bytes

downloaded) to calculate their economic value. The criteria on which the billing system bases its calculations are specified in a product offering.

In PDC, a product offering consists of the following components:

- **Charge Offers and Discount Offers:** Contain the criteria that the billing system uses to determine the cost of a service. For example, a charge offer for a mobile phone service might include criteria for calculating the following charges:
  - A setup charge
  - A monthly subscription charge
  - Usage charges for phone calls

  A discount offer for a mobile phone service might include criteria for calculating the following discounts:
  - 50% off the monthly charge for the first 6 months
  - 25% off for usage over 750 minutes
  - Reduction in usage charges if the subscriber has included minutes

  See "About Charge Offers" and "About Discount Offers" for more information.
- **Bundles:** Contain one or more charge offers, discount offers, or both. You can reuse charge offers and discount offers in multiple bundles.

  For example, you might mix and match charge offers and discount offers for a mobile phone service to create the following bundles:
  - **Bundle A: Basic Voice**
    * **Charge Offer A:** $300 setup fee
    * **Charge Offer B:** $50 monthly fee
    * **Charge Offer C:** $0.50 per minute for all minutes over 300 per month
    * **Discount Offer A:** 25% off usage over 750 minutes per month
  - **Bundle B: Promotional Voice**
    * **Charge Offer A:** $300 setup fee
    * **Charge Offer D:** $20 monthly fee
    * **Charge Offer E:** $1 per minute for all minutes over 300 per month
    * **Discount Offer A:** 25% off usage over 750 minutes per month
    * **Discount Offer B:** No setup fee if the service is purchased before a specified date
    * **Discount Offer C:** 50% off monthly fee for the first 6 months

  All offers in a bundle must be associated with the *same* service.

  See "About Bundles" for more information.
- **Packages:** Contain one or more bundles. You use packages to offer services to customers. For example, if your company provides Internet access and VOIP (voice over Internet protocol), you might create the following packages:
  - **A package that offers only Internet access.** This package would contain one or more Internet access bundles, such as a high-speed Internet access bundle and a mobile Internet access bundle. For example, see the Premium Internet package in Figure 2–2.

- **A package that offers only VOIP.** This package would contain one or more VOIP bundles, such as a standard calling plan bundle and an international calling plan bundle. For example, see the International VOIP package in Figure 2–2.
- **A package that offers Internet access and VOIP.** This package would contain at least one Internet access bundle and at least one VOIP bundle. For example, see the Internet + VOIP Double-Play package in Figure 2–2.

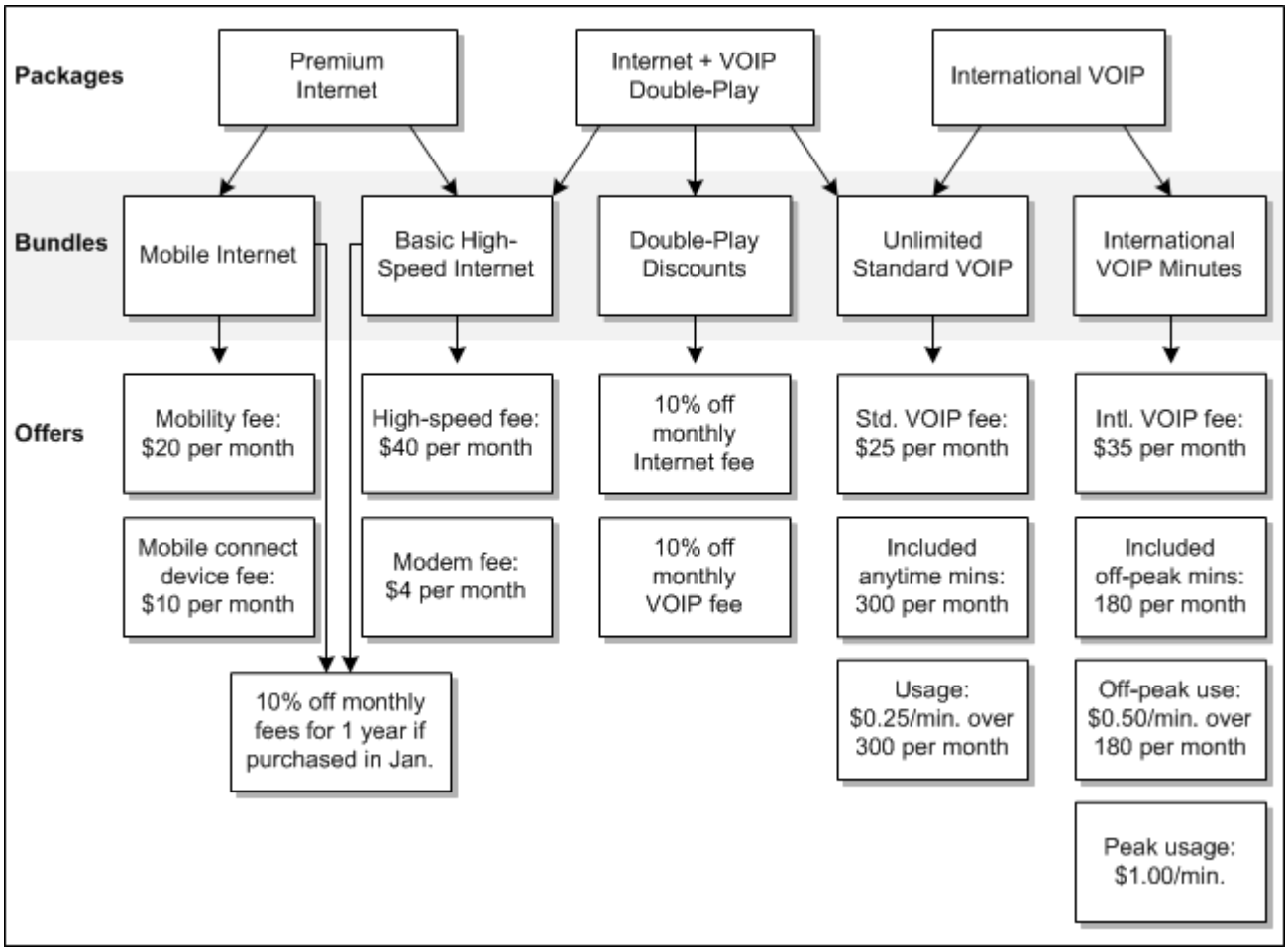
A product offering contains one or more packages. To subscribe to your services, a customer purchases a package.

See "About Packages" for more information.

Example of a Product Offering

Figure 2–2 shows how a product offering is organized.

Figure 2–2 Organization of a Product Offering



Charge offers and discount offers are specified at the lowest level. Therefore, to set up a product offering, you start by creating charge offers and discount offers. See "About Charge Offers" and "About Discount Offers" for more information.

Sample Product Offerings

PDC includes sample product offerings for a broadband service and for a telco service in the *PDC_home*/**apps/samples/data** directory, where *PDC_home* is the directory in which you installed PDC.

For information about importing the sample product offerings into PDC, see the post-installation section of *PDC Installation and System Administration Guide*.

About Charge Offers

Charge offers determine the price of one or more events associated with a service.

This section provides an overview of the information required to configure charge offers. For information on how to create charge offers, see the PDC Help.

Charge Offer Types

PDC supports the following charge offer types, which represent different types of fees:

- **Item:** Contain only a one-time purchase fee. After an item charge offer is used by a billing system, it usually no longer needs to be stored in the database.

  For example, a promotional t-shirt, an installation charge, and a setup charge are all item products.

- **Subscription:** Contain fees for any type and combination of events, including one-time, recurring, usage, rollover, remittance, and fold. These fees apply only to the subscriber who owns the charge offer.

- **System:** Contain fees that apply to all subscribers who use a particular service. System charge offers are not owned by subscribers. They can contain only usage charges.

  Use system charge offers when you do not want to re-create the same charge multiple times in your product offering. For example, use a system charge offer to charge a default usage rate of $0.10 per minute for mobile phone usage when other mobile phone offers are not valid.

Making a Charge Offer Applicable to a Service

When creating a charge offer, you can associate it with a specific service. That is the only service to which the charge offer applies.

Multiple charge offers can be associated with the same service.

After you associate a charge offer with a service, PDC uses the service-event map to limit the events to which the charge offer can apply.

Note: If you create a charge offer that does not apply to a specific service, associate the charge offer with **Account**. For example, you might do this for late charges or for coupons. The charge offer can then be used to rate any event associated with **Account** in the service-event map.

About the Service-Event Map

Before you can create charge offers, you must set up a service-event map. The map lists all the services to which charge offers can apply. For each service, the map

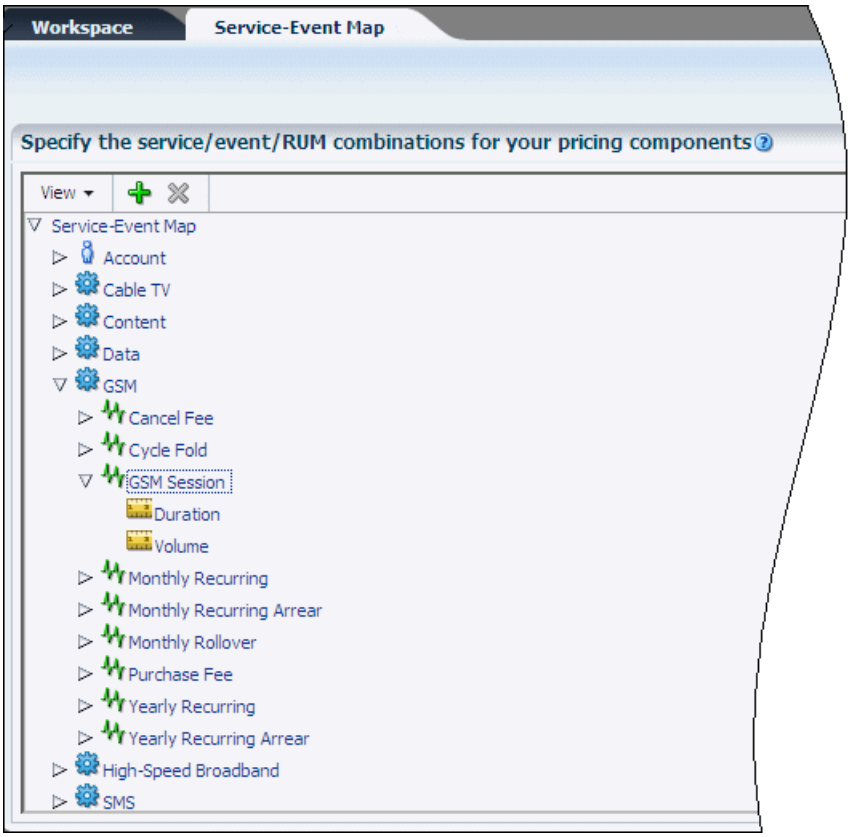
specifies which events can have charges configured for that service. The map also specifies the RUMs to use for each service-event combination.

For events that do not apply to a specific service, you map those events to **Account** in the service-event map.

Note: Remittance charges can be associated only with accounts. Therefore, you map remittance events to **Account** in the service-event map. See "About Charge Categories" for information about remittance charges.

For example, in Figure 2–3, the events Cancel Fee, Cycle Fold, GSM Session, Monthly Recurring, and Monthly Recurring Arrear are mapped to the GSM service. The RUMs, Duration and Volume, are mapped to the GSM Session event.

Figure 2–3 Service-Event Map



See "About Configuring Setup Components" for information about configuring the service-event map.

Specifying When a Charge Offer Can Be Purchased

You can make a charge offer available for purchase as follows:

- Always available
- Available from a future date forward
- Available until a future date

- Available for a specified period in the future

To be added to a bundle, a charge offer must have a purchase period that is the same as or greater than the bundle's purchase period. See "Specifying When a Bundle Can Be Purchased" for more information.

Using Provisioning Tags to Configure Special Charges

In Oracle Communications Billing and Revenue Management (BRM), provisioning tags implement extended rating attributes (ERAs), which provide special charges or discounts based on a specific attribute of a service or an account, such as a telephone number. For example, a provisioning tag could specify that a charge offer discounts the following:

- Calls to friends and family
- Calls to mobile numbers within a closed user group
- Calls made on a subscriber's birthday

For telco services, you can also use provisioning tags to configure service extensions and supplementary services, such as call forwarding and call blocking.

See "About Configuring Setup Components" for information about importing provisioning tags into PDC.

Setting the Priority of Charge Offers

When multiple charge offers apply to the same service-event pair, your billing system can consider the charge offers in the order of their priority, which you specify.

If a portion of the event remains unrated after the billing system applies all the charges in the charge offer with the highest priority, each subsequent priority charge offer is analyzed until the entire event is rated.

Restricting the Ownership of Charge Offers

You can specify the minimum and maximum number of charge offers that a customer can purchase at one time. For example, if a charge offer includes an item such as a t-shirt, you might want to limit the number of t-shirts that can be purchased simultaneously.

You can also specify the minimum and maximum number of charge offers that a customer can own at one time. For example, if a charge offer provides an email service, you might want to limit the number of email login names that a customer can own.

Enabling Partial Purchase of Charge Offers

If your system permits customers to purchase only part of a charge offer, you can specify whether a charge offer qualifies for partial purchase.

For example, if a charge offer gives customers 300 minutes of off-peak calls for $30, customers might be permitted to purchase half that amount for half the price.

Adding Charges to Charge Offers

To enable your billing system to determine the price of a service, you configure charges for events associated with the service. A charge contains detailed information about how to calculate the cost of a particular event. Your billing system uses this information to determine the impact of the credit, debit, or counter balance to apply to

a subscriber's account when that event occurs. A charge offer can include charges for multiple events.

For example, to charge purchase, monthly account, and usage fees for a mobile phone service, you might configure charges for the following types of events:

- Purchase
- Monthly subscription
- Session

Whether the preceding charges are configured in the same charge offer or in separate charge offers depends on the services you offer and how you manage your business and your customers. For example, you might change your pricing often, provide sponsored services, or provide numerous types of discounts. Before you finalize a product offering, test it against some discounting and customer management scenarios to determine whether your charge offers are organized in a way that best supports your business.

You can add charges to a charge offer as follows:

- **New Charge:** Enables you to create a charge.
- **Existing Charge:** Enables you to reuse a charge that was created for another charge offer. You can reuse a charge in either of the following ways:
  - Share the charge with other charge offers and charge selectors. Changes made to the charge affect all the components that share it.
  - Use a copy of the charge. Copying enables you to modify a charge in ways that are not applicable to the components using the original charge.
- **Existing Charge Selector:** Adds a prioritized list of preconfigured charges to the charge offer. When the event to which the charge selector applies occurs at run time, your billing system selects a charge from the list based on the value of one or more of the event, service, or subscriber (account) attributes. This mechanism enables you to charge different fees for the same event without creating multiple charge offers. See "About Selectors" for more information.

The following sections provide an overview of the information required to configure charges:

- Specifying Charge Details
- Configuring Pricing in Charges

For more information, see the PDC Help.

Specifying Charge Details

When creating a charge, you must specify the information described in this section.

For more information, see the PDC Help.

About Charge Categories

You must specify a category for each charge. The charge category helps determine which events the charge can be configured for.

PDC supports the following charge categories:

- **One-time:** Nonrecurring charges, such as setup or cancellation fees.

- **Recurring:** Ongoing charges that are not generated or affected by usage, such as a monthly subscription fee.
- **Usage:** Charges for the use of a service, such as telephone calls or Internet sessions.
- **Rollover:** Charges that extend the validity of unused balances to succeeding cycles. For example, included minutes are often rolled over.
- **Remittance:** Charges that calculate the manner in which you share revenue with third parties, such as resellers or service providers. For example, you could pay them a percentage of your subscriber fees or a flat amount per subscriber. This category is available only for charge offers that apply to accounts.
- **Fold:** Charges used to zero-out a balance or convert one balance into another. For example, you could configure a fold charge to zero-out unused included hours at the end of each month or convert frequent flyer miles to a dollar amount. Fold charges must be associated with an existing charge selector whose event matches the fold event (see "About Selectors").

About Charge Types

The charge type is the event to which the charge applies. The service to which the charge offer applies and the charge category determine which charge types are available for selection. See "About Events" for more information.

About Associating Pricing Profiles with Charges

A pricing profile tells PDC which features to support in its user interface. The charge category and charge type determine which profiles are available for selection. See "Working with Profiles" for more information.

About Measuring Charges

In each charge, you must specify the RUM to use to measure the applicable event (see "About Measuring Events"). When you create or modify a charge, the charge type determines which RUMs are available for selection.

About Charge Currencies

The currency specified for a charge, such as U.S. dollars or euro, is the only currency balance element that you can select when configuring pricing for the charge (see "About Balance Elements").

By default, charges use the system currency. If necessary, you can select an alternative currency for a charge when you create the charge.

About Using Extended Pricing Features

Extended pricing features (such as impact categories, time periods, price overrides, and price selectors) enable you to use the same charge to apply different prices based on various attributes of the event, such as where and when it occurred (see "Charges Based on Attributes").

One-time, recurring, usage, and remittance charges support extended pricing features.

About Cycle Settings for Recurring Charges

The following options apply only to recurring charges:

- **In-Advance Billing:** Use the in-advance billing option to specify how many months or days to charge customers for in their first bill.

For example, if a customer purchases an offer on May 1 and the offer's $10 monthly fee is billed three months in advance, the total charge in the customer's first bill is $30. The total charge in the next (June) bill is $10, but from an accounting perspective, the $10 fee applies to August, not to June.

- **Cycle Alignment:** The cycle alignment option enables you to apply recurring charges in either of the following ways:
  - On the product purchase date. For example, if the billing date is the 1st of the month and the charge offer is purchased on January 10, the charge is applied on the 10th of every month (for the interval January 10 to February 10, February 10 to March 10, and so on).
  - On the customer's current billing date (default). Using the previous example, the charge is prorated and applied on January 10 for the interval January 10 to February 1. For subsequent cycles, the charge is applied on the billing date (for the interval February 1 to March 1, March 1 to April 1, and so on).

About Prorating Recurring Charges and Rollovers

Customers typically purchase and cancel charge offers at some point in the middle of a billing cycle. When you set up recurring charges and rollovers in PDC, you can specify whether the charge or rollover amounts are prorated for the first and last billing cycles based on the number of days in the cycles that the charge offer is owned.

For recurring charges, you can specify whether to charge the full amount, prorate the charge, or apply no charge.

For rollovers, you can specify whether to roll over the entire amount, prorate the rollover amount, or not roll over any of the available balance.

About Charge Date Ranges

The charge date range is the period during which a charge is effective. By default, this period starts immediately and never ends.

You can modify the default date range and add more date ranges. Adding date ranges enables you to make structural changes to a charge for different time periods.

For example, if you plan to add an impact category to the charge in January, you can change the current date range to end January 1 (end dates are exclusive) and add a date range that starts January 1 (start dates are inclusive) and never ends. The charge information from the preceding range is copied into the new range, where you can add the impact category.

PDC supports the following types of charge date ranges:

- **Fixed:** Specifies a period that starts and ends on particular dates. For example:
  - Immediately through 6/1/2012
  - 6/1/2012 through 1/1/2013
  - 1/1/2013 through never ends

  Fixed date ranges cannot overlap.
- **Relative:** Begins at a time relative to the time the charge is purchased and continues for a specified length of time, such as days, hours, minutes, or seconds. The purchase date is the day the charge offer is added to the account.

  Relative date ranges are primarily used as price incentives. For example, you might offer a broadband Internet connection for $49 for the first six months and then raise the price to $99.

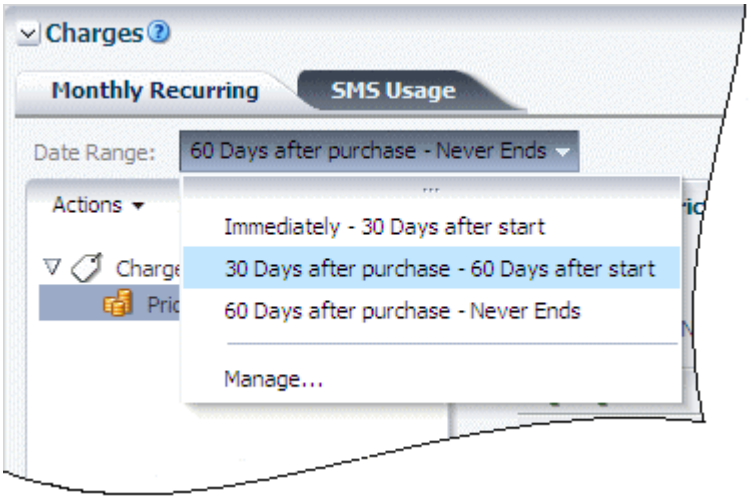
Unlike fixed date ranges, relative date ranges can overlap. For example, a usage charge for a mobile phone service might contain the following relative date ranges:

- 3000 included minutes: Starts Immediately until 90 days after start
- Unlimited included SMS messages: Starts Immediately until 30 days after start
- $0.15 per SMS message: 30 days after start and never ends

The pricing profile associated with a charge determines which types of date ranges can be used for the charge (fixed, relative, or both). A charge can have multiple instances of both types.

When a charge has multiple date ranges, a **Date Range** list appears at the top of the charge tab to enable you to display the Charges tree and pricing details for each date range. By default, the current fixed date range is displayed.

Figure 2–4 Charge Date Range List with Relative Ranges



Configuring Pricing in Charges

After you enter the details for a charge (see "Specifying Charge Details"), you can configure pricing for the charge. Before configuring pricing, however, you should understand the concepts described in this section.

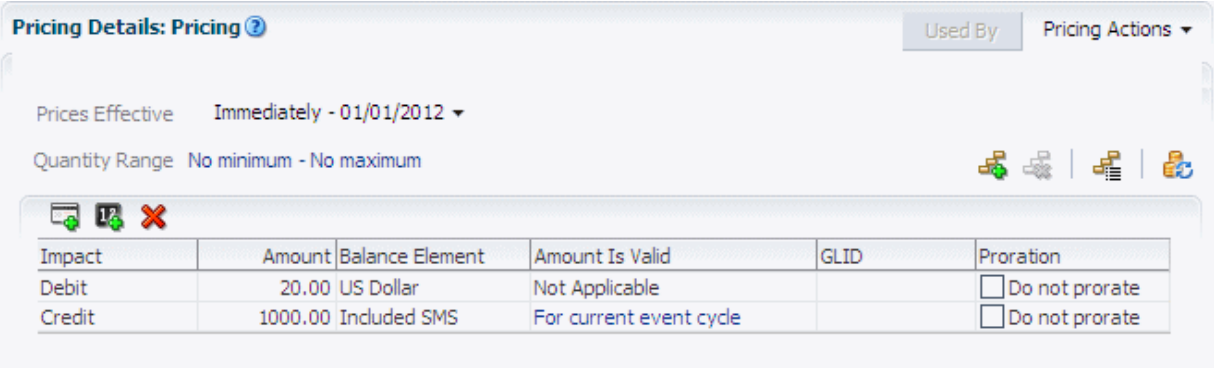
For more information, see the PDC Help.

About the Pricing Table

The pricing table is used to configure the pricing of an event associated with a charge. Each row in the table represents a balance impact. Simple charges that do not use extended pricing features contain only one pricing table. Charges that use extended pricing features contain multiple pricing tables, though only the table or tables associated with the pricing instance selected in the Charges tree are displayed (see "About the Charges Tree").

Figure 2–5 shows an example of the pricing table for a charge.

Figure 2–5 Pricing Table for a Charge



Each row in the pricing table is a debit, credit, or counter balance impact. The information required for each impact depends on the charge category and the pricing profile associated with the charge.

About Effective Periods for Pricing

By default, pricing takes effect immediately and never ends.

Some pricing profiles, however, support multiple effective periods for pricing. This enables you to implement future price changes for a charge without creating multiple versions of the charge. Instead, a copy of the pricing is created within the same charge for each effective period and is modified as necessary.

When pricing has multiple effective periods, a list of the periods appears above the pricing table (see Figure 2–5). Selecting a period in the list displays the pricing table for that period. By default, the table for the current period is displayed.

Note: To implement price changes for a charge whose profile does *not* support multiple pricing effective periods, you must create multiple versions of the charge itself and revise the pricing in each version. See "About Charge Date Ranges" for more information.

About Quantity Ranges in Pricing

To configure volume-based pricing, such as pricing based on amount of usage or frequency of occurrence, you can add quantity ranges and configure different prices for each range.

For example:

- Pricing for a telephony service might contain the following quantity ranges based on the total duration of calls during a month: 0 to 500 minutes, 500 to 1000 minutes, and 1000 minutes and up.
- Pricing for a media streaming service might contain the following quantity ranges based on the total volume of data downloaded during a month: 0 to 6 GB, 6 to 36 GB, and 36 GB and up.

Important: If you enable ECE to generate midsession rated events, note that each midsession event marks the end of a subsession of the network session. For the subsequent subsession, the network session's duration and volume counters are reset to 0. Therefore, if you use midsession rated events, do not base your pricing on cumulative duration or volume across an entire network session.

For more information, see the discussion about midsession rated events in *ECE Concepts*.

About Minimum Charges

You can set a minimum charge for each balance element impacted by a pricing instance. Minimum charges are configured per price tier (see "About Price Tiers") and effective period.

About the Charges Tree

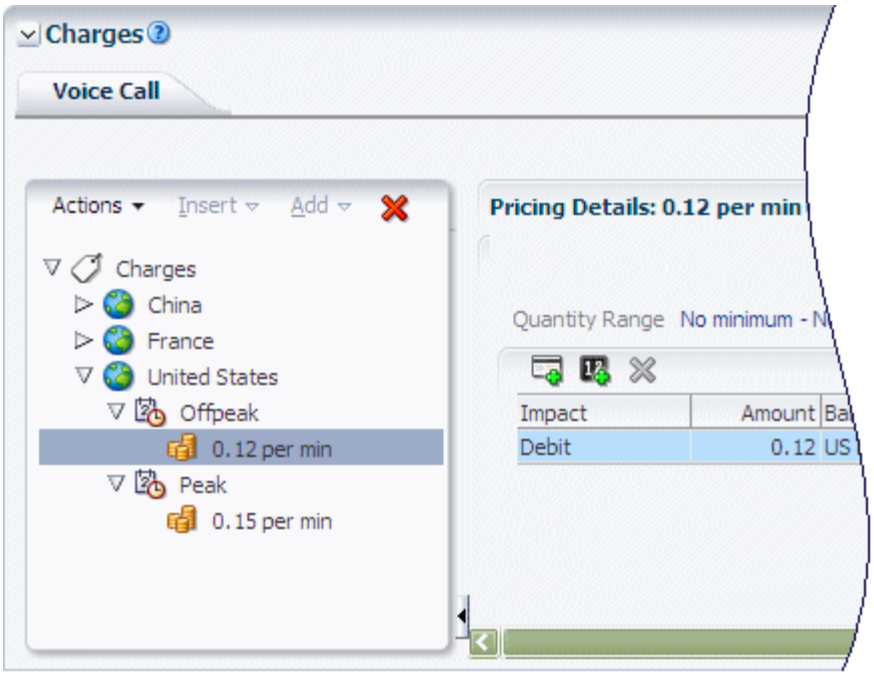
The Charges tree shows the hierarchical structure of a charge.

You use the Charges tree to add extended pricing features, such as impact categories and time periods, to a charge (see "About Using Extended Pricing Features"). When you select a pricing node in the tree, its corresponding pricing table is displayed (see "About the Pricing Table"). When you add or select another node in a branch, read-only information about that node is displayed.

Each date range in a charge has a Charges tree. For charges that do not use extended pricing features, the tree is usually hidden because only one pricing instance needs to be created, and that is done in a single pricing table. You can display the tree by clicking the **Restore Pane** arrow in the lower-left corner of the Pricing Details section.

Figure 2–6 shows an example of a Charges tree.

Figure 2–6 Charges Tree



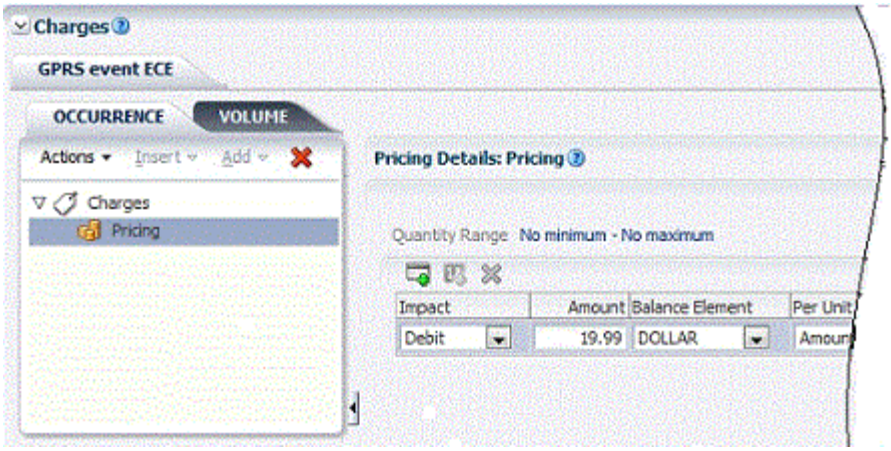
To create charges that use extended pricing features, you add elements such as impact categories, time periods, and pricing to the tree. For example, in Figure 2–6, the Charges tree contains three impact categories (China, France, and United States). The United States impact category contains two time periods (Offpeak and Peak). Under each time period is a pricing leaf node. When you select a pricing leaf node, an editable version of its associated pricing table, price selector, or price override appears.

The Charges tree is typically used to configure usage charges, but it supports all charge categories except folds and rollovers. The pricing profile associated with a charge determines which elements can be added to the tree and the order in which they can be added.

For some rating systems, multiple RUMs within a charge are supported with separate charge trees for each RUM.

Figure 2–7 shows a charge with multiple charge trees, each representing the charge for a different RUM, in this case, Occurrence and Volume.

Figure 2–7 Multiple Charge Trees for Multiple RUMs



When you select the charges node, the RUM rounding details are displayed.

To add or remove a RUM, you must add or remove the RUM from the charge details.

About Impact Categories

You use impact categories to enable the same charge to apply different pricing based on the values of various event attributes (see "Charges Based on Attributes"). For example, to configure different pricing for calls made to different countries, you add impact categories for each destination country to the charge. When a call occurs, the pricing associated with the impact category for that call's destination is applied to the call. For example, to charge usage for a mobile phone service, you might create the following impact categories:

- Impact Category: **Bolivia**
  Balance Impact: **.10 per minute**
- Impact Category: **France**
  Balance Impact: **.05 per minute**

PDC supports impact categories for the following components:

- **Zones:** Zone impact categories are used as follows:
  - *Base* zone impact categories are used as the results of rules in zone models and in Usage Scenario (USC) selectors.
  - *Derived* zone impact categories are used as the results of rules in USC selectors and Access Point Name (APN) selectors. They are considered derived because the rules in USC and APN selectors use additional attributes to remap base zone impact categories to different zone impact categories. For example, a USC selector rule might specify that if the base zone impact category is China and the usage type is **Friends & Family**, the selector returns the Friends & Family derived zone impact category.

    To create derived zone impact categories, you must select the **Derived Only** option in the Impact Categories for Zones table.

  See "About Zone Models" and "About Selectors" for more information.

The available impact categories depend on the pricing profile associated with the charge or selector (see "Working with Profiles"). For example, if a profile supports zoning, the available impact categories are determined by the selected zone and, optionally, the selected USC and APN selectors.

You can group multiple impact categories into a single group node to apply the same pricing to all of them.

See "About Configuring Setup Components" for information about configuring impact categories in PDC.

About Time Periods

To charge different prices for the same service depending on the day and time the service is used, add time periods to a Charges tree. Time periods are blocks of day and time combinations. For example, you might add the following time periods:

- **Peak:** Monday–Friday 8:00 a.m. to 5:00 p.m.
- **Off-peak:** Monday–Friday 5:00 p.m. to 8:00 a.m. and all day Saturday and Sunday

Time periods are defined in time models. To add time periods to a tree, you select a time model and then select one or more of its time periods. The available time models depend on the pricing profile associated with the charge.

Time periods can be grouped and ungrouped.

A different time model can be used for each impact category in the charge.

See "About Time Models" for more information.

About Charge Pricing

You can add the following types of pricing to a charge:

- **Pricing:** Enables you to create one or more balance impacts. By default, each new pricing instance is named *Pricing*. You can use the pricing name in the discount filter to filter a charge when the charge is associated with the Online Usage pricing profile
- **Price Selector:** Adds a prioritized list of preconfigured pricing to the charge. When the event to which the charge applies occurs at run time, your billing system selects a pricing instance from the list based on the value of one or more event, service, or subscriber attributes. This mechanism enables you to charge different prices for the same event without creating multiple pricing instances. See "About Selectors" for more information.
- **Price Override:** Enables you to replace the balance impact of a record imported from another system or to adjust the balance impact by a specified percentage or a fixed amount.

About Fold Charge Pricing

A fold charge is used to zero-out a balance, such as unused included minutes or to convert one balance to another.

A fold is configured using a charge selector. The event for the charge selector must be the **Fold** event. After you have configured the charge selector, select it in the charge offer.

The pricing in a fold charge should have a debit for the balance that you are converting and one or more credits for the balances that you are converting to.

Figure 2–8 shows the pricing for a fold that zeros-out Included Minutes.

Figure 2–8 Fold Pricing to Zero-Out a Balance

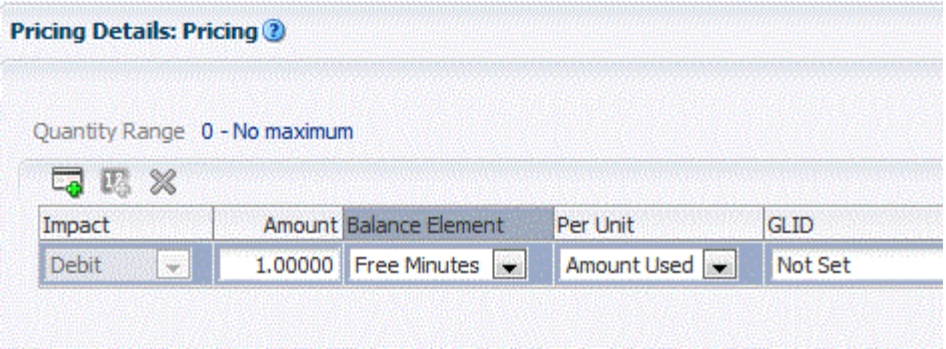
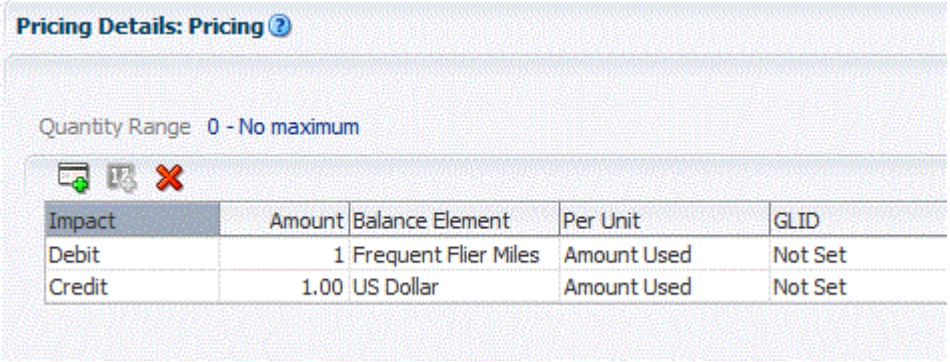


Figure 2–9 shows the pricing for a fold that converts Frequent Flyer Miles balance to US Dollar balance, $1 for each frequent flyer mile.

Figure 2–9 Fold Pricing to Convert One Balance to Another

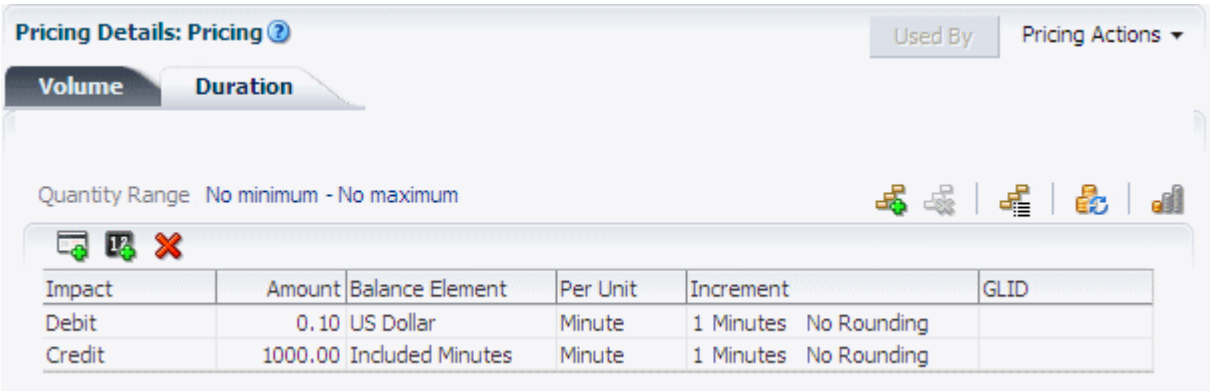


About Price Tiers

By default, each charge has at least one price tier. A price tier contains the pricing for a single RUM (see "About Measuring Events"). To configure pricing for multiple RUMs within a charge, you first select the desired RUMs when you create the charge. Then within the charge, you configure different pricing for each RUM. When a charge contains multiple price tiers, each tier appears in its own tab.

Figure 2–10 shows a charge that contains two price tiers, Volume and Duration.

Figure 2–10 Volume and Duration Price Tiers



When you create a charge, price tiers for all the RUMs listed in the charge when it is created are automatically included in any pricing added to the charge.

To add or remove a RUM from an existing charge, you must add or remove the RUM from the charge details.

> **Note:** Multiple price tiers are not supported in all rating systems. In these cases, multiple RUMs within a charge are supported with separate charge trees for each RUM (see "About the Charges Tree").

About Modifying Pricing without Inadvertently Affecting Other Components

When you use an existing charge, any changes you make to the charge, including price changes, will affect all the charge offers and charge selectors where the charge is used. To avoid the unintended consequences of changing prices in a reusable charge, you use the Change Price feature to specify which of those components you want your changes to affect.

- *Change the original charge.* This affects every component that uses the charge that contains the modified pricing.
- *Make a copy of the charge and change the copy.* This affects only the charge offer or charge selector that contains the charge with the modified pricing. All other components that use the charge continue to use the previous version of the charge.

About Conditional Balance Impacts

A conditional balance impact is a balance impact that credits or debits a customer's balance only when the customer uses a charge offer for the first time within a specified period; for example, the first time in a day or the first time in two days. For example, you can use conditional balance impacts to grant daily included minutes to a customer, instead of using recurring events.

You can align the start time of the conditional balance impact period with the start time of the associated charge offer, the start of a calendar day, or the event occurrence. You can also specify whether the balance is available from the start time of the period or from the time the event occurs.

Conditional balance impacts can only be used with charge offers that are associated with a service, not with the charge offers associated with an account.

The following example shows how to configure conditional balance impacts for granting daily included minutes.

In this example:

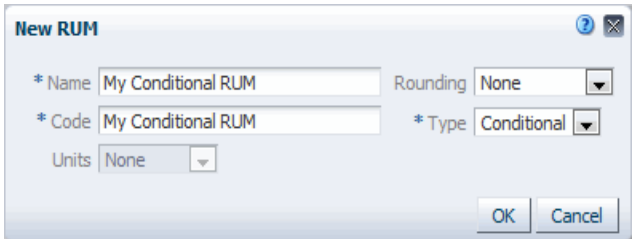
- The customer is granted 40 included minutes at $5 per day.
- The included minutes are valid for a day.
- The customer is not charged on the days that the offer is not used.

To configure conditional balance impacts for granting daily included minutes:

1. Create a RUM of type **Conditional** for the desired service and usage event. See the discussion about creating a new RUM configuration in the PDC Help.

   Figure 2–11 shows a RUM of type **Conditional**.

Figure 2–11 New Rum Configuration



2. Create a charge offer. See the discussion about creating a charge offer in the PDC Help.
3. Add a new charge to the charge offer. When you add the charge, ensure that you select the following values:

   Charge Category: **Usage**

   Pricing Profile: **Convergent Usage**

   Measured By: The RUM that you created in step 1. Optionally, you can select additional RUMs.

   See the discussion about adding a new charge in the PDC Help.
4. Configure conditional balance impacts to grant 40 minutes at $5 per day on the days the charge offer is used.

   Figure 2–12 shows the conditions configured for granting daily included minutes.

Figure 2–12 Conditions for Granting Daily Included Minutes

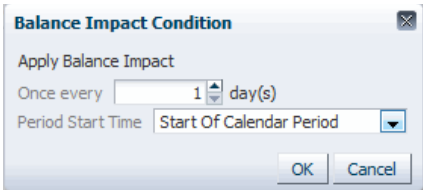


Figure 2–13 shows the validity period for the conditional balance impact.

Figure 2–13 Validity Period for a Conditional Balance Impact

Figure 2–14 shows the conditional balance impacts used in this example for granting daily included minutes.

Figure 2–14 Conditional Balance Impacts for Granting Daily Included Minutes

Pricing Details: Pricing

Quantity Range 0 - No maximum

Impact	Condition	Amount	Balance Element	Per Unit	Amount Is Valid	GLID	Discounting
Debit	Once every 1 day(s)	5.00	US Dollar	Fixed Charge	Not Applicable	Not Set	Do not discount
Credit	Once every 1 day(s)	40	Included Minutes	Fixed Charge	From Period Start - 1 Day(s) After Period Start	Not Set	Do not discount

See the discussion about configuring conditional balance impacts in the PDC Help.

About Discount Offers

A discount offer contains one or more discounts (see "About Discounts"). Discount offers are separate, purchasable items similar to a charge offers. You can include discount offers with charge offers in bundles.

This section provides an overview of the information required to configure discount offers. For more information, see the PDC Help.

