

**Oracle® Communications
Billing and Revenue Management**

Concepts

Release 7.5

E23300-09

August 2015

Copyright © 2011, 2015, 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

Preface	vii
Audience	vii
Accessing Oracle Communications Documentation	vii
Documentation Accessibility	vii
Document Revision History	vii
1 Introducing BRM	
What Is BRM?	1-1
About Setting Up Your Services	1-3
About Charging Customers for Your Services	1-3
About Billable Events	1-3
About Rating	1-4
About Real-Time Rating	1-4
About Batch Rating	1-5
Defining How to Rate Billable Events in Your Price List	1-5
Ways to Measure and Charge for Billable Events	1-6
About Price Lists	1-6
About Calculating Taxes	1-6
About Authentication, Authorization, and Accounting	1-7
Real-Time Postpaid AAA	1-7
Real-Time Prepaid AAA	1-7
Batch Rating AAA	1-8
About Registering Customers	1-8
Ways to Register Customers	1-8
About Billing Customers	1-9
Dealing with Corrections to Bills	1-10
About Billing Types	1-11
About Account Groups	1-11
How Billing Is Different from Rating	1-11
About Managing Payments	1-11
About Managing Accounts Receivable	1-12
About Managing Customers	1-12
About Managing Brands	1-13
About Collecting Revenue Data	1-13
About Running Database Reports	1-13

What Is in a BRM system?	1-13
How the BRM Database Stores Data	1-14
How BRM Processes a Typical Billable Event	1-15
How a Typical Internet Service Provider Uses BRM	1-16
Customer Registration.....	1-16
Custom Profile Information	1-16
Managing Customer Accounts.....	1-16
Pricing.....	1-17
G/L Accounts for Pricing Elements	1-17
Billing and Payments.....	1-18
About the BRM Client Applications	1-18
About BRM Command-Line Utilities.....	1-21
Implementation Process	1-21
Ways to Use and Customize BRM.....	1-22
About Integrating Your Services with BRM	1-22
About Creating Custom Account Management and Billing Tools.....	1-24
About Connecting BRM to External Data Processing Sources.....	1-25
About the Data Displayed in BRM Client Applications.....	1-25
About BRM Defaults.....	1-25
When Is Programming Required?	1-25
Summary of Key Terms	1-26

2 Using BRM with Oracle Application Integration Architecture

About Oracle Application Integration Architecture	2-1
Setting Up BRM to Work with Oracle Application Integration Architecture	2-1
Installing and Configuring the Required BRM Components	2-2
Integrating BRM Features with External CRM Applications	2-2
Integrating Collections with External CRM Applications	2-2
Integrating Friends and Family Promotions with External CRM Applications.....	2-3
Displaying Siebel CRM Promotion Names on Invoices.....	2-3
Integrating BRM Features with External CRM Applications in a Multischema System	2-4
Integrating Account Migrations with External Applications in a Multischema System.....	2-4
Integrating Collections with External Applications in a Multischema System	2-5
Creating Products and Discounts for an External CRM	2-5
Synchronizing Products That Have Multiple Rate Plans with External CRM Applications	2-5
Validating Customer Contact Information	2-5

3 BRM System Architecture

About the Four-Tier Architecture	3-1
Application Tier	3-2
Business Process Tier	3-3
About Facilities Modules (FMs).....	3-4
About External Modules (EMs)	3-5
About Connection Manager Master Processes (CMMPs)	3-5
Data Management Tier	3-6
About Data Managers (DMs)	3-6
About Data Manager (DM) Back Ends	3-6

About IMDB Cache DM.....	3-6
Data Tier	3-7
About the BRM Database.....	3-7
About the Residency Type.....	3-8
About the Multischema Architecture	3-9
About Multidatabase Manager	3-9
About Oracle RAC for a High-Availability BRM System	3-9
About Oracle IMDB Cache	3-9
About Objects in an IMDB Cache-Enabled System	3-10
Transient Objects.....	3-10
Reference Objects	3-10
Database Objects	3-10
About Active Session Objects.....	3-11
About the Object Residency Type	3-11
About the Pipeline Manager System Architecture	3-12
About the Pipeline System Components.....	3-13
About the Controller	3-13
About the EDR Factory	3-13
About the Transaction ID Controller	3-14
About the Sequencer.....	3-14
About the Event Handler.....	3-14
About the Data Pool.....	3-15
About Pipelines	3-15
About Using Multiple Pipelines	3-16
About the Pipeline Controller	3-16
About Thread Handling.....	3-16
About the Pipeline Manager Database	3-16
About Configuring Pipeline Manager	3-16
Configuring the Four-Tier Architecture	3-17
Communication Between System Components	3-17
Four-Tier Architecture and Failure Recovery	3-17

Preface

This document provides an overview of the Oracle Communications Billing and Revenue Management (BRM) system.

Audience

This document is intended for anyone who installs, configures, administers, or customizes BRM.

Accessing Oracle Communications Documentation

BRM documentation and additional Oracle documentation; such as Oracle Database documentation, is available from 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.

Document Revision History

The following table lists the revision history for this book.

Version	Date	Description
E23300-01	November 2011	Initial release.

Version	Date	Description
E23300-02	May 2012	Documentation updates for BRM 7.5 Patch Set 1. <ul style="list-style-type: none"> ■ Made minor formatting and text changes.
E23300-04	December 2012	Documentation updates for BRM 7.5 Patch Set 3. <ul style="list-style-type: none"> ■ Added "Synchronizing Products That Have Multiple Rate Plans with External CRM Applications" section. ■ Updated "About Integrating Your Services with BRM" for policy-driven charging. ■ Added "Integrating BRM Features with External CRM Applications in a Multischema System" section. ■ Updated the pin_collections_process utility configuration step in "Integrating Collections with External CRM Applications" section.
E23300-05	March 2013	Documentation updates for BRM 7.5 Patch Set 4. <ul style="list-style-type: none"> ■ Replaced multidatabase information with multischema information.
E23300-06	August 2014	Documentation updates for BRM 7.5 Patch Set 9. <ul style="list-style-type: none"> ■ Documentation added for RADIUS Manager.
E23300-07	January 2015	Documentation updates for BRM 7.5 Patch Set 11. <ul style="list-style-type: none"> ■ Made minor formatting and text changes.
E23300-08	June 2015	Documentation updates for BRM 7.5 Patch Set 12. <ul style="list-style-type: none"> ■ Updated the following sections: <ul style="list-style-type: none"> "How the BRM Database Stores Data" "Ways to Use and Customize BRM"
E23300-09	August 2015	Documentation updates for BRM 7.5 Maintenance Patch Set 1. <ul style="list-style-type: none"> ■ Updated the following sections: <ul style="list-style-type: none"> About Calculating Taxes Data Management Tier About Data Managers (DMs)

Introducing BRM

This chapter provides an introduction to the Oracle Communications Billing and Revenue Management (BRM) system.

WARNING: Do not use or modify this product except as explicitly instructed in this documentation. Assumptions should not be made about functionality that is not documented or use of functionality in a manner that is not documented. Use or modification of this software product in any manner or for any purpose other than as expressly set forth in this documentation may result in voidance or forfeiture of your warranties and support services rights. Consult your software license agreement for more details. If you have any questions regarding an intended use or modification of this product, contact your Oracle BRM account executive.