Note: Configuration information that is common to both charge offers and discount offers is documented in "About Charge Offers".

Discount Offer Types

When creating a discount offer, you must specify its type. PDC supports the following discount offer types:

- **Subscription:** Contain discounts that apply only to the subscriber that owns the discount offer.
- **System:** Contain discounts that can apply to all subscribers of the specified service. System discount offers are not owned by subscribers.

Applying Multiple Discount Offers to a Charge

The same charge or portion of a charge can be eligible for multiple discount offers. When this occurs, your billing system uses the following discount offer attributes to determine how to apply the discount offers to the charge:

- **Priority:** To control the order in which multiple discount offers are applied to a charge, you can assign a priority number to each discount offer. Offers with higher priority numbers are applied before those with lower priority numbers.
- **Mode:** To specify whether a discount offer is applied to the original charge amount or only to the amount that remains after previous discount offers have been applied, you can assign one of the following modes to the discount offer:
  - **Original Charge:** The discount offer is applied to the original charge amount, regardless of whether that amount was reduced by previous discount offers.
  - **Remaining Charge:** The discount offer is applied to whatever charge amount remains after previous discount offers are applied.
  - **Remaining Charge and Quantity:** The discount offer is applied only to the part of the charge and quantity that has not been used as the basis for a previous discount offer. Discount offers with this mode can be used only if part of the charge has not yet been evaluated for a discount.

This mode is for discount offers that consume noncurrency balance elements or that discount currency charges, not for discount offers that grant balance elements.

For example:

- A bundle includes two discount offers that apply to a mobile phone service. The offer with the higher priority gives a 10% discount. The other offer gives a 20% discount.

  For a $10 call, the 10% discount is processed first because it has a higher priority. This results in a $1 discount, which reduces the charge amount to $9.

  - If the second discount offer is set to **Original Charge**, the 20% discount is applied to the entire original charge amount ($10). This results in a second discount of $2, for a total charge of $7.
  - If the second discount is set to **Remaining Charge**, the 20% discount is applied only to the remaining $9 charge. This results in a second discount of $1.80, for a total charge of $7.20.
  - If the second discount is set to **Remaining Charge and Quantity**, it cannot be applied because the entire original charge was the basis for the first discount. The total charge for the call is $9.

- Another bundle includes the following discount offers for a mobile phone service: a discount offer for 50 included minutes and a discount offer for 20% off all minutes.

  The customer makes a 100-minute call at $0.10 a minute for a total charge of $10. The first discount offer has a higher priority, so it is processed first. The offer determines that there are 50 minutes that can be provided for free and provides them.

  - If the second discount offer is set to **Original Charge**, the 20% discount applies to the entire 100-minute call, even though there was no charge for the first half. The discount is 20% of $10. This results in another discount of $2. The total charge for the call is $3.
  - If the second discount offer is set to **Remaining Charge**, it applies only to the remaining $5 charge, resulting in a total charge of $4.
  - If the second discount offer is set to **Remaining Charge and Quantity**, the customer gets 20% off the remaining portion of the call (50 minutes) because the first discount did not evaluate that part of the charge. The total charge for the call is $4.

About Mutually Exclusive Discount Offers

When creating a discount offer, you can establish a mutually exclusive relationship between that discount offer and other specified discount offers. When the current discount offer and any of its mutually exclusive discount offers are applicable to the same event, only the current discount offer can be used to discount the event.

Note: To exclude a billing-time discount, you must also exclude the counter discount associated with the billing-time discount (see "About Billing-Time Discounts").

You use discount offers to set up mutually exclusive run-time relationships between discount offers. Such relationships govern the application of discount offers.

To set up mutually exclusive purchase relationships between discount offers, you use packages (see "Restricting Discounts in Packages"). Such relationships govern the ownership of discount offers.

About Discounts

Each discount offer contains one or more discounts. A discount can do the following:

- Reduce the charges associated with billable events.

  For example, you could offer a bundle called VOIP Plus that includes basic VOIP service with 300 included peak and 500 included off-peak minutes for a $100 setup fee, a $40 monthly fee, and usage charges. The bundle already contains one discount (the bonus minutes) but you can add another promotional discount to the bundle that reduces the charges even more:

  - 50% off the monthly fee for the first 6 months
  - Waiver of the setup fee (a 100% discount)
  - 25% off usage charges

- Grant or consume nonmonetary balance elements, such as included minutes or loyalty points.

  When you grant balance elements, discounts and charge offers can work together:

  - In a charge offer, you use recurring charges to grant the free balance elements. For example, you can configure a charge offer to grant 100 minutes each week or 500 minutes on a one-time basis.
  - You then use discount offers to consume the free balances and to reduce the charge based on the amount consumed.

- Track usage or spending by using counters.

  For example, when you set up billing-time discounts, you can use a discount to update a counter balance, such as a balance that tracks total downloaded megabytes. You use a second discount to apply a percentage off based on the total counter balance. See "About Billing-Time Discounts" for more information.

The following sections provide an overview of the information required to configure discounts. For more information, see the PDC Help.

> **Note:** Configuration information that is common to both charges and discounts is documented in "Adding Charges to Charge Offers".

About Discount Categories

You must specify a category for each discount. PDC supports one-time, recurring, and usage discounts.

For recurring discounts, note the following:

- Although a charge offer cannot contain both cycle forward and cycle arrears charges, a discount offer can contain both types of cycle discounts as long as their frequency (monthly, yearly, and so on) is the same.

- The same recurring discount can be applied to multiple charge offers that support different services as long as each charge offer contains a charge for the type of recurring event supported by the discount.

See "About Charge Categories" for more information about these categories.

About Discount Profiles

A discount profile tells PDC which discount features to support in its user interface. PDC includes a profile for standard discounts and a profile for billing-time discounts.

See "Working with Profiles" for more information about discount profiles.

About Billing-Time Discounts

Billing-time discounts are determined at the end of the billing cycle. This enables you to grant discounts based on the aggregation of a balance element during a billing cycle. For example, you can create a billing-time discount to do the following:

- Reduce a usage charge by $10 if the total usage charge for the billing cycle is more than $100.
- Grant 10 included minutes if the total minutes used during the billing cycle is more than 500.
- Grant a free month of service if a customer has subscribed to the service for 12 months.

Billing-time discounts are granted based on balances accumulated by a standard discount that uses a counter. The standard discount increments the counter when usage occurs to track the total amount of a balance element used during a specified period.

To create a billing-time discount, you set up the following items:

- A noncurrency balance for the counter to track. Include this balance in the package (see "Tracking Balances by Service").

  For example, you might create a Dollars Spent balance. As charges occur during a billing period, they impact that balance instead of the US Dollars balance. At the end of the billing cycle, the billing-time discount uses the amount in the Dollars Spent balance minus the applicable discount to calculate the impact of the US Dollars balance.

- Two discounts:
  - *A standard discount* that increments a counter to track the accumulation of a balance, such as total fees charged or total units consumed (minutes, text messages, bytes, and so on). For example, if the billing-time discount is based on total monthly charges, create a discount that updates the counter when charges are applied. The incremented counter balance is always a noncurrency balance regardless of whether the aggregated amount is a charge or a nonmonetary unit.
  - *A billing-time discount* that uses the counter balance as a basis for granting the discount. For example, 10% off all usage for the month or 1000 bonus points for every year of subscription. The calculated discount is applied to the appropriate *account* balance, such as the currency balance or bonus points balance.

To create billing-time discounts, use the Billing-Time Discounting profile (see "About Default Pricing Profiles").

About Applying Discounts Retroactively

Some billing systems recalculate charges to apply rate changes or corrections retroactively. For such systems, you can specify whether a discount that was active when a charge was originally calculated but is now inactive or canceled can be included in retroactive calculations as follows:

- Never stop applying the discount
- Do not apply the discount if it is canceled
- Do not apply the discount if it is inactive
- Do not apply the discount if it is inactive or canceled

About Snowball Discounts

A snowball discount is a type of shared billing-time discount that distributes a percentage discount to all accounts in a discount sharing group.

To implement a snowball discount, you use two discounts:

- A standard discount that increments a counter when accounts incur usage.
- A billing-time discount that calculates and distributes the discount based on the counter balance. In PDC, you designate this a snowball discount.

The percent granted to each account can be distributed evenly or based on how much usage each account accrues.

See "About Billing-Time Discounts" and "About Discount Sharing" for more information.

About Discount Date Ranges

A discount date range is the period during which a discount is effective. By default, this period starts immediately and never ends.

Unlike charge date ranges, discount date ranges must be absolute; they cannot be relative.

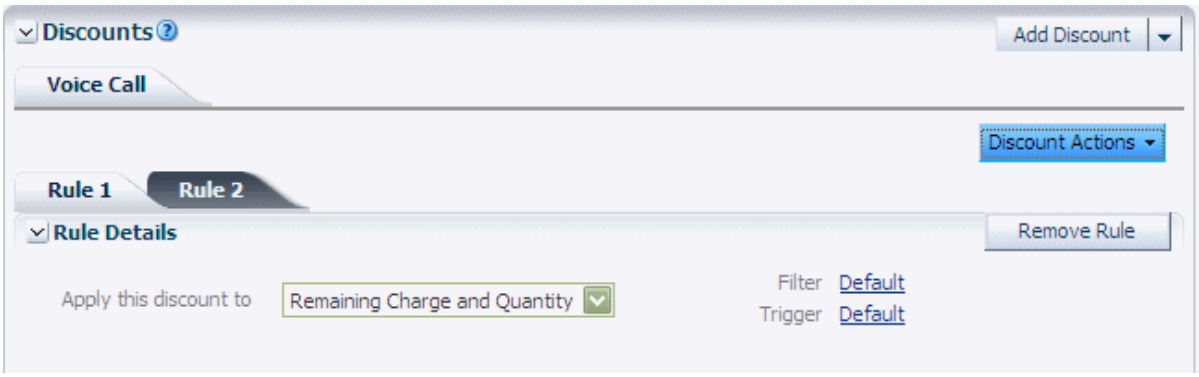
See "About Charge Date Ranges" for more information about absolute date ranges and how to customize them.

About Discount Rules

A discount rule specifies how to calculate and apply a discount. A discount can have one or more rules. For example, when you use multiple RUMs to measure an event, you configure the discount rule for each RUM to calculate and apply the discount using each RUM configured for that event.

When a discount contains multiple rules, each rule appears in its own tab as shown in Figure 2–15.

Figure 2–15 Multiple Rule Tabs in a Discount Offer



When you create multiple discount rules, the first rule that you create appears in the **Rule 1** tab, the second rule appears in the **Rule 2** tab, and so on. You cannot change the default name of a rule tab. Therefore, the sequence in which you create the rules is important because they are evaluated in numerical order.

Each discount rule contains the following elements:

- Rule conditions:
  - Discount filter
  - Discount trigger
  - Discount mode
- Pricing

This section provides an overview of the information required to configure discount rules. For more information, see the PDC Help.

About Discount Filters

A discount filter specifies criteria that a charge or a portion of a charge must satisfy to be eligible for the discount. For example, a discount might apply only to calls made on weekends or during December. The default filter, **Currency Charges**, applies to all charges that impact a currency balance.

If a discount has multiple filters, it applies to all charges that satisfy the criteria of at least one of those filters. When you use multiple RUMs to measure an event, you configure the discount filter for each RUM and specify the RUM as a criteria for applying the discount.

Conditions can be defined by specific values (such as a time period from a specified time model) or by expressions (see "About Discount Expressions"). The default value for most criteria is a dot followed by an asterisk (.*), which means that any value is valid.

The ability to create filters depends on the profile associated with a discount (see "About Discount Profiles"). By default, you can create filters for standard discounts but not for billing-time discounts. A filter can be reused by other discounts.

About Discount Triggers

A discount might require usage or balances to reach certain levels before the discount is applied. In that case, you create a discount trigger to specify one or more conditions, such as the following, that must be met before the discount can be applied:

- A specified charge amount must be reached, such as $50 of usage.
- Usage must be less than a particular value, such as the number of international call minutes must be less than 120.
- A specified quantity must be reached, such as a discount for the first 100 minutes and another discount for the second 100 minutes.
- A specified number of events must occur, such as a discount for the first 100 downloads.
- A combination of conditions must be met, such as the bytes used in a GPRS session must be greater than 1 MB and the session must last longer than 60 minutes.
- A balance must have an available amount, such as included minutes, that can be consumed.

A trigger condition consists of the following items:

- **Expression:** Results in a value to compare with the preset **Value** decimal. See "About Discount Expressions" for more information.
- **Operator:** Specifies how the result of the expression is compare with a value. Valid operators are greater than, greater than or equal to, less than, less than or equal to, equal to, and not equal to.
- **Value:** Specifies the decimal to compare with the result of the expression.

For example, you could configure the following trigger condition:

- **Expression:** Charge (the total charge of the event)
- **Operator:** Is greater than
- **Value:** 0.0

If a trigger includes multiple conditions, all conditions must be met for the discount to be applied. For example, the trigger for a GPRS discount can include conditions specifying that the total charge must be greater than $5 and the total quantity of received data must be greater than 10,000 bytes.

A discount can have only one trigger. A trigger can be reused by other discounts. Not every discount requires a condition to trigger it. For such discounts, you use the default **Always Trigger** trigger.

Applying Multiple Discount Rules to a Charge

The same modes that determine how multiple discount offers apply to a charge are used to determine how multiple discount rules apply to a charge:

- **Original Charge**
- **Remaining Charge**
- **Remaining Charge and Quantity**

See "Applying Multiple Discount Offers to a Charge" for more information about these modes.

By default, a discount rule inherits the discount mode from the discount offer that contains the rule, but you can change the mode of the discount rule.

The following example shows how discount offer and discount rule modes interact:

Discount Offer 1: any mode

Discount Rule A: any mode; 10% off if charge is $0 through $50

Discount Offer 2: **Original Charge** mode

Discount Rule B: 20% off; **Remaining Charge** mode

Discount Rule C: 10% off; **Remaining Charge** mode

The account incurs a charge of $100.

The following rules apply:

- Discount Rule A applies to charges up to $50, so only $50 is considered for discounting. The discount is $50 * 10% = $5.
- Discount Offer 2 is for the original charge, so it ignores previous discounting and applies to the original charge of $100.
- Discount Rule B is for the remaining charge, because it is the first discount in Discount Offer 2, it applies to the original charge. The discount is $100 * 20% = $20.
- Discount Rule C is for the remaining charge, because it applies to the charge that remains in Discount Offer 2 after Discount Rule B is applied. The discount is $80 * 10% = $8.

The final charge is (($100 - $5) - $20) - $8 = $67.

About Discount Pricing

For each discount rule, you configure pricing to specify which balances are impacted by the rule and how they are impacted.

Discount pricing can be used in multiple discount rules.

This section provides an overview of the information required to configure discount pricing. For more information, see the PDC Help.

About Discount Quantity Ranges

To vary a discount based on the amount of usage or frequency of occurrence, you can create discount quantity ranges and assign different pricing to each range. For example, a discount for a mobile phone service might contain the following quantity ranges:

- 0 through 500: No discount
- 500 through 1000: 10% off
- 1000 through No Maximum: 15% off

Each range has its own pricing (see "About Configuring Discount Balance Impacts"). The ranges and their tables appear sequentially in the discount rule.

Gaps are not allowed between ranges, so the end value of one range always matches the start value of the next range. If you leave a gap between quantity ranges, PDC automatically creates a range to fill in the gap.

With one exception, the start and end values of all quantity ranges must be a decimal. The exception is that a discount expression can be used to specify the end value of the last quantity range (see "About Discount Expressions"). Typically, the expression references a balance, such as the number of included minutes.

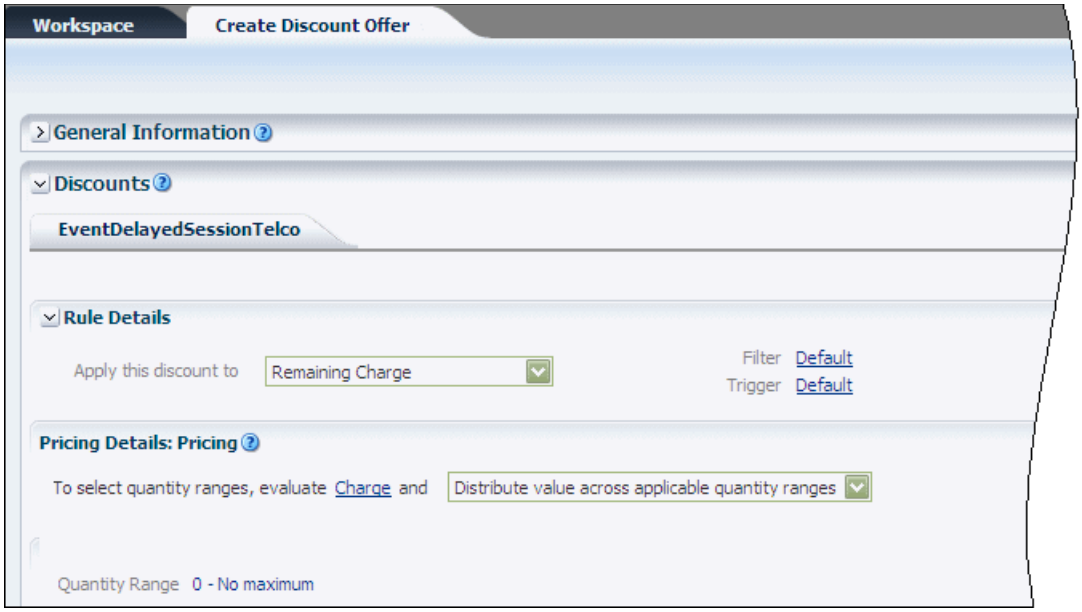
About Selecting Discount Quantity Ranges

To use one or more discount quantity ranges to calculate a discount, PDC selects the ranges as follows:

1. Evaluates the quantity range expression (see "About Discount Expressions").

   The quantity range expression is specified at the top of the Pricing Details section in a discount rule. For example, in Figure 2–16, the quantity range expression is **Charge**. For more information, see the PDC Help.

Figure 2–16 Location of the Quantity Range Expression in a Discount Rule



2. Uses the absolute value of the expression result and the specified selection type to select one or more quantity ranges. (As shown in Figure 2–16, the selection type is specified at the top of the Pricing Details section in a discount rule.)

   PDC provides the following selection types:

   - **Pick the quantity range containing the value:** Only the pricing in the range containing the result value is used to determine the discount. For example, if the value is 750, the ranges are applied as follows:
     - The **0–500** range does not qualify.
     - The **500–1000** range qualifies. The discount pricing configured for this range is applied to the entire result.
     - The **1000–No Maximum** range does not qualify.
   - **Distribute value across applicable quantity ranges:** The result is distributed across the ranges, and the pricing in each range is applied to the amount of the result that intersects that range. For example, if the value is 700, the ranges are applied as follows:
     - The **0–500** range qualifies. The discount pricing configured for this range is applied to the portion of the result that falls within this range.
     - The **500–1000** range qualifies. The discount pricing configured for this range is applied to the portion of the result that falls within this range.

– The **1000–No Maximum** range does not qualify.

When multiple discount quantity ranges qualify, each range is applied to the proportion of the quantity range basis that intersects with the range.

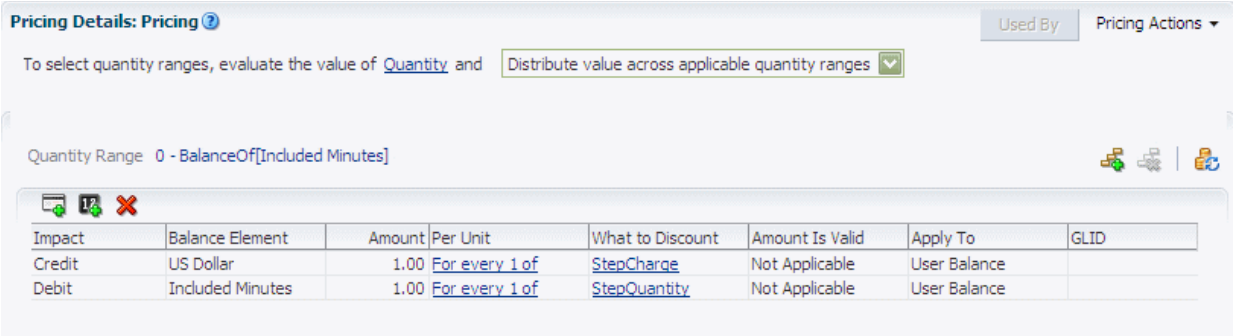
For this selection type, the discount base expression is typically either StepCharge or StepQuantity (see "What to Discount" and "About Discount Expressions").

About Configuring Discount Balance Impacts

The pricing table is used to specify which balances are affected by the discount and how the balances are impacted. For example, you can specify that a customer's balance of included minutes be debited by the length of a call and that charges for usage not covered by included minutes be discounted by 10%.

Figure 2–17 shows an example of the pricing table in a discount. Each row in the table is a balance impact.

Figure 2–17 Pricing Table for a Discount



Impact	Balance Element	Amount	Per Unit	What to Discount	Amount Is Valid	Apply To	GLID
Credit	US Dollar	1.00	For every 1 of	StepCharge	Not Applicable	User Balance	
Debit	Included Minutes	1.00	For every 1 of	StepQuantity	Not Applicable	User Balance	

When you set up a discount balance impact, you enter information such as the following:

- **Impact:** Specifies how the balance element is affected by the discount. For example:
  - Select **Debit** to reduce a noncurrency balance. For example, after charging a customer for a call, use a discount impact to debit any available included minutes along with another impact to credit the corresponding currency amount to the charge.
  - Select **Credit** to add noncurrency balance elements (such as loyalty points for calls of a certain duration) or to credit currency charges (such as to apply a 10% discount).
  - Select **Increase** or **Decrease** to increment or decrement a counter. See "About Billing-Time Discounts" for more information about counters.
- **Amount:** Specifies the discount percentage or amount. This value is applied to the value of the discount base expression (see "What to Discount") to compute the discount.

  When the discount is a percentage, the percentage is multiplied by the value of the base expression to determine the balance impact. For example, a discount percentage is 10 and the base expression refers to a balance of $120. The 10% discount is calculated on $120, which results in a balance impact of $12.

When the discount is an absolute amount, that amount can be one of the following:

- The actual amount to apply to the account balance. For example, to grant 100 included minutes as a birthday bonus, the amount is 100.
- An amount that should be applied to portions of the usage. For example, to grant 10 frequent flyer miles for every hour of usage, the amount is 10 and the unit is 60 (assuming the balance element is minutes).
- An amount that should equal the usage. For example, to consume one free minute for every minute used, the amount is 1 and the unit is 1.

- **Balance Element:** Specifies the balance element that is impacted, such as currency or included minutes. You can specify any noncurrency balance element defined in your system or the currency balance element specified for the charge.
- **Per Unit:** Specifies how to apply the balance element amount to the value of the discount base expression (as a percentage, as an absolute value, or incrementally; that is, for every *x* number of the result value).

  When you apply the amount incrementally, you can specify whether to round the final increment down. It cannot be rounded up.
- **What to Discount:** Specifies the discount base expression. This expression determines the basis of the discount calculation. The basis can be the total charge or quantity of an event, the total charge or quantity in a discount quantity range, a balance, a variable, or a query result. The discount is applied to the result of the base expression multiplied by the amount per unit, and the resulting balance impact is applied to the account balance. See "About Discount Expressions" for more information.

  When the discount mode is **Remaining Charge and Quantity**, the discount basis is preset, and you cannot change it.
- **Amount Is Valid:** For credit and counter (increase/decrease) balance impacts, specifies when the discount is available, such as from the time the event occurs, from the first time the balance element is consumed by a subscriber's usage, from a specified date to a specified date, for a period relative to the event occurrence, and so on.
- **Apply To:** Specifies whether the discount should be applied to the balances of the account that generates the event (the discount user) or the account that owns the discount. Typically, the discount user is also the discount owner. But in discount sharing and charge sharing, the account that generates the event can be different from the account that owns the discount. See "About Discount Sharing" for more information.

In the expanded version of the pricing table, you can also specify the general ledger ID and impact category for each balance impact.

About Discount Expressions

A discount expression is a mathematical formula that your billing system evaluates to produce a value. You can use expressions to define the following discount components:

- **Trigger condition** (see "About Discount Triggers")
- **End value of the last quantity range** (see "About Discount Quantity Ranges")
- **Quantity range basis** (see "About Selecting Discount Quantity Ranges")

- **Discount basis** (see "About Configuring Discount Balance Impacts")

To help you create discount expressions, PDC provides a discount expression builder. PDC supports the following special elements in discount expressions:

- **Charge:** Specifies the total charge of an event or part of an event.
- **Quantity:** Specifies the total quantity in an event, such as the number of minutes talked or megabytes downloaded.
- **StepCharge:** (For discount base expressions only) Specifies the portion of the charge that corresponds to a quantity range.
- **StepQuantity:** (For discount base expressions only) Specifies the portion of the event quantity that corresponds to a quantity range.

  For example, suppose the event quantity is 20 minutes and the quantity range is from 0 to the remaining balance of included minutes. If the remaining balance of included minutes is 10, the portion of the event quantity that falls within the quantity range is 10.
- **Balance[***balance_element***]:** Specifies the account balance for a particular balance element, such as frequent flyer miles or included minutes.

  The balance element in the discount base expression can be different from the balance element that is impacted. For example, to add bonus points based on minutes used, the base expression refers to a counter balance for the total minutes used, but the discount impacts the bonus points balance.

  When the base expression references a specific balance element, it always refers to a balance of the account that owns the discount.
- **Function["***function_name***"]:** Specifies an iScript function or a ECE pre-rating extension that retrieves the data required for the discount. The iScript function or a ECE pre-rating extension is used to base a discount on an amount that is not directly available to the billing system by performing a database query. For example, a function can retrieve the total charges for the 10 most frequently called numbers during the billing cycle.
- **Round:** Specifies whether to round up or truncate the element value. If the digit to the right of the specified precision is equal to or greater than 5, the last significant digit is rounded up to the next highest digit. Otherwise, all digits to the right of the specified precision are truncated. For example, if the expression is Round(Charge; 2) and the Charge is 1.131, the Charge is rounded to 1.13.
- **Round Up:** Specifies whether to round up the element value. If the digits to the right of the specified precision are non-zero, the last significant digit is always rounded up to the next highest digit. For example, if the expression is Round Up(Charge; 2) and the Charge is 1.151, the Charge is rounded to 1.16.
- **Round Down:** Specifies to truncate the element value to the specified precision. For example, if the expression is Round Down(Charge; 2) and the Charge is 1.159, the Charge is rounded to 1.15.
- **Round Bankers:** Specifies whether to round up to a nearest even number or truncate the element value. If rounding up the digit at the specified precision results in an even number, the digit at the specified precision is rounded up and the digits to its right are truncated. Otherwise, the digits to the right of the specified precision are truncated. For example, if the expression is Round Bankers(Charge; 2) and the Charge is 1.159, the charge is rounded to 1.16. If the Charge is 1.149, the Charge is rounded to 1.14.

A discount expression can include one or more individual elements such as decimal constants, standard arithmetic operators, and other discount expressions.

When the discount is an absolute amount that is applied regardless of the usage level, the base expression can be any positive number because you do not need to calculate the discount. For example, to apply a promotional discount of 50 included minutes for the first six months, the quantity range expression is 1, and the base expression is 1. The discount amount (50 included minutes) is then applied directly to the specified balance.

For more information, see the PDC Help.

About Using Temporary Balance Elements in Discounts

Temporary balance elements are used to pass values between multiple discounts for a single event. Typically, you use temporary balance elements to create temporary balances when you need the results of one discount to calculate another discount for the same event.

For example, consider a discount awards 5 SMSs if the amount of data transferred in a data usage event exceeds 1 megabyte and the session duration exceeds 30 minutes. In this situation, Discount 1 would determine the amount of data sent or received and store it in Temporary Balance A. Discount 2 would determine the duration of the session and store it in Temporary Balance B. Discount 3 would grant 5 SMSs if the data in Temporary Balance A exceeds 1 megabyte and the session length in Temporary Balance B exceeds 30 minutes.

Unlike other balances, temporary balances are maintained only while a single event is being discounted. If multiple standard discounts are applied to a single event, the temporary balance is maintained until all the discounts are processed.

For billing-time discounts, temporary balances are maintained only while a single discount is processed.

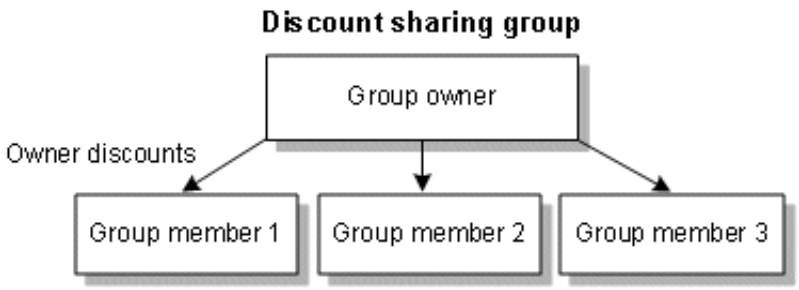
You can create temporary balance elements by using the PDC UI. See the discussion about creating balance elements in the PDC Help for more information.

About Sharing Discounts or Charges

Some billing systems enable accounts to share discounts or charges by joining groups that consist of an owner account and one or more member accounts. In BRM, a sharing group can be one of the following types:

- **Discount sharing group:** The owner shares its discounts with the members. Figure 2–18 shows a group owner providing discounts to three group members:

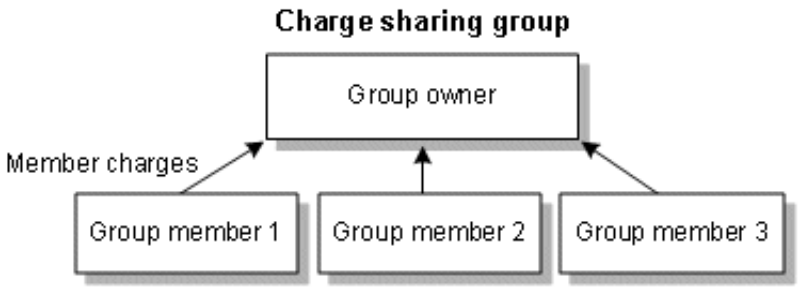
Figure 2–18 Discount Sharing Group



See "About Discount Sharing" for more information.

- **Charge sharing group:** The owner assumes charges that are incurred by the members. Figure 2–19 shows a group owner receiving charges from three group members:

Figure 2–19 Charge Sharing Group



See "About Chargeshare Offers" for more information.

About Discount Sharing

Discount sharing occurs when an account shares its discounts with other accounts. For example, a group of employees might share a pool of included minutes in their company's mobile phone account, or a parent might purchase discounts on her email service and want those discounts to apply to her children's email services as well.

To share discounts in BRM, you create discount sharing groups in Customer Center or a third-party client application. You set up the discounts in PDC.

PDC supports shared discounts by enabling a discount to apply either to the account that generates the event (the discount user), to the account that owns the discount, or to both. See the discussion on **Apply To** in "About Configuring Discount Balance Impacts" for more information.

With discount sharing, you sometimes need several discounts to specify how balance elements are granted or consumed for each account. For example, to offer 20% discount on usage to each member account when the total usage for all member accounts exceeds 1,000 minutes, you set up a discount with a counter to track the usage for each account and another discount to apply the percentage off based on the aggregated usage recorded in the counter balance.

About Chargeshare Offers

Charge sharing enables an account to sponsor the charges of other accounts. The owner account in a charge sharing group receives the balance impact of sponsored charges incurred by the member accounts. For example, charge sharing enables a company to pay for all of its employees' mobile phone services or a parent to pay for his child's SMS and email services.

To share charges in BRM, you create charge sharing groups in Customer Center or a third-party client application. You then set up chargeshare offers and chargeshares in PDC to determine how charges are shared among the members of the charge sharing groups.

A chargeshare uses rules that contain filters to determine whether an event qualifies for charge sharing, triggers to specify conditions that must be met before the

chargeshare applies, and pricing to determine the charge sharing amounts and balance impacts.

Setting up chargeshare offers and chargeshares is similar to setting up discount offers and discounts. See "About Discount Offers" for more information.

About Zone Models

You use zone models to charge for calls based on their origin and destination. PDC supports the following types of zone models:

- **Standard:** A zone model based on the origin and destination numbers of a call. It contains rules that associate a pair of origin and destination numbers with a zone impact category (see "About Impact Categories").

  To specify origin and destination numbers, you must include an international access code (the code used to dial *out of* the country in which the phone number is located). Optionally, you can also include a country code (the code used to dial *in to* the country in which the phone number is located), an area code, a region code, a city code, a phone number prefix, and so on, up to and including the entire phone number.

  Figure 2–20 shows zone rules associated with the same impact category.

Figure 2–20 Multiple Zone Rules Associated with the Same Impact Category

Zone Rules

View Detach

Service Type	* Origin	* Destination	Results		Validity
			* Zone Impact Category	Alternate Zone Model	
*	910	303	Local 2	None	Immediately - Never Ends
*	408	530	Long Distance 1	None	Immediately - Never Ends
*	408	303	Mobile	None	Immediately - Never Ends
*	408	415	Mobile	None	Immediately - Never Ends
*	510	942	New Zealand	None	Immediately - Never Ends
*	525	312	New Zealand	None	Immediately - Never Ends
*	512	303	New Zealand	None	Immediately - Never Ends
*	650	408	South Asia	None	Immediately - Never Ends
*	548	320	Taiwan	None	Immediately - Never Ends
*	717	530	Thailand	None	Immediately - Never Ends

- **Geographical:** A zone model based on the distance between the origin and the destination of a call. Geographical zone models include the following:
  - Zone rules that associate a distance with a zone impact category.
  - A list of area codes. Each area code is associated with one or more pairs of longitude and latitude coordinates.

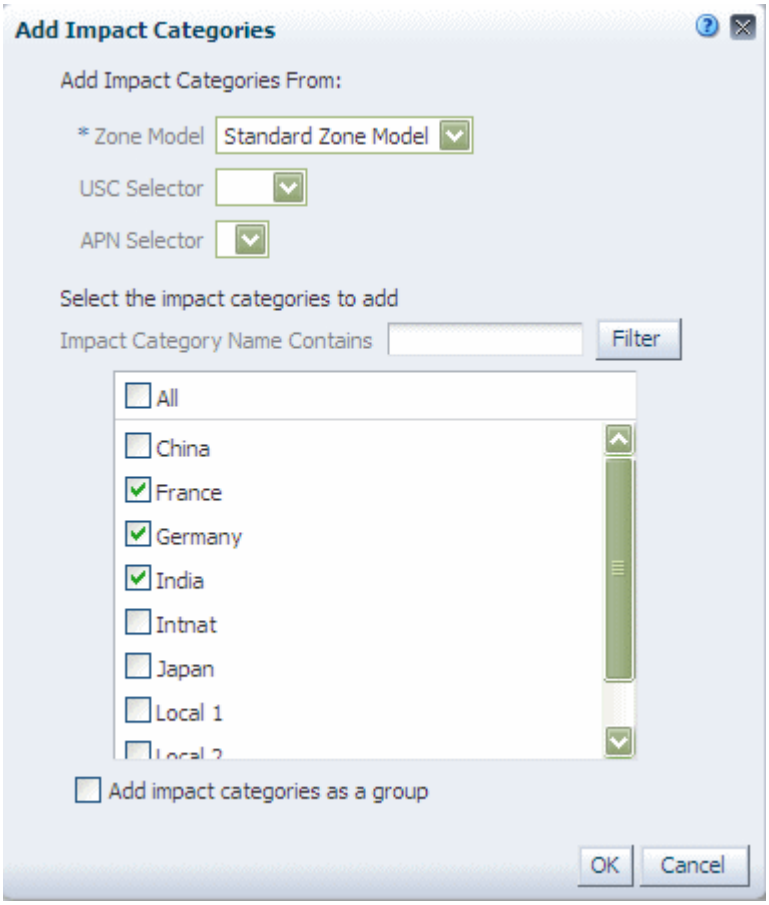
  When a customer makes a call, your billing system uses the data in the area code list to compute the distance between the origin number and the destination number. The billing system then assigns an impact category to the call event by using the zone rule whose distance most closely matches the computed distance.

  Geographical zone models are useful in the following situations:
  - Customers are located close to the border between two area codes. For example, if a customer in one area code calls a person two blocks away in another area code, you do not want to charge for a long-distance call.
  - The distance covered by an area code is very large and you want to use several rates within the same area code. You do this by associating different pairs of

latitude and longitude coordinates with the same area code in a geographical zone model.

When creating charges associated with profiles that support zoning, users must select a zone model to get a list of impact categories that can be added to the charge. For example, Figure 2–21 shows how the selected zone model determines which impact categories are available.

Figure 2–21 Using Zone Models to Select Impact Categories



Note: Optionally, users can also select a USC or APN selector to provide enhanced zone impact categories. See "About Selectors" and "About Impact Categories" for more information.

Alternate Zone Models

In both standard and geographical zone models, each rule results in the application of a zone impact category and, optionally, an alternate zone model (see the rules in Figure 2–20). Alternate zone models enable you to define more granular zone rules and reuse them in multiple zone models.

For example, in a zone model for calls that originate in the U.S., you might include a rule with the following parameters:

- **Origin:** 011
- **Destination:** 00

- **Impact Category:** 00 General
- **Alternate Zone Model:** US to 00 Country Codes

In this example, the alternate zone model might contain rules for various combinations of calls from the U.S. (011 international access code) to numbers with the 00 international access code and a particular country code, such as **00 44** for calls to the U.K., **00 91** for calls to India, and so on).

When a call is made from the U.S. to a 00 international access code in this example, your billing system first checks to see whether the country code of the destination number matches a rule in the alternate zone model. If it does, the billing system uses the impact category associated with the rule in the alternate zone model to calculate the call's charge. If it does not, the billing system uses the 00 General impact category to calculate the charge.

See "About Configuring Setup Components" for more information about configuring zone models.

About Value Maps

A value map is a hierarchical structure that associates names with values.

You use value maps in charge selectors to group event attribute values into manageable categories. For example, to apply the same charge to all calls made from the San Francisco Bay Area to Los Angeles, you might use a California Area Codes value map that includes the following values:

- **Northern California**
  - **San Francisco Bay Area**
    - 408
    - 415
    - 510
    - 650
    - 707
- **Southern California**
  - **Los Angeles**
    - 213
    - 323

Instead of creating charge selector rules for all possible combinations of those area codes, you associate the California Area Codes value map with the charge selector and then create only one rule in which **San Francisco Bay Area** is the origin call value and **Los Angeles** is the destination call value.

> **Note:** In PDC, value maps are used only with charge selectors (see "About Selectors").

See "About Configuring Setup Components" for information about configuring value maps.

About Selectors

A selector is a series of rules that associates the values of event attributes, service attributes, account attributes, or custom rules to a result. Different selectors have different types of results. Selectors can be used in charge offers to determine the price of an event.

When an event occurs, the selector rules are evaluated in order of priority. The first rule that applies to the event determines which result is returned by the selector.

For example, when the rating engine calculates the cost of a telephone call, a charge selector can determine which charge to use based on call origins and destinations.

You can create the following types of selectors:

- **Charge selector:** Determines the charge to use based on the values of specified event attributes, service attributes, account attributes, or custom rules.
- **Discount selector:** Determines the discount to use based on the values of specified event attributes, service attributes, account attributes, or custom rules. You can use a discount selector instead of a discount in a discount offer.
- **Generic selector**: Determines the result name based on the values of specified event attributes, service attributes, account attributes, or custom rules.
- **Price selector:** Determines the appropriate pricing to use based on the values of specified event, service, or account attributes.

  You can use a price selector instead of a pricing instance in a charge offer.
- **Usage Scenario (USC) selector:** Determines a new impact category based on the zone impact category and specific event attributes, such as differentiated network services, or custom rules, such as **Friends & Family**.
- **Access Point Name (APN) selector:** Determines a new impact category based on the zone impact category and access point name that applies to the event.

When creating a selector, you choose the attributes to be used in the rules. Each rule then uses the same fields, and you must specify a value for all the fields in every rule. For some rules, the value of a field might not be relevant, so you can use a wildcard to indicate that any value is acceptable.

The sequence of the rules is important, so you can reorder them as necessary. The rules tables include a search mechanism to make it easy to find a rule.

In all selectors except charge selectors, rules have an effective period. By default, the period starts immediately and never ends. You can modify and add effective periods.

For information about creating selectors, see the PDC Help.

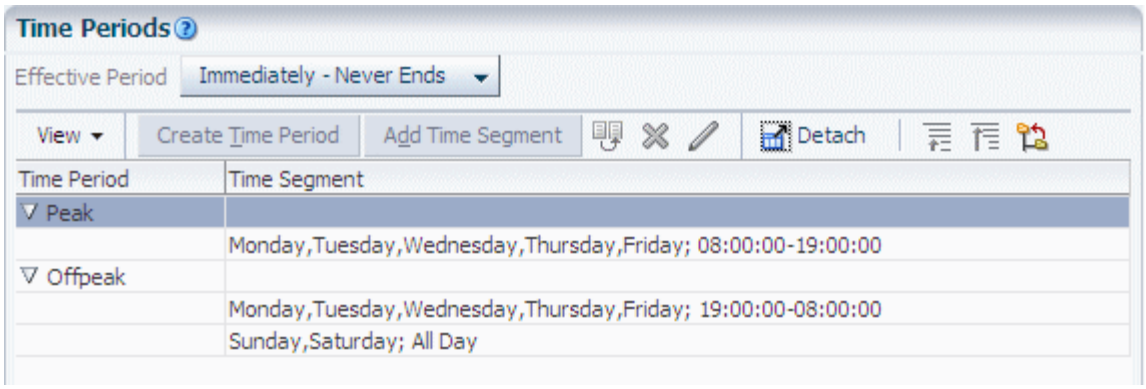
About Time Models

You use time models to charge different prices for the same service depending on the day and time the service is used.

A time model is a set of time periods. Each time period contains one or more time segments. A time segment represents a particular time, such as a day of the week or a range of several hours.

Depending on the pricing profile associated with the time model, you can define time segments by using months, days of the month, days of the week, time of day, and a calendar of special days. For example, Figure 2–22 shows an Offpeak and a Peak time period whose time segments are defined by days of the week and time of day.

Figure 2–22 Peak and Off-Peak Time Periods in a Time Model



To use a time model in a charge, you add its time periods to a Charges tree (see "About the Charges Tree"). If the Charges tree contains multiple impact categories, you can add time periods from a different time model to each impact category. You can then associate a different pricing instance with each time period in the tree.

About the Effective Period of a Time Period

By default, a time period is immediately and forever effective. To add effective periods to a time period, you specify only the start date of the new period. That date becomes the end date of the previous period. The end date of the final effective period is always **Forever**.

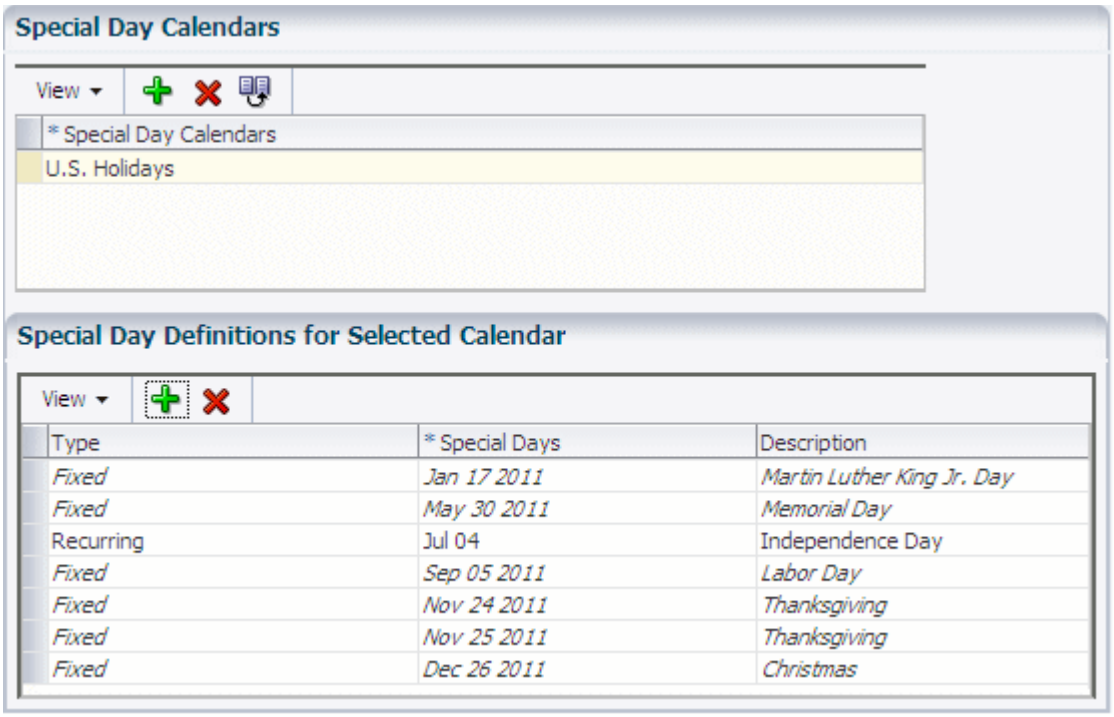
About Special Day Calendars

A special day calendar is a set of dates, such as holidays, for which you want to charge special prices for your services. Each date must be one of the following types:

- **Fixed:** A specific date valid only in *one year,* such as May 8, 2011, for Mother's Day in the U.S.
- **Recurring:** A date that is valid *every year,* such as July 4 for Independence Day in the U.S.

Figure 2–23 shows a special day calendar that has fixed and recurring dates.

Figure 2–23 Special Day Calendar



To configure pricing for special days in a charge, you associate a special day calendar with a time model. You must then configure at least one time period that applies to the special days. The time model should cover all 24 hours of the special days.

Note: The same time period cannot apply to both regular days of the week and special days. If you configure a time period that applies to both, the time model will receive a validation error.

The same calendar can be associated with multiple time models, but you can also create different calendars for different time models.

For more information, see the PDC Help.

About Bundles

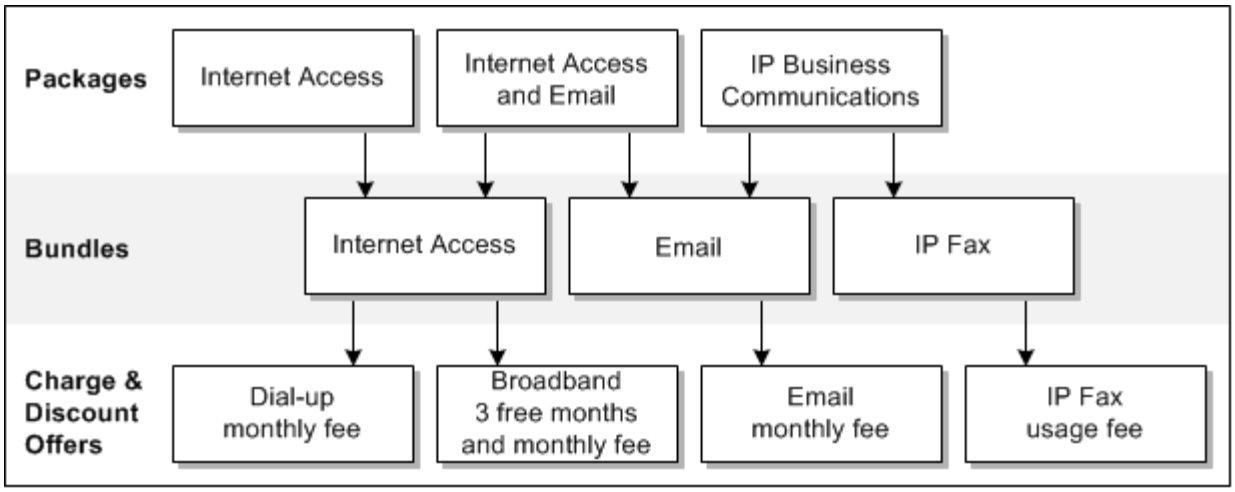
A bundle is a set of charge offers, discount offers, or both. Bundles are typically used to group offers that you want to sell together.

Each bundle is associated with a single service. Only offers that apply to that service can be included in the bundle. For example:

- A package that provides only Internet access includes only Internet access bundles (such as cable and premium cable).
- A package that provides Internet access and VOIP includes at least two bundles, one for Internet access and one for VOIP.
- A package that provides VOIP and cable TV includes at least two bundles, one for VOIP and one for cable TV.

Figure 2–24 shows bundles that are associated with a single service.

Figure 2–24 Bundles Are Always Associated with a Single Service



One bundle can contain any number of charge and discount offers, and two different bundles can contain the same offer. Grouping offers in different ways in different bundles adds flexibility to your pricing structure without requiring you to create additional charge or discount offers.

Before creating bundles, you should understand the concepts described in this section. For more information, see the PDC Help.

Making a Bundle Applicable to a Service

When creating a bundle, you can associate it with a specific service. That is the only service to which the offers in the bundle apply.

If you create a bundle that does not apply to a specific service, associate the bundle with **Account**. For example, you might do this to create a bundle for late charges or for coupons. The offers in the bundle can then be used to rate any event associated with **Account** in the service-event map (see "About the Service-Event Map"). Each account can own only one bundle that applies to **Account**.

Specifying When a Bundle Can Be Purchased

By default, the time period during which a bundle can be purchased starts immediately and never ends. You can change the default start date, end date, or both.

To be added to a bundle, an offer must have a purchase period that is the same as or greater than the bundle purchase period. If the purchase period of an offer in a bundle exceeds the bundle purchase period, the bundle purchase period overrides the offer purchase period. See "Specifying When a Charge Offer Can Be Purchased" for information about charge offer purchase periods.

Billing Customers on Bundle Purchase

Usually, you bill a customer at the end of the customer's billing cycle. *On-purchase billing*, however, enables you to bill a customer immediately for a purchase, even if the customer's billing cycle has not ended.

When you create a bundle, you can flag it for on-purchase billing. When a customer purchases the bundle, a bill is generated immediately for the purchase fees associated with the bundle.

Note: On-purchase billing works with purchase fees only, not with recurring, usage, or cancellation fees.

See "Billing Customers on Package Purchase" for more information.

Aligning Validity of Balance Impacts on First Use

When you create a bundle, you can flag it to synchronize the start date of all balance impacts whose validity period starts on first usage. This ensures that all such balance impacts in the bundle's charge offers are set to the same validity period when one of them is activated for the first time.

Permitting CSRs to Customize Bundles

If your customer service representatives (CSRs) can discount or change the effective period of charges in bundles at purchase time, you can specify whether to prohibit, allow, or require such modifications in a particular bundle.

For example, if customer input is required to set the date on which an offer's purchase fee is applied, you can specify that a bundle must be modified.

About Offer Quantities

You can configure bundles to provide more than one of the same charge offer or discount offer.

For example, if a bundle for a cable service includes a charge offer that provides one set-top box and you want to include three set-top boxes with the cable service, enter **3** for the charge offer in that bundle.

About Charge Offer and Discount Offer Status at Purchase

When you add a charge offer or discount offer to a bundle, you also specify whether the offer is *active* or *inactive* at the time of purchase. For example, an offer might be inactive at purchase so that the purchase or first month's fee is not applied until you get confirmation that the hardware was received and successfully configured.

By default, offers are active.

For offers with inactive status, you can specify a reason for that status.

About the Effective Periods of Offers in Bundles

Charge offers and discount offers have the following effective periods:

- The period specified in each offer that defines when the offer is generally available for purchase.
- The periods specified in a bundle that define when the offers in the bundle are effective.

The effective periods specified in a bundle take precedence over the effective periods specified in the offers themselves.

Setting the Effective Periods of Offers in Bundles

In bundles, you set the effective period of offers by specifying the start and end times of the offer and of the offer's cycle and usage charge periods.

- **Offer start and end times:** Specify when a customer can use the service or benefit from the discount. The offer start time is when the purchase fee is charged. It is also the earliest time that the offer's fees can begin to accumulate in an account balance.
- **Recurring and usage charge periods:** Specify when recurring and usage events can be charged or discounted. These periods must not begin before the offer start time.

You can set *start* times as follows:

- **Immediately:** (the default) The offer or charge is effective and can be activated immediately. The purchase fee is charged as soon as the offer is added to the account.
- **Relative to Purchase:** The purchase fee is charged when the offer is added to the account, but the customer cannot use the service or benefit from the discount until the relative period ends.

  When a charge offer's charge period has a relative start time, the events are not rated and the fees are not charged until the relative period ends, even if the service has been activated. This option enables you to waive subscription or usage fees for a period of time.

  When a discount offer's charge period has a relative start time, the discount is not applied to cycle or usage fees until the relative period ends.
- **First Usage:** The offer is activated and the purchase fee is charged when the customer first uses the service, such as by making a phone call. The charge periods also begin at that time.

You can set the *end* times as follows:

- **Never:** (the default) After it is activated, the offer is effective indefinitely, and its recurring and usage fees can be charged or discounted indefinitely.
- **Relative to Start:** After the relative period ends, the offer is not effective and the recurring or usage fees are not charged or discounted.

If the offer end time specified in the bundle is earlier than the recurring or usage charge end time, the offer end time overrides the charge end time.

Specifying Discounts in Bundles

In addition to adding discount offers to a bundle, you can reduce one-time and recurring charges in charge offers by specifying a flat percentage discount for each charge category in the bundle itself. For example, in Figure 2–25, all charges in the offer are discounted by 10%.

Figure 2–25 Flat Percentage Discounts Applied to Charges in a Bundle

When you discount a charge category in a bundle, the discount applies to all charge types that apply to the charge category. For example, you might have a charge offer that charges for two recurring charges; Monthly Fee and Included Minutes. If you specify a percentage discount for recurring charges in the bundle, the discount applies to both recurring charges. To discount only the Monthly Fee charge, you must make the balance impact for the monthly fee discountable and the balance impact that grants Included Minutes nondiscountable.

Unlike discount offers, discounts configured in bundles cannot be tracked in the general ledger because they are not associated with a general ledger (G/L) ID. In addition, unlike discount offers, bundle-configured discounts are difficult to display on a customer's bill because they are not separate items.

Note: Because of the limitations associated with discounts configured in bundles, Oracle recommends that you use discount offers instead. See "About Discount Offers" for more information.

Creating Dependencies for Bundles

You can set up the following dependent relationships between bundles:

- **Prerequisite:** Specifies that an account must own a particular bundle to be able to purchase another particular bundle. A prerequisite can include bundles for different services. For example, to own a GPRS bundle, an account might be required to own a GSM bundle.

- **Mutually Exclusive:** Sets up a mutually exclusive relationship between two bundles so that if an account owns one of the bundles, it cannot own the other. For example, if you set up a mutually exclusive relationship between a Corporate Voice bundle and a Residential Voice bundle, customers who purchase one cannot purchase the other.

Transitioning between Bundles

You can configure rules for transitioning from one bundle to another. PDC provides the following types of transitions:

- **Upgrade** to a bundle that is typically more expensive and has more features
- **Downgrade** to a bundle that is typically less expensive and has fewer features

Transition rules enable you to limit the bundles that customers can transition to and remain fully provisioned.

While transitioning from one bundle to another, your customers retain their devices, such as phone numbers, and their services.

Note: Transitioning between bundles affects only the charge offers and discount offers related to the service to which the bundle applies; the service itself remains the same. To define a transition that adds a new service, you create a transition for a package (see "Transitioning between Packages").

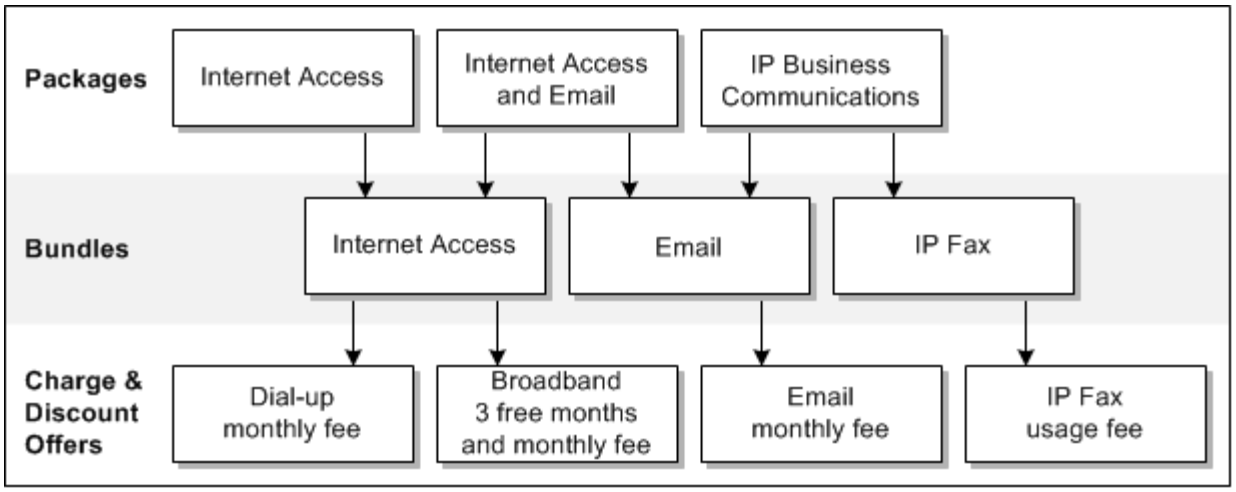
About Packages

A package (sometimes known as a plan or a service plan) is a collection of bundles. You use packages to offer your services to customers. For example, if your company sells Internet access, email, and IP fax service, you might offer the following packages:

- A package that sells only Internet access (dial-up or broadband)
- A package that sells Internet access and email
- A package that sells email and IP fax service

Figure 2–26 shows how you can include bundles and the services to which they apply in a variety of packages.

Figure 2–26 A Bundle Can Be Included in Multiple Packages



A package can contain any number of bundles, and two different packages can share the same bundle. By grouping bundles into packages, you simplify the choices presented to customers.

In addition to bundles that apply to services, a package can contain one bundle that applies to the customer's account instead of to a service. See "Making a Bundle Applicable to a Service" for more information.

Before creating a package, you should understand the concepts described in this section. For more information, see the PDC Help.

Billing Customers on Package Purchase

Usually, you bill a customer at the end of the customer's billing cycle. On-purchase billing, however, enables you to bill a customer immediately for a purchase, even if the customer's billing cycle has not ended.

When you create a package, you can flag it for on-purchase billing. When a customer purchases a package that is flagged for on-purchase billing, a bill is generated immediately for the purchase fees associated with the package.

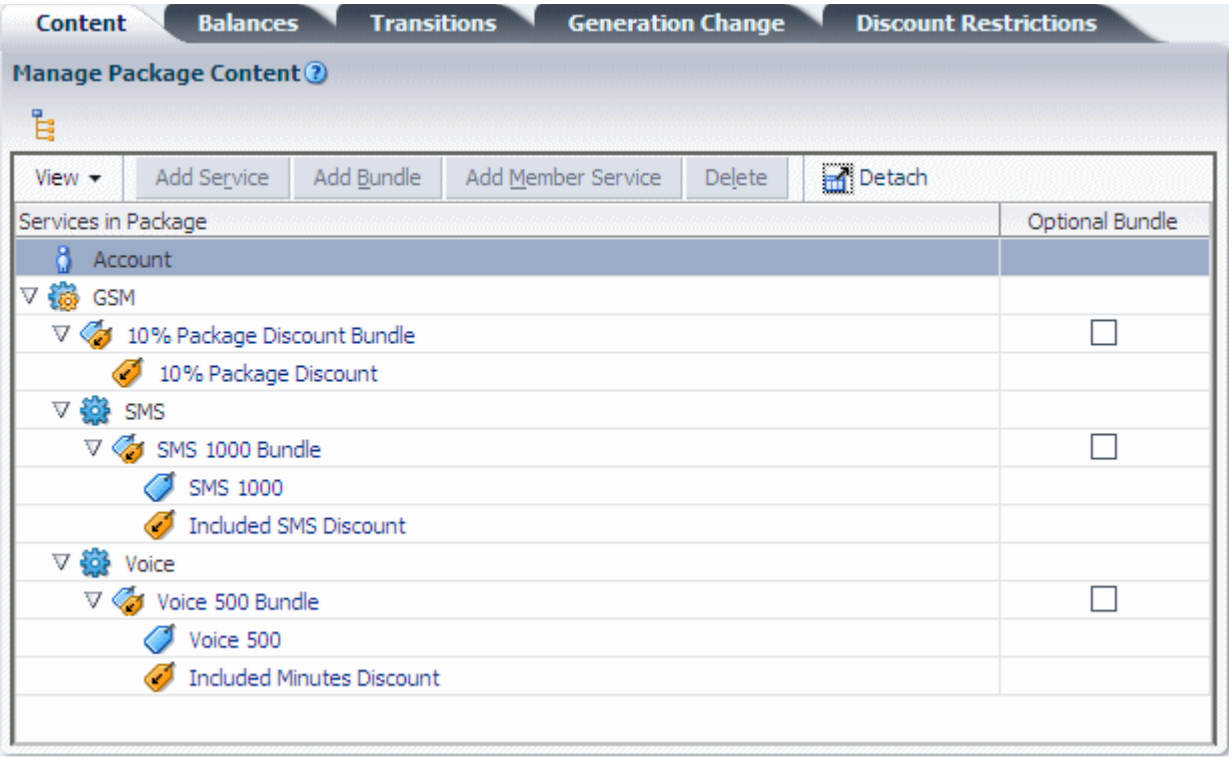
Note: On-purchase billing works with purchase fees only. It does not work with recurring, usage, or cancellation fees.

See also "Billing Customers on Bundle Purchase" for more information.

Adding Bundles to Packages

To add a bundle to a package, you must first add a service to the package. For example, the package content shown in Figure 2–27 includes the GSM, SMS, and Voice services.

Figure 2–27 Add Services to a Package and Then Add Bundles to the Services



After adding services to the package, you add bundles to the services. For example, the package contents shown in Figure 2–27 includes the following bundles:

- 10% Package Discount Bundle
- SMS 1000 Bundle
- Voice 500 Bundle

> **Note:** The content of a package always contains an Account node. To add an account bundle to the package, you use that node. See "Making a Bundle Applicable to a Service" for more information.

Including Optional Bundles in Packages

Bundles can be optional or required for packages:

- *Required bundles* are automatically purchased when a package is purchased. By default, all bundles are required.
- *Optional bundles* can be purchased at a later time.

Grouping Services

In packages, you can combine services into service groups, which are sets of services associated with a subscription, such as telephony and text messaging services associated with a wireless connection.

To create a service group, you add a service that represents the subscription (the subscription service) and then add member services to that service. The subscription

service can be a service that subscribers use, such as telephony or broadband Internet access, or it can be a representational service with no associated usage fees.

Service groups provide the following benefits:

- Member services can benefit from discounts owned by the subscription service.
- Member services can be associated with the devices owned by the subscription service.
- Customers can purchase services as a group.

To group services, you add a service that represents the group to the package (the *subscription service*) and then add *member services* to that service. Together, the subscription service and member services form a *service group*.

For example, in Figure 2–27, the GSM service is a subscription service, and SMS and Voice are its member services.

Grouped services usually apply to a particular device, such as a cable box. An account can contain multiple service groups.

To subscribe to a service group, customers subscribe to its subscription service.

Tracking Balances by Service

A balance is the amount of a balance element that a customer owes to your company or that your company owes to a customer. A balance can also be a quantity tracked by a counter. (See "About Balance Elements" for more information about balance elements.)

Often, purchasing and using a service affects multiple balances. For example, a mobile phone service might have an impact on two balances: US Dollars and Minutes.

Balance groups are collections of balances for various balance elements, such as currency, minutes, bytes, and frequent flyer miles, that apply to one or more services.

By default, a package contains one balance group: the account balance group. Balances for every service in the package share that balance group, which means that balances such as included minutes are shared among all services.

To track and control the allocation and consumption of balances for specific services, you can create multiple balance groups and assign services to their own balance groups, or group sets of services by balance group.

If you do not create additional balance groups, balances for every service in the package share the account balance group. This means that balances such as included minutes are shared among all services. By creating a balance group for each service, you can control the allocation and consumption of balances by an individual service.

For example, consider a family of four that has a mobile phone service for each family member. The bundle for each service includes 300 minutes. If the package has only the default account balance group, all four family members' included minutes go into the same account balance, which would contain 1200 included minutes. One teenager could then use 1000 minutes, leaving only 200 minutes to be shared by the other three family members.

To avoid that situation, you could create a package with two mobile phone services in the same bundle for the parents (because they know how to share) and an optional add-on package in its own balance group. When the parents purchase add-on packages for the children, each child's included minutes are tracked separately as sub-balances in the included minutes balance element.

You can set different credit limits for the same balance element in each balance group. For example, the parents might request a credit limit of $10 on the US Dollars balance element in each child's balance group to control overage charges but request a $100 credit limit on the US Dollars balance in the balance group for their mobile phone services. See "Applying Credit Limits to Balance Elements" for more information.

In service groups, the subscription service and member services typically belong to the same balance group so that they share their balances. See "Grouping Services" for more information.

Applying Credit Limits to Balance Elements

A credit limit is the maximum amount of a balance element that can accumulate in a balance group. When a credit limit is reached, businesses typically deny customers access to the services associated with the balance group. For example, you might set a credit limit of $100 for a telephony package and deny service when customers who reach that limit try to place a call.

You set credit limits for balance elements in packages.

About Credit Thresholds and Floors

You can use credit thresholds to notify customers when they are approaching the credit limit of a balance. A credit threshold specifies the balance total that triggers a notification to the customer. You can specify the threshold in the following ways:

- As a *fixed value,* such as $100 or 30 minutes.
- As a *percentage* of the credit limit, such as 90%. For example, if the credit limit is $100 and the threshold is 90%, the threshold amount is reached when the customer has a balance of $90 (that is, when the customer has used 90% of the balance).

The *credit floor* is the starting point for a credit threshold and is the lowest number that the balance can be (that is, the number that represents no use of the balance).

For currency balances, the credit floor is 0.

For noncurrency balance elements, such as prepaid hours, you must specify a credit floor. You can use a negative number for the floor. For example, suppose you give 100 prepaid hours and set the credit limit to 0. When the credit limit is reached, the customer has no hours remaining and cannot use the service. To notify the customer when only 10 hours remain, set the credit threshold and floor as follows:

- Set the credit floor to -100. This number indicates none of the balance has been used.
- Set the credit threshold to 90%.

The threshold is reached at 90% of -100 hours (that is, when the customer has 10 prepaid hours left).

The credit threshold can be triggered both when a balance is increasing and when it is decreasing. You can customize your billing system to perform different actions in each case. For example, if the credit threshold is crossed when the balance is increasing, service could be turned off. If the threshold is crossed when the balance is decreasing, service could be restored.

Specifying the Order in Which Sub-Balances Are Consumed

You might need to use consumption rules to manage sub-balances.

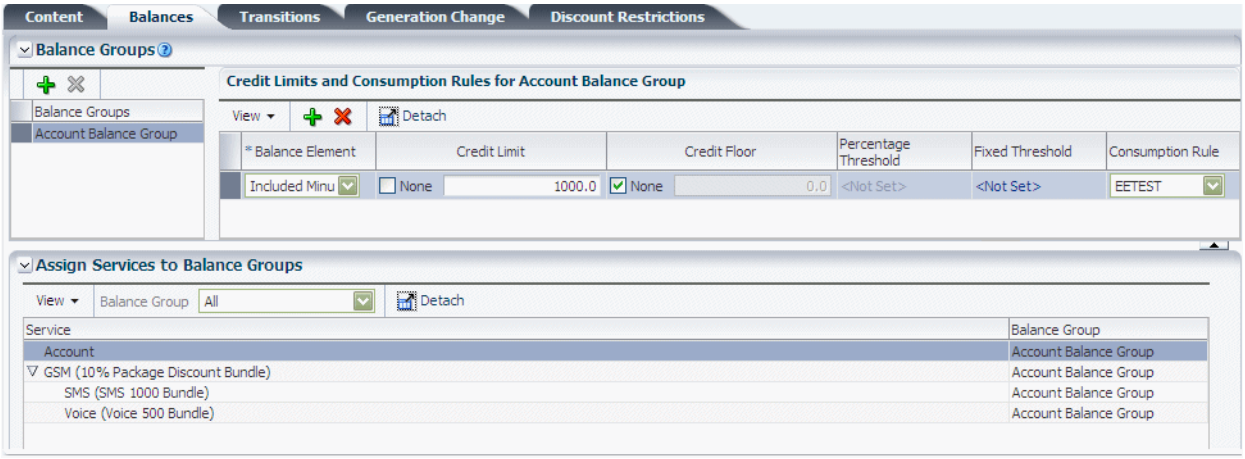
In a balance group, balances for the same type of balance element are combined into a sub-balance. For example, an account that owns two charge offers that each include 300 minutes has a starting included minute balance of 600 minutes, providing the services are associated with the same balance group and have the same validity period for the included minutes.

When portions of a balance element are valid at different times, however, BRM creates multiple sub-balances for that balance element in the balance group. For example, a balance group might include 300 minutes that are valid only for the current billing cycle and 1000 minutes that never expire. Because the included minutes have different validity periods, they are tracked in different sub-balances.

When a customer uses a service, BRM must know which sub-balance to use first. To specify the order in which sub-balances are consumed, you use consumption rules. The rules are based on the start and end times of the sub-balances.

As shown in Figure 2–28, you associate consumption rules with balances in packages.

Figure 2–28 Consumption Rules for Balance Groups



For example, you can specify whether a subscriber with the following minute sub-balances uses the Anytime Minutes or the rollover Anytime Minutes first:

- 100 Anytime Minutes that are valid March 1 to April 30
- 50 rollover Anytime Minutes that are valid February 1 to March 30

If the minute balance is associated with a rule that says to consume the sub-balance that expires first, the 50 rollover Anytime Minutes are used first.

If the minute balance is associated with a rule that says to consume the sub-balance with the latest start time, the 100 Anytime Minutes are used first.

For more information, see the PDC Help.

Transitioning between Packages

You can configure rules for transitioning from one package to another. Each transition rule applies to a particular service. The package being transitioned to and the package being transitioned from must both contain that service, though each package can contain additional services that the other package does not contain.

PDC provides the following types of transitions:

- **Upgrade** to a package that is typically more expensive and has more features. For example, a customer might transition from a package that provides Internet and cable TV services to a package that provides Internet, cable TV, and VOIP services. That transition adds a service and a bundle and possibly changes the bundles for the existing services.
- **Downgrade** to a package that is typically less expensive and has fewer features.

Transitions enable you to limit the packages that customers can switch to while remaining fully provisioned. When transitioning to a designated package, your customers retain their devices, such as phone numbers, and their services.

Defining Generation Changes

A generation change enables you to transition customers between 2G (second generation) and 3G (third generation) wireless packages and services. Packages are called 2G or a 3G depending on whether the wireless service they provide runs on a 2G or 3G wireless network.

When creating a 2G or 3G package, you can set up generation change rules to specify which packages can replace the package when it is transitioned to a different generation.

For generation changes, the packages do not have to provide the same service.

> **Note:** The same package cannot be used in both a transition rule and a generation change.

Restricting Discounts in Packages

When creating a package, you can prohibit specified discount offers from being owned or purchased while the package is owned.

If an exclusion relationship exists between a discount and a package, a customer can own the discount or the package, but not both. Further, the customer cannot own any discounts associated with the package if the customer owns the excluded discount.

Creating CSR Packages in BRM

This section applies only to BRM users.

In BRM, packages provide services not only to customers, but also to CSRs. CSR packages serve the following purposes:

- They control access to Customer Center.
- They enable customer management events to be recorded, including information about the CSR who generated the event. This information can be helpful when researching customer complaints.

To create a CSR package, you use PDC to create a package that has the following attributes:

- Uses the **admin_client** service. That service provides access to Customer Center users.
- Includes no bundles.

You then add the CSR package to the **CSR - New** package list (see "About Package Lists").

About Package Lists

A package list is a group of packages that is usually offered to a single type of customer. For example, you might create the following package lists:

- A package list that includes packages for customers above a certain age.
- A package list that includes packages for customers in a particular location, such as Canada.
- A package list that includes promotional discounts offered for a limited time.

You can create any number of package lists for your system, and each package list can contain any number of packages. Different package lists can contain the same package.

The package list does not have to include all your packages. You can create packages and not include them in a package list until you need them. Or you can offer one set of packages to one group of potential customers and another set of packages to another group.

To make a package available for customers to purchase, you must include the package in a package list.

Note: For implementations covering a large geographic area, you might need package lists containing *regional product offerings*, each with variations in the pricing structure.

Specifying the Package List Segment and Type

The package list segment identifies the customer segment that the package list is offered to, and the package list type identifies the type of packages that are added to the package list.

The package list segment and type determines the package list name in Customer Center when the package list is used in BRM. For Customer Center, by default, the package list segment must be **CSR**. The package list type can be either **New** or **Add-on**. For example:

- The **CSR-New** package list contains packages that register customers and add the services and bundles in the packages to the new accounts.
- The **CSR-Add-on** package list contains packages that add services and bundles to existing accounts.

The package list segment and type are case sensitive and together uniquely identify package lists in Customer Center. For example, **CSR-New** and **CSR-new** are two different package lists.

Specifying the Package List Status

You can assign the following statuses to a package list:

- **Active:** Use this status to indicate that customers can purchase packages in the list as soon as the list is added to your system.
- **Inactive:** Use this status to indicate that the list is not visible to customers and customers cannot purchase packages from the list.

About Setting Up a Product Offering

When setting up a product offering, consider the following:

- Which services your company offers and how much to charge for them. For example, you should decide the following:
  - The types of fees to use for a service, such as a flat monthly fee, hourly usage fees, or both
  - The rates to charge for monthly fees, hourly usage, setup, and so forth
  - Which discounts to offer
  - Whether to offer special pricing options, such as sponsored rates
- How to implement your services and their associated balance elements, events, G/L IDs, and tax codes in PDC. See "Prerequisites for Creating a Product Offering" for more information.
- How to structure your product offering. For example, determine how many packages and bundles to create and the charge offers and discount offers that make up the packages.

> **Tip:** Because product offerings are built by setting up relationships among packages, bundles, charge offers, and discount offers, it is helpful to draw a diagram of your product offering before you create it in PDC. For example, see "Example Product Offering".

After planning your product offering, you use PDC to create it. Because charge offers and discount offers are the basic units of a product offering, they are created first. See "About Using PDC to Create Product Offerings" for more information.

Prerequisites for Creating a Product Offering

Before you create a product offering, you must complete the following tasks. See "About Configuring Setup Components" for information about configuring these items in PDC.

- **Create services and events**

  You might need to add or modify services before you create your product offering. In addition, you must configure a list of events to track for each service. If you add services, you might need to create events to track the services.

- **Collect account attribute data**

  To structure pricing, you might want to use account information, such as the location of a customer and the type of account (such as Premium or Standard).

- **Define RUMs for events**

  Before your billing system can apply a charge to an event, it must measure the event. To enable your billing system to measure events, you configure RUMs, which specify the units to measure and how to calculate the measurement.

- **Map event types to services**

  Each charge offer applies to a particular service. When you create a charge offer, you select events related to that service to configure charges for. To prevent you from selecting an event that does not occur for a service, PDC uses the service-event map.

- **Create balance elements**

  Charges require balance elements. Use PDC to add or modify balance elements.

- **Define impact categories**

  You use impact categories to specify that a particular group of balance impacts in a charge should be used. If you plan to base charges on the values of event attributes, you must define some impact categories.

- **Configure zones**

  Set up value maps and zone models.

- **Define tax codes and tax suppliers**

  To calculate taxes, you must define tax codes and tax suppliers.

- **Create G/L IDs**

  You use G/L IDs to collect general ledger information from your database and to export it to your accounting application. You must decide how to track the revenue for each type of charge and create the appropriate G/L IDs.

- **Define provisioning tags**

  If you use charge offer or discount offer provisioning, you must define provisioning tags.

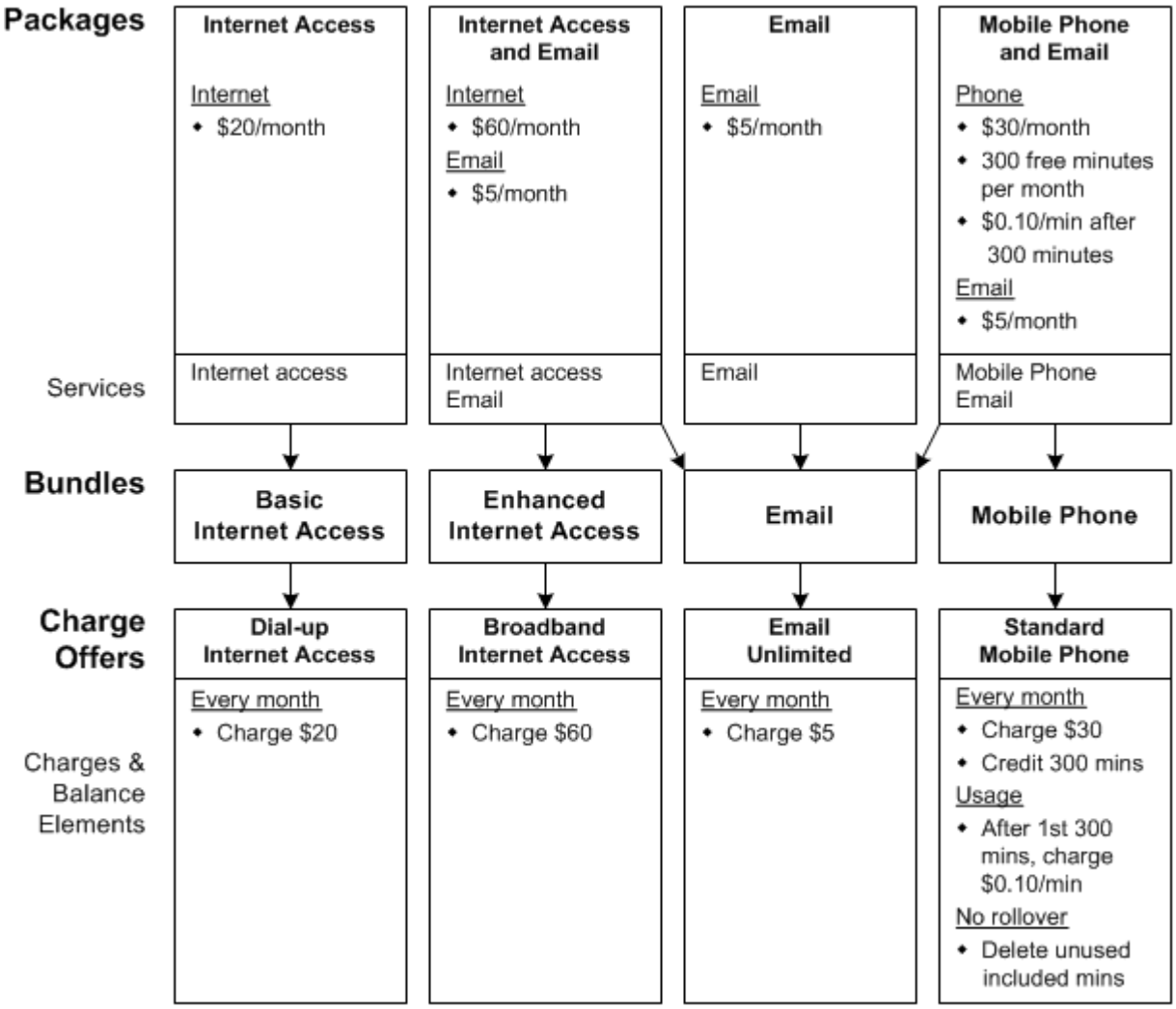
- **Configure time models and special day calendars**

  You use time models and special day calendars to charge different prices for the same service depending on the day and time the service is used.

Example Product Offering

The example product offering in Figure 2–29 offers three services: Internet Access, Email, and Mobile Phone.

Figure 2–29 Example Product Offering for Three Services



Note the following points about the organization of the product offering in Figure 2–29:

- The offer contains two charge offers that have different monthly fees for Internet access. These charge offers are used by different bundles.
- Two of the packages include two bundles; the other two packages include only one bundle each.

About Using PDC to Create Product Offerings

With PDC, you can perform the following tasks:

- Configure setup components, which contain data used to create pricing components. Setup components include balance elements, impact categories, RUMs, the service-event map, and special day calendars.

  See "About Configuring Setup Components" for more information.

- Create and modify pricing components, which are elements that define product offerings. Pricing components include charges, discounts, chargeshares, charge

offers, discount offers, chargeshare offers, bundles, packages, package lists, time models, and selectors.

- Configure complex charges for your services based on event attributes and quantities, impact categories, time periods, special dates, and prioritized selection rules.
- Set credit limits and discounts.
- Reuse charges and discounts, charge and discount offers, bundles, and packages so that you do not need to re-create them for each pricing component that references them.
- Create and modify pricing components in draft form in your private workspace before submitting them to public view.

> **Note:** When you start PDC, the Workspace page is open by default. You use this page to manage the setup and pricing components in your changesets. For more information, see "About Changesets" and the PDC Help.

- Use the **ImportExportPricing** utility to export setup components from PDC into XML files and to import setup components from XML files into PDC. See "Importing and Exporting Pricing and Setup Components" for more information.

See "Example of Using PDC to Create a Product Offering" for more information.

Who Uses PDC?

Generally, the services you offer and how you charge for them are defined by your marketing and finance personnel. The product offering is typically created by operations personnel using PDC. You should outline, in detail, your entire pricing structure before you use PDC to implement it.

About Using the XML Pricing Interface to Create a Product Offering

As an alternative to using PDC, you can create and modify product offerings in an XML file and then use the **ImportExportPricing** utility to do the following:

- Import an entire or partial product offering configured in an XML file into the PDC database.

  The utility creates any new pricing components and modifies any changed components in the PDC database.

- Export an entire or partial product offering from the PDC database into an XML file for editing.

  If you export the data into an XML file that contains pricing components, the utility overwrites the entire XML file.

To create or modify a product offering in an XML file, you use a text editor or an XML editor. The XML product offering must follow the format specified in the appropriate XML Schema Definition (XSD) file. See "ImportExportPricing" for more information.

You can also use the **ImportExportPricing** utility to move a product offering from a PDC database to another PDC database.

See "Importing and Exporting Pricing and Setup Components" for more information.

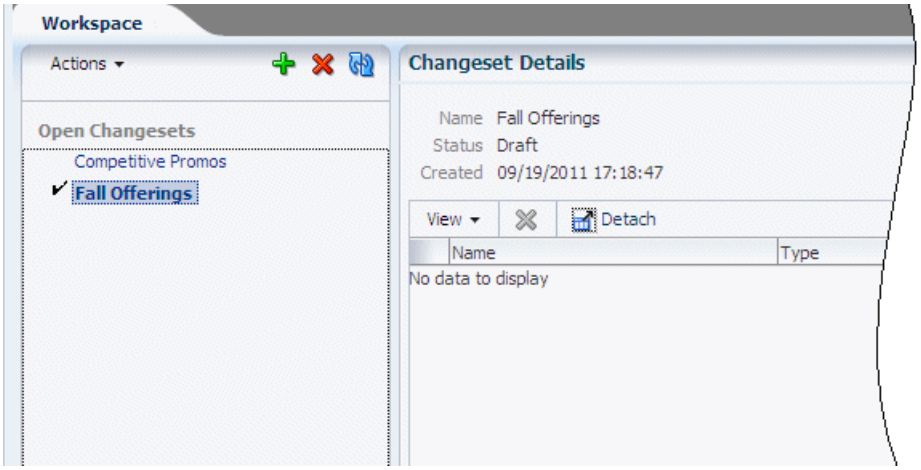
Example of Using PDC to Create a Product Offering

When you are ready to use PDC to create a product offering, the product offering should be planned, and all necessary prerequisites, such as balance elements, G/L IDs, and RUMs, should be in place. See "About Setting Up a Product Offering" for more information.