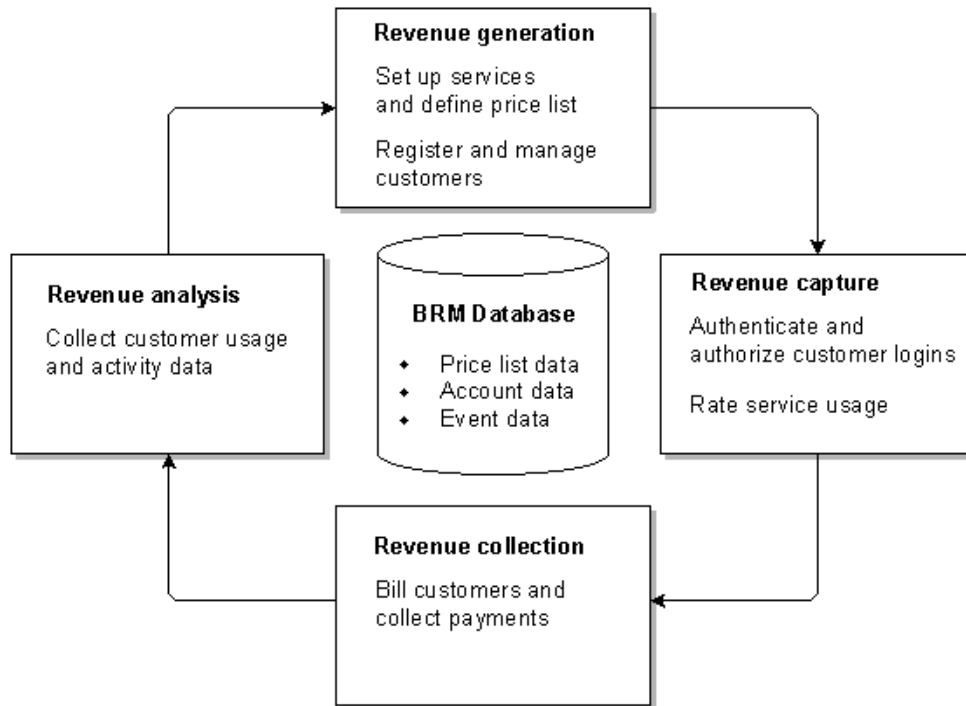
What Is BRM?

BRM is an end-to-end revenue management system for communications and media service providers. When you build your business around BRM, you can more effectively manage the revenue life cycle:

- BRM *revenue generation* enables you to deliver services with optimal pricing to your customers. With BRM, service plans can be complex, yet you can change them quickly to respond to changing market conditions. BRM customer management tools enable you to continue generating revenue from existing customers by maintaining a positive and proactive relationship.
- BRM *revenue capture* enables you to rate service usage with a high level of control and responsiveness. The BRM system is integrated with network authentication, authorization, and accounting.
- BRM *revenue collection* enables you to invoice and bill customers, collect payments, manage accounts receivable (A/R), and collect general ledger (G/L) data.
- BRM *revenue analysis* enables you to audit processes for revenue leakage, generate business intelligence reports, and set up safeguards for data consistency and high availability. Collecting revenue analysis data helps you adjust your service plans for maximum revenue generation.

Figure 1–1 shows the BRM revenue management life cycle:

Figure 1–1 BRM Revenue Management Life Cycle



Setting Up Services. When you set up your services, you use optional components such as Global System for Mobile Communications (GSM) Manager and Remote Authentication Dial-In User Service (RADIUS) Manager to capture data about service usage (for example, the number of hours in a customer’s dialup session or length of a telephone call).

Defining a Price List. You create a price list to define how much to charge for your services. For example, you can charge for wireless telephony or Internet access usage, monthly subscription fees, and setup fees.

Registering Customers. Customers register for accounts by calling a customer service representative (CSR) or by using a Web page. A customer typically signs up for one or more services, such as Internet access and email. Customer account data, such as telephone numbers, Internet service logins, and billing information, is stored in the BRM database.

Managing Customers. When customer account information changes (for example, if a customer changes the credit card), you use customer management tools to update account information or you can provide Web pages for customers to manage their own accounts.

Authenticating and Authorizing Customer Logins. For services where usage can be tracked in real time, BRM uses customer data stored in the BRM database to authenticate the customer’s identity and to verify the customer’s authorization to use that service.

Rating Service Usage. BRM measures service usage and rates it (for example, 10 cents per minute for a wireless call). The charge is added to the customer’s account balance.

Billing Customers and Collecting Payments. Customers are billed at intervals that you choose (typically every month). You can charge customers to their credit cards directly or send invoices and process payments by checks and other payment methods.

Collecting Customer Usage and Activity Data. You use data in the BRM database to report on all facets of your business (for example, revenue generated from each service). As your business grows and changes, you can add pricing and provisioning support for new services.

About Setting Up Your Services

Businesses that use BRM typically provide services such as telephony, Internet access, and email. To integrate with your services, you use optional service integration components such as GSM Manager and RADIUS Manager. For example:

- To implement GSM wireless services, GSM Manager provides the ability to rate and manage GSM services and supplementary services, such as voice mail.
- To implement dialup Internet access, RADIUS Manager uses data stored in the BRM database to authenticate and authorize the customer. RADIUS Manager also captures data about the customer's dialup session. BRM uses this data to measure, rate, and bill for service usage.

Other services use optional service integration components in a similar way; for example, you use Email Manager to provide email service.

When you integrate your services with BRM, you can manage customer access in real time. For example, because BRM tracks Internet usage and updates the customer account balance in real time, you can prevent service authorization as soon as a credit limit is reached.

About Charging Customers for Your Services

You define how to charge for your services by creating a *price list*. For example, your price list might specify charges for wireless usage as follows:

- 10 cents per minute for peak-time usage.
- 5 cents per minute for off-peak usage.

Or you can charge for an IP dialup service as follows:

- \$20 per month subscription fee.
- \$1 per hour usage fee.

You can create complex pricing structures (for example, 10 cents per minute for the first 60 minutes and 5 cents per minute for the next 100 minutes).

You use Pricing Center to create and modify your price list.

About Billable Events

In a typical business day, thousands of customers make phone calls or access the Internet, generating millions of interactions that must be managed and charged for. BRM manages those interactions by creating and storing events.

An *event* is a record in the BRM database of a customer or administrative action. For example:

- When a customer logs in, or makes a telephone call, BRM creates a *session* event, which stores data about the session, such as the start time and end time.
- When a CSR changes a customer's password, BRM creates an *activity* event, which stores information such as the login name of the CSR who made the password change.

In general, the end time of the event is used for rating, billing, and other BRM processes. For example, if an event begins before midnight on the 23rd of the month and ends after midnight on the 24th, BRM treats the event as occurring on the later date.

When you set up your price list, you define which events you want to charge for. These are called *billable events*. To determine how much to charge a customer for a billable event, BRM rates the event.

About Rating

To rate an event, BRM does the following:

1. Measures the event.
2. Applies a charge to the resulting measurement.
3. Adds the charge to the customer's account balance.

You can rate usage in two ways:

- *Real-time rating* monitors and rates the service usage as it happens, such as Internet access.
- *Batch rating* rates events that have been recorded in files, such as telephony call detail records (CDRs).

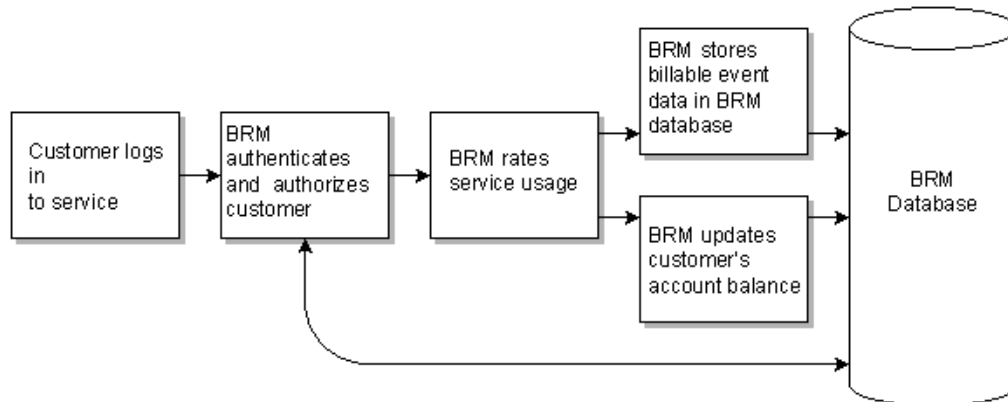
About Real-Time Rating

Real-time rating is used for services such as dialup access and email. For example, to rate Internet access in real time:

1. BRM determines the length of the session by comparing the start time and end time.
2. The BRM rating opcodes apply a charge to the amount of time. For example, if you specify a charge of \$1 per hour for Internet access, a 10-hour session costs \$10. This charge is called the *balance impact*.
3. BRM adds the total charge for the event to the customer's account balance.

Figure 1-2 shows how a billable event is captured, rated, and recorded in the BRM database:

Figure 1-2 Capturing, Rating and Recording of Events in BRM



About Batch Rating

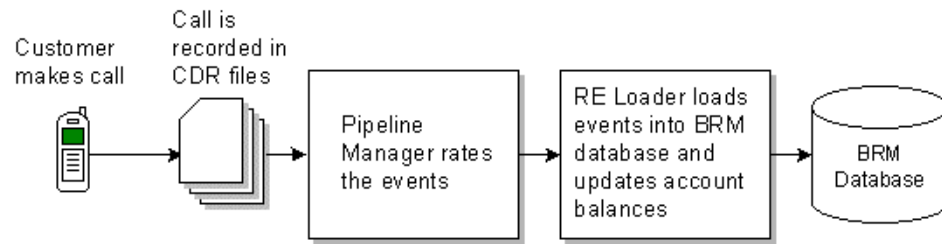
Batch rating is typically used for rating telephony services, where large amounts of events are recorded in files. Batch rating is usually performed by Pipeline Manager, but you can also perform batch rating by using Universal Event (UE) Loader.

For example, to rate telephone calls:

1. Pipeline Manager reads the data from CDRs to determine the time and duration of the calls.
2. Pipeline Manager rating modules apply a charge to the amount of time. For example, if you specify a charge of 10 cents per minute, a 10-minute call costs \$1. This charge is called the *balance impact*.
3. Rated Event (RE) Loader loads the rated events into the BRM database.
4. BRM adds the total charge for the event to the customer's account balance.

Figure 1-3 shows how a billable event is rated by Pipeline Manager and recorded in the BRM database:

Figure 1-3 Pipeline Manager Event Rating and Recording to BRM Database



Note: You can also use the UE Loader to load data from an event log file as billable events and rate them using BRM real-time rating opcodes. The event log files can include data from Web servers or Internet Protocol (IP) telephony CDR files. When the events are loaded, BRM rates the events and updates the customers' account balances.

Defining How to Rate Billable Events in Your Price List

When you create your price list, you specify the following:

1. What types of events to rate (for example, telephone calls or dialup usage events).
2. The unit of measurement to rate (for example, the amount of time for a telephone call or the bytes downloaded).
3. The balance impact (for example, 10 cents per minute or 1 cent per byte).

In addition to rating usage events, you can rate the following types of events:

- Product purchase events, such as a \$25.00 setup fee.
- Monthly subscription events, such as a \$19.95 monthly account fee.
- Product cancellation events, such as a \$20.00 fee for closing an account after holding it for less than one year.
- Administrative events, such as a \$5.00 fee for changing a password.

You can set up your price list to use multiple types of fees. For example, a typical Internet service provider price list might include the following fees:

- A one-time account setup fee.
- A monthly account subscription fee.
- An hourly usage fee.

Ways to Measure and Charge for Billable Events

You can create pricing models that measure and rate billable events in a variety of ways:

- You can charge based on time of day (for example, Internet access for \$2 per hour for prime time and \$1 per hour between midnight and 6 a.m.). If a session spans the prime time and off-peak time, BRM rates the session accurately in each time period.
- You can charge for Internet access based on how long a customer is connected in a single session. For example, you can charge \$2 per hour for the first hour and \$1 per hour for the second hour.
- You can charge for telephony based on the number of calls. For example, you can charge 10 cents per call for the first 100 calls and 5 cents per call for additional calls.
- You can provide real-time discounts based on the number of accounts. For example, if a company has more than 100 dialup accounts, you can offer a 10% discount.

About Price Lists

You create a price list by using Pricing Center. In addition to defining which events to rate and how to rate them, your price list organizes all of the fees you charge into *plans*. When customers register for an account, they choose from a set of plans. For example, a customer can choose a plan that includes Internet access, email, or both as shown in [Figure 1-4](#):

Figure 1-4 Price List with Plans



About Calculating Taxes

BRM calculates taxes by integrating with tax databases, such as Vertex. To calculate taxes:

1. When BRM rates a billable event, it collects information about the event that can be used for calculating taxes, such as the location of your business, the location of the customer, and customer tax exemptions.
2. BRM compiles the tax calculation data into a transaction and sends it to the tax database.

3. Based on the data you provide, the tax database calculates the taxes on the event, then returns the tax amount.
4. BRM adds the tax amount to the customer's account balance.
5. The tax amount is recorded as G/L data.

About Authentication, Authorization, and Accounting

Authentication, authorization, and accounting (AAA) define how you interact with your customers:

- Authentication provides a way for you to identify customers (for example, by requiring a login name and a password).
- Authorization provides a way to control what customers can do with their services. For example, you can restrict access based on products owned or credit limits.
- Accounting tracks the resources used by the customer and enables you to rate usage.

To interface BRM with IMS networks, you use Oracle Communications Service Broker (OCSB) Online Mediation Controller (OCMC). For more information, see the following documents:

- *Oracle Communications Service Broker Concepts Guide*
- *Oracle Communications Service Broker Online Mediation Controller Implementation Guide*

These documents are part of the Oracle Communications Service Broker 6.0 documentation set.

Real-Time Postpaid AAA

For services that use real-time rating with postpaid accounts, such as IP access, BRM uses AAA as follows:

- BRM authenticates customers by checking the login name and password that the customer provides with the login name and password stored in the BRM database.
- BRM authorizes customers by making sure they own the service being requested and that they have not surpassed a credit limit.
- BRM performs accounting by recording sessions in real time. When the session is closed, BRM rates the session.

Real-Time Prepaid AAA

For services that use real-time rating with prepaid accounts, such as prepaid telecommunications (Telco) services, BRM uses AAA as follows:

- BRM authenticates customers by verifying their device ID with the ID in the BRM database.
- BRM authorizes customers to use a service for a specified duration or volume (for example, authorizing customers to make a 20-minute wireless call or download 200 bytes of data).

In a prepaid scenario, BRM must determine whether the customer has sufficient resources before authorizing the usage. To do this, BRM estimates the usage cost

by applying rating and discounting and comparing the estimate to the customer's prepaid resources.

- BRM reauthorizes customers for extended usage, if needed. This enables customers to continue their prepaid sessions when the initially approved quantity is consumed.
- BRM performs accounting to collect usage information to rate the prepaid session.

Batch Rating AAA

For services that are not rated in real time, such as postpaid Telco services, AAA is handled by the network; BRM is not involved. For example, for a telephone call:

- The network authenticates and authorizes the customer.
- The network performs accounting to collect usage data and store it in CDR files. For example, the network records the call start and end times and the originating and destination numbers. BRM uses that data to rate the usage.

About Registering Customers

When you register a customer, you create a customer account in the BRM database. Registration typically follows this process:

1. The customer calls the CSR or accesses a Web page.
2. The customer chooses a plan. Plans are usually based on services; for example, a customer might choose a plan that offers Internet access, email, or both.
3. The customer provides information such as name, address, and credit card number. In addition, a CSR might give discounts to the customer or collect billing information, such as a billing contact name.
4. The account is created in the database. At account creation, various events occur:
 - The customer information is checked by the system to see whether it is valid and to ensure that all required information has been supplied.
 - If the customer entered a credit card number, it is validated by a credit card processing service.
 - If there is a purchase fee, the customer is automatically charged.
 - If the service includes a phone number, the phone number is provisioned on the network.
 - A welcome message is automatically sent to the customer.