The following procedure provides a high-level example of how to use PDC to create a product offering:

1. Start by creating an active changeset to contain the components of the product offering.

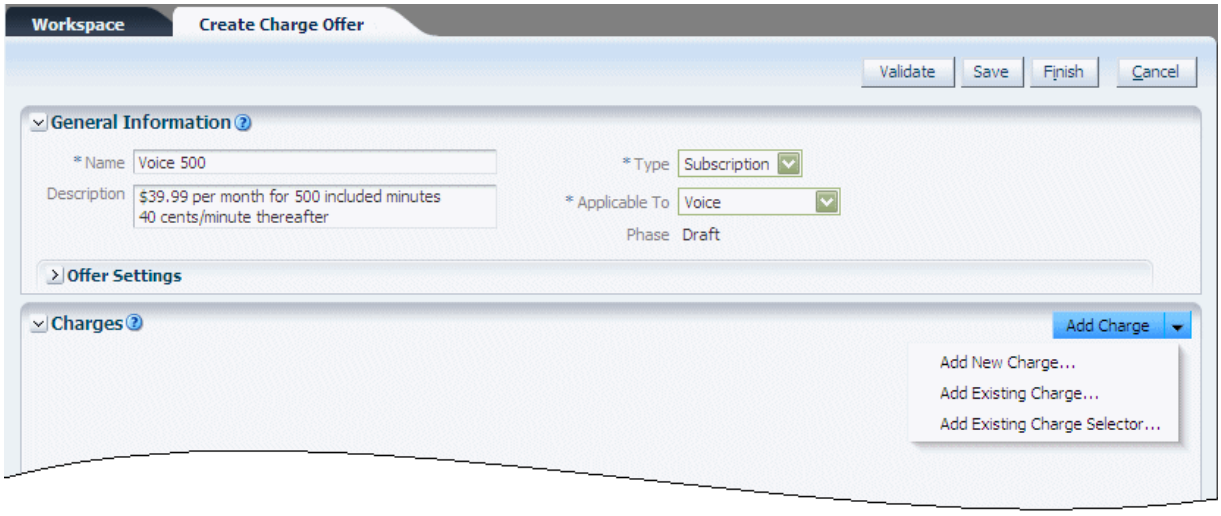
   The following example shows an active changeset named Fall Offerings in the Open Changesets section of the Workspace page:



   See "About Changesets" for more information.

2. Then create charge offers.

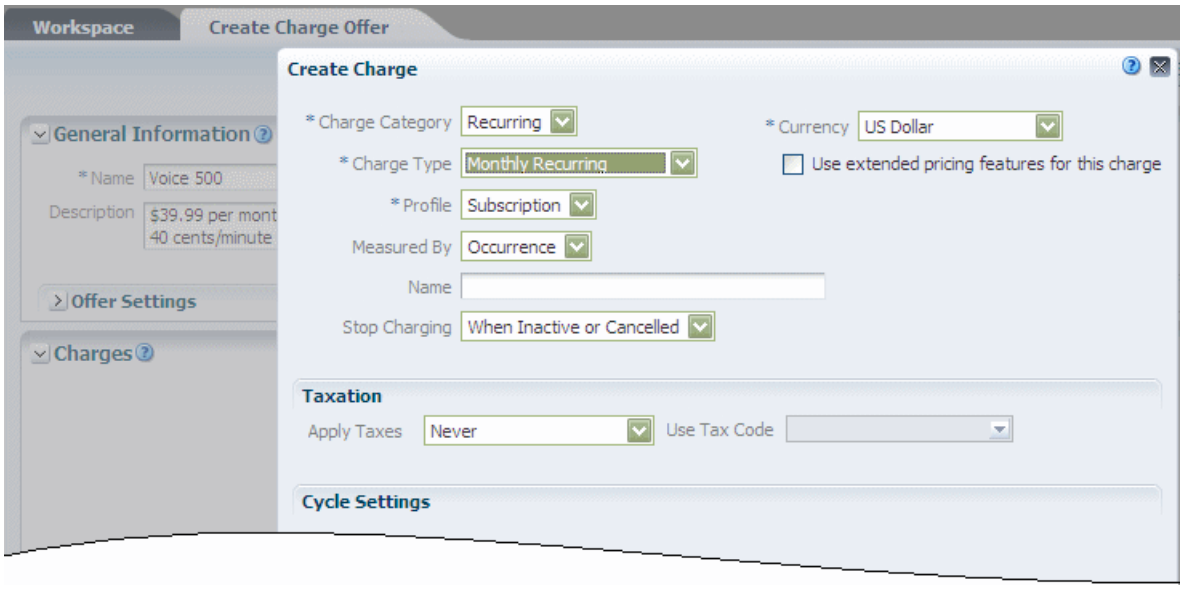
   The following example shows a charge offer named Voice 500 being created in the Create Charge Offer page:



   See "About Charge Offers" for more information.

3. After specifying general information for the charge offer, add charges to specify the cost of the events associated with the service to which the charge offer applies.

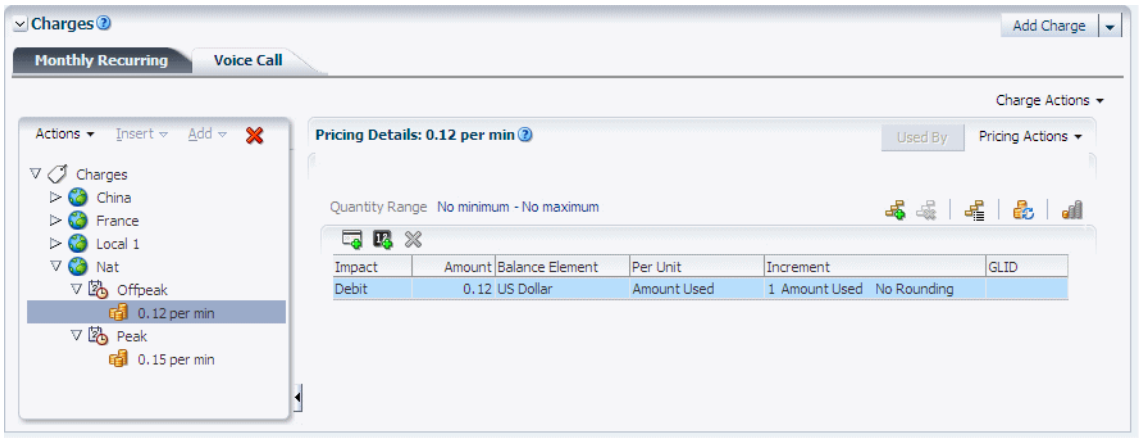
The following example shows a monthly service charge being created for the Voice 500 charge offer:



See "Adding Charges to Charge Offers" for more information.

4. Configure balance impacts for each charge.

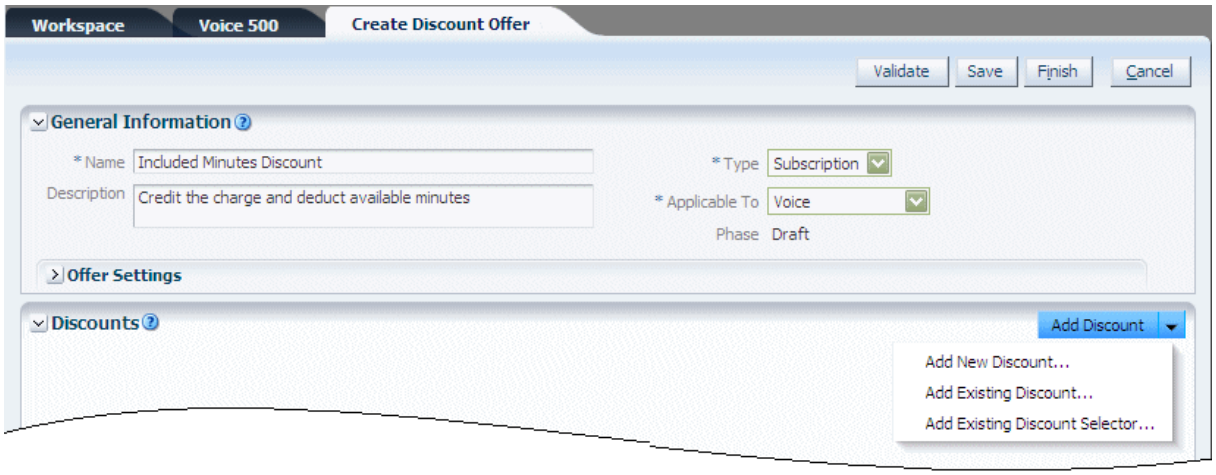
   The following example shows a balance impact for the Nat impact category and Offpeak time period in a usage charge for the Voice 500 charge offer:



See "Configuring Pricing in Charges" for more information.

5. Optionally, create discount offers for the service.

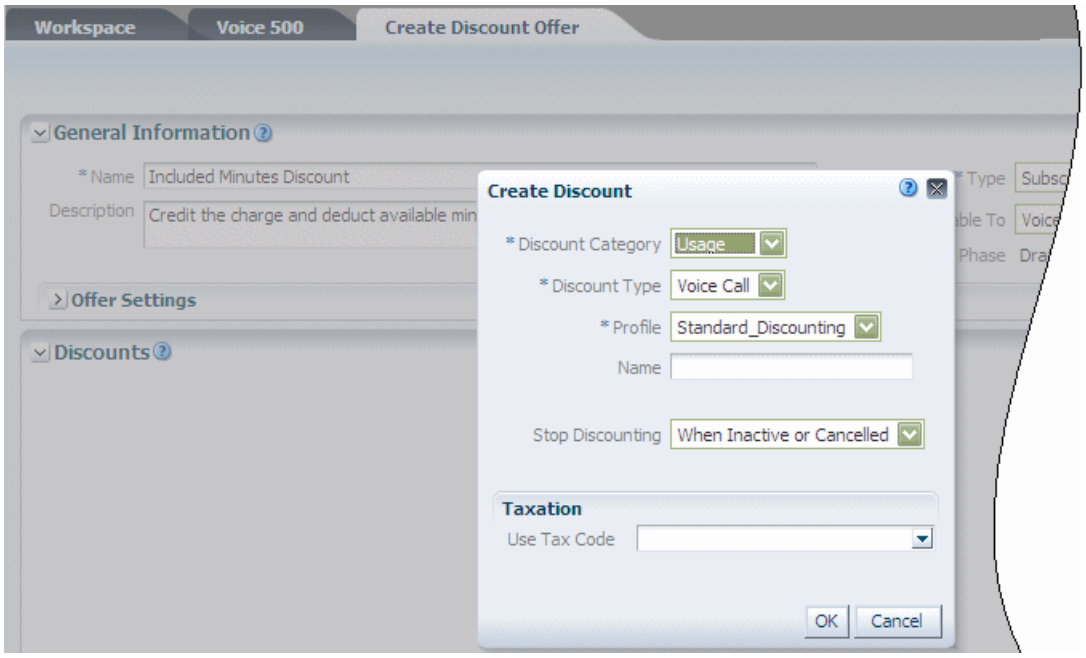
   The following example shows a discount offer named Included Minutes Discount being created in the Create Discount Offer page:



See "About Discount Offers" for more information about creating discount offers.

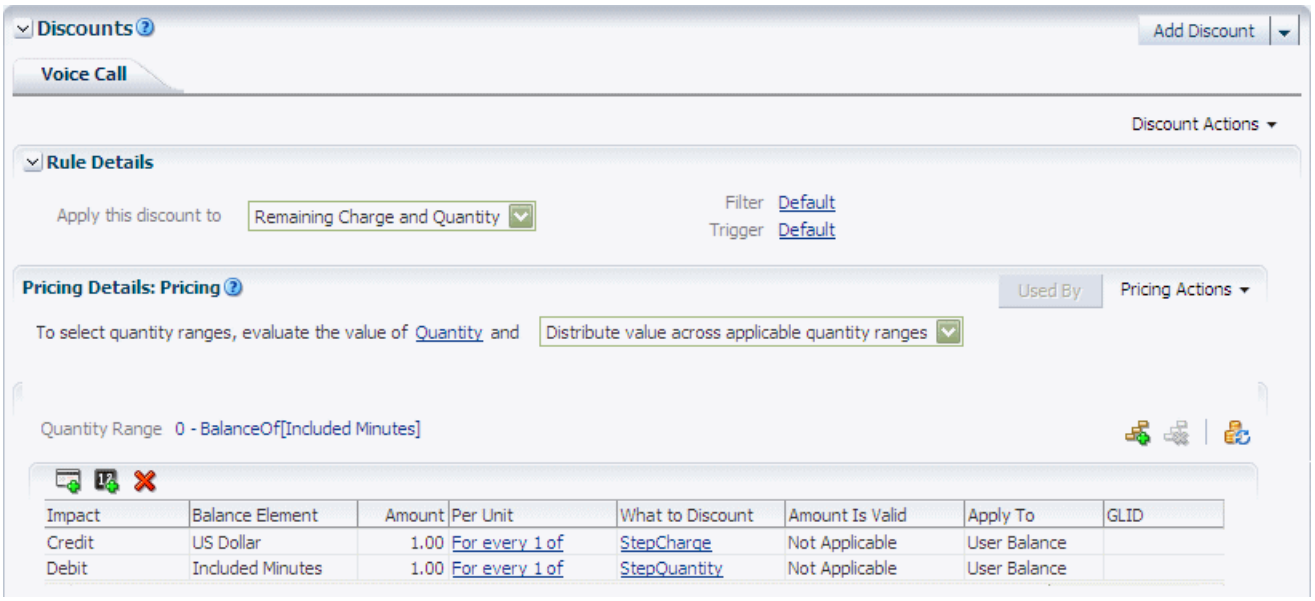
6. After specifying general information for the discount offer, add discounts to reduce the cost of events associated with the service to which the discount offer applies.

   The following example shows a usage discount being created for the Included Minutes Discount discount offer:



See "About Discounts" for more information.

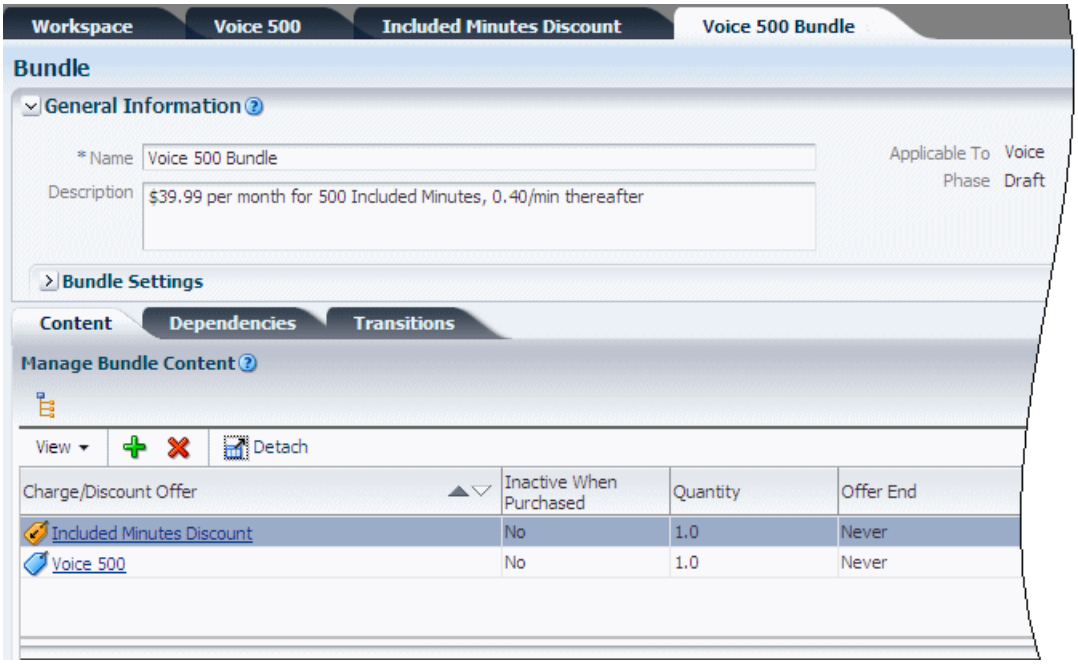
7. In each discount, configure rules and balance impacts:



See "About Discount Rules" and "About Discount Pricing" for more information.

8. When the charge offers and discount offers are finished, create bundles and add the offers to them.

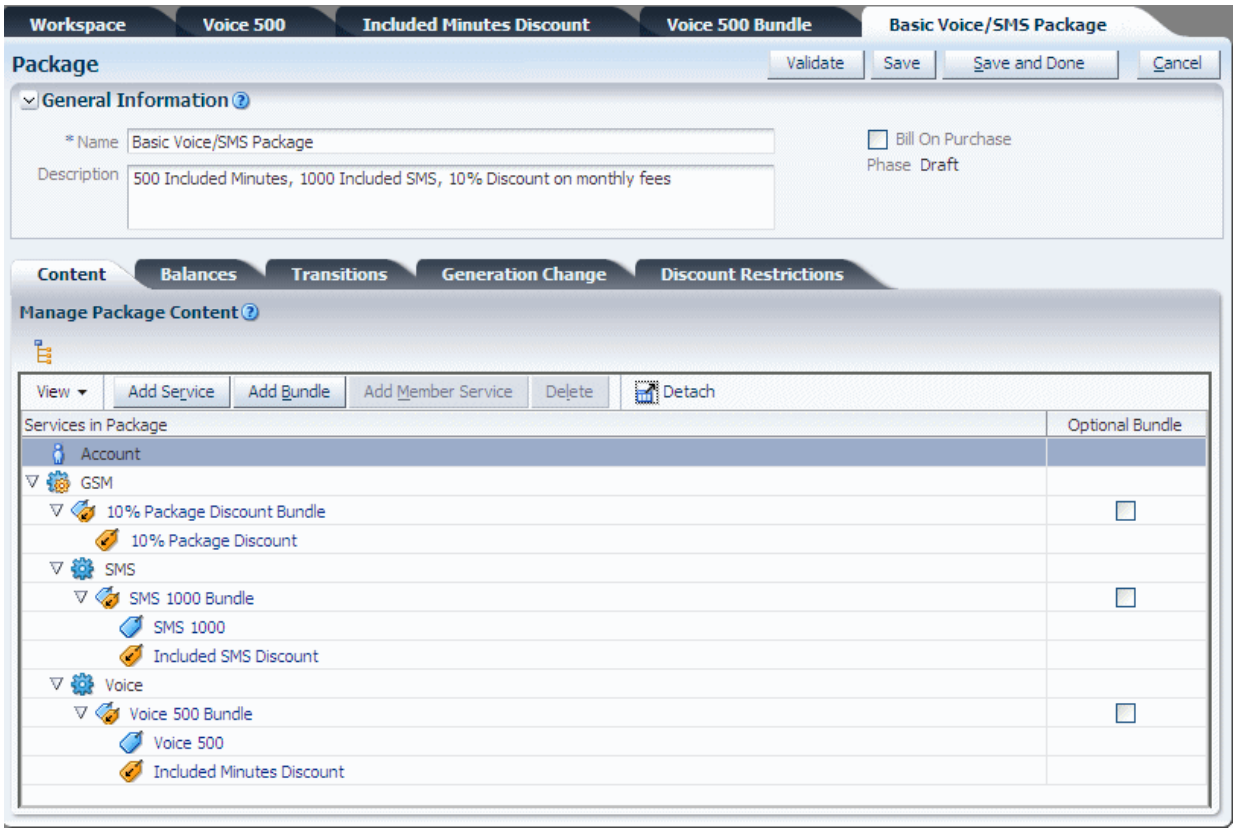
   In the following example, the Voice 500 charge offer and the Included Minutes Discount discount offer have been added to Voice 500 Bundle:



See "About Bundles" for more information.

9. Then create packages, specify the services they provide, and add bundles for those services to the packages.

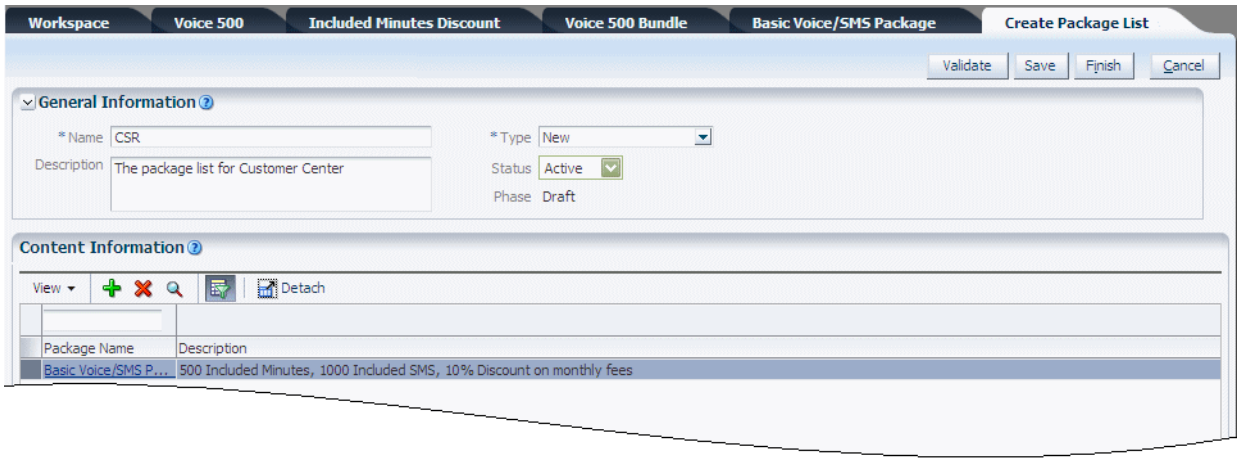
   In the following example, Voice 500 Bundle has been added to a package that provides voice and SMS services for mobile phone users:



See "About Packages" for more information.

10. After creating packages, add them to one or more package lists.

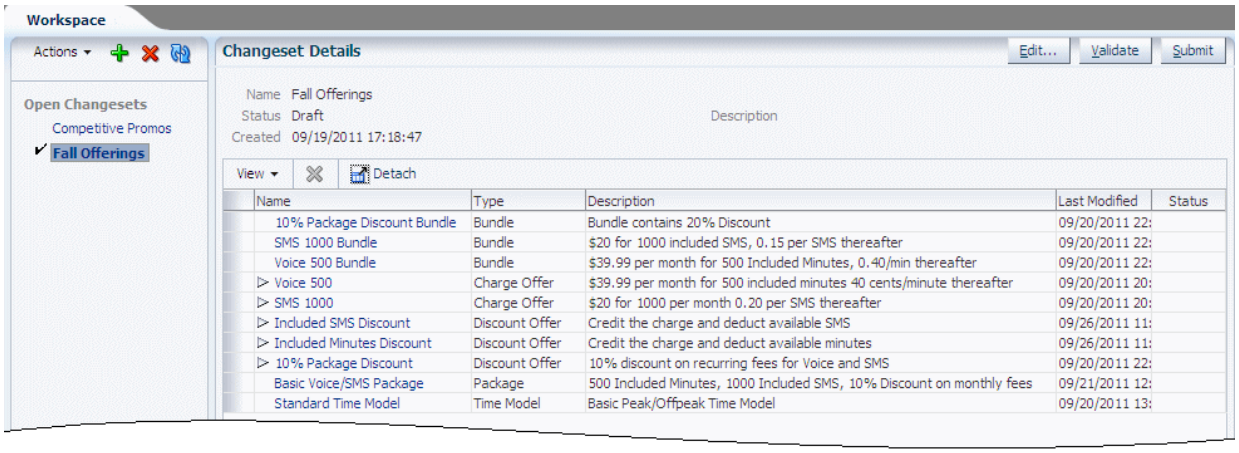
    In the following example, Basic Voice/SMS Package has been added to the new CSR package list:



See "About Package Lists" for more information.

11. After creating all the components in the product offering, validate the changeset that contains them, fix any validation errors, and then submit the changeset to a rating engine.

    For example, the components shown in this procedure are part of the Fall Offerings changeset shown in the following figure:



For information about validating changesets, see the PDC Help.

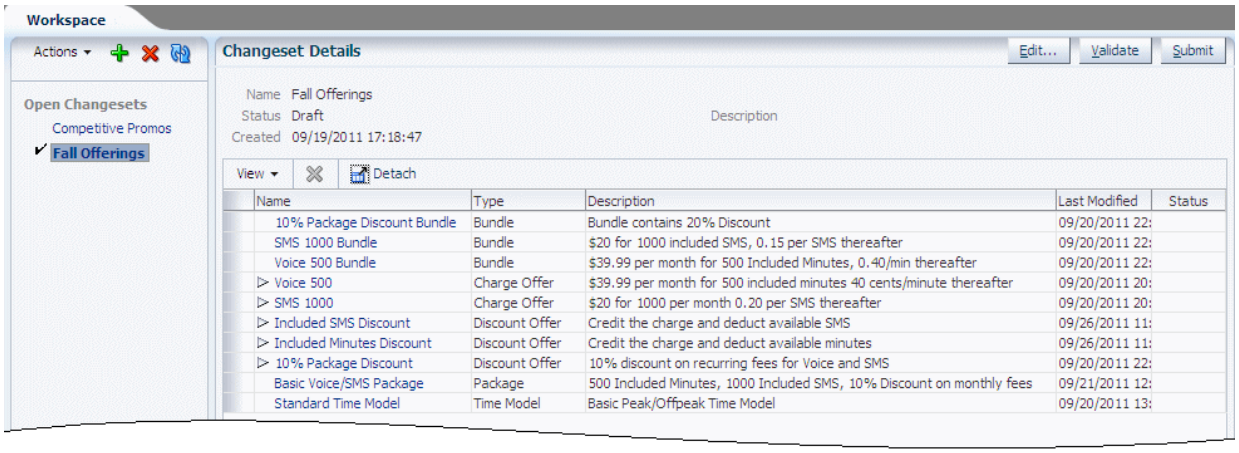
See "About Changesets" for information about submitting changesets.

12. Test the product offering by creating accounts that use your new package, generating activity, and running billing to ensure that the account balances are correctly impacted.

About Changesets

In PDC, you create and modify all setup and pricing components in the context of a changeset, which is a group of pricing components. Each user has one or more changesets, which only the user can access. As shown in Figure 2–30, changesets are listed and displayed in the user's workspace, which is the page that appears by default when PDC starts. When you select a changeset in a changeset list, its components are listed in the Changeset Details section of the page.

Figure 2–30 Selected Changeset Displayed in the Changeset Details Section of Workspace Page



If you do not have a changeset, PDC automatically creates one for you when you log in. This changeset becomes your active changeset. When you create a setup or pricing component, a draft of the new component is added to your active changeset. When you modify a component, the updated component remains in the changeset in which it was created. If you have multiple changesets, only one can be active, but you can designate a different changeset as your active changeset at any time.

Changeset States and Statuses

Changesets can have the following states:

- **Open:** Contains draft versions of new and modified pricing components. You can edit, validate, and submit open changesets.
- **Submitted:** Has been submitted for publication to a billing system and is undergoing transformation (see "About Publishing Pricing Components").
- **Closed:** Has been successfully transformed.

Changesets can have the following statuses:

- **Invalid:** An open changeset that failed validation.

  Before the components in a changeset can be published to a billing system, PDC must *validate* them to ensure that they are correctly configured for that billing system. Components that are incorrectly configured receive validation errors. For more information, see the PDC Help.
- **Pending:** A submitted changeset that is waiting for its transformation to be completed. Components in pending changesets are read only.
- **Successful:** A changeset whose components have been transformed into the appropriate format and published to the database of the target engine.
- **Failed:** A submitted changeset that contains one or more pricing components that failed transformation.

When a changeset is submitted, the **Pending** and **Successfully Submitted** links are displayed in the Workspace task pane. You can view the list of the pending or successfully submitted changesets by moving the mouse over the **Pending** or **Successfully Submitted** link.

About Publishing Pricing Components

To publish the setup or pricing components in a changeset to a billing system, you submit the changeset. You must submit an entire changeset; you cannot submit only some of its components.

When a changeset is initially submitted, the status of its components is changed from Draft to Promoted.

If the submission is successful, the components are transformed into the appropriate format and published to the database of the target engine associated with the components' pricing profile (see "Working with Profiles"). The changeset is then closed and removed from your workspace. You no longer have access to it. You can, however, access its promoted pricing components by using the PDC search functionality.

If the submission fails:

- The changeset status changes from Pending to Failed.
- The Changeset Details section of the Workspace page lists the errors that caused the failure. An error tab is displayed for each transformation engine to which the changeset was submitted.

  If the failure was due to a system error, you must check the load utility log and correct the error.

  If the failure was due to a pricing setup error, you can fix the changeset.

  If you choose to fix it, PDC performs one of the following operations, based on its analysis of the errors:

- Resubmits the failed changeset.
- Removes the failed changeset from your workspace and adds a new open changeset containing only the unpublished components. The new changeset includes a history of its predecessor changesets and a list of the errors that caused the original changeset to fail. After fixing the errors, resubmit the changeset.

  Tip: To see the latest status of a changeset, right-click the changeset name in the Workspace page and choose **Refresh**.

If you submit your last open changeset, PDC creates another open changeset for you.

About Importing Components into Changesets

You can import components from an XML file into a new or an existing open changeset. The imported components can be immediately submitted for publication.

For more information about using changesets, see the PDC Help.

About Exporting Components from Changesets

To share your changesets with other users, you can export the components from a changeset into an XML file.

Optionally, you can include all the components referenced by the components in the exported changeset, whether or not the components are part of that changeset.

3

Working with Profiles

This chapter describes the profiles used by Oracle Communications Pricing Design Center (PDC). It also explains how to create custom profiles.

You should have a basic understanding of the following:

- PDC (see "About Pricing Design Center")
- Product offerings (see "About Creating Product Offerings")

To create custom profiles, you must also understand the following:

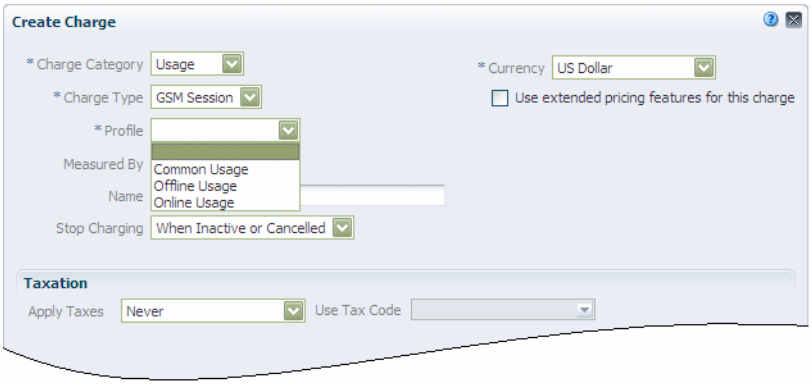
- Extensible markup language (XML) programming
- XML schema definition (XSD)

About Pricing Profiles

A *pricing profile* specifies the pricing features that are displayed in the PDC UI. You use pricing profiles to adapt the UI to your business needs and to simplify the user experience. For example, if you do not use time to determine how much to charge for the use of your services, you can set up a pricing profile that does not include the time feature.

Pricing profiles are used with pricing, charges, discounts, chargeshares, selectors, and time models. You select a pricing profile for all those components except pricing when creating them. (PDC selects a profile for pricing that matches the pricing profile of the component that contains the pricing, such as a charge.) For example, Figure 3–1 shows the list of profiles available for a new usage charge. After a pricing profile is selected, the features it supports are the only options that appear in the UI used to create or modify the usage charge.

Figure 3–1 Selecting a Pricing Profile for a Charge



A set of default pricing profiles is included with PDC (see "About Default Pricing Profiles"). You can also create custom pricing profiles (see "About Customizing Pricing Profiles").

Each pricing profile is associated with one or more *target engine profiles*. A target engine profile specifies all the pricing features supported by a *target engine*, which is the rating or subscription engine in the billing system to which you plan to publish your pricing components. Typically, a pricing profile contains a subset of the pricing features in the associated target engine profiles.

PDC includes several target engine profiles (see "About the PDC Target Engine Profiles").

When you validate and submit a component for publication, PDC uses the target engine profiles associated with the component's pricing profile to determine which rating or subscription engines to validate the component for and to publish the component to.

About Default Pricing Profiles

During PDC installation, depending on whether you selected to support Oracle Communications Elastic Charging Engine (ECE) or not, a set of default pricing profiles is loaded in XML format. The XML files containing the default pricing profiles are located in the *PDC_home*/**apps/Samples/Examples** directory, where *PDC_home* is the directory in which the PDC software is installed.

Table 3–1 lists the default pricing profiles that are loaded during PDC installation if ECE is not selected. These pricing profiles are supported by Oracle Communications Billing and Revenue Management (BRM) real-time and batch rating engines:

Table 3–1 Default Pricing Profiles That Are Loaded When ECE Is Not Selected

Profile Name	Description
Billing-time Discount	Contains all the features required to configure billing-time discounts in BRM (see Table 3–6, " Discounting Capabilities in the BRM Batch Rating Target Engine Profile"). This profile can be used with a discount or a discount selector.
ChargeSharing	Contains all the chargeshare features supported by the BRM batch rating engine (see Table 3–7, " Chargesharing Capabilities in the BRM Batch Rating Target Engine Profile"). This profile can be used with a chargeshare.
Offline Usage	Contains all the pricing features supported by the BRM batch rating engine (see Table 3–5, " Charging Capabilities in the BRM Batch Rating Target Engine Profile"). This profile can be used with a usage charge, time model, or selector.
Online Usage	Contains all the pricing features supported by the BRM real-time rating engine (see Table 3–4, " Capabilities in the BRM Real-Time Rating Target Engine Profile"). This profile can be used with a usage charge, time model, or selector.
Standard Discount	Contains all the discount features supported by the BRM batch rating engine (see Table 3–6, " Discounting Capabilities in the BRM Batch Rating Target Engine Profile"). This profile can be used with a discount or a discount selector.

Table 3–2 lists the default pricing profiles that are loaded during PDC installation if ECE is selected. These pricing profiles are supported by ECE rating engine:

Table 3–2 Default Pricing Profiles That Are Loaded When ECE Is Selected

Profile Name	Description
Billing-time Discount	Contains all the features required to configure billing-time discounts in BRM (see Table 3–6, " Discounting Capabilities in the BRM Batch Rating Target Engine Profile"). This profile can be used with a discount or a discount selector.
Convergent Usage	Contains all the pricing features supported by ECE (see Table 3–9, " Charging Capabilities in the ECE Rating Target Engine Profile"). This profile can be used with a usage charge, time model, or generic selector.
Standard Chargesharing	Contains all the chargeshare features supported by ECE (see Table 3–11, " Chargesharing Capabilities in the ECE Target Engine Profile"). This profile can be used with a chargeshare.
Standard Discount	Contains all the discount features supported by ECE (see Table 3–10, " Discounting Capabilities in the ECE Rating Target Engine Profile"). This profile can be used with a discount or a discount selector.
Subscription ChargeSharing	Contains all the chargeshare features supported by the BRM batch rating engine (see Table 3–7, " Chargesharing Capabilities in the BRM Batch Rating Target Engine Profile"). This profile can be used with a chargeshare.
Subscription Discount	Contains all the discount features supported by the BRM batch rating engine (see Table 3–6, " Discounting Capabilities in the BRM Batch Rating Target Engine Profile"). This profile can be used with a discount or a discount selector.

See "About Customizing Pricing Profiles" for information about modifying the default pricing profiles.

About Customizing Pricing Profiles

Typically, you create a custom pricing profile by copying one of the PDC default pricing profiles and then removing features that you do not need from the copy.

Note: In profile XML files, pricing features are defined in **<capability>** and **<subCapability>** elements.

Note: You cannot modify the Subscription pricing profile.

For example, if your business does not charge for services by time, you could remove the **TimeModeling** capability from the Online Usage default pricing profile.

> **Note:** If a capability is mandatory in the target engine profile associated with a pricing profile, the capability must be included in the pricing profile. Do not remove it. See "About the PDC Target Engine Profiles" for a list of capabilities supported by each PDC target engine profile. The list identifies the mandatory capabilities.
>
> If two or more mandatory capabilities are mutually exclusive, you can remove all but one of them from a custom profile. If you include more than one, you must also include the exclusive constraint (**<exclusiveConstraint>**) between them. See the default pricing profiles for examples.

The following rules apply for *subcapabilities*:

- If only one subcapability exists for a particular capability in a target engine profile, the pricing profile must include that subcapability. Do not remove it.
- If multiple subcapabilities exist for a particular capability in a target engine profile, at least one of those subcapabilities must be included in the pricing profile. The custom pricing profile does not, however, have to use the same subcapability as the default pricing profile.

  For example, suppose a target engine profile includes Capability X, which has Subcapabilities A, B, C, and D, and a default pricing profile uses Capability X with Subcapabilities A and B. If you customize a copy of that default pricing profile, the copy could include Capability X with Subcapability D. In this case, the custom pricing profile does not use the same subcapabilities as the default pricing profile.
- If you *add* capabilities to a custom pricing profile, ensure the profile contains capabilities for only one pricing type. See "About Pricing Types" for more information.
- Only capabilities that are supported by all the target engine profiles associated with a pricing profile can be added to a custom pricing profile. See "About the PDC Target Engine Profiles" for a list of capabilities supported by each PDC target engine profile. See "About Associating Custom Pricing Profiles with Target Engine Profiles" for information about associating pricing profiles with target engine profiles.

> **WARNING: After pricing components are associated with a pricing profile, you should not modify that pricing profile. Doing so might invalidate the pricing components that reference the profile.**

See "Setting Up Custom Pricing Profiles" for instructions on how to create custom profiles.

About Pricing Types

Each pricing profile can support only one pricing type. A *pricing type* performs the following functions:

- Indicates the type of pricing, such as a usage charge or a discount, that can be configured when the pricing profile is associated with a pricing component
- For custom profiles, determines the set of capabilities that are available for inclusion in the profile

In each pricing profile's XML file, the pricing type is specified in the **<pricingTypeValue>** element.

Table 3–3 shows the pricing types supported by the default pricing profiles:

Table 3–3 Pricing Types Supported by the Default Pricing Profiles

Default Pricing Profile	Supported Pricing Type
Online Usage	USAGE_CHARGE
Offline Usage	USAGE_CHARGE
(For BRM) Standard Discount	DISCOUNT
Billing-time Discount	RECURRING_DISCOUNT
ChargeShare	CHARGE_SHARE
Convergent Usage	USAGE_CHARGE
(For ECE) Standard Discount	DISCOUNT
Standard Chargesharing	CHARGE_SHARE

About the PDC Target Engine Profiles

PDC uses the following target engine profiles:

> **Note:** During PDC installation, all the target engine profiles are loaded irrespective of whether ECE is supported or not.

- **BRM real-time rating:** Contains all the pricing capabilities (features) supported by the BRM real-time rating engine. See "BRM Real-Time Rating Target Engine Profile" for a list of the capabilities in this profile.
- **BRM batch rating:** Contains all the pricing capabilities supported by the BRM batch rating engine. See "BRM Batch Rating Target Engine Profile" for a list of the capabilities in this profile.
- **Subscription:** Contains capabilities required to create components supported by the BRM subscription engine, such as recurring charges, one-time charges, rollovers, folds, and bundles. See "Subscription Engine Profile" for a list of the capabilities in this profile.
- **ECE rating:** Contains all the pricing capabilities supported by ECE. See "ECE Rating Target Engine Profile" for a list of the capabilities in this profile.
- **ECE subscription:** Contains capabilities to support product offerings. See "ECE Subscription Engine Profile" for a list of the capabilities in this profile.

> **Note:** If a target engine capability is identified as mandatory in the following tables, it must be included in any pricing profile that is associated with the target engine profile.
>
> If two or more mandatory capabilities are mutually exclusive, you can remove all but one of them from a custom profile. If you include more than one, you must also include the exclusive constraint (**<exclusiveConstraint>**) between them. See the default pricing profiles for examples.

Note: You cannot customize a target engine profile.

BRM Real-Time Rating Target Engine Profile

The name of the BRM real-time rating target engine profile is RRE_RATING.

The pricing type of all the capabilities in this profile is USAGE_CHARGE (see "About Pricing Types").

Table 3–4 lists the capabilities contained in this target engine profile.

Table 3–4 Capabilities in the BRM Real-Time Rating Target Engine Profile

Capability	Description
RateplanSelector	Enables you to use a charge selector when configuring a charge (see "About Selectors"). Cannot be used with the EnhancedZoneModeling capability.
ChargeRateplan	(Mandatory) Enables you to configure charges for a charge offer (see "Adding Charges to Charge Offers").
SubscriberCurrency	(Mandatory) Enables you to specify a currency for a charge. **Multicurrency:** This subcapability enables you to select one or more currencies in the **Currency** field in the Create Charge dialog box (see "About Charge Currencies").
MultiRumGraph	Enables you to select one or more RUMs in the **Measured By** field in the Create Charge dialog box (see "About Measuring Events").
AbsoluteRelativeDateTimeRange	Enables users to specify absolute and relative date ranges when configuring a charge (see "About Charge Date Ranges").

Table 3–4 (Cont.) Capabilities in the BRM Real-Time Rating Target Engine Profile

Capability	Description
EnhancedZoneModeling	Enables you to use zone models and Usage Scenario (USC) selectors when configuring a charge (see "About Zone Models"). **USCSelector:** (Mandatory) This subcapability enables you to use USC selectors when configuring a charge (see "About Selectors"). Cannot be used with the RateplanSelector capability.
TimeModeling	Enables you to use time models when configuring a usage charge (see "About Time Models"). The following subcapabilities specify which attributes of a time model can be used: ■ **MonthOfYear** ■ **DayOfWeek** ■ **TimeOfDay** ■ **DayOfMonth**
UsageBI	(Mandatory if PriceSelector is not included) Enables you to configure one or more balance impacts for a usage charge (see "Configuring Pricing in Charges"). **CreditLimitCheck:** This subcapability enables you to specify whether to continue rating the remaining quantity after the credit limit is exceeded or to search for a charge that impacts a different balance element whose credit limit has not been exceeded and use that charge instead (see "Applying Credit Limits to Balance Elements"). **StepsBasedOnUsageQuantity:** This subcapability enables you to create price steps based on event quantity. **StepsBasedOnBalanceElement:** This subcapability enables you to create price steps based on balance element quantity (see "About Quantity Ranges in Pricing"). **SingleRum:** This subcapability enables you to select only one ratable usage metric (RUM) in the **Measured By** field in the Create Charge dialog box (see "About Measuring Events"). Cannot be used with the PriceSelector capability.

BRM Batch Rating Target Engine Profile

The name of the BRM batch rating target engine profile is BRE_RATING.

This profile contains capabilities for the following pricing types:

- USAGE_CHARGE (see Table 3–5, " Charging Capabilities in the BRM Batch Rating Target Engine Profile")
- DISCOUNT and RECURRING_DISCOUNT (see Table 3–6, " Discounting Capabilities in the BRM Batch Rating Target Engine Profile")
- CHARGE_SHARE (see Table 3–7, " Chargesharing Capabilities in the BRM Batch Rating Target Engine Profile")

See "About Pricing Types" for more information.

Table 3–5 lists the charging capabilities contained in this target engine profile:

Table 3–5 Charging Capabilities in the BRM Batch Rating Target Engine Profile

Capability	Description
ChargeRateplan	(Mandatory) Enables you to configure charges for a charge offer (see "Adding Charges to Charge Offers"). **MultiRum:** This subcapability enables you to select one or more RUMs in the **Measured By** field in the Create Charge dialog box (see "About Measuring Events").
SubscriberCurrency	(Mandatory) Enables you to specify the currency to use in the charge. **SingleCurrency** This subcapability enables you to select only one currency in the **Currency** field in the Create Charge dialog box (see "About Charge Currencies").
AbsoluteDateRange	Enables you to specify only absolute (not relative) date ranges when configuring a charge (see "About Charge Date Ranges").
ZoneModeling	Enables you to use zone models when configuring a charge (see "About Zone Models"). Cannot be used with the EnhancedZoneModeling capability.
EnhancedZoneModeling	Enables you to use zone models, USC selectors, and Access Point Name (APN) selectors when configuring a charge (see "About Zone Models"). **USCSelector:** This subcapability enables you to use a USC selector when configuring a charge (see "About Selectors"). **APNSelector:** This subcapability enables you to use an APN selector when configuring a charge (see "About Selectors"). Cannot be used with the ZoneModeling capability.
TimeModeling	Enables you to use time models when configuring a usage charge (see "About Time Models"). The following subcapabilities specify which attributes of a time model can be used: ■ **Holiday:** Special day calendars ■ **DayOfWeek:** Days of the week ■ **TimeOfDay:** Time of day ■ **UsesEffectivity:** Effective periods

Table 3–5 (Cont.) Charging Capabilities in the BRM Batch Rating Target Engine Profile

Capability	Description
PriceSelector	(Mandatory if UsageBI or PriceOverrideBI is not included) Enables you to use price selectors based on event, service, or account attributes when configuring a charge (see "About Selectors"). **AttributeBased:** This subcapability provides support for results based on specified event, service, or account attributes or a combination of such attribute types. Cannot be used with the UsageBI or PriceOverride capabilities.
UsageBI	(Mandatory if PriceSelector or PriceOverrideBI is not included) Enables you to configure one or more balance impacts for a usage charge (see "Configuring Pricing in Charges"). **MultiRum:** This subcapability enables you to select one or more RUMs in the **Measured By** field in the Create Charge dialog box (see "About Measuring Events"). **StepsBasedOnUsageQuantity:** This subcapability enables you to create price steps based on event quantity (see "About Quantity Ranges in Pricing"). **MinimumCharge:** This subcapability enables you to set a minimum charge for each balance impact (see "About Minimum Charges"). **ChargingIncrement:** This subcapability enables you to specify a charging increment (see "About Measuring Events"). **CreditWithoutValidity:** This subcapability enables you to specify a credit balance impact without using validity periods (see "About the Pricing Table"). **PriceTierUsesEffectivity:** This subcapability enables you to set effective dates for the prices in a charge (see "About Effective Periods for Pricing"). Cannot be used with the PriceSelector or PriceOverrideBI capabilities.
PriceOverrideBI	(Mandatory if PriceSelector or UsageBI is not included) Enables you to process prerated events by adding to or replacing the existing charge (see "About Charge Pricing"). Cannot be used with the PriceSelector and UsageBI capabilities.

Table 3–6 lists the discounting capabilities contained in this target engine profile:

Table 3–6 Discounting Capabilities in the BRM Batch Rating Target Engine Profile

Capability	Description
DiscountRateplanSelector	Enables you to use a discount selector when configuring a discount (see "About Selectors").
DiscountRateplan	(Mandatory) Enables you to configure discounts for a discount offer (see "About Discount Offers").
AbsoluteDateRange	(Mandatory) Enables you to specify absolute date ranges when configuring a discount (see "About Discount Date Ranges").

Table 3–6 (Cont.) Discounting Capabilities in the BRM Batch Rating Target Engine Profile

Capability	Description
DiscountOrChargeShareTrigger	Enables you to specify conditions that must be met before a discount can be applied to a charge (see "About Discount Triggers").
DiscountOrChargeShareFilter	Enables you to specify criteria that a charge must satisfy to be eligible for a discount (see "About Discount Filters"). This capability is included in the Standard Discount pricing profile but not in the Billing-time Discount pricing profile. **EventBasedChargeSelectorSpec:** This subcapability supports the use of event attributes to determine whether a charge qualifies for a particular discount.
DiscountOrChargeShareBI	(Mandatory) Enables you to configure one or more balance impacts for a discount (see "About Configuring Discount Balance Impacts").

Table 3–7 lists the chargesharing capabilities contained in this target engine profile:

Table 3–7 Chargesharing Capabilities in the BRM Batch Rating Target Engine Profile

Capability	Description
ChargeShareRateplan	(Mandatory) Enables you to configure chargeshares for a chargeshare offer (see "About Chargeshare Offers").
AbsoluteDateRange	(Mandatory) Enables you to specify absolute date ranges when configuring a chargeshare.
DiscountOrChargeShareTrigger	Enables you to specify conditions that must be met before a chargeshare can be applied to a charge.
DiscountOrChargeShareFilter	Enables you to specify criteria that a charge must satisfy to be eligible for a chargeshare. **EventBasedChargeSelectorSpec:** This subcapability supports the use of event attributes to determine whether a charge qualifies for a particular chargeshare.
DiscountOrChargeShareBI	(Mandatory) Enables you to configure one or more balance impacts for a chargeshare.

Subscription Engine Profile

The name of the subscription engine profile is RRE_SUBSCRIPTION.

This profile contains capabilities for the PRODUCT_OFFERING pricing type (see Table 3–8).

See "About Pricing Types" for more information.

Table 3–8 lists the product offering capabilities contained in this engine profile:

Table 3–8 Product Offering Capabilities in the Subscription Engine Profile

Capability	Description
ChargeOffering	Enables you to configure a charge offer (see "About Charge Offers").
AlterationOffering	Enables you to configure a discount offer (see "About Discount Offers").
DistributionOffering	Enables you to configure a chargeshare offer (see "About Chargeshare Offers").

ECE Rating Target Engine Profile

The name of the ECE rating target engine profile is ECE_RATING.

This profile contains capabilities for the following pricing types:

- USAGE_CHARGE (see Table 3–9, " Charging Capabilities in the ECE Rating Target Engine Profile")
- DISCOUNT (see Table 3–10, " Discounting Capabilities in the ECE Rating Target Engine Profile")
- CHARGE_SHARE (see Table 3–11, " Chargesharing Capabilities in the ECE Target Engine Profile")

See "About Pricing Types" for more information.

Table 3–9 lists the charging capabilities contained in this target engine profile:

Table 3–9 Charging Capabilities in the ECE Rating Target Engine Profile

Capability	Description
ChargeRateplan	Enables you to configure charges for a charge offer (see "Adding Charges to Charge Offers").
SubscriberCurrency	(Mandatory) Enables you to specify the currency to use in the charge. **SingleCurrency:** This subcapability enables you to select only one currency in the **Currency** field in the Create Charge dialog box (see "About Charge Currencies").
MultiRumGraph	Enables you to select one or more RUMs in the **Measured By** field in the Create Charge dialog box (see "About Measuring Events").
AbsoluteRelativeDateTimeRange	Enables users to specify absolute and relative date ranges when configuring a charge (see "About Charge Date Ranges").
ZoneModeling	Enables you to use zone models when configuring a charge (see "About Zone Models").
GenericSelector	Enables you to use generic selectors when configuring a charge (see "About Selectors").

Table 3–9 (Cont.) Charging Capabilities in the ECE Rating Target Engine Profile

Capability	Description
TimeModeling	Enables you to use time models when configuring a usage charge (see "About Time Models"). The following subcapabilities specify which attributes of a time model can be used: ▪ **Holiday:** Special day calendars ▪ **MonthOfYear:** Months of the year ▪ **DayOfWeek:** Days of the week ▪ **TimeOfDay:** Time of day ▪ **DayOfMonth:** Days of month
UsageBI	(Mandatory) Enables you to configure one or more balance impacts for a usage charge (see "Configuring Pricing in Charges"). **SingleRum:** This subcapability enables you to select only one RUM in the **Measured By** field in the Create Charge dialog box (see "About Measuring Events"). **MinimumCharge:** This subcapability enables you to set a minimum charge for each balance impact (see "About Minimum Charges"). **StepsBasedOnBalanceElement:** This subcapability enables you to create price steps based on balance element quantity (see "About Quantity Ranges in Pricing"). **StepsBasedOnUsageQuantity:** This subcapability enables you to create price steps based on event quantity (see "About Quantity Ranges in Pricing"). **StepsBasedOnBalanceElementWithThreshold:** This subcapability enables you to create price steps based on balance element with threshold (see "Applying Credit Limits to Balance Elements"). **StepsBasedOnUsageQuantityWithThreshold:** This subcapability enables you to create price steps based on event quantity with threshold. **PriceTierUsesEffectivity:** This subcapability enables you to set effective dates for the prices in a charge (see "About Effective Periods for Pricing").
ConditionalBI	Enables you to configure conditional balance impacts for a usage charge (see "Configuring Pricing in Charges"). **SingleRum:** This subcapability enables you to select only one RUM in the **Measured By** field in the Create Charge dialog box (see "About Measuring Events"). **StepsBasedOnUsageQuantity:** This subcapability enables you to create price steps based on event quantity (see "About Quantity Ranges in Pricing"). **ConditionalImpact:** This subcapability enables you to specify when the balance impact should be applied (see "About Conditional Balance Impacts").

Table 3–10 lists the discounting capabilities contained in this target engine profile:

Table 3–10 Discounting Capabilities in the ECE Rating Target Engine Profile

Capability	Description
DiscountRateplan	(Mandatory) Enables you to configure discounts for a discount offer (see "About Discount Offers").
AbsoluteDateRange	(Mandatory) Enables you to specify absolute date ranges when configuring a discount (see "About Discount Date Ranges").
DiscountOrChargeShareTrigger	Enables you to specify conditions that must be met before a discount can be applied to a charge (see "About Discount Triggers").
DiscountOrChargeShareFilter	Enables you to specify criteria that a charge must satisfy to be eligible for a discount (see "About Discount Filters"). This capability is included in the Standard Discount pricing profile but not in the Billing-time Discount pricing profile. **EventBasedChargeSelectorSpec:** This subcapability supports the use of event attributes to determine whether a charge qualifies for a particular discount.
DiscountOrChargeShareBI	(Mandatory) Enables you to configure one or more balance impacts for a discount (see "About Configuring Discount Balance Impacts").

Table 3–11 lists the chargesharing capabilities contained in this target engine profile:

Table 3–11 Chargesharing Capabilities in the ECE Target Engine Profile

Capability	Description
ChargeShareRateplan	(Mandatory) Enables you to configure chargeshares for a chargeshare offer (see "About Chargeshare Offers").
AbsoluteDateRange	(Mandatory) Enables you to specify absolute date ranges when configuring a chargeshare (see "About Discount Date Ranges").
DiscountOrChargeShareTrigger	Enables you to specify conditions that must be met before a chargeshare can be applied to a charge (see "About Discount Triggers").
DiscountOrChargeShareFilter	Enables you to specify criteria that a charge must satisfy to be eligible for a chargeshare (see "About Discount Filters"). **EventBasedChargeSelectorSpec:** This subcapability supports the use of event attributes to determine whether a charge qualifies for a particular chargeshare.
DiscountOrChargeShareBI	(Mandatory) Enables you to configure one or more balance impacts for a chargeshare (see "About Configuring Discount Balance Impacts").

ECE Subscription Engine Profile

The name of the ECE subscription engine profile is ECE_SUBSCRIPTION.

This profile contains capabilities for the PRODUCT_OFFERING pricing type (see Table 3–12).

See "About Pricing Types" for more information.

Table 3–12 lists the product offering capabilities in this engine profile:

Table 3–12 Product Offering Capabilities in the ECE Subscription Engine Profile

Capability	Description
ChargeOffering	Enables you to configure a charge offer (see "About Charge Offers").
DiscountOffering	Enables you to configure a discount offer (see "About Discount Offers").
ChargeShareOffering	Enables you to configure a chargeshare offer (see "About Chargeshare Offers").

Target Engine Profiles for the Default Pricing Profiles

The pricing capabilities (features) in a pricing profile are based on the capabilities in the target engine profiles with which it is associated. A pricing profile can be associated with one or more target engine profiles (see "About Associating Custom Pricing Profiles with Target Engine Profiles").

Table 3–13 shows the default target engine profile or profiles for each default pricing profile:

Table 3–13 Target Engine Profiles Associated with the Default Pricing Profiles

Default Pricing Profile	Default Target Engine Profile	Target Engine Profile Pricing Capabilities
Online Usage	RRE_RATING	See Table 3–4, " Capabilities in the BRM Real-Time Rating Target Engine Profile".
Offline Usage	BRE_RATING	See Table 3–5, " Charging Capabilities in the BRM Batch Rating Target Engine Profile".
Standard Discount	BRE_RATING	See Table 3–6, " Discounting Capabilities in the BRM Batch Rating Target Engine Profile".
Billing-time Discount	BRE_RATING	See Table 3–6, " Discounting Capabilities in the BRM Batch Rating Target Engine Profile".
ChargeShare	BRE_RATING	See Table 3–7, " Chargesharing Capabilities in the BRM Batch Rating Target Engine Profile".
Convergent Usage	ECE_RATING	See Table 3–9, " Charging Capabilities in the ECE Rating Target Engine Profile".
Standard Discount	ECE_RATING	See Table 3–10, " Discounting Capabilities in the ECE Rating Target Engine Profile".
Standard Chargesharing	ECE_RATING	See Table 3–11, " Chargesharing Capabilities in the ECE Target Engine Profile".

About Associating Custom Pricing Profiles with Target Engine Profiles

Before you can publish pricing components that use a custom pricing profile to a rating engine, you must associate the pricing profile with at least one target engine profile. The associated target engine profiles determine which rating or subscription engines the components are published to.

The target engines must support all the capabilities in the pricing profile. See the tables in "About the PDC Target Engine Profiles" for the capabilities supported by each PDC target engine profile.

Note: When a component that uses a pricing profile is submitted for publication, PDC validates the pricing profile against its associated target engine profiles to ensure that the capabilities in the pricing profile are supported by the target engines.

Important: PDC does not support a common pricing profile. Hence, do not create a custom pricing profile with capabilities supported by RRE_RATING and ECE_RATING or BRE_RATING and ECE_RATING profiles.

If a pricing profile is not associated with a target engine profile, any pricing components that use that profile are not published to a target engine.

See "Target Engine Profiles for the Default Pricing Profiles" for information about the target engine profiles associated with the default pricing profiles.

See "Associating Custom Pricing Profiles with Target Engine Profiles" for instructions on how to associate pricing profiles with target engine profiles.

About Importing Custom Pricing Profiles into PDC

You define a custom pricing profile in an XML file and use the **ImportExportPricing** utility to import the XML file into PDC. The XML file containing the custom pricing profile must conform to the format specified in the **profiles.xsd** file in *PDC_home*/**apps/xsd**.

See "Prerequisites for Using the ImportExportPricing Utility" for information about prerequisites for using the **ImportExportPricing** utility.

Setting Up Custom Pricing Profiles

To set up a custom pricing profile in PDC:

1. Create a custom pricing profile (see "Creating Custom Pricing Profiles").
2. Associate the profile with a target engine profile (see "Associating Custom Pricing Profiles with Target Engine Profiles").
3. Import the profile into PDC (see "Importing Custom Pricing Profiles into PDC").

Creating Custom Pricing Profiles

To create a custom pricing profile:

1. Make a copy of the default pricing profile to customize.

   See "About Default Pricing Profiles" for descriptions and locations of the default pricing profiles.
2. Open the copy in a text or XML editor.
3. In the copy, modify the pricing profile name to distinguish the copy from the original:

```
<pricingProfile>
   <name>CustomPricingProfileName</name>
   . . .
```

where *CustomPricingProfileName* is the new name for the copy.

4. Add or delete pricing capabilities as necessary.

   Note the following:

   - If only one subcapability exists for a particular capability in the target engine profile, the pricing profile must include that subcapability.
   - If multiple subcapabilities exist for a particular capability in a target engine profile, at least one of the subcapabilities must be included in the pricing profile. The custom pricing profile does not, however, have to use the same subcapability as the default pricing profile.

   See "About Customizing Pricing Profiles" for more information.

5. Save your changes.

Associating Custom Pricing Profiles with Target Engine Profiles

To associate a custom pricing profile with a target engine profile:

1. Open the file containing the custom pricing profile in a text or XML editor.
2. Add the following entry to the **<profiles>** element:

```
<profileUsedBy>
   <pricingProfileName>CustomPricingProfile</pricingProfileName>
   <targetEngineProfileName>TargetEngineProfile1</targetEngineProfileName>
   <targetEngineProfileName>TargetEngineProfile2</targetEngineProfileName>
   <priceListName>Default</priceListName>
</profileUsedBy>
```

   where:

   - *CustomPricingProfile* is the name of the pricing profile that you are associating with the target engine profile.
   - *TargetEngineProfile#* is the name of the target engine profile that you are associating with the custom pricing profile.

   Note: For each additional target engine that you want to associate with the pricing profile, add another **<targetEngineProfileName>** entry to the **<profileUsedBy>** element.

3. Save and close the file.

See "About Associating Custom Pricing Profiles with Target Engine Profiles" for more information.

Importing Custom Pricing Profiles into PDC

To import a custom pricing profile into PDC, you must have the **Pricing Design Admin** role.

To import a custom pricing profile into PDC:

1. Verify that the custom pricing profile XML file is complete and follows the guidelines specified in the **profiles.xsd** file in the *PDC_home*/**apps/xsd** directory.

2. Go to the *PDC_home*/**apps/Samples/Examples** directory, and enter the following command:

   UNIX

   ```
   ImportExportPricing -import -profile ProfileFileName
   ```

   where *ProfileFileName* specifies the full path and file name of the XML file containing the custom profile data.

   > **Note:** To run this command, you must have *PDC_home*/**apps/bin** in your PATH environment variable.

For example, the following command imports a custom pricing profile from the **MyPricingProfile.xml** file into the PDC database in a UNIX environment:

```
ImportExportPricing -import -profile /MyPricingProfile.xml
```

4

About Configuring Setup Components

This chapter describes how to configure the setup components that are required to create a product offering in Oracle Communications Pricing Design Center (PDC).

You should have a basic understanding of the following:

- PDC: See "About Pricing Design Center" for more information.
- Product offerings: See "About Creating Product Offerings" for more information.

To configure setup components, you must also understand the following:

- Extensible markup language (XML) programming
- XML schema definition (XSD)

About Setup Components

Setup components are the prerequisite data that you configure before creating a product offering. For example, before creating a product offering, you must configure the services that your customers can purchase.

Some setup components are defined in PDC and some components are mastered in Oracle Communications Billing and Revenue Management (BRM) and the rating system and need to be loaded into PDC.

The following setup components are defined in PDC. Any modifications to these components are also handled by PDC and are transformed to BRM and rating system.

- **Ratable Usage Metrics (RUMs)**: A RUM specifies the units to measure events and how to calculate the measurement. You can base a measurement on any data captured in an event, such as how long a session lasted or the number of bytes downloaded during a session.
- **Service-event maps**: The service-event map lists all the services to which charge offers and discount offers can apply. For each service, the map specifies which events can have charges configured for that service. The map also specifies the RUMs to use for each service-event combination.
- **Balance elements**: A balance element represents a currency or noncurrency asset of economic value, such as U.S. dollars. When you configure pricing in a charge offer, you specify the balance elements that are debited or credited when the charge offer is used to rate an event. For example, a charge of one dollar per minute for a phone call affects the US Dollars balance element. You can add multiple currency balance elements for subscriber currencies under a single pricing tier by using the ImportExportPricing utility.

  See "Enabling Single Currency Graph" for more information.

- **Impact categories**: Impact categories enable you to use the same charge to apply to different balance impacts based on various event attributes. For example, to configure different impacts for calls made to different countries, you create impact categories for each country. When a call occurs, the impact category determines which pricing to apply.
- **Zone models**: A zone model is a set of zone rules that associate the origin and destination of a call to an impact category.

  You can configure the following types of zone models:

  – **Standard Zone Model:** Associates the origin and destination area codes of a call to an impact category.

  – **Geographical Zone Model:** Associates the distance between the origin and destination area codes of a call to an impact category.
- **Value maps**: A value map is a hierarchical structure of zones and corresponding values. Each zone contains a group of values that apply to a single event attribute, such as an area code, an IP address, or a broadband bit rate.
- **Special day calendars**: A special day calendar is a set of dates, such as holidays, for which you want to charge special prices for your services. These dates are used in a time model to define a time period. A time period can be used in a charge offer to determine a price.
- **Custom rules**: A custom rule is a named expression that is used in a generic selector rule to guide to a price.
- **Profile attribute specifications**: A profile attribute specification specifies the name of a profile that is used to assign special rating attributes to a subscriber. A profile attribute specification can be associated to one or more services or to an account. Profile attribute specifications are referenced in custom rules.
- **Item type selector**: An item type selector contains rules and item specifications for assigning balance impacts to bill items. These rules and specifications enable you to assign different bill items for balance impacts within the same event. For example, you can define separate bill items for international and national calls and apply a different tax to each bill item at billing time.

  Note: Item type selectors are designed to be used only with Oracle Communications Billing and Revenue Management (BRM) Elastic Charging Engine (ECE).

- **Policy Specification**: A policy specification is made up of one or more counter policies each of which defines a gradation in the QoS based on the subscriber's service usage. For example, you can have a policy specification called *Platinum* for a data service and a balance element for it as *Megabytes Used*. You can define a counter policy labeled *Fair Usage*, which has three levels, *Low QoS*, *Medium QoS*, and *High QoS*, with each level containing a usage range valid for that quality of service, such as *100-150*, *150-200*, and *200-No Maximum* megabytes.

> **Important:** If you are using ECE for usage charging, ensure that you migrate the existing offer profile data from BRM into PDC and use only PDC for configuring policy specifications. After you migrate the offer profile data into PDC, do not create any offer profiles in BRM.
>
> For more information on migrating the offer profile data, see *PDC Installation and System Administration Guide*.

The following setup components are mastered in BRM and the rating system. Any modifications to these components are also handled by BRM and the rating system and must be synchronized with PDC.

- **Service definitions**: A set of attributes pertaining to a type of service. Service definitions are used for provisioning a service or for determining a price. For example, the service definition can be used to hold the user identity or the quality of a service.
- **Event definitions**: A set of attributes pertaining to a type of transaction. Event definitions are used in charging and billing. For example, you can use the time the call was made, the origin and destination of the call, or the duration of the call to determine how to charge for the event.
- **Account definition**: A set of attributes that define the characteristic of a customer account, such as location, type, and age.
- **General ledger (G/L) IDs**: A G/L ID is an ID assigned to the balance impact. The G/L ID ensures that the balance impact is reported to the correct account in your company's G/L. For example, you keep a record of revenue from usage fees by mapping usage balance impacts to G/L IDs.
- **Provisioning tag names**: In BRM, provisioning tags implement extended rating attributes (ERAs), which provide special charges or discounts based on a specific attribute of a service or an account, such as a telephone number. For example, a provisioning tag could specify that a charge offer uses a discount for calls to friends and family on a subscriber's birthday.
- **Tax codes**: A tax code is a code that identifies a type of charge offer, such as a subscription or a physical commodity. You map tax codes to taxation packages that calculate the tax on the charge offers or to custom tax rates that you define. A tax code indicates which tax to apply based on the PDC charge offer.
- **Tax suppliers**: A tax supplier is a company or company division responsible for collecting taxes. Taxes can be calculated differently depending on the location of the tax supplier.
- **Business Profiles**: A business profile defines the requirements and key-value pairs that are used by BRM to determine how the bill units are billed. This enables you to aggregate balance impacts in the bill items based on how the bill items are used. For example, business profiles can be used to classify bills as prepaid or postpaid to determine how bills and payments are handled.

See "About Creating Product Offerings" for a description of each of the setup components.

Configuring Setup Components

You configure the setup components in PDC by using the following applications:

- **PDC UI**: Use to create and modify setup components using a graphical user interface. See the PDC Help for more information.
- **ImportExportPricing utility**: Use to import setup components defined in an XML file into PDC. See "Importing and Exporting Pricing and Setup Components" for more information.
- **SyncPDC utility**: Use to synchronize setup components defined in BRM and the rating system with PDC. See "Synchronizing Setup Components" for more information.

Table 4–1 lists the setup components, the order in which you should configure the data, whether the setup component is required or optional, and the application that you use to configure them.

Note: Some of the pricing setup data is optional. You must configure the optional pricing setup data only if you plan to use the related feature or perform certain business functions. For example, you configure a zone model only if you use zoning to determine a price in your charge offer.

Table 4–1 Setup Components and the Application You Use to Configure Them

Pricing Setup Data	Required/Optional	Application
RUMs	Required	PDC UI or **ImportExportPricing** utility
Service-event maps	Required	PDC UI or **ImportExportPricing** utility
Impact categories	Optional	PDC UI or **ImportExportPricing** utility
Zone models	Optional	PDC UI or **ImportExportPricing** utility
Special day calendars	Optional	PDC UI or **ImportExportPricing** utility
Balance elements	Required	PDC UI or **ImportExportPricing** utility
Services	Required	**SyncPDC** utility
Events	Required	**SyncPDC** utility
Account attributes	Required	**SyncPDC** utility
Tax codes	Optional	**SyncPDC** utility
Tax suppliers	Optional	**SyncPDC** utility
G/L IDs	Required for some target engines	**SyncPDC** utility
Provisioning tag names	Optional	**SyncPDC** utility
Value maps	Optional	**ImportExportPricing** utility
Custom rules	Optional	**ImportExportPricing** utility
Profile attribute specifications	Optional	**ImportExportPricing** utility
Item type selector	Required for ECE	**ImportExportPricing** utility
Business Profile	Required for ECE	**SyncPDC** utility
Policy Specifications	Optional	PDC UI or **ImportExportPricing** utility

Sample XML files for each setup component are in the *PDC_home*/**apps/Samples/Examples** directory, where *PDC_home* is the directory in which the PDC software is installed.

The **Examples** directory contains the following sample and ready-to-use files:

- **PDCSampleBE.xml.** Contains all currency balance elements and some frequently used non-currency balance elements. You can load this file to quickly create balance elements to use when configuring pricing.
- **PDCSampleRum.xml.** Contains commonly used ratable usage metrics (RUMs). You can load this file to quickly create RUMs to use when creating a service-event map and when configuring pricing.
- **OOB_ProfileSpecifications.xml.** Contains profile attribute specifications referenced in the following custom rules: **Friends&Family**, **SpecialDay**, and **ClosedUserGroup**. This XML file references the names used in the provisioning tags provided by default with the BRM installation.

  Note: You must load the **OOB_ProfileSpecifications.xml** file before loading the **OOB_CRs.xml** file.

- **OOB_CRs.xml.** Contains the custom rules: **Friends&Family**, **SpecialDay**, and **ClosedUserGroup**. You can use these custom rules in a generic selector when configuring a charge based on a subscriber's Friends & Family, Special Day, or Closed User Group profile.
- **Sample_ServiceCUG_ProfileSpecification.xml.** Contains the profile attribute specification for configuring a service-level closed user group. See "About Configuring Closed User Groups" for more information.
- **Sample_ServiceCUG_CR.xml.** Contains the custom rule for configuring a service-level closed user group. See "About Configuring Closed User Groups" for more information.
- **OOB_ItemTypeSelector.xml.** Contains the default billing item assignments that are supported by BRM without any additional configuration. You must load this file if you are using ECE. See "Loading the OOB_ItemTypeSelector.XML File" for more information.
- **Sample_ItemTypeSelector.xml.** Contains the item type selector rules for defining separate bill items for international and national calls. You can use this as a template to quickly create the item type selector for tracking international calls and national calls separately. See "Configuring Item Type Selectors" for more information.
- **SamplePolicyLabel.xml.** Contains policy labels referenced in the policy specifications. You can use this as a template to add and modify the elements as required. See "Configuring Policy Specifications".
- **SamplePolicySpecification.xml.** Contains policy specifications referenced in the charge offers and discounts offers used for policy-driven charging. You can use this as a template to add and modify the elements as required. See "Configuring Policy Specifications".

You can load the updated sample XML files into PDC by using the **ImportExportPricing** utility. See "ImportExportPricing" for the utility's syntax and parameter descriptions.

Note: The **Examples** directory also contains sample alias files and sample pricing profile files that are provided only for reference and for use as templates for creating custom alias events or pricing profiles. You should not load the sample alias files and sample pricing profile files after installing PDC.

About Configuring Service and Event Definitions

BRM provides a preconfigured set of service and event definitions. When you run the **SyncPDC** utility, the preconfigured service and event definitions are stored in PDC. For the list of preconfigured events, see the *BRM_Home*/**sys/data/pricing/example/pin_event_map** file, where *BRM_Home* is the directory in which you installed BRM.

Note: Before you run the **SyncPDC** utility for the first time, ensure that only those services and events that are relevant to your business are available in the **pin_event_map** file.

If you are using ECE for usage rating, the preconfigured event definitions are automatically enriched with required information for ECE when you run the **SyncPDC** utility. You can use these definitions for pricing and charging without any further enrichment.

Enriching includes adding user-defined charging attributes, defining charge operation types, and mapping event attributes to the external network format. For example, you add specific attributes that ECE uses to apply session-based charge for a transaction.

You can also create your own service and event definitions:

- To offer new services and events
- To add additional attributes for provisioning, networking, charging, or billing

You can define the custom services and events in Development Center and use the **SyncPDC** utility to synchronize the definitions with PDC. If you are defining a new service or event, ensure that you do the following:

1. Create a subclass for the new service or event in Development Center. See the discussion about creating storable classes for services and events in *BRM Developer's Guide* for more information.
2. Create a substruct in the event subclass; for example, USAGE_INFO, and add USER_IDENTITY and CALLED_ID fields in Development Center. See the discussion about substructure and creating custom fields in *BRM Developer's Guide* for more information.
3. Map the event to the corresponding service by editing the **pin_event_map** file and then running the **load_event_map** utility. See the discussion about mapping event types to services in *BRM Setting Up Pricing and Rating* for more information.
4. Synchronize the service and event definitions with PDC by running the **SyncPDC** utility. See "Synchronizing Setup Components" for more information.

   The service and event definitions are loaded into PDC.
5. If you are using ECE for usage rating, enrich the event definitions with the information ECE requires for processing usage requests. See "About Enriching Event Definitions"for more information.

   The enriched event definition is published to ECE.

6. If you are using ECE for usage rating and you want to use any service or account attributes for usage rating, ensure that only those service or account attributes are loaded into the ECE cache. See "Using Service and Account Attributes with ECE for Usage Rating" for more information.

Using Service and Account Attributes with ECE for Usage Rating

If you are using ECE for usage rating and you want to use any service or account attributes for usage rating, ensure that only those service or account attributes are loaded into the ECE cache by doing the following:

1. Export the service or account definition into an XML file by running the following command:

   ```
   ImportExportPricing -export -metadata -n PRODUCT_ATTRIBUTE_SPEC|CUSTOMER_
   ATTRIBUTE_SPEC "ObjectName"
   ```

   where *ObjectName* is the name of the service or account to be exported into an XML file.

   For example:

   ```
   ImportExportPricing -export -metadata PRODUCT_ATTRIBUTE_SPEC -n TelcoGSM
   ```

   Exports the definition for TelcoGSM to the **productattr_export_config.xml** file.

2. Open the XML file in a text editor.

3. Search for the attributes that you want to use for usage rating and add the following entries for each attribute:

   ```
   <attributeItemSupportedBy>
      <targetApplicationSpecName>Pricing</targetApplicationSpecName>
      <targetApplicationSpecName>Convergent Charging</targetApplicationSpecName>
   </attributeItemSupportedBy>
   ```

4. Save and close the file.

5. Import the XML file into PDC by running the following command:

   ```
   ImportExportPricing -import -config FileNamePrefix_export_config.xml-ow
   ```

   For example:

   ```
   ImportExportPricing -import -config productattr_export_config.xml
   ```

   The service or account definition is loaded into PDC. The ECE Pricing Updater publishes only the service or account attributes that you want to use for usage rating to ECE and only those attributes are loaded into the ECE cache.

About Configuring Closed User Groups

You can configure closed user groups for applying special prices for events between members of a closed user group. For example, you can configure a closed user group for applying special rates on calls between members of that group.

To identify the members of a closed user group and apply special prices, you configure a custom rule and add it to a generic selector. The custom rule evaluates the groups to which the originating and terminating subscribers belong. If the originating and terminating subscribers belong to even one common closed user group, the custom rule evaluates as true. The generic selector guides to the price based on how the custom rule is evaluated at runtime.

You can configure closed user groups at the account level or at the service level. To configure closed user groups at the account level, use the **OOB_ ProfileSpecifications.xml** and **OOB_CRs.xml** files as templates. To configure closed users groups at the service level, use the **Sample_ServiceCUG_ ProfileSpecification.xml** and **Sample_ServiceCUG_CR.xml** files as templates. See Table 4–2, " Elements in the Custom Rules XML File" for the usage and description of each element.

If you configure a closed user group at the account level, associate the closed user group names with the appropriate service identifiers for the account in BRM Customer Center when the customer purchases the charge offer with a closed user group. For example, map the phone numbers to the appropriate closed user group names.

If you configure a closed user group at the service level, configure only the appropriate closed user group names in BRM Customer Center when the customer purchases the charge offer with a closed user group. See the discussion about configuring a closed user group promotion in the BRM Customer Center Help.

Table 4–2 lists the elements in the custom rules XML file, the usage of each element, and a description of how to specify each element in the XML file.

Table 4–2 Elements in the Custom Rules XML File

Element	Syntax	Description
customAnalyzerRules	`<customAnalyzerRules>`	The root element of the rule.
name	`<name>`*`RuleName`*`</name>`	*RuleName* specifies the custom rule name.
description	`<description>`*`RuleDescription`*`</description>`	*RuleDescription* contains the description of the custom rule.
priceListName	`<priceListName>`*`PriceListName`*`</priceListName>`	*PriceListName* specifies the price list name. The valid value is **Default**.
profileSpecName	`<profileSpecName>`*`ProfileAttributeSpecificationName`*`</profileSpecName>`	*ProfileAttributeSpecificationName* specifies the name of the profile attribute specification used for configuring the rule.

Table 4–2 (Cont.) Elements in the Custom Rules XML File

Element	Syntax	Description

customRuleExpression	`<customRuleExpression>`
	`<dynamicFieldCRExpression>`
	`<operator>Operator</operator>`
	`<FieldIdentifier>`
	`...`
	`</FieldIdentifier>`
	`</dynamicFieldCRExpression>`
	`</customRuleExpression>`
		Contains the expression to evaluate closed user groups, where *Operator* specifies the operator to be used in the expression. The valid value is **INTERSECT**.

FieldIdentifier	`<FieldIdentifier>`
	`<qualifier>Qualifier</qualifier`
	`<fieldSelector>`
	`...`
	`</fieldSelector>`
	`</FieldIdentifier>`
		Contains the details to identify the fields for evaluating closed user groups, where *Qualifier* specifies the qualifier for identifying the fields. The valid values are:
		▪ **ORGINATING**
		▪ **TERMINATING**
		In the **<FieldIdentifier>** element, you can define fields from the service, account, and event attribute specifications.

FieldSelector	`<fieldSelector>`
	`<operator>ConditionOperator</operator>`
	`<selectorFieldName>SelectorFieldName</selectorFieldName>`
	`<selectorFieldKind>SelectorFieldKind</selectorFieldKind>`
	`<conditionFieldName>ConditionFieldName</conditionFieldName>`
	`<conditionFieldKind>ConditionFieldKind</conditionFieldKind>`
	`</fieldSelector>`
		Identifies the closed user group names that match the given product identifier (PUID); for example, a phone number, for both the originating and terminating subscribers.
		Note: You can use the **<FieldSelector>** element for configuring account-level closed user groups only.
		where:
		▪ *ConditionOperator* specifies the operator to be used in the condition. The valid value is **EQUAL_TO**.
		You must specify the *ConditionOperator* if you want to define the conditions for selecting fields. If the *ConditionOperator* is null, you cannot specify *ConditionFieldName* and *ConditionFieldKind*.
		▪ *SelectorFieldName* specifies the selector field name. You must define the selector fields based on the condition fields.
		▪ *SelectorFieldKind* specifies the type of attributes used for field selection. You can specify any service, event, or account attribute.
		▪ *ConditionFieldName* specifies the condition field name.
		▪ *ConditionFieldKind* specifies the type of attributes used for selection. You can specify any service, event, or account attribute.

About Configuring Item Type Selectors

You configure item type selectors in PDC so that balance impacts can be tracked appropriately for different bill items at run time in ECE.

Note: Item type selectors can be used only with ECE.

By default, BRM tracks balances for the following bill items: cycle arrears items, cycle forward items, cycle forward arrears items, cycle tax items, cycle incentive items, and usage items. You can load the **OOB_ItemTypeSelector.xml** file into PDC to configure the item type selectors for these bill items in PDC. See "Loading the OOB_ ItemTypeSelector.XML File" for more information.

If you want to use custom bill items, you can configure item type selectors to create rules for ECE to assign balance impact to the custom bill items. You can use the **Sample_ItemTypeSelector.xml** file as a template to configure the item type selectors for custom bill items. See "Configuring Item Type Selectors" for more information.

Table 4–3 lists the elements that are supported in the item type selector XML file, the usage of each element, and a description of how to specify each element in the XML file.

Table 4–3 Elements Supported in the Item Type Selector XML File

Element	Syntax	Description
itemTypeSelectors	`<itemTypeSelectors>`	The root element of **ItemTypeSelector.xml**.
name	`<name>`*`ItemTypeSelectorName`*`</name>`	*ItemTypeSelectorName* specifies the item type selector name.
description	`<description>`*`ItemTypeSelectorDescription`*`</description>`	*ItemTypeSelectorDescription* contains the description of the item type selector.
priceListName	`<priceListName>`*`PriceListName`*`</priceListName>`	*PriceListName* specifies the price list name. The valid value is **Default**.
applicableToName	`<applicableToName>`*`ApplicableToName`*`</applicableToName>`	*ApplicableToName* specifies the service name or **Account**. The attributes of this service or account are used in the rules.
eventSpecName	`<eventSpecName>`*`EventName`*`</eventSpecName>`	*EventName* specifies the name of any event. The attributes of this event are used in the rules.
itemGroup	`<itemGroup>`*`ItemGroupName`*`</itemGroup>`	Specifies the item group to be used for aggregating the balance impacts; where *ItemGroupName* specifies the name of the item group. The **<itemGroup>** element is associated with the **<ItemTag>** element in BRM. To aggregate balance impacts based on the item configuration value in the business profile, specify the item configuration value as *ItemGroupName*. See the discussion about improving performance by using multiple item configurations in *BRM System Administrator's Guide* for more information on item configuration. **Note:** If you specify the item configuration value as *ItemGroupName*, Oracle recommends that you do not change that value in the business profile. The default value is **Default**.
applicableToAllBalImpact	`<applicableToAllBalImpact>`*`BalanceImpactsOption`*`</applicableToAllBalImpact>`	Specifies whether this item type selector is applicable to all balance impacts, where *BalanceImpactsOption* is: ▪ **true** if the item type selector applies for all the balance impacts ▪ **false** if the item type selector applies only to specific balance impacts

Table 4–3 (Cont.) Elements Supported in the Item Type Selector XML File

Element	Syntax	Description

rule	`<rule>`
	`<ruleName>`*`RuleName`*`</ruleName>`
	`<ruleOrder>`*`RuleOrder`*`</ruleOrder>`
	`<expression>`
	`Expression`
	`</expression>`
	`<specName>`*`ItemSpecificationName`*`</specName>`
	`</rule>`
		Contains the rules for assigning the balance impacts to the bill items, where:
		▪ *RuleName* specifies the rule name.
		▪ *RuleOrder* specifies the order in which the rules must be applied.
		▪ *Expression* contains the details to derive which items to assign balance impacts to.
		Note: You can add multiple rules in the item type selector. Each rule points to the item specification in the same item type selector.
		Each rule in the item type selector can refer to a different attribute. For example, Rule 1 can use an event attribute, Rule 2 can use an attribute of the charging result, and Rule 3 can use a service attribute.