Ways to Register Customers

You can register customers by using these methods:

- Have CSRs create accounts by using Customer Center or a third-party customer management program.
- Set up Web-based account registration. Customers use their browsers to create accounts by filling out HTML forms on your Web page.
- For telephony, give customers disposable calling cards, which they use to access accounts you have already created in your BRM database.

About Billing Customers

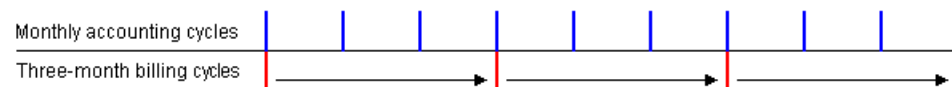
Each customer account has a monthly billing date, typically the date that the account was created. For example, if the account was created on January 15, the billing date is typically the 15th day of each month.

To bill customers, you run a set of billing utilities. One of the billing utilities (**pin_bill_accts**) finds every account that has a billing date of the previous day or earlier. If the account's billing date is due, BRM does the following:

1. **BRM performs monthly accounting.** BRM compiles the total amount of balance impacts that have occurred in the past month. This can include usage fees and subscription fees. This monthly accounting is called the *accounting cycle*.
2. **BRM creates the bill.** The bill includes all of the charges collected in the monthly accounting cycle. Typically, a bill is created monthly, at the same time as the accounting cycle. However, you can bill for multiple months (for example, every two months, quarterly, or annually). In that case, the bill includes balance impacts from multiple accounting cycles. The collection of balance impacts into a bill is called the *billing cycle*.

Figure 1–5 shows how a billing cycle can include multiple accounting cycles:

Figure 1–5 Billing Cycle with Multiple Accounting Cycles



3. **BRM requests a payment.** BRM supports two types of payments:
 - You process *BRM-initiated payments* by automatically requesting payments from a payment processor.
 - You process *externally initiated payments* by sending invoices, receiving the payments, and processing the payments in batches.

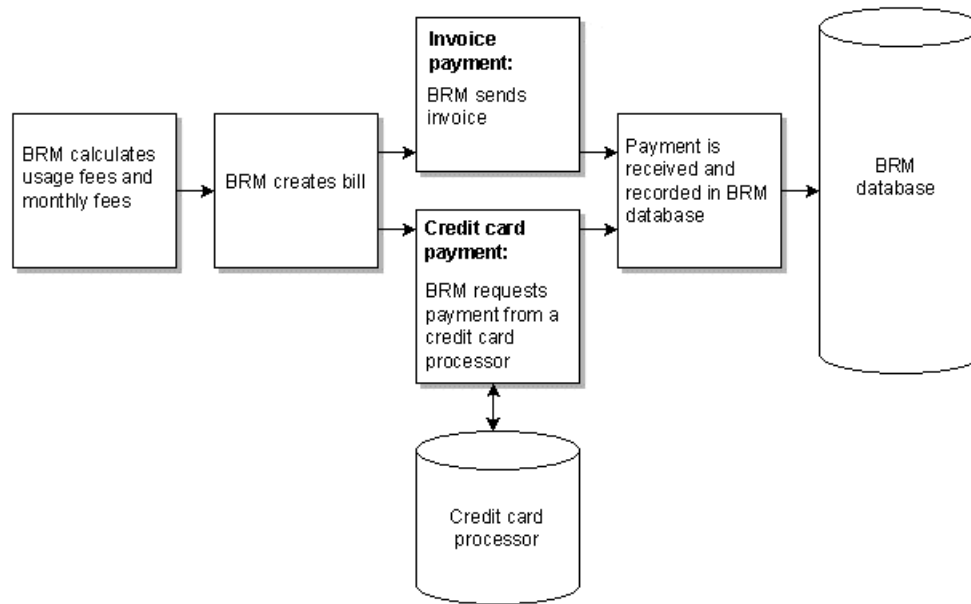
An invoice lists the events that were charged for, plus the customer's total account balance.

When the payment is recorded in the BRM database, the customer's account balance is updated automatically.

Note: BRM does not request a payment for some accounts. For example, the charges for accounts with nonpaying (subordinate) bill units are paid by a parent account, so those child accounts do not receive a request for payment. However, all accounts are billed; that is, all current balance impacts are compiled and stored as a bill in the BRM database.

For more information on payment processing, see "[About Managing Payments](#)".

Figure 1–6 shows how BRM compiles bills and requests payments from customers.

Figure 1–6 BRM Bill Compilation and Payment Request

Dealing with Corrections to Bills

Whenever the customer's account is modified in a manner that affects an invoice already sent to the customer, BRM enables you to generate a new type of bill and invoice:

- **Corrective bill.** A bill containing changes to item charges that affect the amount due in a finalized bill. Corrective bills are associated with the bill they correct, but they do not have the same bill number. However, they share the same object identifier in the BRM database with the bill they correct, enabling you to retrieve all the information on a set of bills.
- **Corrective invoice.** An invoice containing changes to an invoice already generated and sent to a customer. Corrective invoices are issued for simple corrections, such as address or language changes, and for charge corrections that affect the amount due in a finalized bill.

CSRs can use Customer Center to generate corrective bills for a specified account and billing period. The `pin_make_corrective_bills` utility generates corrective bills based on user-specified criteria such as bill number, account number, A/R item type, adjustment reason, threshold amount, date range, correction reason, and corrective invoice type. This utility can generate corrective bills individually or, when large numbers of bills are changed by processes such as rerating and backdating, in a batch.

The `pin_inv_accts` utility generates both regular and corrective invoices. By default, BRM supports either of the following formats for a corrective invoice:

- **Replacement invoice.** A **replacement invoice** includes all A/R actions, bill items (corrected and uncorrected), and events (corrected and uncorrected) for the specified billing period. Adjustments are included for the corrected items and events that apply to the current corrective bill.
- **Invoice correction letter.** An **invoice correction letter** includes only the A/R actions that occurred after the preceding bill for the specified billing period was finalized and only the corrected bill items and events that apply to the current corrective bill (plus their corresponding adjustments).

BRM provides a summary and a detailed version of both invoice formats.

About Billing Types

A customer's billing type determines the charges that are included in a bill. There are two billing types:

- With *open item accounting*, a customer is billed only for charges in the current bill. If a customer does not pay a bill, the next bill does not include charges for the bill that the customer did not pay.

You typically use open item accounting for invoice accounts. Each invoice includes the charges that apply to a single billing cycle. If a customer does not pay a bill, the next invoice includes the charges from the previous invoice.

- With *balance forward accounting*, a customer's bill includes all the charges that a customer owes, including those from previous billing cycles. If a customer does not pay a bill, the next bill includes the charges from the previous bill.

Accounts for customers who pay by credit card should always use balance forward accounting. Balance forward accounting is the default.

About Account Groups

You can create account groups where one customer pays the bill for a group of accounts. For example, a corporate account might include multiple dialup accounts, but all the billing is handled by the corporation.

How Billing Is Different from Rating

Rating calculates how much a customer should be charged for usage, subscription fees, and so forth. Rating determines the charges, stores the charges in bill items, and updates the customer's account balance.

Billing does not involve any measuring of events; billing simply collects existing balance impacts and creates a bill. The service or action that generated the events is irrelevant to creating the bill.

About Managing Payments

By default, BRM is configured to accept several different types of payment methods (for example, credit card, direct debit, cash, and check). Usually, each customer account is set up to use one payment method; however, you can set up an individual customer with multiple payment methods.