Table 4–3 (Cont.) Elements Supported in the Item Type Selector XML File

Element	Syntax	Description

expression	`<expression>`
	`<separator>`*`Delimiter`*`</separator>`
	`<operation>`*`Operator`*`</operation>`
	`<fieldName>` *`FieldName`*`</fieldName>`
	`<fieldKind>`*`FieldKind`*`</fieldKind>`
	`<fieldValue>`*`FieldValue`*`</fieldValue>`
	`</expression>`
		Contains the details to derive which items to assign balance impacts to.
		▪ *Delimiter* specifies the character that is used to separate the field values.
		▪ *Operator* specifies the operator to be used in the expression. The valid values are:
		EQUAL_TO
		NOT_EQUAL_TO
		GREATER_THAN
		GREATER_THAN_EQUAL
		LESS_THAN
		LESS_THAN_EQUAL
		IN_LIST
		REGEX
		▪ *FieldName* specifies the field name. If you specify **CHARGING_RESULT_SPEC** as the *FieldKind*, the valid values are:
		TAX_CODE
		GLID
		RUM
		BALANCE_ELEMENT
		ZONE_RESULT
		TIMEMODEL_TAG_NAME
		GENERIC_SELECTOR_RESULT
		ORIG_ZONE_RESULT
		PRICING_NAME
		TIME_MODEL_NAME
		ZONE_MODEL_NAME
		EVALUATED_ZONE_MODEL_NAME
		EVALUATED_ZONE
		CHARGE_RATE_PLAN_NAME
		▪ *FieldKind* specifies the type of attributes to be selected for assigning the balance impacts. The valid values are:
		PRODUCT_SPEC_FIELD
		EVENT_SPEC_FIELD
		CUSTOMER_SPEC_FIELD
		CHARGING_RESULT_SPEC
		▪ *FieldValue* specifies the field value that is used for tracking balance impacts.

Table 4–3 (Cont.) Elements Supported in the Item Type Selector XML File

Element	Syntax	Description

itemSpec	`<itemSpec>`
	`<name>`*ItemSpecficationName*`</name>`
	`<description>`*ItemSpecificationDescription*`</description>`
	`<priceListName>`*PriceListName*`</priceListName>`
	`<type>`*ItemType*`</type>`
	`<default>`*DefaultOption*`</default>`
	`<aggregationType>`*AggregationType*`</aggregationType>`
	`<category>`*ChargeCategory*`</category>`
	`</itemSpec>`
		Contains the details about the item specification, where:
		▪ *ItemSpecficationName* specifies the item specification name. The item specification name is associated with the item tag field in BRM.
		▪ *ItemSpecificationDescription* contains the description of the item specification.
		▪ *PriceListName* specifies the price list name. The valid value is **Default**.
		▪ *ItemType* specifies the type of the bill item for assigning balance impacts. This must match the item type created in BRM.
		▪ *DefaultOption* specifies whether a item specification is the default specification for the bill item. The valid values are **true** and **false**.
		▪ *AggregationType* specifies whether the item accumulates charges or tracks each charge separately. The valid values are:
		CUMULATIVE_NONPRECREATE, which specifies that the bill item accumulates charges through out the billing cycle and the bill item need not be pre-created in BRM
		INDIVIDUAL_NONPRECREATE, which specifies that the bill item stores a charge for a single event and the bill item need not be pre-created in BRM
		CUMULATIVE_PRECREATE, which specifies that the bill item accumulates charges through out the billing cycle and the bill item needs to be pre-created in BRM
		▪ *Category* specifies the charge category. See "About Charge Categories" for more information.
		Note: You can add multiple item specifications, but you can add only one default item specification.
		You can use the same item specification in multiple item type selectors, but the item specification name must be unique.

Loading the OOB_ItemTypeSelector.XML File

To load the **OOB_ItemTypeSelector.xml** file:

1. If you have created custom bill items in BRM, make a backup copy of the customized **config_item_tags.xml** and **config_item_types.xml** files in BRM.
2. Go to the *PDC_home*/**apps/Samples/Examples** directory.
3. Import the default billing item assignments from the **OOB_ItemTypeSelector.xml** file into the PDC database by running the following command:

   ```
   ImportExportPricing -import -config /OOB_ItemTypeSelector.xml -ow
   ```

   The item type selector is imported into the PDC database and the existing item tag-to-item type mapping in the BRM database is overwritten.

Configuring Item Type Selectors

To configure an item type selector:

1. Create the custom bill item in the database by subclassing the **/item** storable object. For example, you can create an **/item/national** object for tracking the charges for national calls. See the discussion about creating custom fields and storable classes in *BRM Developer's Guide*.
2. Make a copy of the *PDC_home***/apps/Samples/Examples/Sample_ ItemTypeSelector.xml** file to customize.
3. Open the file in a text editor or XML editor.
4. Add or modify the elements as required. See Table 4–3, " Elements Supported in the Item Type Selector XML File" for the usage and description of each element.
5. Save and close the file.

   Note: You can save the file with a different name and location or use the original file.

6. Go to the directory where you have saved the file.
7. Import the item type selector from the customized XML file into the PDC database by running the following command:

   ```
   ImportExportPricing -import -config filename
   ```

   where *filename* is the name of the customized XML file.

   The item type selector is imported into the PDC database and transformed to BRM and ECE.

About Configuring Dynamic Quota

Note: You can use Dynamic Quota Configuration only in Oracle Communications Billing and Revenue Management Elastic Charging Engine (ECE).

Dynamic quota allows you to allocate the available quota dynamically for each parallel session of a subscriber based on the rules you configure. This enables the subscribers to run concurrent online charging sessions effortlessly. This also enables you to optimize the network usage effectively. For more information on dynamic quota, see *ECE Release Notes*.

You can define the rules for determining the dynamic quota allocation by configuring dynamic quota selectors in PDC. The rules can reference any of, or a combination of, the event, service, and customer attributes, and profile attribute specifications. You can configure the dynamic quota selector for each service and event combination.

ECE evaluates the rules defined in the dynamic quota selector at run time to derive the quota to be allocated for a session and the quota attributes, such as quota holding time (QHT), volume quota threshold (VQT), and quota validity time (VT). Dynamic quotas change in real time based on the dynamic quota selector rules as ECE grants and redistributes the quotas.

You configure dynamic quota selectors by importing the rules in XML files by using the **ImportExportPricing** utility. You can use the **SampleDynamicQuotaSelector.xml** file in the *PDC_home***/apps/Samples/Examples** directory (where *PDC_home* is the

directory in which the PDC software is installed) to create the XML files for configuring dynamic quota selectors. See "Configuring Dynamic Quota Selectors" for more information.

Table 4–4 lists the elements that are supported in the dynamic quota selector XML file, the usage of each element, and a description of how to specify each element in the XML file.

Table 4–4 Elements Supported in the Dynamic Quota Selector XML File

Element	Syntax	Description
dynamicQuotaSelectors	`<dynamicQuotaSelectors>`*`DynamicQuotaSelector`*`</dynamicQuotaSelectors>`	*DynamicQuotaSelector* specifies the root element of the **DynamicQuotaSelector.xml** file.
name	`<name>`*`DynamicQuotaConfiguration`*`</name>`	*DynamicQuotaConfiguration* specifies the dynamic quota selector name. For example, Generic Selector ABC.
description	`<description>`*`DynamicQuotaSelectorDescription`*`</description>`	*DynamicQuotaSelectorDescription* specifies the description of the dynamic quota selector.
priceListName	`<priceListName>`*`PriceListName`*`</priceListName>`	*PriceListName* specifies the price list name. The valid value is **Default**.
applicableToName	`<applicableToName>`*`ApplicableToName`*`</applicableToName>`	*ApplicableToName* specifies the service name. The attributes of this service are used in the rules.
eventSpecName	`<eventSpecName>`*`EventName`*`</eventSpecName>`	*EventName* specifies the name of an event. The attributes of this event are used in the rules.
validityPeriod	`<validFrom>`*`Validfromdate`*`</validFrom>`	*Validfromdate* specifies the date from which the rule is valid. The valid format is *YYYYMMDD*. **Note:** It is mandatory to mention the date from when the rule is valid. If no date is specified, the rule is valid from immediate effect.
applicableToAllChildEvent	`<applicableToAllChildEvent>`*`ChildEventsOption`*`</applicableToAllChildEvent>`	Specifies whether the dynamic quota selector is applicable to the child events, where *ChildEventsOption* is: ■ **True** if the dynamic quota selector applies to all the child events. ■ **False** if the dynamic quota selector applies to only the specific event type.

rule	`<rule>`
	`<ruleName>`*`RuleName`*`</ruleName>`
	`<ruleOrder>`*`RuleOrder`*`</ruleOrder>`
	`<configuration>`*`DyamicQuotaConfiguration`*`</configuration>`
	`<requestedUnits>`*`RequestedUnitsForAllocation`*`</requestedUnits>`
	`<dynamicQuotaFieldToValueExpression>`*`FieldToValueExpression`*`</dynamicQuotaFieldToValueExpression>`
	`<dynamicQuotaComplexExpression>`*`ComplexExpression`*`</dynamicQuotaComplexExpression>`
	`</rule>`
		Specifies the rules for allocating dynamic quotas, where:
		■ *RuleName* specifies the rule name.
		■ *RuleOrder* specifies the order in which the rules must be applied.
		■ *DyamicQuotaConfiguration* contains the configurations for dynamic quotas.
		■ *RequestedUnitsForAllocation* contains the requested units for allocating dynamic quotas.
		■ *FieldToValueExpression* contains the attributes used for deriving the quota.
		■ *ComplexExpression* contains the complex conditions for deriving the quota.
		Note: You can add multiple rules in the dynamic quota selector.
		Each rule in the dynamic quota selector can refer to a different attribute. For example, Rule 1 can use an event attribute, Rule 2 can use a balance expression, and Rule 3 can use a service attribute.

Table 4–4 (Cont.) Elements Supported in the Dynamic Quota Selector XML File

Element	Syntax	Description

DyamicQuotaConfiguration	`<configuration>`
	`  <key>`*key*`</key>`
	`  <value>`*value*`</value>`
	`  <unit>`*unit*`</unit>`
	`</configuration>`
		Specifies the quota attribute configurations, where:
		▪ *key* specifies the quota attribute. The valid values are:
		VOLUME_QUOTA_THRESHOLD. Specifies how long a granted quota can be idle before the reservation is released.
		QUOTA_HOLDING_TIME. Specifies how much of the granted quota must be consumed before a subscriber can request additional quota. This attribute is configured per service, event, and number of granted units.
		VALIDITY_TIME. Specifies whether validity time can be set to a fixed value per service-event combination at runtime. This attribute is independent of the number of units in the granted quota.
		▪ *value* specifies the quota attribute value.
		▪ *unit* specifies the unit for measurement. The valid values are:
		Seconds
		Minutes
		Hours
		Days
		Bytes
		Kbytes
		Mbytes
		Gbytes
		None
		Note: You must add the *DyamicQuotaConfiguration* element for each valid quota attribute.

RequestedUnitsForAllocation	`<requestedUnits>`
	`  <fieldName>`*fieldname*`</fieldname>`
	`  <unit>`*unit*`</unit>`
	`  <dynamicQuotaBinaryExpression>`*dynamicQuotaBinaryExpression*`</dynamicQuotaBinaryExpression>`
	`</requestedUnits>`
	
	`<requestedUnits>`
	`  <fieldName>`*fieldname*`</fieldname>`
	`  <unit>`*unit*`</unit>`
	`  <numberExpression>`*numberExpression*`</numberExpression>`
	`</requestedUnits>`
		Specifies the details about the requested units, where:
		▪ *fieldName* specifies the fully qualified requested attribute field name. This is specified in the event specification. For example, *EventDelayedSessionTelcoGsm.REQUESTED_UNITS.INPUT_VOLUME*.
		▪ *unit* specifies the unit for measurement. The valid values are:
		Seconds
		Minutes
		Hours
		Days
		Bytes
		Kbytes
		Mbytes
		Gbytes
		None
		▪ *dynamicQuotaBinaryExpression* specifies the condition for deriving the quota.
		▪ *numberExpression* specifies a valid number; for example, 10.0.

Table 4–4 (Cont.) Elements Supported in the Dynamic Quota Selector XML File

Element	Syntax	Description

dynamicQuotaBinaryExpression	`<dynamicQuotaBinaryExpression>`
	`<leftOperand>`
	
	`<numberExpression>`*`numberExpression`*`</numberExpression>`
	
	`<balanceExpression>`*`balanceExpression`*`</balanceExpression>`
	`</leftOperand>`
	
	`<rightOperand>`
	
	`<balanceExpression>`*`balanceExpression`*`</balanceExpression>`
	
	`<numberExpression>`*`numberExpression`*`</numberExpression>`
	`</rightOperand>`
	
	`<dynamicQuotaBinaryOperator>`*`Operator`*`</dynamicQuotaBinaryOperator>`
	`</dynamicQuotaBinaryExpression>`
		Specifies the condition for deriving the quota, where:
		▪ *numberExpression* specifies a valid number; for example, 10.0.
		▪ *balanceExpression* specifies the currency or noncurrency balance element defined in PDC.
		▪ *Operator* specifies the arithmetic operators. The valid values are:
		Add
		Subtract
		Multiply
		Divide

FieldToValueExpression	`<dynamicQuotaFieldToValueExpression>`
	`<separator>`*`Delimiter`*`</separator>`
	`<operation>`*`Operator`*`</operation>`
	`<fieldName>`*`FieldName`*`</fieldName>`
	`<fieldKind>`*`FieldKind`*`</fieldKind>`
	`<fieldValue>`*`FieldValue`*`</fieldValue>`
	`</dynamicQuotaFieldToValueExpression>`
		Specifies the attributes used for deriving the quota, where:
		▪ *Delimiter* specifies the character that is used to separate the field values.
		Note: This is applicable only when the *operator* is **IN_LIST**.
		▪ *Operator* specifies the operator to be used in the expression. The valid values are:
		EQUAL_TO
		NOT_EQUAL_TO
		GREATER_THAN
		GREATER_THAN_EQUAL
		LESS_THAN
		LESS_THAN_EQUAL
		IN_LIST
		CONTAINS
		▪ *FieldName* specifies the fully qualified field name. For example, EventSessionTelcoGsm.REQUESTED_UNITS.NUMBER_OF_UNITS.
		▪ *FieldKind* specifies the type of attributes. The valid values are:
		PRODUCT_SPEC_FIELD
		EVENT_SPEC_FIELD
		CUSTOMER_SPEC_FIELD
		PROFILE_SPEC_FIELD
		▪ *FieldValue* specifies the field value that is used to track the balance impacts.
		Note: You can add multiple *FieldToValueExpression* elements in a rule to derive the quota based on a combination of attributes.

Table 4–4 (Cont.) Elements Supported in the Dynamic Quota Selector XML File

Element	Syntax	Description

ComplexExpression	`<dynamicQuotaComplexExpression>`
	`<operator>`*Operator*`</operator>`
	`<value>`*Value*`</value>`
	
	`<dynamicQuotaBinaryExpression>`*dynamicQuotaBinaryExpression*`</dynamicQuotaBinaryExpression>`
	`</dynamicQuotaComplexExpression>`
		Specifies complex conditions for deriving the quota, where:
		▪ *Operator* specifies the operator to be used in the expression. The valid values are:
		EQUAL_TO
		NOT_EQUAL_TO
		GREATER_THAN
		GREATER_THAN_EQUAL
		LESS_THAN
		LESS_THAN_EQUAL
		▪ *Value* is a numeric value. For example, 0.004.
		▪ *dynamicQuotaBinaryExpression* specifies the condition for deriving the quota.

Configuring Dynamic Quota Selectors

To configure a dynamic quota selector:

1. Make a copy of the *PDC_home*/**apps/Samples/Examples/Sample_DynamicQuotaSelector.xml** file to customize.
2. Open the file in a text editor or XML editor.
3. Add or modify the elements as required. See Table 4–4, " Elements Supported in the Dynamic Quota Selector XML File" for the usage and description of each element.
4. Save and close the file.

   Note: You can save the file with a different name.

5. Go to the directory where you have saved the file.
6. Import the dynamic quota selector from the customized XML file into the PDC database by running the following command:

   ```
   ImportExportPricing -import -config filename
   ```

   where *filename* is the name of the customized XML file. For example, the **Dynamic_QuotaSelector.xml** file.

   The dynamic quota selector is imported into the PDC database and published to ECE.

About Configuring Policy Specifications

You configure policy specifications in PDC for policy-driven charging.

Note: Policy specifications can be used only with ECE.

After you configure policy specifications, you can import the policy specifications and policy labels from the customized XML files into the PDC database by running the following command:

```
ImportExportPricing -import -config filename
```

where *filename* is the name of the customized XML file.

The policy specification is imported into the PDC database and transformed to BRM and ECE.

Configuring Policy Specifications

To configure policy specifications:

1. Make a copy of the *PDC_home*/**apps/Samples/Examples/SamplePolicyLabel.xml** file to customize.
2. Open the file in a text editor or XML editor.
3. Add or modify the elements as required.
4. Save and close the file.
5. Make a copy of the *PDC_home*/**apps/Samples/Examples/SamplePolicySpecification.xml** file to customize.
6. Open the file in a text editor or XML editor.
7. Add or modify the elements as required.
8. Save and close the file.
9. Go to the directory where you have saved the files.
10. Import the policy specifications and policy labels from the customized XML files into the PDC database by running the following command:

    ```
    ImportExportPricing -import -config filename
    ```

    where *filename* is the name of the customized XML file.

The policy specifications and policy labels are imported into the PDC database and transformed to ECE.

After you configure policy specifications, you can export the policy specifications and policy labels in PDC to XML files by running the following command:

```
ImportExportPricing -export -config POLICY_SPECIFICATION
ImportExportPricing -export -config POLICY_LABEL
```

You can load the updated sample XML files into PDC by using the

ImportExportPricing utility. See "ImportExportPricing" for the utility's syntax and

parameter descriptions.

Configuring Dynamic Pricing

Dynamic pricing allows you to override the price specified in the product offerings at run time.

To override the price, you create a pricing XML file with dynamic tags and import the file into the PDC database by using the **ImportExportPricing** utility.

Dynamic tags are the XML elements that are used for overriding the value of the pricing attributes. ECE uses these attributes to determine the price when processing a usage request.

> **Important:** Ensure that you create a unique dynamic tag across multiple charge offers since the tags are not scoped to charge offers at run time.

Dynamic tags are used for different pricing within a rate plan. PDC provides a sample XML file for dynamic tags in the *PDC_home*/**apps/Samples/Examples** directory, where *PDC_home* is the directory in which you installed PDC. The dynamic tag (**priceTag**) consists of the following fields:

- **attributeName.** The name of the attribute. The value of the attribute is dynamically determined during rating.
- **tagName.** The unique identifier of the dynamic tag.
- **tagScope.** The scope of the dynamic tag. Following are the available tag scopes:
  - EVENT_PROFILE
  - PRODUCT_PROFILE
  - CUSTOMER_PROFILE

> **Note:** Only the EVENT_PROFILE tag scope is supported for charges with the Convergent Usage pricing profile.

- **Description.** (Optional) The description of the dynamic tag.

For example:

```
<fixedCharge>
     <price>20.0</price>
     <unitOfMeasure>NONE</unitOfMeasure>
     <balanceElementNumCode>840</balanceElementNumCode>
     <discountable>true</discountable>
     <priceType>CONSUMPTION</priceType>
     <priceTag>
          <attributeName>price</attributeName>
          <tagName>PRICE_TAG_FIXED_PRICE</tagName>
          <tagScope>EVENT_PROFILE</tagScope>
     </priceTag>
 </fixedCharge>
```

In this example, the default price is 20.0. This is used by default if an overriding price is not specified.

For information on importing the pricing XML file into the PDC database, see "Importing and Exporting Pricing and Setup Components".

For information on enabling ECE to override the default value of the pricing attributes at run time, see the discussion about configuring ECE to override a product price in *BRM Elastic Charging Engine Implementation Guide*.

Enabling Single Currency Graph

To enable a single currency graph:

> **Note:** Single currency graph can be used only with ECE.

1. Go to the *PDC_Home*/**apps/bin** directory.
2. Export the pricing data by running the **ImportExportPricing** utility.

   For example, to export the data to a file named **chargeoffer_pricing.xml**, run the following command:

   ```
   ./ImportExportPricing -export chargeoffer -pricing
   ```

3. In the **chargeoffer_pricing.xml** file, search for the **balanceElementNumCode** attribute. By default, only one currency code is available. You can add more currency codes.

   For example:


```
-<subscriberCurrency>
  <currencyCode>USD</currencyCode>
  <currencyCode>EUR</currencyCode>
  -<applicableRum>
      <applicableRumName>Duration</applicableRumName>
      <minQuantity>1.0</minQuantity>
      <minQuantityUnit>NONE</minQuantityUnit>
      <incrementQuantity>1.0</incrementQuantity>
      <incrementQuantityUnit>NONE</incrementQuantityUnit>
      <roundingMode>NEAREST</roundingMode>
      -<crpRelDateRange>
         -<absoluteDateRange>
             <startDate>0</startDate>
             <endDate>inf</endDate>
         </absoluteDateRange>
        -<crpCompositePopModel>
             <name>Pricing</name>
             -<usageChargePopModel>
               -<priceTier>
                   <distributionMethod>FROM_BAL_IMPACT</distributionMethod>
                   -<tierBasis>
                      <rumTierExpression/>
                    </tierBasis>
                    <enforceCreditLimit>false</enforceCreditLimit>
                    <rumName>Duration</rumName>
                    <currencyCode>USD</currencyCode>
                   -<priceTierValidityPeriod>
                       <lowerBound>0</lowerBound>
                       <validFrom>0</validFrom>
               -<priceTierRange>
                  <upperBound>NO_MAX</upperBound>
                 -<scaledCharge>
                   <price>0.0</price>
                   <unitOfMeasure>NONE</unitOfMeasure>
                   <balanceElementNumCode>840</balanceElementNumCode>
                   <discountable>true</discountable>
                   <priceType>CONSUMPTION</priceType>
                   <incrementStep>1.0</incrementStep>
                   <incrementRounding>NONE</incrementRounding>
                  </scaledCharge>
               </priceTierRange>
    </priceTierValidityPeriod>
<applicableQuantity>ORIGINAL</applicableQuantity>
</priceTier>
</usageChargePopModel>
</crpCompositePopModel>
</crpRelDateRange>
```

```
</applicableRum>
</subscriberCurrency>
```

4. Save the XML file as **import_pricing.xml** file.
5. Import the **import_pricing.xml** file by using the **ImportExportPricing** utility.

   For example:

   ```
   ./ImportExportPricing -import -pricing import_pricing.xml -ow
   ```

6. Add, edit, or remove balance impacts for all the supported subscriber currencies.

5

Synchronizing Setup Components

This chapter describes how to synchronize setup components from Oracle Communications Billing and Revenue Management (BRM) and rating systems with Oracle Communications Pricing Design Center (PDC).

About Synchronizing Setup Components

When configuring a new PDC system, you use the **SyncPDC** utility to synchronize the setup components defined in billing and rating systems with PDC. See "About Configuring Setup Components" for more information.

Note: Ensure that **/event/realtimeDiscount** does not exist in BRM during synchronization.

SyncPDC synchronizes the following setup components:

- Service definitions
- Event definitions
- Account definition
- General ledger (G/L) IDs
- Provisioning tags
- Tax codes
- Tax suppliers
- Business profiles

After these setup components are synchronized with PDC, any modifications to these setup components must only be done in the system in which they were initially defined and then resynchronized.

SyncPDC synchronizes only the changes from the previous synchronization. It generates reports for the components that it synchronizes. See "Generating Synchronization Reports" for more information.

When synchronizing setup components, **SyncPDC** creates display names in PDC for the corresponding data objects and their fields. For example, it creates the **GsmTelephony** PDC name for the **/service/gsm/telephony** service name in BRM. You can change the PDC display name in PDC. For example, you can change **EventTelcoGsmVoice** to **Voice Call**. If you later run **SyncPDC** to resynchronize this event, it retains the name change. See "Changing Display Names" for more

information about updating PDC display names.

If you delete a setup component (or any of its fields) from the billing or rating system, you must manually remove the corresponding PDC component (or its fields) from the PDC database. See "Deleting Components Mastered in BRM" for more information.

About the SyncPDC Utility

The **SyncPDC** utility runs as a server process in the background, continuously checking for data to synchronize from BRM or rating system with PDC. You can schedule it to run immediately, at a specified time, or regularly at a specified time. At any given time, you can stop the synchronization by stopping **SyncPDC**. See "Stopping SyncPDC" for more information.

Prerequisites for Running SyncPDC

Before you run **SyncPDC**, do the following:

- Start the transformation engines for the rating systems. See the discussion of starting the transformation engines in *PDC Installation and System Administration Guide*.
- Include the *PDC_home***/apps/bin** directory in your PATH environment variable, where *PDC_home* is the directory in which you installed the PDC software.
- Configure the *BRM_Integration_Pack_Home***/apps/syncpdc/SyncPDCConfiguration.xml** file. See Table 5–1, " Elements in the SyncPDCConfiguration.xml File", for more information.

To run **SyncPDC**, you must have **Pricing Design Admin** role privileges.

About Configuring SyncPDC

SyncPDC uses the **SyncPDCConfiguration.xml** file, which contains connection information for the Oracle WebLogic Server Administration, the PDC server, and the cross-reference database and configuration settings for **SyncPDC**. This file is generated during PDC installation and is located in the *BRM_Integration_Pack_Home***/apps/syncpdc** directory. By default, this file contains the values that you provide during BRM Integration Pack installation. You can change the default values by updating this file.

Table 5–1 lists the elements in **SyncPDCConfiguration.xml**, the usage of each element, and a description of how to specify each element based on the default version of the file.

Table 5–1 Elements in the SyncPDCConfiguration.xml File

Element	Syntax	Description
Configuration	`<Configuration>`	The root element of **SyncPDCConfiguration.xml**.

xrefDatabase	`<xrefDatabase>`
	`  <connectionInfo>`
	`  <login>`*CrossRefUserName*`</login>`
	`  <password>`*CrossRefPassword*`</password>`
	`  <hostName>`*CrossRefHostName*`</hostName>`
	`  <port>`*CrossRefPort*`</port>`
	`  <serviceName>`*CrossRefServiceName*`</serviceName>`
	`  </connectionInfo>`
	`</xrefDatabase>`
		Contains the details about the cross-reference database, where:
		■ *CrossRefUserName* specifies the cross-reference database user name
		■ *CrossRefPassword* specifies the encrypted cross-reference database password
		■ *CrossRefHostName* specifies the IP address or the host name of the computer on which the cross-reference database is configured
		■ *CrossRefPort* specifies the port number assigned to the cross-reference database
		■ *CrossRefServiceName* specifies the name of the cross-reference database service

pricingServer	`<pricingServer>`
	`  <connectionInfo>`
	`  <hostName>`*PricingServerHostName*`</hostName>`
	`  <port>`*PricingServerPort*`</port>`
	`  <adminUser>`*AdminUserName*`</adminUser>`
	`  <adminPassword>`*AdminPassword*`</adminPassword>`
	`  <pdcUser>`*PDCUser*`</pdcUser>`
	`  <pdcUserPassword>`*PDCUserPassword*`</pdcPassword>`
	`  </connectionInfo>`
	`</pricingServer>`
		Contains the PDC server information, where:
		■ *PricingServerHostName* specifies the IP address or the host name of the computer on which PDC is deployed
		■ *PricingServerPort* specifies the port number of the domain on which PDC is deployed
		■ *AdminUserName* specifies the user name of the PDC server administrator
		■ *AdminPassword* specifies the encrypted PDC server administrator password
		■ *PDCUser* specifies the user name of the PDC user
		■ *PDCUserPassword* specifies the encrypted PDC user password

transformationHome	`<transformationHome>`*TransformationHome*`</transformationHome>`	Specifies the path to the directory that stores the transformation process IDs, where *TransformationHome* is the complete path to the directory.
syncPDCLogFileLocation	`<syncPDCLogFileLocation>`*SyncPDCLogFile*`</syncPDCLogFileLocation>`	Specifies the path to the directory that stores **SyncPDC** log files, where *SyncPDCLogFile* is the complete path and the name of the log file.
brand	`<brand>`*BrandOption*`</brand>`	Specifies whether PDC supports branding, where *BrandOption* is: ■ **enabled** to specify that PDC supports branding ■ **disabled** to specify that PDC does not support branding
pdcSSL	`<pdcSSL>`*SSLOption*`</pdcSSL>`	Specifies whether the PDC server supports SSL, where *SSLOption* is: ■ **enabled** to specify that PDC supports SSL. If SSL is enabled, **SyncPDC** uses the **t3s://***Host***:***Port* URL to access PDC ■ **disabled** to specify that PDC does not support SSL. If SSL is disabled, **SyncPDC** uses the **t3://***Host***:***Port* URL to access PDC

Table 5–1 (Cont.) Elements in the SyncPDCConfiguration.xml File

Element	Syntax	Description

retryInfo	`<retryInfo>`
	`<retryInterval>`*`RetryInterval`*`</retryInterval>`
	`<maxRetries>`*`MaxRetries`*`</maxRetries>`
	`</retryInfo>`
		Contains the information about when and how many times **SyncPDC** should retry the synchronization process if it fails initially, where:
		■ *RetryInterval* specifies the interval in seconds after which **SyncPDC** again tries to synchronize the data that was not previously synchronized
		■ *MaxRetries* specifies the maximum number of retries

ECESync	`<ECESync>`*`ECESyncOption`*`</ECESync>`	Specifies to synchronize BRM data with PDC, where *ECESyncOption* is: ■ **true** to synchronize BRM data with PDC to use ECE for usage rating. ■ **false** to synchronize BRM data with PDC to use real-time and batch rating engines for usage rating.
pdcXML	`<pdcXML>`*`PDCData`*`</pdcXML>`	Specifies the path to the directory where **SyncPDC** creates the XML files containing the extracted data in PDC format, where *PDCData* is the complete path to the directory.
reportFileLocation	`<reportFileLocation>`*`ReportLocation`*`</reportFileLocation>`	Specifies the path to the directory where **SyncPDC** stores the report of the synchronization process, where *ReportLocation* is the complete path to the directory.
brmExtractedXML	`<brmExtractedXML>`*`BRMData`* `</brmExtractedXML>`	Specifies the path to the directory where **SyncPDC** creates the XML files containing the data extracted from the BRM database, where *BRMData* is the complete path to the directory.
pdcExtractedXML	`<pdcExtractedXML>`*`PDCData`* `</pdcExtractedXML>`	Specifies the path to the directory where **SyncPDC** creates the XML files containing the data extracted from the PDC database, where *PDCData* is the complete path to the directory.
archiveFileLocation	`<archiveFileLocation>`*`ArchiveFile`*`</archiveFileLocation>`	Specifies the path to the directory where a successfully processed file is archived, where *ArchiveFile* is the complete path to the directory. No archive file is created if there is no change from the previous synchronization.
skipBREMigration	`<skipBREMigration>`*`SkipOption`*`</skipBREMigration>`	Specifies to skip synchronization of pipeline configuration data, where *SkipOption* is either true or false.

Table 5–1 (Cont.) Elements in the SyncPDCConfiguration.xml File

Element	Syntax	Description

fieldSelection	`<fieldSelection>`
	`<targetEngine>`*`TargetRatingEngine`*`</target Engine>`
	`<eventFields>`
	`<eventName>`*`EventName`*`</eventName>`
	`<fullyQualifiedName>`*`EventFieldName`*`</full yQualifiedName>`
	`</eventFields>`
	`</fieldSelection>`
		Contains the information about the target rating engine and the BRM event fields provided as input to the target rating engine for usage rating.
		<targetEngine> element specifies the target rating engine used for usage rating, where: *TargetRatingEngine* is one of the following:
		■ **Convergent Charging**. Specifies that ECE is used for usage rating.
		■ **Realtime Charging**. Specifies that the real-time rating engine is used for usage rating.
		■ **Batch Charging**. Specifies that the batch rating engine is used for usage rating.
		<eventFields> element specifies the fields that are provided as input to *TargetRatingEngine*, where:
		■ *EventName* specifies the class name of the BRM event; for example,/event.
		■ *EventFieldName* specifies the fully qualified name of the BRM event field that is provided as input to *TargetRatingEngine*; for example, PIN_FLD_ NAME.
		Note: Each **<fieldSelection>** element can have multiple **<eventFields>** elements but only one **<targetEngine>** element. To support multiple target rating engines, add the **<fieldSelection>** element for each target rating engine.
		Similarly, each **<eventFields>** element can have multiple **<fullyQualifiedName>** elements but only one **<eventName>** element. To rate multiple events, add the **<eventFields>** element for each event.

Table 5–1 (Cont.) Elements in the SyncPDCConfiguration.xml File

Element	Syntax	Description

syncPDCBREConfig	`<syncPDCBREConfig>`
	`<containerDesc>`
	`<param>`
	`<paramname>`*`ParamName`*`</paramname>`
	`<paramvalue>`*`ParamValue`*`</paramvalue>`
	`</param>`
	`</containerDesc>`
	`<eventExtension>`
	`<param>`
	`<paramname>`*`BRMBatchRatingEventName`*`</paramname>`
	`<paramvalue>`*`EventExtensionBlockName`*`</paramvalue>`
	`</param>`
	`</eventExtension>`
	`<serviceExtension>`
	`<param>`
	`<paramname>`*`BRMServiceName`*`</paramname>`
	`<paramvalue>`*`ServiceExtensionBlockName`*`</paramvalue>`
	`</param>`
	`</serviceExtension>`
	`</syncPDCBREConfig>`
		Contains the information about EDR container description data and delayed events and services for batch rating engine.
		The **<containerDesc>** element specifies the EDR container description data that is used by the batch rating engine, where:
		■ *ParamName* specifies the field name that defines the EDR container description name
		■ *ParamValue* specifies the value of *ParamName*. The EDR container description data that matched *ParamValue* is used as the base to synchronize EDR fields for events and services
		The **<eventExtension>** element specifies the mapping of a delayed event to the extension block name used in the batch rating engine, where:
		■ *BRMBatchRatingEventName* specifies the delayed event name
		■ *EventExtensionBlockName* specifies the extension block name to which the delayed event is mapped
		Add **<param>** elements to the **<eventExtension>** elements for each *BRMBatchRatingEventName* and *EventExtensionBlockName* to include.
		The **<serviceExtension>** element specifies the mapping of a BRM service to the extension block name used in the batch rating engine, where:
		■ *BRMServiceName* specifies the BRM service name
		■ *ServiceExtensionBlockName* specifies the extension block name to which the BRM service is mapped
		Add **<param>** elements to the **<serviceExtension>** elements for each *BRMServiceName* and *ServiceExtensionBlockName* to include.
		For more information about the EDR container description, see the BRM documentation.

scheduling	`<scheduling>`
	`<runOnStartup>`*`StartupOption`*`</runOnStartup>`
	`<startAt>`*`StartTime`*`</startAt>`
	`<interval>`*`Interval`*`</interval>`
	`</scheduling>`
		Contains the scheduling configurations for **SyncPDC**.
		If the **<scheduling>** element is not present or its child elements are not defined correctly, **SyncPDC** runs immediate only once.
		StartupOption specifies whether **SyncPDC** should run immediately:
		■ **true** specifies that **SyncPDC** runs immediately in addition to the scheduled time
		■ **false** specifies that **SyncPDC** does not run immediately but at the scheduled time
		StartTime specifies the time in *HH*:*MM* format when you want **SyncPDC** to start, where:
		■ *HH* specifies hours between 00 and 23
		■ *MM* specifies minutes between 00 and 59.
		Interval specifies the frequency in *N*:*A* format at which you want to run **SyncPDC**, where:
		■ *N* is an integer
		■ *A* is **D** for days, **H** for hours, or **M** for minutes
		For example, to run **SyncPDC** every two days, enter:
		`<interval>2:D</interval>`
		
		Note: 24:H is not same as **1:D** due to daylight saving time.

Updating the SyncPDCConfiguration.xml File

To update **SyncPDCConfiguration.xml**:

> **Note:** To change an encrypted password in the **SyncPDCConfiguration.xml** file, see the discussion about changing the encrypted passwords in the configuration files in *PDC Installation and System Administration Guide*.

1. Open the *BRM_Integration_Pack_Home***/apps/syncpdc/SyncPDCConfiguration.xml** file in a text editor or an XML editor.
2. Edit the file. See Table 5–1, " Elements in the SyncPDCConfiguration.xml File", for more information.
3. Save and close the file.
4. Run the **SyncPDC** utility.

   See "Running SyncPDC" for more information.

Running SyncPDC

To run **SyncPDC**:

1. Ensure that transformation engines are running. See the discussion of starting the transformation engines in *PDC Installation and System Administration Guide*.
2. Go to the *BRM_Integration_Pack_Home***/apps/syncpdc** directory.
3. Enter the following command:

   ```
   startSyncPDC
   ```

   The **Enter Key Password** prompt appears.
4. Enter the password PDC uses for accessing the PDC alias key in the keystore (*BRM_Integration_Pack_Home***/apps/conf/pdc.jks**).

   A series of messages appears on the command prompt that indicate the synchronization status.

   For example:

   ```
   Clean up work item SYNC_EVENT...
   Work item SYNC_EVENT started (item 1 of 8).
     Processing EXTRACT work action...
     Work action EXTRACT completed.
     Processing ANALYZE work action...
     Work action ANALYZE completed.
     Processing TRANSFORM work action...
     Work action TRANSFORM completed.
   Processing LOAD work action...
     Work action LOAD completed.
   Work item SYNC_EVENT completed.
   ```

   After the synchronization is complete, check the synchronization report available in the directory defined in the **<reportFileLocation>** element of the **SyncPDCConfiguration.xml** file.

Scheduling Synchronization

You can schedule the data synchronization from BRM or rating systems with PDC by:

- Running SyncPDC Immediately
- Running SyncPDC at a Scheduled Time
- Running SyncPDC Immediately and at a Scheduled Time
- Running SyncPDC at Recurring Scheduled Time

Running SyncPDC Immediately

To run **SyncPDC** immediately instead of waiting for the next scheduled run time:

1. Open the *BRM_Integration_Pack_Home*/**apps/syncpdc/SyncPDCConfiguration.xml** file in a text editor or an XML editor.
2. Set the **<runOnStartup>** element to **true**:

   ```
   <runOnStartup>true</runOnStartup>
   ```

3. Save and close the file.
4. (Optional) If **SyncPDC** is already running in the background, stop it.

   See "Stopping SyncPDC" for more information.
5. Run **SyncPDC**. See "Running SyncPDC" for more information.

Running SyncPDC at a Scheduled Time

To run **SyncPDC** at a schedule time:

1. Open the *BRM_Integration_Pack_Home*/**apps/syncpdc/SyncPDCConfiguration.xml** file in a text editor or an XML editor.
2. Set the **<runOnStartup>** element to **false**:

   ```
   <runOnStartup>false</runOnStartup>
   ```

3. Set the **<startAt>** element to the start time:

   ```
   <startAt>HH:MM</startAt>
   ```

   where:

   - *HH* specifies hours between 00 and 23
   - *MM* specifies minutes between 00 and 59

   For example, to run **SyncPDC** at 11:50 pm, enter:

   ```
   <startAt>23:50</startAt>
   ```

4. Save and close the file.
5. Run **SyncPDC**.

   See "Running SyncPDC" for more information.

Running SyncPDC Immediately and at a Scheduled Time

To run **SyncPDC** immediately and at a schedule time:

1. Open the *BRM_Integration_Pack_Home*/**apps/syncpdc/SyncPDCConfiguration.xml** file in a text editor or an XML editor.

2. Set the **<runOnStartup>** element to **true**:

   ```
   <runOnStartup>true</runOnStartup>
   ```

3. Set the **<startAt>** element to the start time:

   ```
   <startAt>HH:MM</startAt>
   ```

   where:

   - *HH* specifies hours between 00 and 23
   - *MM* specifies minutes between 00 and 59

   For example, to run **SyncPDC** immediately and at 11:50 pm, enter:

   ```
   <runOnStartup>true</runOnStartup>
   <startAt>23:50</startAt>
   ```

4. Save and close the file.
5. Run **SyncPDC**.

   See "Running SyncPDC" for more information.

Running SyncPDC at Recurring Scheduled Time

To run **SyncPDC** at recurring scheduled time:

1. Open the *BRM_Integration_Pack_Home*/**apps/syncpdc/SyncPDCConfiguration.xml** file in a text editor or an XML editor.
2. Set the **<runOnStartup>** element to **false**:

   ```
   <runOnStartup>false</runOnStartup>
   ```

3. Set the **<startAt>** element to the start time:

   ```
   <startAt>HH:MM</startAt>
   ```

   where:

   - *HH* specifies hours between 00 and 23
   - *MM* specifies minutes between 00 and 59

4. Specify the **<interval>** element as follows:

   ```
   <interval>N:A</interval>
   ```

   where:

   - *N* is an integer
   - *A* is **D** for days, **H** for hours, or **M** for minutes

   For example, to run **SyncPDC** at 11:50 pm everyday, enter:

   ```
   <runOnStartup>false</runOnStartup>
   <startAt>23:50</startAt>
   <interval>1:D</interval>
   ```

5. Save and close the file.
6. Run **SyncPDC**.

   See "Running SyncPDC" for more information.

Synchronizing BRM Data with PDC to Use ECE for Usage Rating

To synchronize BRM data with PDC to use ECE for usage rating:

1. Open the *BRM_Integration_Pack_Home*/**apps/syncpdc/SyncPDCConfiguration.xml** file in a text editor or an XML editor.
2. Set the **<ECESync>** element to **true**:

   ```
   <ECESync>true</ECESync>
   ```

3. Save and close the file.
4. Run **SyncPDC**.

   See "Running SyncPDC" for more information.

Stopping SyncPDC

You can stop **SyncPDC** at any time. If you stop **SyncPDC** while it is synchronizing data, it finishes the synchronization before stopping.

To stop **SyncPDC**, enter the following command:

```
stopSyncPDC
```

A series of messages appears, indicating progress.

For example:

```
Previous process id :4381
Killing child processes ...
Killing process ...
Cleaning temporary files ...
[1]  + Killed
```

Generating Synchronization Reports

SyncPDC generates reports for the components that it synchronizes. If there are no changes from the previous synchronization, **SyncPDC** does not create a report.

You specify the location of the **SyncPDC** report in the **<reportFileLocation>** element of the **SyncPDCConfiguration.xml** file.

When synchronizing data from BRM with PDC, the **SyncPDC** report lists the object type for each setup component that is synchronized. It lists the PDC display name and the corresponding BRM or rating system name for each of the setup component objects. It also provides the total number of object types for each setup component that are synchronized.

Figure 5–1 shows a sample **SyncPDC** report for BRM data.

Figure 5–1 Sample SyncPDC Report for BRM Data

SyncPDC Report. Generated on 20120730_235025_832

Report Type: SyncPDC, Object Type: Event

PDC Internal Id	PDC Event Name	BRM Event Name
d672f756-ed1c-4be8-93dd-ea4f787ea809	Event	/event
9a34af04-9f86-4c5f-8b99-b884c6c47490	EventActivityContent	/event/activity/content
2c65f382-ad3a-4920-b30d-63785e3ef954	EventActivitySettlement	/event/activity/settlement
b7338796-3c4c-4564-88ca-15cc5e289ecd	EventActivityTelco	/event/activity/telco
51a87729-e4ab-466f-80de-9d37f7890aa8	EventBillingCycleDiscount	/event/billing/cycle/discount
c0576f18-54cc-4f8a-9ed6-754ca44dbead	EventBillingCycleDiscountMostcalled	/event/billing/cycle/discount/mostcalled
b05c6917-305d-4b43-bb3a-b9f6faa6b9c1	EventBillingCycleFold	/event/billing/cycle/fold
b260e6d5-601e-442b-9f96-f56c58e11df7	EventBillingCycleRolloverMonthly	/event/billing/cycle/rollover/monthly
51b0ae22-7101-4b47-af55-1a122ea48419	EventBillingFeeFailed_payment	/event/billing/fee/failed_payment
c37e96ed-0202-47ee-a45a-043dc8699adb	EventBillingIncentive	/event/billing/incentive
f13ad160-3093-4a60-9574-a3957952ea86	EventBillingProductFeeCancel	/event/billing/product/fee/cancel
3bad8520-2c95-4b1c-a22b-78c87b3e0cfb	EventBillingProductFeeCycleCycle_arrear	/event/billing/product/fee/cycle/cycle_arrear
927f9fca-f02a-4e16-98ce-ea8c12d193ea	EventBillingProductFeeCycleCycle_forward_annual	/event/billing/product/fee/cycle/cycle_forward_annual

Total number of BRM event objects synchronized: 13

When synchronizing data from ECE with PDC, the **SyncPDC** report provides the status of loading the XML files containing ECE data.

Figure 5–2 shows a sample **SyncPDC** report for ECE data.

Figure 5–2 Sample SyncPDC Report for ECE Data

Start Load ECE Report:

Import File Name	Status
PdcService.xml	SUCCESS
Customer.xml	SUCCESS
PdcEvent.xml	FAILED

Successfully Imported Files: 2

Failed to Import Files: 1

End Load ECE Report.

Handling Synchronization Failures

The most common reasons for a synchronization failure are as follows:

- The billing or rating system is down
- After billing or rating system data was modified, the billing or rating system was not restarted. Hence, no data is available for synchronization with PDC
- The PDC system is down

- The transformation engines are down
- The transformation engines failed to process the PDC data and update the cross-reference database
- The billing or rating system database or the PDC database is down

Reprocessing a Failed Synchronization

To reprocess a failed synchronization:

1. Identify and fix the cause of the synchronization failure.

   For information about synchronization failures, check the **SyncPDC** log file located in the directory specified in the **<syncPDCLogFileLocation>** element of the **SyncPDCConfiguration.xml** file.

2. Run **SyncPDC**.

   See "Running SyncPDC" for more information.

   SyncPDC synchronizes the data that failed to synchronize earlier and ignores the data that was already synchronized.

Changing Display Names

When synchronizing setup components, **SyncPDC** creates display names in PDC for the corresponding data objects and their fields. You can change the default display names that **SyncPDC** creates.

To change the display name of a setup component:

1. Export the setup component to an XML file by using the **ImportExportPricing** utility.

   See "Exporting Pricing and Setup Components from PDC" for more information.

2. Open the XML file in a text editor or an XML editor.
3. Change the display name of the setup component in the XML file.
4. Save and close the file.
5. Import the updated XML file to PDC by using **ImportExportPricing**.

   See "Importing Pricing and Setup Components" for more information.

   If you later run **SyncPDC** to synchronize any changes from the billing or rating system with PDC, **SyncPDC** retains the updated component name.

6

Importing and Exporting Pricing and Setup Components

This chapter describes how to use the **ImportExportPricing** utility of Oracle Communications Pricing Design Center (PDC) to import pricing and setup components to and export pricing and setup components from the PDC database by using XML files. See "ImportExportPricing" for the utility's syntax and parameter descriptions.

You should have a basic understanding of the following:

- PDC. See "About Pricing Design Center" for more information.
- Product offerings. See "About Creating Product Offerings" for more information.

To use **ImportExportPricing**, you must also understand the following:

- Extensible markup language (XML) programming
- XML schema definition (XSD)

About the ImportExportPricing Utility

The **ImportExportPricing** utility is a command-line interface for importing pricing and setup components to and for exporting pricing and setup components from the PDC database by using XML files.

ImportExportPricing is a role-based utility that authenticates and authorizes the users based on their role. The role of the users determines what tasks the users can perform and the information they can see.

The users of this utility can have the following roles:

- **Pricing Design Admin**: The user having this role can import and export all the pricing and setup components.
- **Pricing Reviewer**: The user having this role can only export all the pricing and setup components.
- **Pricing Analyst**: The user having this role can import only pricing components and cannot import setup components and profile data. The user can export all the components.

ImportExportPricing allows you to filter the data that you want to import to or export from PDC. For example, you can export all pricing and setup components from PDC or specify specific pricing and setup components to export.

You can use **ImportExportPricing** to:

- Export pricing and setup components from the PDC database to an XML file. See "Exporting Pricing and Setup Components from PDC" for more information.
- Import pricing and setup components defined in an XML file into the PDC database. See "Importing Pricing and Setup Components" for more information.
- List pricing or setup components available in the PDC database. See "Listing Pricing or Setup Components" for more information.
- Delete setup components from the PDC database if it is not prerequisite data for any other component. See "Deleting Components Mastered in BRM" for more information.
- Create pricing and setup components in an XML file and load the data into the PDC database. See "Creating the Pricing and Setup Components by Using XML Files" for more information.
- Obsoletes pricing components from the PDC database if it is not prerequisite data for any other component. See "Obsoleting PDC Components" for more information.

About Dependencies between Pricing and Setup Components

Before importing a pricing component, you must import the setup components that the pricing component references. You must ensure that the setup components referenced by the pricing components are either in the XML file or are already imported into PDC.

For example, to import a charge, you must ensure that the balance elements that it references already exist in PDC.

This section assumes that the following prerequisite data is already available in PDC:

- Service
- Event
- Account Attributes
- Service-Event Map
- Profiles

Table 6–1 lists the prerequisites for importing the pricing components into PDC. Some of the prerequisites are optional depending on your business functions and the rating system used with Oracle Communications Billing and Revenue Management (BRM).

Table 6–1 Prerequisite Data for Importing Pricing Components

Pricing Component	Prerequisite Pricing Component	Prerequisite Setup Component
Discount Filter	None	None
Time Model	None	None
APN Selector	None	Impact Categories and Zone Models
Pricing	None	RUMs, Balance Elements, G/L IDs, and Tax Codes
USC Selector	None	Impact Categories and Zone Models
Discount Trigger	None	Balance Elements
Generic Selector	None	Custom Rules and Profile Attribute Specification

Table 6–1 (Cont.) Prerequisite Data for Importing Pricing Components

Pricing Component	Prerequisite Pricing Component	Prerequisite Setup Component
Price Selector	Pricing	None
Charge	Time Models, Pricing, USC Selectors, APN Selectors, and Price Selectors	RUMs, Balance Elements, Tax Codes, G/L IDs, and Zone Models
Charge Selector	Charges	Value Maps and Zone Models
Charge Offer	Charge Selectors and Charges	RUMs, Tax Codes, Tax Suppliers, and Provisioning Tags
Discount	Pricing, Discount Trigger, and Discount Filter	Balance Elements, Tax Codes, and G/L IDs
Discount Selector	Discounts	None
Discount Offer	Discount Selectors and Discounts	Provisioning Tags
Discount Offer Exclusion	Discount Offer	None
Bundle	Charge Offers and Discount Offers	None
Best Pricing	Bundles	None
Bundle Dependency	Bundles	None
Bundle Transition	Bundles	None
Package	Bundles	Balance Elements
Package Discount Restriction	Bundles	None
Package List	Packages	None
Package Transition	Packages	None
Chargeshare	Pricing, Trigger, Filter, and Charge Selector Spec	Balance Elements, Tax Codes, and G/L IDs
Chargeshare Offer	Distribution Rate Plan	None

About the XSD Files

ImportExportPricing uses separate XSD files for importing or exporting pricing components, setup components, and profile data. The XSD files are available in their respective directories in the *PDC_home*/**apps/xsd** directory, where *PDC_home* is the directory in which the PDC software is installed. The XSD files describe the structure of the XML document. The XML file you create must comply with the structure defined in the XSD.

The XSD defines the following items for an XML file:

- The elements and attributes, their data types, and the default and fixed values for the elements and attributes.
- Elements that are child elements, and the number and order of the child elements.
- If an element can be empty or can include text.

Prerequisites for Using the ImportExportPricing Utility

To use **ImportExportPricing**, you must:

- Ensure that the XML files used by this utility conform to the format detailed in the corresponding XSD files. See "About the XSD Files" for more information.
- Ensure *PDC_home*/**apps/bin** is in your PATH environment variable.

Exporting Pricing and Setup Components from PDC

Exporting by using **ImportExportPricing** involves extracting the pricing and setup components from the PDC database into XML files.

Before exporting pricing or setup components, consider the following:

- You can export all or a subset of pricing and setup components.
- When exporting pricing components, the utility does not export the setup component referenced by pricing components. You must export the setup components explicitly.

ImportExportPricing enables you to export the following from a PDC database to an XML file:

- All pricing and setup components. See "Exporting All the Pricing and Setup Components" for more information.
- A subset of pricing and setup components. See "Exporting a Subset of Pricing and Setup Components" for more information.

Exporting All the Pricing and Setup Components

To export all the pricing and setup components to XML files, enter the following command:

```
ImportExportPricing
                          -export [FileNamePrefix]
                          -config
                          -pricing
                          -profile
                          -appsvruser ApplicationServerUserName
                          -pdcuser PdcUserName
```

See "ImportExportPricing" for a description of parameters used in this command.

For example:

```
ImportExportPricing -export MyData -config -pricing -profile -appsvruser weblogic
-pdcuser pdcuser
```

exports all the pricing components, setup components, and profile data in the PDC database in a UNIX environment to the **MyData_export_pricing.xml**, **MyData_export_config.xml**, and **MyData_export_profile.xml** files, respectively.

Exporting a Subset of Pricing and Setup Components

You can export specific pricing and setup components based on the object type, modification time, and the user who modified the data. You can also filter the exported data based on the object name.

To export a subset of pricing and setup components, enter the following command:

```
ImportExportPricing
                        -export [FileNamePrefix]
                        -config [SetupObjectType1,SetupObjectType2,…]
                        -pricing [PricingObjectType1,PricingObjectType2,…]
                        [-n "ObjectName1, ObjectName2,…"]
                        [-productSpecName ProductSpecName1,ProductSpecName2,…]
                        [-expRefs | -expAllRefs]
                        [-ma time]
                        [-mby user1, user2,… ]
                        -appsvruser ApplicationServerUserName
                        -pdcuser PdcUserName
```

See "ImportExportPricing" for a description of parameters used in this command.

See "About Supported Pricing and Setup Components" for more information about the command line representations for the pricing components and setup components.

Exporting a Subset of Pricing Components

For example:

```
ImportExportPricing -export MyPricingComponents -pricing ALTERATION_RATE_PLAN,
CHARGE_RATE_PLAN -appsvruser weblogic -pdcuser pdcuser
```

exports the ALTERATION_RATE_PLAN and CHARGE_RATE_PLAN pricing components from the PDC database in a UNIX environment to the **MyPricingComponents_export_pricing.xml** file.

Exporting Only the Setup Components that are Mastered in PDC

For example:

```
ImportExportPricing -export -brmobject
```

exports all the setup components that are mastered in PDC from the PDC database in a UNIX environment to the **MySetupComponents_export_pdc_config.xml** file.

Exporting Only the Setup Components that are Mastered in BRM

For example:

```
ImportExportPricing -export -brmobject GLID GLID
```

exports the G/L IDs into the GLID.XML file.

If you do not specify *BRMObjectType*, the utility exports all the components mastered in BRM into the XML file.

Exporting All Pricing Reference Objects

For example:

```
ImportExportPricing -export MyChargeOffers -pricing  CHARGE_OFFERING -name
ChargeOffer-RealtimeUsage, ChargeOffer-Batch -expAllRefs
```

exports the ChargeOffer-RealtimeUsage, ChargeOffer-Batch including the corresponding metadata and setup components from the PDC database in a UNIX environment to the **MyChargeOffers_export_pricing.xml**, **MyChargeOffers_export_metadata.xml** and **MyChargeOffers_export_config.xml** files.

Exporting Only the Metadata Objects

For example:

```
ImportExportPricing -export MySetupComponents -metadata
```

exports all the event, service, account, and profile attribute specifications from the PDC database in a UNIX environment to the **MySetupComponents_export_meta_config.xml** file.

Exporting Setup Components Modified After a Specific Time

For example:

```
ImportExportPricing -export MySetupComponents -config -ma
2010-01-05T19:05:09GMT+05:30 -appsvruser weblogic -pdcuser pdcuser
```

exports all the setup components modified after the 2010-01-05T19:05:09GMT+05:30 time from the PDC database in a UNIX environment to the **MySetupComponents_export_config.xml** file.

Exporting Pricing Components Modified by Specific Users

For example:

```
ImportExportPricing -export MyPricingComponents -pricing -mby User1, User2
-appsvruser weblogic -pdcuser pdcuser
```

exports the modified pricing components (by User1 and User2) from the PDC database in a UNIX environment to the **MyPricingComponents_export_pricing.xml** file.

Exporting Only Successfully Promoted and Transformed Components

For example:

```
ImportExportPricing -export MyPromotedComponents -pricing CHARGE_OFFERING -name
"COTest" -expRefs
```

exports the latest version of only the successfully promoted and transformed pricing components in **ChargeOffer COTest** along with the reference components from PDC in a UNIX environment to the **MyPromotedComponents_export_pricing.xml** file.

Exporting Only Obsoleted Components

For example:

```
ImportExportPricing -exp -pricing -expObsolete MyPricingObsoleteComponents
-appsvruser weblogic -pdcuser pdcuser
```

exports all the obsoleted pricing components (including metadata and setup components) from PDC in a UNIX environment to the **MyPricingObsoleteComponents_export_pricing.xml** file.

About Supported Pricing and Setup Components

Table 6–2 lists the type of pricing components and their representation in the command line and the XML file that you can import to or export from PDC by using **ImportExportPricing** utility.

Table 6–2 Supported Pricing Components and Their Representation in the Command Line and XML File

Pricing Component Type	Represented in the Command Line and XML File As ...
Charge	CHARGE_RATE_PLAN
Discount	ALTERATION_RATE_PLAN
Chargeshare	DISTRIBUTION_RATE_PLAN
Rollover	ROLLOVER_RATE_PLAN
Discount Selector	ALTERATION_RATE_PLAN_SELECTOR
Generic Selector	GENERIC_SELECTOR
Charge Selector	CHARGE_RATE_PLAN_SELECTOR
Price Selector	PRICE_MODEL_SELECTOR
APN Selector	APN_MAP
USC Selector	USC_MAP
Time Model	TIME_MODEL
Trigger	TRIGGER_SPEC
Filter	CHARGE_SELECTOR_SPEC
Discount Pricing	ALTERATION_POP_MODEL
Rollover Rules	ROLLOVER_POP_MODEL
Recurring Pricing	RECURRING_POP_MODEL
One-Time Pricing	ONE_TIME_POP_MODEL
Charge Offer	CHARGE_OFFERING
Discount Offer	ALTERATION_OFFERING
Chargeshare Offer	DISTRIBUTION_OFFERING
Discount Exclusion	ALTERATION_EXCLUSION
Bundle	BUNDLED_PRODUCT_OFFERING
Bundle Transition	BUNDLE_TRANSITION
Bundle Dependency	BUNDLE_DEPENDENCY
Package	PACKAGE_OBJ
Discount Restriction	PACKAGE_EXCLUSION
Package Transition	PACKAGE_TRANSITION
Package List	PACKAGE_LIST
Best Pricing	BEST_PRICING

Table 6–3 lists the type of setup components and their representation in the command line and the XML file that you can import to or export from PDC by using **ImportExportPricing**.

Table 6–3 Supported Setup Components and their Representation in the Command Line and XML File

Setup Component Type	Represented in the Command Line and XML File As ...
Value Map	ZONE_MAP
Event	EVENT_ATTRIBUTE_SPEC
Service	PRODUCT_ATTRIBUTE_SPEC
Account Attribute	CUSTOMER_ATTRIBUTE_SPEC
RUM	RUM_CONFIGURATION
Impact Category for Zones	ZONE_RESULT_CONFIGURATION
Service-Event Map	ATTRIBUTE_SPEC_MAP
Balance Element	BALANCE_ELEMENT
G/L ID	GLID
Tax Code	TAX_CODE
Tax Supplier	TAX_SUPPLIER
Geographical Zone Model	GEO_ZONE_MODEL
Standard Zone Model	STANDARD_ZONE_MODEL
Provisioning Tag	PROVISIONING_TAG
Special Day Calendar	HOLIDAY_CALENDAR
Custom Rule	CUSTOM_ANALYZER_RULE
Profile Attribute Specification	PROFILE_ATTRIBUTE_SPEC
Item Type Selector	ITEM_TYPE_SELECTOR
Business Profile	BUSINESS_PROFILE

Importing Pricing and Setup Components

Importing pricing and setup components by using the **ImportExportPricing** utility involves retrieving data from an XML file and loading it into the PDC database. For the **ImportExportPricing** utility to import pricing and setup components, the XML file must conform to the format detailed in the XSD files for pricing or setup components. See "About the XSD Files" for more information.

Before importing pricing and setup components, consider the following:

- If you import the pricing and setup components at the same time, the utility imports setup components before importing the pricing components. If the import of setup components is not successful, the utility does not import the pricing components. When importing data, you must specify at least **-pricing**, **-config**, or **-profile** in the command.
- During import, the utility modifies the objects in the database:
  - If the **-ow** parameter is used and the objects already exist in the database, the utility overwrites the existing objects.
  - If the **-ow** parameter is not used and the objects already exist in the database, the utility generates an error.
  - If the object does not exist in the database, the utility creates the object, regardless of the usage of the **-ow** parameter.

– If the **-ignoreID** parameter is used, the utility ignores the internal IDs in the specified XML file and imports the components based on the input file name.

To import pricing and setup components, enter the following command:

```
ImportExportPricing
                    -config SetupFileName
                    -pricing PricingFileName
                    -profile ProfileFileName
                    -appsvruser ApplicationServerUserName
                    -pdcuser PdcUserName
```

See "ImportExportPricing" for a description of parameters used in this command.

For example:

```
ImportExportPricing -import -config /MySetupComponents.xml -appsvruser weblogic
-pdcuser pdcuser
```

imports the setup components from the **MySetupComponents.xml** file to the PDC database in a UNIX environment.

The **ImportExportPricing** utility imports the data from the XML file into the PDC database and commits the data. If there is a failure, the **ImportExportPricing** utility rolls back the data and logs the errors in the log file.

Listing Pricing or Setup Components

You can use the **ImportExportPricing** utility to display the pricing or setup components available in the PDC database.

To list the pricing or setup components, enter the following command:

```
ImportExportPricing
                        -t config | pricing
                        -appsvruser ApplicationServerUserName
                        -pdcuser PdcUserName
```

See "ImportExportPricing" for a description of parameters used in this command.

Publishing Pricing and Setup Components

You can use the **ImportExportPricing** utility with the **-publish** and **-target** parameters to publish the components from PDC to the specified target engines, such as batch rating engine, real-time rating engine, and Elastic Charging Engine (ECE).

To publish setup components, enter the following command:

```
ImportExportPricing
                        -publish [Component] [ObjectType1,ObjectType2,…]
                        -target Target_Engine
                        -appsvruser ApplicationServerUserName
                        -pdcuser PdcUserName
```

See "ImportExportPricing" for a description of parameters used in this command.

For example:

```
ImportExportPricing -publish BALANCE_ELEMENT -target ece
```

All the balance elements in PDC are published to ECE.

Deleting Components Mastered in BRM

You can use the **ImportExportPricing** utility to delete components mastered in BRM from the PDC database if they are not being used by any other setup component or pricing component.

To delete components mastered in BRM, enter the following command:

```
ImportExportPricing
                        -d FilePath
                        -appsvruser ApplicationServerUserName
                        -pdcuser PdcUserName
```

See "ImportExportPricing" for a description of parameters used in this command.

For example:

```
ImportExportPricing -d /Temp/BRMMasteredComponentsToDelete.xml -appsvruser
weblogic -pdcuser pdcuser
```

deletes the components mastered in BRM defined in the **/Temp/BRMMasteredComponentsToDelete.xml** file in PDC in a UNIX environment.

Deleting Pricing Profiles

You can also use the **ImportExportPricing** utility to delete pricing profiles if they are not being used by any pricing component.

To delete pricing profiles, enter the following command:

```
ImportExportPricing
                        -dp FilePath
                        -appsvruser ApplicationServerUserName
                        -pdcuser PdcUserName
```

See "ImportExportPricing" for a description of parameters used in this command.

For example:

```
ImportExportPricing -dp /Temp/ProfilesToDelete.xml -appsvruser weblogic -pdcuser
pdcuser
```

deletes the pricing profile data defined in the **/Temp/ProfilesToDelete.xml** file in PDC in a UNIX environment.

Deleting Old Versions of PDC Components

You can delete old versions of PDC components that are obsolete in PDC or in the associated target engines, such as Oracle Communications Billing and Revenue Management (BRM) Elastic Charging Engine (ECE), and keep only the latest versions of successfully promoted PDC components.

To delete old versions of PDC components:

1. Navigate to the *PDC_home*/**apps/bin** location.
2. Run the following command:

```
ImportExportPricing -keep [metadata | config | pricing | all] -numVersion N
```

Where *N* is a positive integer that specifies the number of latest successful versions to be kept in the PDC database.

Note: PDC keeps *N*+1 versions of the successfully promoted components if they are available in the PDC database. For example, if you want to keep the latest two versions of successfully promoted components and there is only one version available in the PDC database, PDC keeps only that one version of successfully promoted components.

Obsoleting PDC Components

You can use the **ImportExportPricing** utility to obsolete PDC components in the Promoted status if they are not referenced by any other pricing components.

To obsolete PDC components:

1. Export the PDC component that you want to obsolete into an XML file. See "Exporting Pricing and Setup Components from PDC" for more information.
2. In the XML file, set the **<obsolete>** element for the PDC component to **true**:

   ```
   <obsolete>true</obsolete>
   ```

3. Import the PDC component into the PDC database with the **-ow** parameter. See "Importing Pricing and Setup Components" for more information.

   For example:

   ```
   ImportExportPricing -import -pricing export_pricing.xml -ow
   ```

   The imported PDC component is obsoleted.

   Note: If you want to reuse an obsoleted PDC component, you must set the **<obsolete>** element for that component to **false** and import it into the PDC database.

Creating the Pricing and Setup Components by Using XML Files

You can use XML files to create pricing and setup components in the PDC database.

Creating the pricing and setup components by using the XML files involves:

- Creating the XML file containing the pricing or setup components. See "Creating the XML Files" for more information.
- Loading the data in the XML file into PDC by using **ImportExportPricing**. See "Importing Pricing and Setup Components" for more information.

Creating the XML Files

You create the XML file containing the pricing or setup components in an XML editor or a text editor.

You can create:

- One consolidated XML file containing all the pricing or setup components
- Separate XML files for each type of pricing or setup component

The XML files that you create must conform to the format detailed in the XSD files for pricing or setup components. See "About the XSD Files" for more information.

Some pricing components have dependencies on other pricing and setup components. Before loading such pricing components, ensure that the prerequisite data is available in the PDC database. See Table 6–1, " Prerequisite Data for Importing Pricing Components", for more information.

If you create XML files for specific pricing or setup components, you must load the XML files in the order based on the dependencies. See "About Dependencies between Pricing and Setup Components" for more information.

7

About Enriching Event Definitions

This chapter describes how to enrich the event definitions in Oracle Communications Pricing Design Center (PDC) with information required by Oracle Communications Billing and Revenue Management (BRM) Elastic Charging Engine (ECE) to process usage charging requests from the network.

You should have a basic understanding of the following:

- PDC. See "About Pricing Design Center" for more information.
- **ImportExportPricing**. See "Importing and Exporting Pricing and Setup Components" for more information.
- Creating services and events in BRM. See the discussion about creating storable classes for services and events in *BRM Developer's Guide*.

To use the **ImportExportPricing** utility, you must also understand the following:

- Extensible markup language (XML) programming
- XML schema definition (XSD)

Overview of Enriching Event Definitions

An event definition is a set of attributes pertaining to a type of transaction. You define the event definitions in BRM Development Center and then enrich them in PDC with additional information that is required for ECE. Enriched event definitions are used by ECE for rating events.

You do the following to enrich an event definition:

- Add **USER_IDENTITY** and **CALLED_ID** attributes, which are required for ECE to process custom events.
- Add transient attributes if required. These are attributes that are used specifically for usage charging and they need not be stored persistently in the event for rerating. For example, you can add the CELL_ID attribute in the event definition to retrieve cell load information for usage charging, but the CELL_ID value need not be stored persistently in the event.
- Add charging operation types to be used for usage charging (for example, initiating a session-based charge for an event). See "About Charging Operation Types" for more information.
- Define the network mapping for the events; for example, if you are using Diameter Gateway for network integration for online charging, you map the request attributes that are received from the Diameter network to event attributes. This mapping is used by the ECE charging client at runtime to populate the event for

charging. See "About Mapping Event Attributes to Network Attributes" for more information.

About Charging Operation Types

A charging operation type is a type of operation that ECE uses for usage charging; for example, the **Initiate** operation type is used for initiating a session-based charge and the **Price_Enquiry** operation type is used for enquiring a price information. Each charging operation type requires a set of event attributes to be specified.

When you add the charging operation type in the event definition, you associate the charging operation type with the group of event attributes that are necessary to perform the charging operation on the event. For example, the attributes for session-based charging are associated with the **Initiate**, **Update**, and **Terminate** operation types. The attributes for event-based charging are associated with the **Debit**, **Refund**, and **Price_Enquiry** charging operation types.

Table 7–1 lists the charging operation types that are used in ECE.

Table 7–1 Charging Operation Types Used in ECE

Operation Type	Description
Initiate	Initiates a session-based charging operation.
Update	Modifies a session-based charging operation.
Terminate	Terminates a single non-session-based charging operation.
Cancel	Cancels a session-based charging operation.
Refund_Amount	Refunds a specific amount to a specific balance resource.
Refund_Unit	Refunds a calculated amount, based on units consumed, to the impacted resources.
Debit_Amount	Debits a specific amount to a specific balance resource.
Debit_Unit	Debits a calculated amount, based on Units consumed, to the impacted resources.
Price_Enquiry	Generates a price estimation without any balance reservations occurring. It is used when there isn't a high probability of receiving a charging request. For example, Price_Enquiry might be called to get the price of an event charge to display in a content portal.
Start_Accounting	Starts tracking usage without incurring balance impacts.
Update_Accounting	Continues tracking usage without incurring balance impacts.

See "Enriching Event Definitions" for information on how to add charging operation types and associating them with the event attributes.

About Charging Operation Type Versions

A set of charging operation types has a version, which is indicated by the **<externalVersion>** element.

When you modify the charging operation types in PDC, use the same version number to replace the existing set of charging operation types with the updated set of charging operation types. The updated charging operation types are published to ECE.

You increase the charging operation type version number if you want to use both the existing set of charging operation types and the updated set of charging operation

types. For example, when incrementally upgrading the network mediation software program (client), you may want both the set of charging operation types active at the same time. Both old and new versions of the charging operation types are stored in PDC and published to ECE.

You must update the version number in your mediation specification file to associate the usage request builder with the event definition to which it applies.

The following is the syntax for the charging operation type version:

```
<opVersion>
   <externalVersion>version_number</externalVersion>
   <opTypes>
      ...
   </opTypes>
   <opTypes>
      ...
   </opTypes>
</opVersion>
```

where *version_number* is the version number for the set of charging operation types.

About Mapping Event Attributes to Network Attributes

Event attributes that are used for charging can be mapped to their respective attributes in the usage requests received from the network; such as the Diameter network.

The event attribute to network attribute mapping is used to populate the event attributes of usage requests at run time. For example, if you are using Diameter Gateway for online charging, the event attribute to network attribute mapping is used to populate the event attributes of usage requests with the data from your incoming Diameter request.

The following example shows the event attribute, **CALLED_ID**, that is mapped to the network attribute, **Service-Information.PS-Information.Called-Station-Id**, which is used to populate the **CALLED_ID** value with the data from the incoming Diameter request:

```
<simpleAttributeItem>
       <name>CALLED_ID</name>
       <description>CALLED_ID</description>
       <type>STRING</type>
       <persistedName>my_called_id_c</persistedName>
       <attributeItemSupportedBy>
            <targetApplicationSpecName>Pricing</targetApplicationSpecName>
            <targetApplicationSpecName>Convergent
Charging</targetApplicationSpecName>
       </attributeItemSupportedBy>

<networkAttributeItem>Service-Information.PS-Information.Called-Station-Id</networ
kAttributeItem>
</simpleAttributeItem>
```

See the discussion about Diameter Gateway in ECE documentation for more information.

Enriching Event Definitions

Before enriching the event definition, ensure the following:

- A new class (for example, **/cloud**) or a subclass (for example, **/event/cloud/library**) for the new event is created in Development Center. See the discussion about creating storable classes for services and events in *BRM Developer's Guide* for more information.
- A substruct in the event class or subclass; for example, USAGE_INFO, is created in Development Center. See the discussion about substructure in *BRM Developer's Guide* for more information.
- Your custom fields are added to the substruct in the event subclass.

> **Note:** Ensure that the fields for storing the user identity and called ID (if you are using zoning in your charge offer) are added to the substruct in the event class.

- The event is mapped to the corresponding service (for example, **service/cloud/library**) by editing the **pin_event_map** file and then running the **load_event_map** utility. See the discussion about mapping event types to services in *BRM Setting Up Pricing and Rating* for more information.
- The service and event definitions are synchronized with PDC by running the **SyncPDC** utility. See "Synchronizing Setup Components" for more information.

To enrich an event definition:

1. Export the event definition into an XML file by running the following command:

   ImportExportPricing -export*FileNamePrefix* **-metadata EVENT_ATTRIBUTE_SPEC -n "***eventname***"**

   where *eventname* is the name of the event to be exported into an XML file.

   For example:

   ```
   ImportExportPricing -export EventCloudLibrary -metadata EVENT_ATTRIBUTE_SPEC -n
   EventCloudLibrary
   ```

   Exports the definition for EventCloudLibrary to the **EventCloudLibrary_export_metadata.xml** file.

2. Open the event definition file in a text editor.

   The following example shows the event definition for the EventCloudLibrary event:

   ```
   <eventAttributeSpec>
      <name>EventCloudLibrary</name>
      <description>EventCloudLibrary</description>
      <internalId>c06de0cc-6105-4cd3-92c9-ad8011e5fe7b</internalId>
      <priceListName>Default</priceListName>
      <obsolete>false</obsolete>
      <attributeSpecSupportedBy>
         <targetApplicationSpecName>Billing</targetApplicationSpecName>
         <targetApplicationSpecName>Pricing</targetApplicationSpecName>
         <targetApplicationSpecName>Convergent
   Charging</targetApplicationSpecName>
         <targetApplicationSpecName>Realtime Charging</targetApplicationSpecName>
      </attributeSpecSupportedBy>
      <status>ENABLED</status>
      <className>/event/cloud/library</className>
      <complexAttributeItem>
   ```

```
      <name>USAGE_INFO</name>
      <description>USAGE_INFO</description>
      <type>STRUCT</type>
      <persistedName>cloudlibrary_usage_info_t</persistedName>
      <simpleAttributeItem>
         <name>MY_IDENTITY</name>
         <description>MY_IDENTITY</description>
         <type>STRING</type>
         <persistedName>my_identity_c</persistedName>
         <attributeItemSupportedBy>
            <targetApplicationSpecName>Pricing</targetApplicationSpecName>
            <targetApplicationSpecName>ConvergentCharging</targetApplicationSpe
cName>
         </attributeItemSupportedBy>
      </simpleAttributeItem>
      <simpleAttributeItem>
         <name>MY_CALLED_ID</name>
         <description>MY_CALLED_ID</description>
         <type>STRING</type>
         <persistedName>my_called_id_c</persistedName>
         <attributeItemSupportedBy>
           <targetApplicationSpecName>Pricing</targetApplicationSpecName>
           <targetApplicationSpecName>ConvergentCharging</targetApplicationSpec
Name>
         </attributeItemSupportedBy>
      </simpleAttributeItem>
      <simpleAttributeItem>
         <name>SERVICE_PROVIDER</name>
         <description>SERVICE_PROVIDER_C</description>
         <type>STRING</type>
         <persistedName>service_provider_c</persistedName>
         <attributeItemSupportedBy>
            <targetApplicationSpecName>Pricing</targetApplicationSpecName>
            <targetApplicationSpecName>Convergent
Charging</targetApplicationSpecName>
         </attributeItemSupportedBy>
         <length>60</length>
      </simpleAttributeItem>
      <simpleAttributeItem>
         <name>SERVICE_STATUS</name>
         <description>SERVICE_STATUS_C</description>
         <type>STRING</type>
         <persistedName>service_status_c</persistedName>
         <attributeItemSupportedBy>
            <targetApplicationSpecName>Pricing</targetApplicationSpecName>
            <targetApplicationSpecName>ConvergentCharging</targetApplicationSpe
cName>
         </attributeItemSupportedBy>
         <length>60</length>
      </simpleAttributeItem>
      <simpleAttributeItem>
         <name>SERVICE_TYPE</name>
         <description>SERVICE_TYPE_C</description>
         <type>STRING</type>
         <persistedName>service_type_c</persistedName>
         <attributeItemSupportedBy>
            <targetApplicationSpecName>Pricing</targetApplicationSpecName>
            <targetApplicationSpecName>ConvergentCharging</targetApplicationSpe
cName>
         </attributeItemSupportedBy>
```

```
        <length>60</length>
      </simpleAttributeItem>
   </complexAttributeItem>
   <eventType>SESSION_USAGE</eventType>
   <baseEventSpec>EventCloud</baseEventSpec>
   <virtualColumn>false</virtualColumn>
   <unit>MONTHS</unit>
   <frequency>0</frequency>
</eventAttributeSpec>
```

3. Add USER_IDENTITY and CALLED_ID (if you are using zoning in your charge offer) attributes as simple attribute items immediately under the **<className>** element by doing the following:

   a. Search for the following element:

   ```
   <className>EventClassName</className>
   ```

   where *EventClassName* is the class name of the event.

   For example:

   ```
   <className>/event/cloud/library</className>
   ```

   b. Add the **USER_IDENTITY** and **CALLED_ID** attributes as simple attribute items immediately under the **<className>** element:

   ```
   <className>/event/cloud/EventUsage</className>
   <simpleAttributeItem>
      <name>USER_IDENTITY</name>
      <description>USER_IDENTITY</description>
      <type>STRING</type>
      <persistedName>PersistedNameofAttribute</persistedName>
      <attributeItemSupportedBy>
         <targetApplicationSpecName>Pricing</targetApplicationSpecName>
         <targetApplicationSpecName>ConvergentCharging</targetApplicationSpecN
   ame>
      </attributeItemSupportedBy>
   </simpleAttributeItem>
   <simpleAttributeItem>
      <name>CALLED_ID</name>
      <description>CALLED_ID</description>
      <type>STRING</type>
      <persistedName>PersistedNameofAttribute</persistedName>
      <attributeItemSupportedBy>
           <targetApplicationSpecName>Pricing</targetApplicationSpecName>
           <targetApplicationSpecName>ConvergentCharging</targetApplicationSpe
   cName>
      </attributeItemSupportedBy>
   </simpleAttributeItem>
   ```

   where *PersistedNameofAttribute* specifies the name of the BRM database table and column associated with the attributes used for storing the user identity and called ID; for example, cloudlibrary_usage_info_t.my_identity_c and cloudlibrary_usage_info_t.my_called_Id_c.

4. Add the event attributes that you use in ratable usage metric (RUM) expressions for **Initiate** or **Update** charging operations as top-level attributes under the **<className>** element.

   The following example shows the event attributes used in the RUM expressions for charging the EventCloudLibrary event:

```
      <simpleAttributeItem>
         <name>DURATION</name>
         <description>DURATION</description>
         <type>UNIT_VALUE</type>
         <unitType>TimeUnit</unitType>
         <attributeItemSupportedBy>
            <targetApplicationSpecName>Pricing</targetApplicationSpecName>
            <targetApplicationSpecName>ConvergentCharging</targetApplicationSpe
cName>
         </attributeItemSupportedBy>
      </simpleAttributeItem>
      <simpleAttributeItem>
         <name>SPECIFIC_UNIT</name>
         <description>SPECIFIC_UNIT</description>
         <type>UNIT_VALUE</type>
         <unitType>Occurrence</unitType>
         <attributeItemSupportedBy>
            <targetApplicationSpecName>Pricing</targetApplicationSpecName>
            <targetApplicationSpecName>ConvergentCharging</targetApplicationSpe
cName>
         </attributeItemSupportedBy>
      </simpleAttributeItem>
```

5. Add REQUESTED_UNITS and USED_UNITS attributes as complex attribute items in the file:

```
   <complexAttributeItem>
      <name> REQUESTED_UNITS</name>
      <description>REQUESTED_UNITS</description>
      <type>STRUCT</type>
      <attributeItemSupportedBy>
         <targetApplicationSpecName>Pricing</targetApplicationSpecName>
         <targetApplicationSpecName>Convergent
Charging</targetApplicationSpecName>
      </attributeItemSupportedBy>
   </complexAttributeItem>
   <complexAttributeItem>
      <name>USED_UNITS</name>
      <description>USED_UNITS</description>
      <type>STRUCT</type>
      <attributeItemSupportedBy>
         <targetApplicationSpecName>Pricing</targetApplicationSpecName>
         <targetApplicationSpecName>ConvergentCharging</targetApplicationSpecNa
me>
      </attributeItemSupportedBy>
   </complexAttributeItem>
```

6. Copy the event attributes you added in step 4 (if applicable) and paste them in both REQUESTED_UNITS and USED_UNITS complex attribute items:

   For example:

```
   <complexAttributeItem>
      <name>REQUESTED_UNITS</name>
      <description>REQUESTED_UNITS</description>
      <type>STRUCT</type>
      <attributeItemSupportedBy>
         <targetApplicationSpecName>Pricing</targetApplicationSpecName>
         <targetApplicationSpecName>ConvergentCharging</targetApplicationSpecNa
me>
      </attributeItemSupportedBy>
```

```
      <simpleAttributeItem>
         <name>DURATION</name>
         <description>DURATION</description>
         <type>UNIT_VALUE</type>
         <unitType>TimeUnit</unitType>
         <attributeItemSupportedBy>
            <targetApplicationSpecName>Pricing</targetApplicationSpecName>
            <targetApplicationSpecName>ConvergentCharging</targetApplicationSpe
cName>
         </attributeItemSupportedBy>
      </simpleAttributeItem>
      <simpleAttributeItem>
         <name>SPECIFIC_UNIT</name>
         <description>SPECIFIC_UNIT</description>
         <type>UNIT_VALUE</type>
         <unitType>Occurrence</unitType>
         <attributeItemSupportedBy>
            <targetApplicationSpecName>Pricing</targetApplicationSpecName>
            <targetApplicationSpecName>ConvergentCharging</targetApplicationSpe
cName>
         </attributeItemSupportedBy>
      </simpleAttributeItem>
   </complexAttributeItem>
   <complexAttributeItem>
      <name>USED_UNITS</name>
      <description>USED_UNITS</description>
      <type>STRUCT</type>
         <attributeItemSupportedBy>
            <targetApplicationSpecName>Pricing</targetApplicationSpecName>
            <targetApplicationSpecName>ConvergentCharging</targetApplicationSpe
cName>
         </attributeItemSupportedBy>
      <simpleAttributeItem>
         <name>DURATION</name>
         <description>DURATION</description>
         <type>UNIT_VALUE</type>
         <unitType>TimeUnit</unitType>
         <attributeItemSupportedBy>
            <targetApplicationSpecName>Pricing</targetApplicationSpecName>
            <targetApplicationSpecName>ConvergentCharging</targetApplicationSpe
cName>
         </attributeItemSupportedBy>
      </simpleAttributeItem>
      <simpleAttributeItem>
         <name>SPECIFIC_UNIT</name>
         <description>SPECIFIC_UNIT</description>
         <type>UNIT_VALUE</type>
         <unitType>Occurrence</unitType>
         <attributeItemSupportedBy>
            <targetApplicationSpecName>Pricing</targetApplicationSpecName>
            <targetApplicationSpecName>ConvergentCharging</targetApplicationSpe
cName>
         </attributeItemSupportedBy>
      </simpleAttributeItem>
   </complexAttributeItem>
```

7. (Optional) Add transient attributes if required for usage charging:

   For example:

```
<simpleAttributeItem>
```

```
<name>CELL_ID</name>
<description>CELL_ID</description>
<type>STRING</type>
<attributeItemSupportedBy>
<targetApplicationSpecName>Pricing</targetApplicationSpecName>
<targetApplicationSpecName>ConvergentCharging</targetApplicationSpecName>
</attributeItemSupportedBy>
</simpleAttributeItem>
```

8. Add the charging operation types that you intend to use for usage charging and associate them with the group of event attributes that are necessary to perform the charging operation (for example: **Initiate**, **Terminate**, **Update**, and **Cancel**).