The payment method determines how the customer's payment is submitted to BRM: directly by a payment processor, such as a payment gateway, or manually by a payment clerk who uses Payment Tool to load a batch file sent from a bank or automated clearing house (an externally initiated payment).

If the submitted payment passes validation, the customer's account balance is updated accordingly. If the payment fails validation, the payment is handled according to the business policies implemented in your system configuration.

For example, you can configure BRM to process payments as follows:

- If a payment is paid early and in full, provide a credit or other incentive.
- If a payment fails financially, charge a fee.

- If a payment is not valid, save it to the payment suspense account so that the error can be fixed and the payment applied to the correct account at a later time. Payment suspense management must be enabled.

You can configure your business policies to handle overpayments and underpayments, and you can use Payment Tool or Customer Center to allocate payments to accounts or to specific bill and bill items. If you are not sure how to allocate a payment, you can leave it unallocated at the account.

About Managing Accounts Receivable

A/R includes all money owed to your business by customers who have bought your goods or services.

BRM manages A/R by storing different types of charges in *bill items*. For example, the *usage item* stores all the charges from usage fees, and the *cycle forward* and *cycle arrears* items store charges from different types of monthly subscription fees.

To manage A/R, you use Customer Center. Typical A/R tasks include:

- Displaying a customer's balance.
- Applying a credit or debit to a customer's balance.
- Managing billing disputes.
- Adjusting the tax on a bill.

About Managing Customers

Most customer management tasks are accomplished by CSRs using Customer Center or a third-party customer management system. You can set up Web-based customer self-care so that your customers can handle their own administration tasks, such as changing login names and passwords.

Typical customer problems include:

- Cannot log in to a service.
- Wrong amount on a bill.
- Forgotten password or login name.

Typical customer requests include:

- Change an address or phone number.
- Change an email address, password, or login name.
- Add, upgrade, or cancel services.
- Find out the current balance.

A common strategy for assigning tasks to CSRs is to use two levels of CSRs:

- Junior CSRs typically create accounts, check customer account balances, change customer login names and passwords, and answer common customer questions.
- Senior CSRs handle payment disputes, change credit limits, and change account status. Senior CSRs can also validate the work of junior CSRs (for example, by making sure that new accounts are created correctly and have all the necessary information).

About Managing Brands

You can use BRM to host services for *branded service providers* (BSPs). A BSP is a company that offers Internet services under its own brand name but does not actually operate the service.

If you are a host provider, you can set up and maintain many separate brands of Internet service in a single BRM system. You can:

- Create brands, which distinguish separate BSP accounts and other data, such as price lists.
- Control which personnel can access customer accounts and other data for each brand.
- Create brand-specific Web pages where customers can register and modify their accounts.

You need the optional Brand Manager component to host services for BSPs.

About Collecting Revenue Data

To record your business results, you export a monthly summary of all G/L data to your accounting software.

- You can create G/L reports that show revenue from any type of event (for example, all revenue generated from usage fees, monthly fees, and the revenue impacts of A/R adjustments).
- You can create reports based on revenue recognition (for example, billed, unbilled, earned, and unearned).

About Running Database Reports

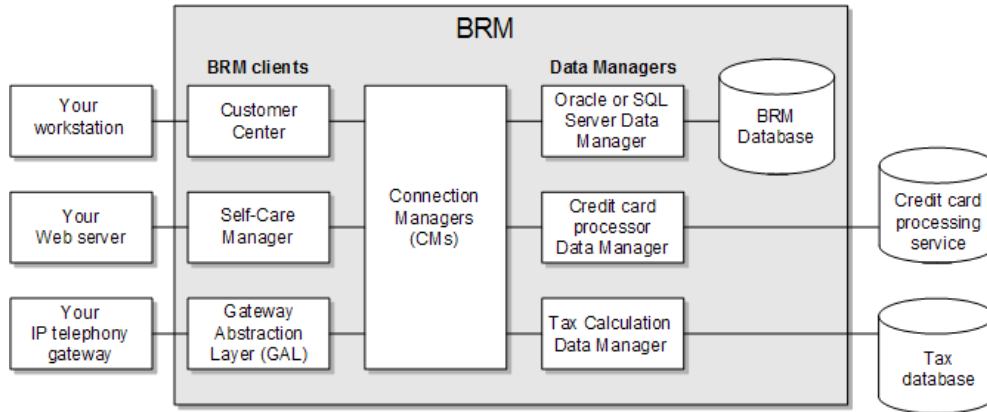
You can run database reports that show your financial data. For example:

- The **ARSummary** report summarizes the amounts currently due in a specified time period.
- The **CCExpire** report shows information about all accounts whose credit cards will expire in a specified period.
- The **UsageTimeCurrent** report displays the number of customer dialup sessions that are open in a specified time period.

What Is in a BRM system?

The BRM system is an integral part of your business. [Figure 1-7](#) shows how BRM components work with your other software components.

Figure 1–7 BRM Components within an Enterprise



In this figure:

- **BRM clients** interface to your services and capture data about customer service usage. For example, RADIUS Manager collects information from your terminal server about the customer’s session. BRM clients also include BRM applications, such as Customer Center and Pricing Center.
- **BRM Connection Managers (CMs)** process the data collected by BRM clients. For example, when a customer logs in, the Connection Manager determines how to validate the login name and sends a validation request to the BRM database.
- **BRM Data Managers (DMs)** translate BRM operations into a language that databases can understand. For example, a request for a login validation or credit card verification is translated into a database query.
- The **BRM database** stores customer account data and business data, such as your price list, G/L data, and records of all events that occur in the BRM system.

For more information about the BRM system, see "[BRM System Architecture](#)".

How the BRM Database Stores Data

The BRM database stores data in *objects*. For example, there are objects that store information about accounts, events, bills, services, and price list elements such as products.

Each object includes fields; each field contains a specific piece of data. For example:

- The account object includes fields that contain the customer’s first name, last name, and account status.
- Event objects include fields that contain the start and stop times of the event, the account that generated the event, and the balance impact of the event.
- Configuration objects contain data used by the system. For example, the `/config/glid` object includes the G/L IDs used for collecting G/L data.

Developers who create custom applications, or customize BRM source code, must know about objects and fields.

Caution:

- Always use the BRM API to manipulate data. Changing data in the database without using the API can corrupt the data.
 - Do not use SQL commands to change data in the database. Always use the API.
-

If you implement BRM, but are not a developer, it sometimes helps to know about objects and fields to understand how BRM works (for example, how BRM handles A/R).

How BRM Processes a Typical Billable Event

This example shows how BRM processes a billable event; in this case, a dialup session event.

Before your customers can log in to your service, you must do the following:

1. Install your terminal server and modem pool.
2. Install BRM and RADIUS Manager.
3. Create a price list that includes a plan for the IP service. This plan charges \$1 per hour for dialup access on weekdays, with free access on weekends.
4. Register the customer using the Internet Access plan.

When the customer logs in, the following actions occur:

1. The customer logs in at 10 p.m. RADIUS Manager collects the login name and password and sends them to the BRM database to authenticate the user; that is, to see if this user has an account.
2. The login name and password are found in the database. This identifies the account and enables BRM to check if the customer is authorized to use the IP service. In this case, the customer has purchased a plan that includes the service, paid all bills, and has not exceeded a credit limit, so authorization is granted.
3. The customer connects to the Internet for three hours. When the customer logs off, BRM records the start and stop times of the session in a session event:
 - Start time: Friday, 10 p.m.
 - End time: Saturday, 1 a.m.
4. BRM rates the event. The connection time was for three hours, but one hour was on the weekend, when Internet access is free. The total usage fee is \$2.
5. BRM adds \$2 to the usage billing item, which is displayed in the customer's account balance.