   The syntax for adding charging operation types is as follows:

```
<opTypes>
   <opType>ChargingOperationType</opType>
   <action>Input</action>
   <item>
     <attributeItemName>EventAttribute</attributeItemName>
     <optional>[true|false]</optional>
   </item>
</opTypes>
```

   where:

   - *ChargingOperationType* is the charging operation type to be used. See Table 7–1 for the list of charging operation types that are used in ECE.
   - *EventAttribute* is the event attribute that is used as input for performing the charging operation.

   The following example shows the charge operation type and required attributes to initiate a charging session for the EventCloudLibrary event:

```
<opTypes>
   <opType>Initiate</opType>
   <action>INPUT</action>
   <item>
      <attributeItemName>START_T</attributeItemName>
      <optional>false</optional>
   </item>
   <item>
      <attributeItemName>DURATION</attributeItemName>
      <optional>true</optional>
   </item>
   <item>
      <attributeItemName>END_T</attributeItemName>
      <optional>true</optional>
   </item>
   <item>
      <attributeItemName>USAGE_INFO</attributeItemName>
      <optional>false</optional>
   </item>
   <item>
      <attributeItemName>USAGE_INFO.SERVICE_TYPE</attributeItemName>
      <optional>false</optional>
   </item>
   <item>
      <attributeItemName>USAGE_INFO.SERVICE_STATUS</attributeItemName>
      <optional>true</optional>
   </item>
```

```
            <item>
               <attributeItemName>USAGE_INFO.SERVICE_PROVIDER</attributeItemName>
               <optional>true</optional>
            </item>
            <item>
               <attributeItemName>USED_UNITS</attributeItemName>
               <optional>false</optional>
            </item>
            <item>
               <attributeItemName>USED_UNITS.DURATION</attributeItemName>
               <optional>true</optional>
            </item>
            <item>
               <attributeItemName>USED_UNITS.SPECIFIC_UNIT</attributeItemName>
               <optional>true</optional>
            </item>
            <item>
               <attributeItemName>REQUESTED_UNITS</attributeItemName>
               <optional>true</optional>
            </item>
            <item>
               <attributeItemName>REQUESTED_UNITS.DURATION</attributeItemName>
               <optional>true</optional>
            </item>
            <item>
               <attributeItemName>REQUESTED_UNITS.SPECIFIC_
UNIT</attributeItemName>
               <optional>true</optional>
            </item>
         </opTypes>
```

9. (Optional) Update the charging operation type version number. See "About Charging Operation Type Versions" for more information.
10. Define the network mapping for the events used for charging by doing the following:
    a. Search for the event attribute that you want to map to the network attribute.
    b. Add the following entry:

       <networkAttributeItem>*NetworkAttribute***</networkAttributeItem>**

       where *NetworkAttribute* is the attribute of requests received from the network.

    The following example shows the network mapping for the EventCloudLibrary event:

```
         <simpleAttributeItem>
            <name>MY_CALLED_ID</name>
            <description>MY_CALLED_ID</description>
            <type>STRING</type>
            <persistedName>my_called_id_c</persistedName>
            <attributeItemSupportedBy>
              <targetApplicationSpecName>Pricing</targetApplicationSpecName>
              <targetApplicationSpecName>ConvergentCharging</targetApplicationSpec
Name>
            </attributeItemSupportedBy>

<networkAttributeItem>Service-Information.PS-Information.Called-Station-Id</net
workAttributeItem>
         </simpleAttributeItem>
```

```
...
   <complexAttributeItem>
      <name> REQUESTED_UNITS </name>
      <description>REQUESTED_UNITS</description>
      <type>STRUCT</type>
      <attributeItemSupportedBy>
         <targetApplicationSpecName>Pricing</targetApplicationSpecName>
         <targetApplicationSpecName>Convergent
Charging</targetApplicationSpecName>
      </attributeItemSupportedBy>
      <simpleAttributeItem>
         <name>DURATION</name>
         <description>DURATION</description>
         <type>UNIT_VALUE</type>
         <unitType>TimeUnit</unitType>
         <attributeItemSupportedBy>
            <targetApplicationSpecName>Pricing</targetApplicationSpecName>
            <targetApplicationSpecName>ConvergentCharging</targetApplicationSpe
cName>
         </attributeItemSupportedBy>
         <networkAttributeItem>Multiple-Services-Credit-Control.Requested-Servi
ce-Unit.CC-Time</networkAttributeItem>
      </simpleAttributeItem>
      <simpleAttributeItem>
         <name>SPECIFIC_UNIT</name>
         <description>SPECIFIC_UNIT</description>
         <type>UNIT_VALUE</type>
         <unitType>Occurrence</unitType>
         <attributeItemSupportedBy>
            <targetApplicationSpecName>Pricing</targetApplicationSpecName>
            <targetApplicationSpecName>ConvergentCharging</targetApplicationSpe
cName>
         </attributeItemSupportedBy>
         <networkAttributeItem>Multiple-Services-Credit-Control.Requested-Servi
ce-Unit.CC-Service-Specific-Units</networkAttributeItem>
      </simpleAttributeItem>
   </complexAttributeItem>
   <complexAttributeItem>
      <name>USED_UNITS</name>
      <description>USED_UNITS</description>
      <type>STRUCT</type>
         <attributeItemSupportedBy>
            <targetApplicationSpecName>Pricing</targetApplicationSpecName>
            <targetApplicationSpecName>ConvergentCharging</targetApplicationSpe
cName>
         </attributeItemSupportedBy>
      <simpleAttributeItem>
         <name>DURATION</name>
         <description>DURATION</description>
         <type>UNIT_VALUE</type>
         <unitType>TimeUnit</unitType>
         <attributeItemSupportedBy>
            <targetApplicationSpecName>Pricing</targetApplicationSpecName>
            <targetApplicationSpecName>ConvergentCharging</targetApplicationSpe
cName>
         </attributeItemSupportedBy>
         <networkAttributeItem>Multiple-Services-Credit-Control.Used-Service-Un
it.CC-Time</networkAttributeItem>
      </simpleAttributeItem>
      <simpleAttributeItem>
```

```
         <name>SPECIFIC_UNIT</name>
         <description>SPECIFIC_UNIT</description>
         <type>UNIT_VALUE</type>
         <unitType>Occurrence</unitType>
         <attributeItemSupportedBy>
            <targetApplicationSpecName>Pricing</targetApplicationSpecName>
            <targetApplicationSpecName>ConvergentCharging</targetApplicationSpe
cName>
         </attributeItemSupportedBy>
         <networkAttributeItem>Multiple-Services-Credit-Control.Used-Service-Un
it.CC-Service-Specific-Units</networkAttributeItem>
      </simpleAttributeItem>
   </complexAttributeItem>
   <complexAttributeItem>
      <name>USAGE_INFO</name>
      <description>USAGE_INFO</description>
      <type>STRUCT</type>
      <persistedName>cloudlibrary_usage_info_t</persistedName>
      <simpleAttributeItem>
         <name>SERVICE_PROVIDER</name>
         <description>SERVICE_PROVIDER_C</description>
         <type>STRING</type>
         <persistedName>service_provider_c</persistedName>
         <attributeItemSupportedBy>
            <targetApplicationSpecName>Pricing</targetApplicationSpecName>
            <targetApplicationSpecName>ConvergentCharging</targetApplicationSpe
cName>
         </attributeItemSupportedBy>
         <networkAttributeItem>Multiple-Services-Credit-Control.Used-Service-Un
it.CC-Time</networkAttributeItem>
         <length>60</length>
      </simpleAttributeItem>
      <simpleAttributeItem>
         <name>SERVICE_STATUS</name>
         <description>SERVICE_STATUS_C</description>
         <type>STRING</type>
         <persistedName>service_status_c</persistedName>
         <attributeItemSupportedBy>
            <targetApplicationSpecName>Pricing</targetApplicationSpecName>
            <targetApplicationSpecName>ConvergentCharging</targetApplicationSpe
cName>
         </attributeItemSupportedBy>
         <networkAttributeItem>Multiple-Services-Credit-Control.Used-Service-Un
it.CC-Time</networkAttributeItem>
         <length>60</length>
      </simpleAttributeItem>
      <simpleAttributeItem>
         <name>SERVICE_TYPE</name>
         <description>SERVICE_TYPE_C</description>
         <type>STRING</type>
         <persistedName>service_type_c</persistedName>
         <attributeItemSupportedBy>
            <targetApplicationSpecName>Pricing</targetApplicationSpecName>
            <targetApplicationSpecName>ConvergentCharging</targetApplicationSpe
cName>
         </attributeItemSupportedBy>
         <networkAttributeItem>Multiple-Services-Credit-Control.Used-Service-Un
it.CC-Time</networkAttributeItem>
         <length>60</length>
      </simpleAttributeItem>
```

```
</complexAttributeItem>
```

11. Save and close the file.
12. Import the XML file into PDC by running the following command:

    ImportExportPricing -import -metadata *FileNamePrefix*_**export_metadata.xml -ow**

    For example:

    ```
    ImportExportPricing -import -metadata EventCloudLibrary_export_metadata.xml -ow
    ```

    The EventCloudLibrary event definition is loaded into PDC. The ECE Pricing Updater publishes the enriched event definition to ECE.
13. Update the version number in your mediation specification file to associate the usage request builder with the event definition to which it applies. See the discussion about editing the mediation specification file in *BRM Elastic Charging Engine Implementation Guide* for more information.

You may need to restart your charging client to receive usage requests that adhere to the enriched event definition.

Editing Event Definitions

To edit an event definition, update the event definition in PDC by using the **ImportExportPricing** utility. When you edit an event definition, ensure that you use the same event type and product type, and update the charging operation type version as appropriate. See "About Charging Operation Type Versions" for more information.

When you create your usage request builder, you must associate it with the product type, event type, and charging operation type version in the event definition to which it applies. You may need to restart your charging client to receive usage requests that adhere to the enriched event definition.

8

About Creating Pricing Components by Using the PDC Web Service

This chapter describes how to use the Oracle Communications Pricing Design Center (PDC) Web service to create or modify the pricing components in PDC using the pricing components that are created or modified in an external application.

You should have a basic understanding of the following:

- PDC. See "About Pricing Design Center" for more information.
- Product offerings. See "About Creating Product Offerings" for more information.

To use the PDC Web service, you should be familiar with Web services and Oracle WebLogic Server. See your Oracle WebLogic Server documentation for more information.

You should also be familiar with the following:

- Extensible markup language (XML) programming
- XML schema definition (XSD)

About the PDC Web Service API

The PDC Web service application programming interface (API) allows other Oracle applications as well as third-party applications to interact with PDC. The PDC Web service API enables you to create or modify the pricing components in PDC using the pricing components that are created or modified in an external application.

To use the PDC Web service API to create or modify the pricing components in PDC, the external application must call the PDC Web service API through a custom client application. You can use any language that supports Web services to access the API; for example, Java.

Prerequisites for Using the PDC Web Service

To use the PDC Web service, you must:

- Ensure that the SSL port for the WebLogic Server domain is enabled. See the discussion about enabling the SSL port for the WebLogic Server domain in *PDC Installation and System Administration Guide* for more information.
- Ensure that the following system properties are set in WebLogic Server:
  - weblogic.security.TrustKeyStore
  - weblogic.security.CustomTrustKeyStoreFileName

  - weblogic.security.CustomTrustKeyStorePassPhrase
  - weblogic.security.CustomTrustKeyStoreType

  See the discussion about setting the system properties in the WebLogic Server Administration Console Help for more information.

- Ensure that the following BindingProvider properties are set in WebLogic Server:
  - BindingProvider.USERNAME_PROPERTY
  - BindingProvider.PASSWORD_PROPERTY

  See the discussion about setting the BindingProvider properties in the WebLogic Server Administration Console Help for more information.

About Creating or Modifying Pricing Components by Using the PDC Web Service

Creating or modifying pricing components by using the PDC Web service involves:

- Creating the XML file containing the pricing components. See "About Creating the XML Files" for more information.
- Using the PDC Web service to create new pricing components or modify the promoted pricing components as defined in the XML file in PDC. See "About Using the PDC Web Service to Create or Modify Pricing Components" for more information.

About Creating the XML Files

You create the XML file containing the pricing components in an XML editor or a text editor.

You can create:

- One consolidated XML file containing all the pricing components
- Separate XML files for each type of pricing component (for example, charge offers, discount offers, packages, and so on)

The XML files that you create must conform to the format detailed in the XSD files for pricing components.

The PDC Web service uses the **PricingGateway.xsd** file for creating or modifying pricing components in PDC. The XSD files describe the structure of the XML document. The XML file you create must comply with the structure defined in the XSD. The **PricingGateway.xsd** file is available at:

http://*hostName***:***sslPortNumber***/pdc/PricingGatewayPort?xsd=1**

where:

- *hostName* is the host name of the machine on which PDC is deployed.
- *sslPortNumber* is the SSL port number of the domain on which PDC is deployed.

About Using the PDC Web Service to Create or Modify Pricing Components

The PDC APIs are exposed as Web service operations through the **PricingGateway.wsdl** file. The **PricingGateway.wsdl** file defines the Web service that can be called as well as the attributes required to call a specific operation.

You can use the PDC Web service to do the following:

- Create the pricing components defined in an XML file in PDC. See "CreatePricing" for more information.
- Modify the promoted pricing components in PDC as defined in an XML file. See "ModifyPricing" for more information.
- Create the pricing components defined in an XML file in PDC and publish the pricing components to a billing system, such as Oracle Communications Billing and Revenue Management (BRM). See "CreatePricingAndSubmit" for more information.
- Modify the promoted pricing components in PDC as defined in an XML file and publish the pricing components to a billing system. See "ModifyPricingAndSubmit" for more information.

Use the following sample code to call the PDC Web service:

```
PricingGatewayPortType pricingGatewayPortType
=(PricingGatewayPortType)Service.getPort(new
QName("urn:PricingGateWay","PricingGatewayPort"),
PricingGatewayPortType.class);

Map<String, Object> rc =
((BindingProvider)pricingGatewayPortType).getRequestContext();
rc. put(BindingProvider.ENDPOINT_ADDRESS_PROPERTY,
http://hostName:sslPortNumber/pdc/PricingGatewayPort?WSDL
));
rc.put(BindingProvider.USERNAME_PROPERTY,pdcUserName);
rc.put(BindingProvider.PASSWORD_PROPERTY,pdcUserPassword);

UserContextType userContext = new UserContextType();
userContext.setUserid("pdcUserName");

PricingInputXMLType pricingInputXMLType = new PricingInputXMLType();
pricingInputXMLType.setUserContext(userContext);
ByteArrayOutputStream bs = new ByteArrayOutputStream();
ZipOutputStream out = new ZipOutputStream(bs);
        FileInputStream fl = null;
        try {
            fl = new FileInputStream(new File(pricing.xml));
            // Set the compression ratio
            out.setLevel(Deflater.BEST_COMPRESSION);
            ZipEntry ze = new ZipEntry(PDC);
            out.putNextEntry(ze);
            byte[] data = new byte[BUFFER_SIZE];
            int count = 0;
            BufferedInputStream in = new BufferedInputStream(fl);
            while ((count = in.read(data, 0, BUFFER_SIZE)) != -1){
                out.write(data, 0, count);
            }
        } catch (Exception e) {
            throw e;
        } finally {
            if (fl != null) {
                fl.close();
            }
            if (bs != null) {
                bs.flush();
            }
            if (out != null) {
```

```
                out.flush();
                out.close();
            }
        }
        byte[] bytesToRet = null;
        if (bs != null) {
            bytesToRet = bs.toByteArray();
        }
pricingInputXMLType.setXmlBinaryString(bytesToRet);

PDCResponseType pDCResponseType =
pricingGatewayPortType.createPricingAndSubmit(pricingInputXMLType);
```

CreatePricing

This Web service operation validates the input XML by comparing the XML fields and values against the values in the **PricingGateway.xsd** file and the rules for each type of pricing component. If the validation is successful, it retrieves the data from the XML file and creates the pricing components in PDC.

The **CreatePricing** operation does not publish the pricing components to a billing system.

Syntax

```
public oracle.communications.brm.pdc.server.service.types.PDCResponseType
createPricing(oracle.communications.brm.pdc.server.service.types.PricingInputXMLTy
pe param) throws
oracle.communications.brm.pdc.server.service.PricingExceptionResponse;
```

ModifyPricing

This Web service operation validates the input XML by comparing the XML fields and values against the values in the **PricingGateway.xsd** file and the rules for each type of pricing component. If the validation is successful, it retrieves the data from the XML file and updates the existing pricing components in PDC.

The **ModifyPricing** operation does not publish the pricing components to a billing system.

Syntax

```
public oracle.communications.brm.pdc.server.service.types.PDCResponseType
modifyPricing(oracle.communications.brm.pdc.server.service.types.PricingInputXMLTy
pe param) throws
oracle.communications.brm.pdc.server.service.PricingExceptionResponse;
```

CreatePricingAndSubmit

This Web service operation validates the input XML by comparing the XML fields and values against the values in the **PricingGateway.xsd** file and the rules for each type of pricing component. If the validation is successful, it retrieves the data from the XML file, creates the pricing components in PDC, and publishes the pricing components to the billing system.

Syntax

```
public oracle.communications.brm.pdc.server.service.types.PDCResponseType
createPricingAndSubmit(oracle.communications.brm.pdc.server.service.types.PricingI
nputXMLType param) throws
oracle.communications.brm.pdc.server.service.PricingExceptionResponse;
```

ModifyPricingAndSubmit

This Web service operation validates the input XML by comparing the XML fields and values against the values in the **PricingGateway.xsd** file and the rules for each type of pricing component. If the validation is successful, it retrieves the data from the XML file, updates the existing pricing components in PDC, and publishes the pricing components to the billing system.

Syntax

```
public oracle.communications.brm.pdc.server.service.types.PDCResponseType
modifyPricingAndSubmit(oracle.communications.brm.pdc.server.service.types.PricingI
nputXMLType param) throws
oracle.communications.brm.pdc.server.service.PricingExceptionResponse;
```

9

Replicating Data Between PDC Systems

This chapter describes how to replicate data from one Oracle Communications Pricing Design Center (PDC) system to another, such as from a development system to a test system.

> **Important:** You can replicate data between databases of the same patch set only. For example, you can replicate data from a PDC 11.2 database to another PDC 11.2 database only.

Before replicating the data, you must be familiar with the following:

- **PDC**. See "About Pricing Design Center" for more information.
- **ImportExportPricing**. See "Importing and Exporting Pricing and Setup Components" for more information.
- **SyncPDC**. See "Synchronizing Setup Components" for more information.

To use **ImportExportPricing**, you must also understand the following:

- Extensible markup language (XML) programming
- XML schema definition (XSD)

About Replicating Data

You replicate data by extracting data from a *source* PDC database and then loading the data into a *destination* PDC database by using the **ImportExportPricing** utility.

> **Note:** If you are creating a new target system where you want to replicate the data, set up a new Oracle Communications Billing and Revenue Management (BRM) system and a new PDC system before replicating the data. See the discussion about installing and configuring BRM and PDC in *PDC Installation and System Administration Guide* for more information.
>
> Do not load the sample files in the *PDC_home*/**apps/samples/examples** directory after installing PDC, where *PDC_home* is the directory in which the PDC software is installed.

Replicating data between two PDC systems involves the following tasks:

1. Replicating Setup Components Mastered in BRM

2. Replicating Data Between PDC Databases

Replicating Setup Components Mastered in BRM

Some setup components are mastered in BRM and are then synchronized with PDC.

To ensure that all the setup components mastered in the *source* BRM system are synchronized with the *destination* BRM and PDC systems, extract the setup components from the *source* BRM database and load them into the *destination* BRM database by using the BRM load utilities. When you replicate the PDC data, these components will be synchronized with the *destination* PDC system.

Table 9–1 lists the type of setup components that are mastered in BRM and their corresponding load utilities.

Table 9–1 Setup Components Mastered in BRM and Their Corresponding Load Utilities in BRM

Setup Component Type	BRM Load Utility
Service	**pin_deploy**
Account	**pin_deploy**
Event	**pin_deploy**
G/L ID	**load_pin_glid**
Tax Supplier	**load_tax_supplier**
Provisioning Tag	**load_config_provisioning_tags**
Tax Code	NA **Note:** To replicate the tax codes, copy the **taxcodes_map** file in the *source* BRM system to the *destination* BRM system. The default location of the **taxcodes_map** file is the *BRM_Home***/sys/cm** directory, where *BRM_Home* is the directory in which you installed BRM.

See the following for more information:

- About Setup Components
- BRM load utility documentation

Replicating Data Between PDC Databases

To replicate data between PDC databases:

1. Extract the custom fields from the *source* PDC database and load them into the *destination* PDC database by doing the following:

   a. Export the custom fields from the *source* PDC database by running the following command:

   ```
   ImportExportPricing -export -customfields
   ```

   The custom fields are exported to the **export_custom.xml** file.

   b. Ensure that the transformation engines are running. See the discussion about starting the transformation engines in *PDC Installation and System Administration Guide* for more information.

c. Import the exported custom fields from the **export_custom.xml** file into the *destination* PDC database by running the following command:

```
ImportExportPricing -import -customfields export_custom.xml
```

2. Extract the service, account, event, and profile (applicable only if you are using BRM Elastic Charging Engine (ECE) for usage rating) attribute specifications from the *source* PDC database and load them into the *destination* PDC database by doing the following:

   a. Export the service, account, event, and profile (if applicable) attribute specifications from the *source* PDC database by running the following command:

   ```
   ImportExportPricing -export -metadata
   ```

   The service, account, event, and profile (if applicable) attribute specifications are exported to the **export_meta_config.xml** file.

   Important: Do not change the internal IDs of the exported attribute specifications.

   b. Ensure that the transformation engines are running. See the discussion about starting the transformation engines in *PDC Installation and System Administration Guide* for more information.

   c. Import the exported attribute specifications from the **export_meta_config.xml** file into the *destination* PDC database by running the following command:

   ```
   ImportExportPricing -import -metadata export_meta_config.xml
   ```

3. Run the **SyncPDC** utility by doing the following:

   a. Go to the *BRM_Integration_Pack_Home***/apps/syncpdc** directory.

   b. Enter the following command:

   ```
   startSyncPDC
   ```

   The **Enter Key Password** prompt appears.

   c. Enter the password PDC uses for accessing the PDC alias key in the keystore (*BRM_Integration_Pack_Home***/apps/conf/pdc.jks**).

   See "Running SyncPDC" for more information.

4. Extract all the setup components that are defined in PDC from the *source* PDC database and load them into the *destination* PDC database by doing the following:

   a. Export all the setup components from the *source* PDC database by running the following command:

   ```
   ImportExportPricing -export -config
   ```

   The setup components are exported to the **export_config.xml** file.

   b. Import the exported setup components from the **export_config.xml** file into the *destination* PDC database by running the following command:

   ```
   ImportExportPricing -import -config export_config.xml
   ```

5. Extract all the pricing components from the *source* PDC database and load them into the *destination* PDC database by doing the following:

   a. Export all the pricing components from the *source* PDC database by running the following command:

      ```
      ImportExportPricing -export -pricing
      ```

      The pricing components are exported to the **export_pricing.xml** file.

   b. Import all the pricing components from the **export_pricing.xml** file into the *destination* PDC database by running the following command:

      ```
      ImportExportPricing -import -pricing export_pricing.xml
      ```

10

Pricing Design Center Utilities

This chapter provides reference information for the Oracle Communications Pricing Design Center (PDC) utilities.

ImportExportPricing

Use the **ImportExportPricing** utility to import and export pricing and setup components from the PDC database by using XML files. See "Importing and Exporting Pricing and Setup Components" for information on using this utility.

The XML files that are used for importing or exporting data must conform to the format detailed in the XML schema definition (XSD) files.

The utility uses separate XSD files for importing or exporting pricing components, setup components, and profile data. The XSD files are available in their respective directories in the *PDC_home*/**apps/xsd** directory, where *PDC_home* is the directory in which the PDC software is installed. See "About the XSD Files" for more information.

Location

PDC_home/**apps/bin**

Syntax - export

```
ImportExportPricing
                    -export [FileNamePrefix]
                    -config [SetupObjectType1,SetupObjectType2,…]
                    -metadata
                    -expObsolete
                    -pricing [PricingObjectType1,PricingObjectType2,…]
                    -profile
                    [-n "ObjectName1, ObjectName2,…"]
                    [-productSpecName ProductSpecName1,ProductSpecName2,…]
                    [-expRefs -expAllRefs]
                    [-inclFailedObjs]
                    [-brmobject [brmObjectType1, brmObjectType2, …]]
                    [-pricing [PricingObjectType1,PricingObjectType2,…]
                    [-ma time]
                    [-mby user1, user2,… ]
                    [-l loglevel]
                    [-v]
                    [-ow]
                    [-pricingprofilename "Name1, Name2,…"]
                   -appsvruser ApplicationServerUserName
                    -pdcuser PdcUserName
                    [-h]
```

Parameters - export

-export [*FileNamePrefix*]

Exports the data from a PDC database and generates a separate XML file for pricing components, setup components, and pricing profile data in the directory from which you run the utility. The utility generates the file names as:

- *FileNamePrefix*_**export_pricing.xml** for the file containing pricing components.
- *FileNamePrefix*_**export_config.xml** for the file containing setup components.
- *FileNamePrefix*_**export_profile.xml** for the file containing pricing profiles data.

If you do not specify *FileNamePrefix*, the utility generates the file name as follows:

- **export_pricing.xml** for the file containing pricing components. If this file already exists in PDC, the utility generates the file name as **export_pricing_***timestamp***.xml**, where *timestamp* is the server's local time in the format *yyyy-mm-dd_hh-mm-ss*.
- **export_config.xml** for the file containing setup components. If this file already exists in PDC, the utility generates the file name as **export_config_***timestamp***.xml**, where *timestamp* is the server's local time in the format *yyyy-mm-dd_hh-mm-ss*.
- **export_profile.xml** for the file containing pricing profile data. If this file already exists in PDC, the utility generates the file name as **export_profile_***timestamp***.xml**, where *timestamp* is the server's local time in the format *yyyy-mm-dd_hh-mm-ss*.

-config

Exports all the setup components that are defined in PDC from the PDC database to the XML file. See "About Setup Components" for more information about the setup components that are defined in PDC.

-config [*SetupObjectType1*, *SetupObjectType2*,...]

Exports *SetupObjectType* from the PDC database to the XML file, where *SetupObjectType* is the setup component type listed in Table 10–1.

Table 10–1 lists the setup component types and the setup components they represent.

Table 10–1 Setup Component Types and the Setup Components They Represent

Setup Component Type	Setup Component
ZONE_MAP	Value Map
EVENT_ATTRIBUTE_SPEC	Event
PRODUCT_ATTRIBUTE_SPEC	Service
CUSTOMER_ATTRIBUTE_SPEC	Account Attribute
RUM_CONFIGURATION	RUM
ZONE_RESULT_CONFIGURATION	Impact Category for Zones
ATTRIBUTE_SPEC_MAP	Service-Event Map
BALANCE_ELEMENT	Balance Element
GLID	G/L ID
TAX_CODE	Tax Code
TAX_SUPPLIER	Tax Supplier
GEO_ZONE_MODEL	Geographical Zone Model
STANDARD_ZONE_MODEL	Standard Zone Model
PROVISIONING_TAG	Provisioning Tag
HOLIDAY_CALENDAR	Special Day Calendar
CUSTOM_ANALYZER_RULE	Custom Rule
PROFILE_ATTRIBUTE_SPEC	Profile Attribute Specification
ITEM_TYPE_SELECTOR	Item Type Selector

If you do not specify *SetupObjectType*, the utility exports all the setup components from the PDC database to the XML file.

-metadata
Exports the event, service, account, and profile attribute specifications listed in Table 10–1 from the PDC database to the XML file.

-expObsolete
Exports all obsolete pricing components (including the corresponding metadata and setup components).

-brmobject [*BRMObjectType1*, *BRMObjectType2*,...]
Exports the components mastered in BRM from PDC into XML file, where *BRMObjectype* is the setup component type. The following are the BRM component types mastered in BRM that you can export:

- GLID
- TAX_CODE
- TAX_SUPPLIER
- PROVISIONING_TAG

-pricing [*PricingObjectType1*, *PricingObjectType2*,...]
Exports *PricingObjectType* from the PDC database to the XML file, where *PricingObjectType* is the pricing component type listed in Table 10–2.

Table 10–2 lists the pricing component types and the pricing components they represent.

Table 10–2 Pricing Component Types and the Pricing Components They Represent

Pricing Component Type	Pricing Component Represented
CHARGE_RATE_PLAN	Charge
ALTERATION_RATE_PLAN	Discount
DISTRIBUTION_RATE_PLAN	Chargeshare
ROLLOVER_RATE_PLAN	Rollover
ALTERATION_RATE_PLAN_SELECTOR	Discount Selector
GENERIC SELECTOR	Generic Selector
CHARGE_RATE_PLAN_SELECTOR	Charge Selector
PRICE_MODEL_SELECTOR	Price Selector
APN_MAP	APN Selector
USC_MAP	USC Selector
TIME_MODEL	Time Model
TRIGGER_SPEC	Trigger
CHARGE_SELECTOR_SPEC	Filter
ALTERATION_POP_MODEL	Discount Pricing
ROLLOVER_POP_MODEL	Rollover Rules
RECURRING_POP_MODEL	Recurring Pricing
ONE_TIME_POP_MODEL	One-Time Pricing
CHARGE_OFFERING	Charge Offer
ALTERATION_OFFERING	Discount Offer

Table 10–2 (Cont.) Pricing Component Types and the Pricing Components They

Pricing Component Type	Pricing Component Represented
DISTRIBUTION_OFFERING	Chargeshare Offer
ALTERATION_EXCLUSION	Discount Exclusion
BUNDLED_PRODUCT_OFFERING	Bundle
BUNDLE_TRANSITION	Bundle Transition
PACKAGE_OBJ	Package
PACKAGE_EXCLUSION	Discount Restriction
PACKAGE_TRANSITION	Package Transition
PACKAGE_LIST	Package List
BEST_PRICING	Best Pricing

If you do not specify *PricingObjectType*, the utility exports all the pricing components from the PDC database to the XML file.

-profile
Exports the pricing profile data from the PDC database to the XML file.

-n "*ObjectName1*, *ObjectName2,…*"
Filters the pricing or setup components based on the specified component names. You can use an asterisk (*) at the end of the name to search for variations of that name. For example, searching for **chargeoffergsm*** yields variations of that name, such as **chargeoffergsm1** and **chargeoffergsm2**.

> **Note:** When you use the **-n** parameter, you must specify either **-config** or **-pricing**.
>
> The parameters **-n** and **-productSpecName** are mutually exclusive.

-productSpecName *ProductSpecName1, ProductSpecName2,…*
Filters the exported pricing components that are scoped to the specified services.

If you specify this parameter along with *PricingObjectType*, the utility exports all the pricing object types that are scoped to *ProductSpecName*. You can use this parameter only for the following pricing object types:

- ALTERATION_OFFERING
- CHARGE_OFFERING
- DISTRIBUTION_OFFERING
- CHARGE_RATE_PLAN

If you specify this parameter without specifying *PricingObjectType*, the utility exports all pricing components scoped to the specified service.

> **Note:** The parameters **-n** and **-productSpecName** are mutually exclusive.

-expRefs
Exports all the pricing components referenced by *PricingObjectType*. For example, if multiple charge offers are referenced by bundles, this parameter exports all the charge offers when exporting bundles. The utility ignores this parameter if it is used with the **-config** parameter.

-expAllRefs
Exports all the PDC components (including metadata, setup components, and pricing components) referenced by *PricingObjectType*. For example, if setup components, such as ratable usage metrics (RUMs) and tax codes are referenced by charge offers, then this parameter exports RUMs and tax codes when exporting charge offers. This parameter applies only to the **-pricing** parameter.

-ma *time*
Exports pricing components modified on or after *time*. Enter the *time* in the format *yyyy-mm-dd_hh-mm-ss*.

-mby *user1, user2,...*
Exports the pricing components created or modified by the specified users.

Note: The parameters **-mby** and **-productSpecName** are mutually exclusive.

-l *loglevel*
Specifies how much information the utility should log. The logs are stored in the **ImportExportPricing**_*username_timestamp*.**log** file, where *username* is the name of user who used the utility and *timestamp* is the time the log file was created in the format *yyyy-mm-dd_hh-mm-ss*.

Set *loglevel* to one of the following:

- FINE
- FINER
- FINEST
- INFO
- OFF
- SEVERE
- WARNING

The default is **INFO**.

-v
Displays information about successful or failed processing as the utility runs.

-pricingprofilename "*Name1*, *Name2*,…"
Exports pricing components scoped to the pricing profile names specified in the list.

If this parameter is used with the **-pricing** parameter when a list of pricing component types is specified, the utility exports all the pricing components of the specified type scoped to the pricing profiles names specified in the list.

If this parameter is specified with the **-pricing** parameter when no pricing components types are specified, the utility exports all the pricing components that are scoped to a pricing profile.

-appsvruser *ApplicationServerUserName*
Specifies the user name to access the WebLogic Server. If you do not specify this parameter, the utility prompts you to enter the user name at the command prompt.

-pdcuser *PdcUserName*
Specifies the user name to access PDC. If you do not specify this parameter, the utility prompts you to enter the user name at the command prompt.

-h
Displays the syntax for this utility.

See Table 10–2 for the list of the pricing component types and the pricing components they represent.

Syntax - import

```
ImportExportPricing
                    -import
                    -metadata
                    -config SetupFileName
                    -pricing PricingFileName
                    -profile ProfileFileName
                    -ignoreID
                    [-l loglevel]
                    [-v ]
                    [-ow]
                    [-changeSetName ChangeSetName]
                    -submitCS
                    -appsvruser ApplicationServerUserName
                    -pdcuser PdcUserName
                    [-h]
```

Parameters - import

-import
Imports the data from the XML file that conforms to the format detailed in the XSD file to the PDC database. You must specify either **-pricing**, **-config**, or **-profile** along with **-import**.

-metadata
Imports the metadata from the metadata filename to the PDC database.

-config *SetupFileName*
Imports the setup components from the *SetupFileName* to the PDC database, where *SetupFileName* specifies the full path and file name of the XML file containing the setup components.

-pricing *PricingFileName*
Imports the pricing components from the *PricingFileName* to the PDC database, where *PricingFileName* specifies the full path and file name of the XML file containing the pricing components.

-profile *ProfileFileName*
Imports the profile data from the *ProfileFileName* file to the PDC database, where *ProfileFileName* specifies the full path and file name of the XML file containing profile data. If you specify **-pricing** and **-config** parameters along with **-profile**, the utility imports profile data and setup components before importing the pricing components.

If the import of the profile data and setup components is not successful, the utility does not import pricing components.

-ignoreID
Ignores the internal IDs in the specified XML file and imports the components by the input file name.

-l *loglevel*
Specifies how much information the utility should log. The logs are stored in the **ImportExportPricing**_*username_timestamp*.**log** file, where *username* is the name of user who used the utility and *timestamp* is the time the log file was created in the format *yyyy-mm-dd_hh-mm-ss*.

Set *loglevel* to one of the following:

- FINE
- FINER
- FINEST
- INFO
- OFF
- SEVERE
- WARNING

The default is **INFO**.

-v
Displays information about successful or failed processing as the utility runs.

-ow
Overwrites any existing pricing or setup components available in the PDC database with the specified components. If the specified pricing or setup component does not exist in PDC, the utility creates new pricing or setup components.

Note: If you do not use the **-ow** parameter when importing data, the utility generates an error if an object with the same name already exists in PDC.

[-changeSetName *ChangeSetName*]
Specifies the name of the changeset used to import setup and pricing components into the PDC database. If you do not specify *ChangeSetName*, the utility generates the changeset name as **ieChangeSet***timestamp*, where *timestamp* is the current timestamp in the format *yyyy-mm-dd_hh-mm-ss*.

-submitCS
Submits the changeset that is used to import the setup and pricing components to the configured billing system database.

-appsvruser *ApplicationServerUserName*
Specifies the user name to access the WebLogic Server. If you do not specify this parameter, the utility prompts you to enter the user name at the command prompt.

-pdcuser *PdcUserName*
Specifies the user name to access PDC. If you do not specify this parameter, the utility prompts you to enter the user name at the command prompt.

-h
Displays the syntax for this utility.

Syntax - publish - target

```
ImportExportPricing
                    -publish [Component] [ObjectType1,ObjectType2,…]
                    -target Target_Engine
                    -appsvruser ApplicationServerUserName
                    -pdcuser PdcUserName
```

Parameters - publish - target

-publish [*Component*]
Publishes the following components from PDC to the specified target engines:

- **metadata.** Publishes the metadata in PDC to the specified target engine.
- **config.** Publishes the setup components in PDC to the specified target engine.
- **pricing**. Publishes the pricing components in PDC to the specified target engine.
- **all.** Publishes all the components in PDC to the specified target engine.

-Publish [*ObjectType1,ObjectType2,…*]
Publishes specified components to the target engine; for example, BALANCE_ ELEMENT. You can specify [*Component*] or [*ObjectType*] for publishing the components. For the list of component types, see Table 10–1 and Table 10–2.

-target [*target engine*]
Specifies the following target engines for publishing the components:

- **Rre.** Publishes the components or object types to the real-time rating target engine.
- **Bre.** Publishes the components or object types to the batch rating target engine.
- **Ece.** Publishes the components or object types to the ECE rating target engine.
- **All.** Publishes the components or object types to all the target engines mentioned above.

-appsvruser *ApplicationServerUserName*
Specifies the user name to access the WebLogic Server. If you do not specify this parameter, the utility prompts you to enter the user name at the command prompt.

-pdcuser *PdcUserName*
Specifies the user name to access PDC. If you do not specify this parameter, the utility prompts you to enter the user name at the command prompt.

Syntax - keep

```
ImportExportPricing
                    -keep [metadata | config | pricing | all] -numVersion N
                    -appsvruser ApplicationServerUserName
                    -pdcuser PdcUserName
```

Parameters - keep

-keep *metadata* | *config* | *pricing* | *all*
Deletes old versions of PDC components that are obsolete in PDC or the associated target engines (for example, Oracle Communications Billing and Revenue Management (BRM) Elastic Charging Engine (ECE)) and keeps only the latest versions of successfully promoted PDC components.

-numVersion *N*
Keeps only the latest specified version of all the successfully promoted objects in the PDC database and deletes the remaining versions, where *N* is a positive integer that specifies the number of latest successful versions to be kept in the PDC database.

-appsvruser *ApplicationServerUserName*
Specifies the user name to access the WebLogic Server. If you do not specify this parameter, the utility prompts you to enter the user name at the command prompt.

-pdcuser *PdcUserName*
Specifies the user name to access PDC. If you do not specify this parameter, the utility prompts you to enter the user name at the command prompt.

Syntax - delete

```
ImportExportPricing
                    [-d FilePath][-dp FilePath]
                    -appsvruser ApplicationServerUserName
                    -pdcuser PdcUserName
```

Parameters - delete

-d *FilePath*
Deletes the setup components specified in *FilePath* from the PDC database, where *FilePath* specifies the full path and file name of the XML file containing the setup components. The utility displays an error message if the data being deleted is dependent on other data.

-dp *FilePath*
Deletes the pricing profiles specified in *FilePath* from the PDC database provided the profiles are not being used by any pricing component, where *FilePath* specifies the full path and file name of the XML file containing the pricing profiles.

-appsvruser *ApplicationServerUserName*
Specifies the user name to access the WebLogic Server. If you do not specify this parameter, the utility prompts you to enter the user name at the command prompt.

-pdcuser *PdcUserName*
Specifies the user name to access PDC. If you do not specify this parameter, the utility prompts you to enter the user name at the command prompt.

Syntax - type

```
ImportExportPricing
                    -t config | pricing
                    -appsvruser ApplicationServerUsername
                    -pdcuser PdcUserName
```

Parameters - type

-t config | pricing
Displays the list of the setup or pricing component types supported in PDC.

-appsvruser *ApplicationServerUserName*
Specifies the user name to access the WebLogic Server. If you do not specify this parameter, the utility prompts you to enter the user name at the command prompt.

-pdcuser *PdcUserName*
Specifies the user name to access PDC. If you do not specify this parameter, the utility prompts you to enter the user name at the command prompt.

Parameters - ignoreID

-ignoreID
Ignores the internal IDs in the specified XML file and imports the components by the input file name.

-appsvruser *ApplicationServerUserName*
Specifies the user name to access the WebLogic Server. If you do not specify this parameter, the utility prompts you to enter the user name at the command prompt.

-pdcuser *PdcUserName*
Specifies the user name to access PDC. If you do not specify this parameter, the utility prompts you to enter the user name at the command prompt.

Results

The **ImportExportPricing** utility notifies you when it runs successfully. Otherwise, look in the **ImportExportPricing**_*username*_*timestamp*.**log** file, where *username* is the name of user who used the utility and *timestamp* is the timestamp of the log file in the format *yyyy-mm-dd_hh-mm-ss*. This file is in the directory from which the utility was started.

11

BRM and PDC Terminology

The Oracle Communications Pricing Design Center (PDC) term for a component sometimes differs from the Oracle Communications Billing and Revenue Management (BRM) term for the same component. Table 11–1 lists those differences.

Table 11–1 Mapping BRM to PDC Terminology

BRM Term	PDC Term
ChargeShare model	chargeshare
ChargeShare	chargeshare offer
deal	bundle
discount model	discount
discount	discount offer
EVAL("*function_name*") discount expression	Function["*function_name*"] discount expression
item tag and item type	item type selector
plan	package
plan list	package list
product	charge offer
rate plan	charge
rate plan selector	charge selector
rate tag	pricing name
resource	balance element
ROUND_PLAIN	round
subscription group	service group
trunc	round down
zone Map	value map