To bill the customer, the following actions take place:

1. The customer's billing date is January 15. When you run the daily billing utilities on January 16, a billing utility determines that the customer's account is ready for billing.
2. BRM creates the customer's bill, which includes the \$2 usage fee.
3. Because the customer pays by credit card, another billing utility creates a credit card transaction for the total amount of the bill and sends the transaction to a credit card processor.

4. The credit card processor authorizes the payment and tells BRM that the payment was made.
5. BRM records the payment and updates the customer's account balance.
6. BRM automatically sends an email invoice to the customer, notifying the customer of the month's charges. This invoice is for the customer's information only.

How a Typical Internet Service Provider Uses BRM

This example shows how a typical Internet service provider (ISP) implements BRM.

ISP-1 is a full-service ISP company that targets local and regional markets. ISP-1 provides dialup and ISDN Internet access, email, and Web hosting.

ISP-1 chose BRM primarily for the following reasons:

- BRM can handle the rapid growth that ISP-1 is experiencing.
- ISP-1 can offer online registration.
- ISP-1 can offer account self-management.
- ISP-1 can add and change price plans easily.

ISP-1 is currently in negotiations to purchase another ISP. In that case, it can use the optional branding functionality to manage existing accounts separately from accounts inherited from the other ISP.

Customer Registration

ISP-1 has a two-fold approach to finding and registering customers:

- Using various advertising methods, ISP-1 provides a toll-free number that customers can call to register. These customers talk to CSRs, who use Customer Center or a third-party customer management system to create the account.
- ISP-1 distributes a CD that contains a customized registration kit that connects to the ISP-1 self-care Web site. In addition, ISP-1 maintains a Web site for customers who already have Internet access.

To encourage online registration, the purchase fee is waived if the customer registers online. To implement this feature, plans are created without a purchase fee, and the purchase fee is added as an override by the CSR.

Custom Profile Information

To handle corporate accounts, such as those that use the T-1 service, ISP-1 keeps track of sales representative and account owner information. To manage this information, ISP-1 uses profiles to add an **Account Owner** field and an **Account Representative** field.

Managing Customer Accounts

ISP-1 offers a subset of the available BRM administrative functions on its customer self-care Web site. Customers can change passwords, look at their balance, update credit card information, and change name and address information.

All of these customer management tasks can also be handled by CSRs using Customer Center or a third-party customer management system. In addition, CSRs handle product upgrades and billing disputes. CSR support is managed as follows:

- Some CSRs handle account creation and product upgrades only. Other CSRs handle simple maintenance tasks, such as changing passwords and billing information. Advanced CSRs handle billing disputes.
- There are no charges for administrative tasks, such as changing a password.

In addition, ISP-1 has a technical support group dedicated to maintaining its ISP network and dealing with access issues. This technical support group has extended hours, seven days per week.

Pricing

ISP-1 offers three basic plans:

Standard

This minimal plan offers Internet access and Web space, but no email. The target customer is a casual-use individual.

- \$6.95 per month for 20 hours of Internet access and 5 MB Web space.
- Usage over 20 hours per month is charged at \$2 per hour.
- \$10 purchase fee.
- If a customer cancels the plan, the monthly fee is prorated; the customer does not have to pay for the part of the month that was not used.

Unlimited

This plan offers Internet access, Web space, and email. The target customer is a family account.

- \$19.95 per month for unlimited hours, 4 mailboxes, 5 MB Web space.
- \$5 per month for each additional mailbox.
- \$10 purchase fee.
- If the customer cancels the plan, the monthly fees are prorated; the customer does not have to pay for the part of the month that was not used.

Premium

This plan offers Internet access, Web space, email, and additional premiums. The target customer is a high-use individual or family account. The customer pays a yearly rate that is considerably less than the yearly cost of a monthly rate. In exchange, there is a lack of proration and a cancel fee for early cancellation.

- \$100 per year for unlimited hours, 5 mailboxes, 10 MB Web space, and access to free Internet software downloads.
- \$20 purchase fee.
- \$50 cancellation fee if the customer has owned the account for less than one year.
- There is no proration if the customer cancels before owning the account for less than one year.
- The yearly fee is paid in advance. The customer has the option of paying quarterly or yearly.

G/L Accounts for Pricing Elements

Revenue is tracked in the following G/L accounts:

- All usage fees are recorded in the Usage fees G/L account.
- Monthly fees for the Standard and Unlimited plans are recorded in the Monthly subscription G/L account.
- The yearly fee for the Premium account is recorded in the Premium subscription G/L account.
- There is a separate G/L account for each type of purchase fee: Standard purchase fee, Unlimited purchase fee, and Premium purchase fee.
- The Premium cancel fee is recorded in the Premium cancel fee G/L account.

Billing and Payments

ISP-1 accepts payment by check and credit card.

Accounts that pay by credit card use anniversary billing (the billing day of month is the day that the account was created), although customers can change their billing day of month. (This change is handled by a CSR.) Premium customers cannot change their billing day of month.

Invoice accounts are paid by check. Check payments are processed by CSRs using Payment Tool. Payments that cannot be validated are compiled into a separate batch of payments and are processed by a member of the accounting department.

Invoice accounts are all billed on the 15th of each month. A customer changing the payment method from invoice to credit card can choose a different billing day of month.

About the BRM Client Applications

BRM includes the following applications that you use to manage customers, process payments, and configure your system:

- To register customers, manage customer information, and handle billing disputes, use Customer Center.

[Figure 1–8](#) shows the full Customer Center interface, with the **Summary** tab active. By default, this tab is active when you first open an account:

Figure 1–8 Customer Center

The screenshot shows the Customer Center application window. The title bar reads "Customer Center". The menu bar includes "File", "Edit", "View", "A/R", and "Help". The toolbar contains icons for "Refresh", "New Account", "Search", and "Notes", along with "US Dollar" and "East Coast Enterprises" buttons. The main content area displays account information for "Samantha Miranda, Account 0.0.0.1-15759" with a status of "Active".

On the left, a sidebar lists actions and other accounts:

- 0.0.0.1-38582 Ellen Simone
- 0.0.0.1-15759 **Samantha Miranda**
- 0.0.0.1-17623 Gary Cornwall

The main content area has tabs for "Summary", "Balance", "Payments", "Product", "Service", "Hierarchy", and "Sponsorship". The "Summary" tab is active, showing:

Contact Information	Account Summary
Additional contacts: 0	Account status: Active
Samantha Miranda 475 Bergenline Ave. Union City, NJ 07087	Created: Feb 9, 2000
	Deferred actions: 1
	Customer type: Silver
	Security code: _____
	Language: English(US)

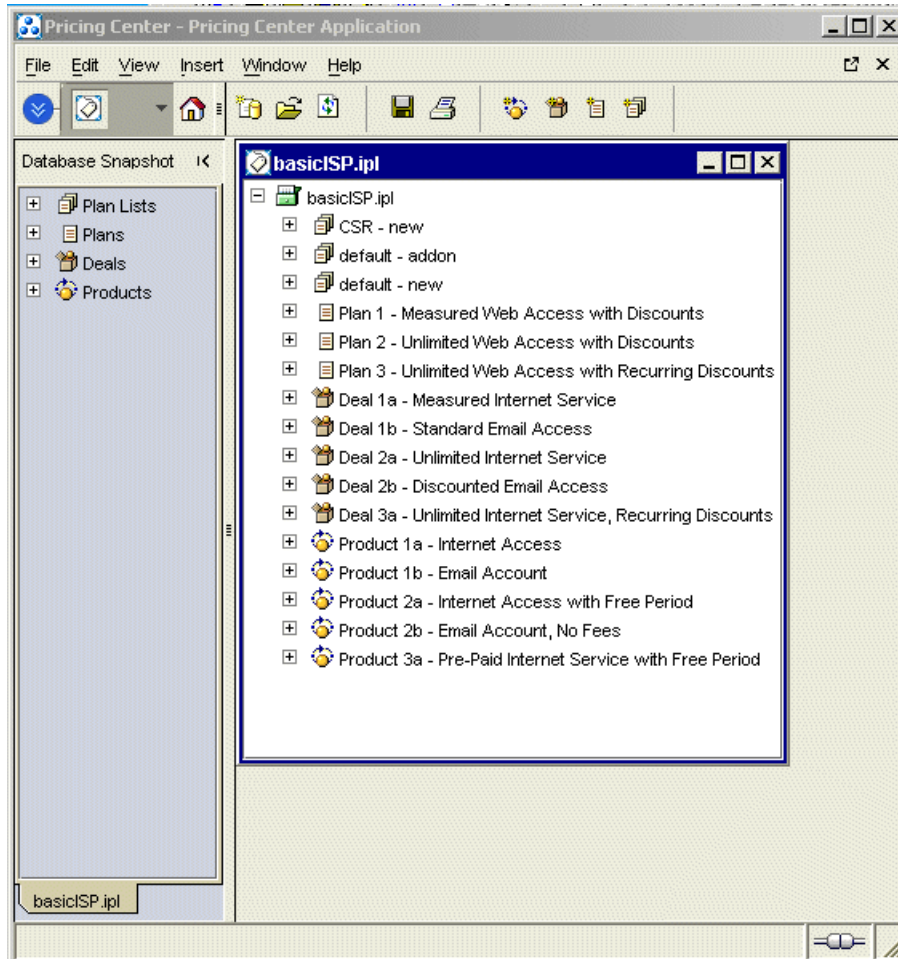
Below this is the "Currency Balance Summary" section:

Amount due for all bills:	\$98.37
Adjustments/Payments not applied:	(\$10.00)
Due now:	\$88.37
Credit limit:	\$500.00
Unresolved disputes:	\$0.00
Payment type:	Invoice
Current billing cycle:	Nov 9, 2001 - Dec 9, 2001

- To set up your price list, use Pricing Center.

Figure 1–9 shows how Pricing Center displays a typical price list:

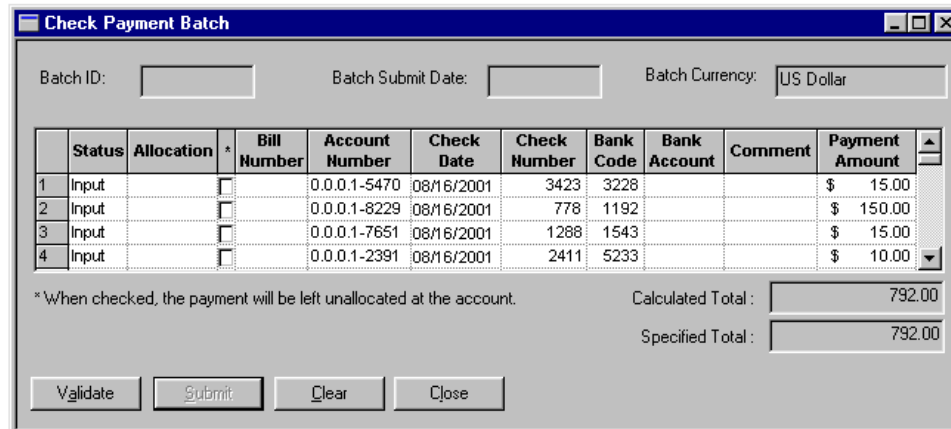
Figure 1–9 Pricing List Displayed in Pricing Center



- To process externally initiated payments, such as payments by check, and to process payment reversals and refunds, use Payment Tool.

Figure 1–10 shows the Check Payment Batch window, where you process a batch of check payments:

Figure 1–10 Check Payment Batch Window



- To perform business management tasks, such as creating brands and setting up CSR and other access privileges, use Configuration Center.

BRM developers and administrators use Configuration Center for advanced business management tasks. Configuration Center can contain any combination of the components listed in [Table 1-1](#), depending on what you have installed:

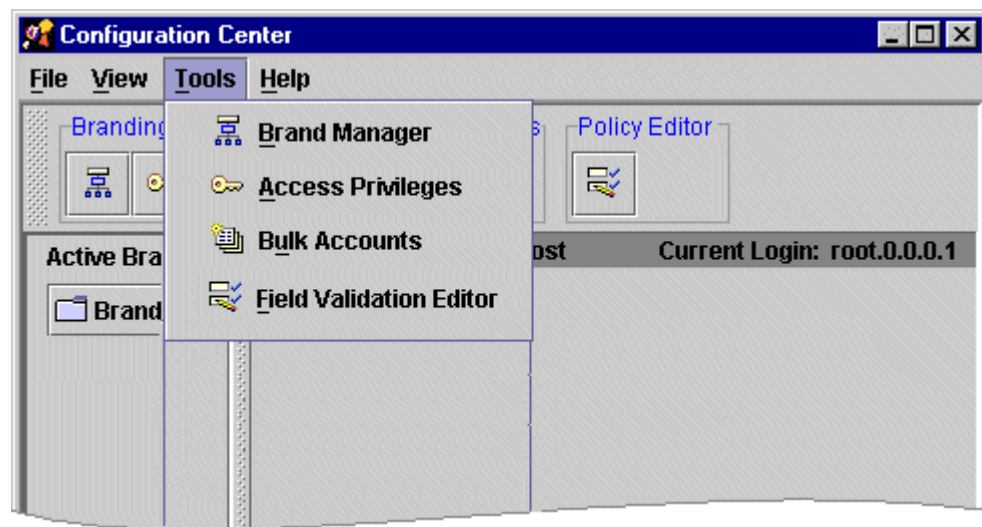
Table 1-1 *Combination of Components*

Component	Application
Policy Editors	Field Validation Editor
Brand Manager	Brand Manager Access Privileges

Brand Manager is an optional component that you acquire separately.

In [Figure 1-11](#), Configuration Center contains all its possible applications:

Figure 1-11 *Configuration Center*



About BRM Command-Line Utilities

In addition to the client applications, BRM includes command-line utilities. These utilities typically have a narrow purpose. For example:

- Use the `pin_bill_accts` utility to run daily billing.
- Use the `load_pin_glid` utility to load G/L IDs into the database.
- Use the `pin_restore` utility to restore archived data.

Implementation Process

A typical BRM implementation process includes these tasks:

1. Installing a test system.
2. Creating services (for example, setting up the hardware necessary for your services, such as terminal servers and IP telephony gateways, and installing and configuring optional service integration components).

3. Creating your price list, including creating resources, and setting up event maps and ratable user metrics (RUMs) so your system can capture and rate customer service usage.
4. Setting up customer registration.
5. Testing your implementation on the test system.
6. Installing a production system.

Note: You should keep a small test system available to test custom components. This test system should be the same as the production system, but it only needs to have a few accounts.

Ways to Use and Customize BRM

You can customize BRM to change the default BRM business functionality. By customizing BRM, you can change how BRM responds to customer activity. For example, you can do the following:

- Change the net due date for invoice accounts.
- Specify the reasons for automatically inactivating an account.
- Specify how and when to validate credit cards.
- Collect additional data about a customer (for example, the customer's bank ABA number, employee number, or member referral information).

You customize business functionality by editing configuration files and by customizing BRM source code. You do not need to be a programmer to edit configuration files, but you must be a programmer to customize source code.

You can use several methods to use and customize BRM features:

- Use BRM client applications, such as Customer Center and Business Configuration Center, to manage customers, record payments, and change BRM default behavior. Client applications run on the Windows platform and have a graphical user interface.
- Use BRM command-line utilities such as the **pin_collect** utility and the **load_pin_glid** utility to run billing utilities and load data into the BRM database.
- Edit configuration files to change the default BRM behavior and tune system performance.
- Customize BRM source code. This requires programming.

Caution:

- Always use the BRM API to manipulate data. Changing data in the database without using the API can corrupt the data.
 - Do not use SQL commands to change data in the database. Always use the API.
-
-

About Integrating Your Services with BRM

A service is a capability that you provide to customers, such as Internet access, email, or telephony.

Integrating your services with BRM is a key part of your BRM implementation. It enables you to do the following:

- Capture, measure, rate, and charge for service usage.
- Authenticate and authorize customers, which enables you to activate and inactivate accounts automatically (for example, as soon as a credit limit is reached).
- Charge customers based on levels of service. For example:
 - You can charge different fees for different connection speeds.
 - You can charge based on the number of mailboxes.
- Provision services based on the products that a customer owns. For example, you can create products that determine the size of a customer’s mailbox or connection bandwidth.

To integrate BRM with your services, you use service integration components, such as RADIUS Manager. For example:

- When customers log in, RADIUS Manager uses data in the BRM database to authenticate the customer’s login name and password.
- RADIUS Manager uses BRM data to authorize service usage. For example, if the customer has exceeded a credit limit, BRM can deny authorization to use the service.
- RADIUS Manager collects terminal server accounting data, such as session start and stop times. BRM uses this data to measure, rate, and charge for service usage.

Figure 1–12, Figure 1–13, and Figure 1–14 show how RADIUS Manager authenticates, authorizes, and collects data for rating.

Figure 1–12 RADIUS Manager Authentication Process

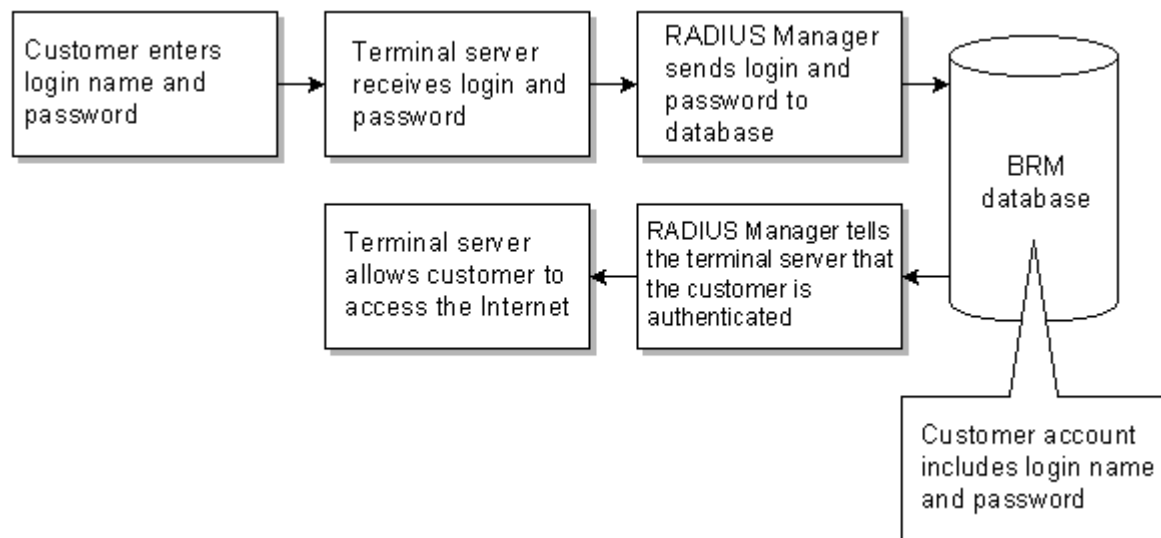
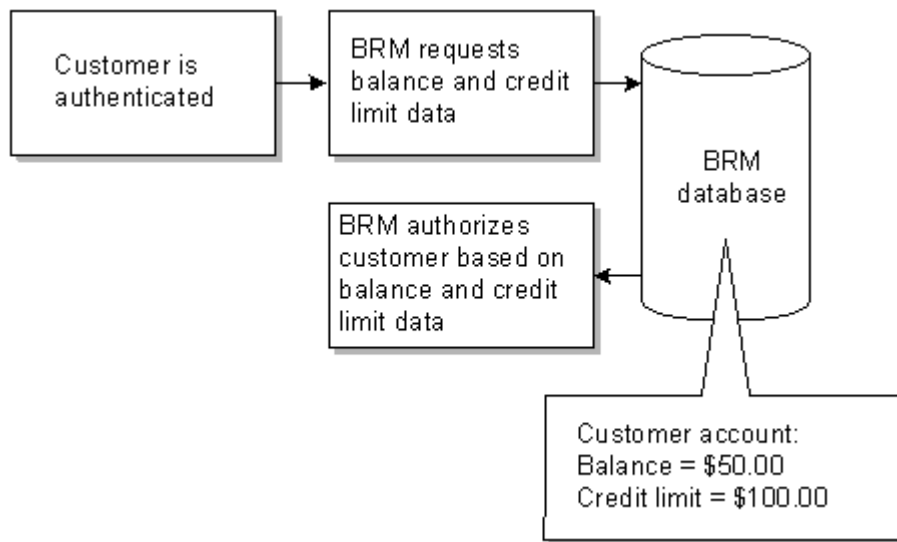
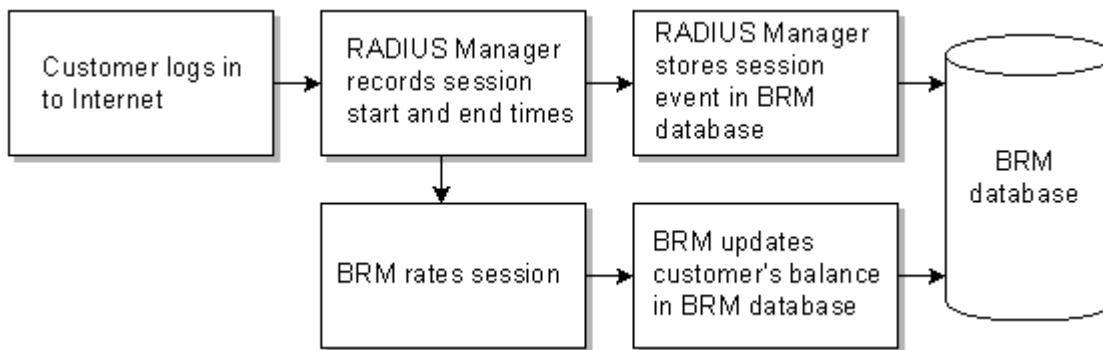


Figure 1–13 RADIUS Manager Authorization Process**Figure 1–14 Collecting Usage Data with RADIUS Manager**

About Creating Custom Account Management and Billing Tools

Using the BRM APIs, you can create custom client applications and customer interfaces. For example:

- Web-based applications that customers use to change their account status, upgrade their services, and pay their bills.
- An online registration package requiring no existing Internet access and distributed on a variety of shareware CDs.
- An application that runs nightly, searches for accounts that have a free trial account, and sends email when the free trial account is about to expire.
- An application that runs nightly to check the age of an account. If the account is over two years old, the customer is automatically upgraded to a discounted plan.
- An application that checks customer profiles and applies a discounted rate for teachers.
- An application that monitors the customer's disk space on a server and charges more if allotted disk space is exceeded.

About Connecting BRM to External Data Processing Sources

BRM is modular and extensible by design. Many customers add custom interfaces to external components. Some customers use external components instead of standard BRM components. For example, some customers use BRM rating but use their own proprietary billing system. Other customers use a third-party customer management system.

Examples of external integrations:

- Tax calculation services.
- Credit card processing services.
- Legacy billing systems.
- Customer management systems.
- Credit checking services.
- Accounting applications to handle A/R and G/L data.
- Business information distribution services (for example, services that automatically distribute the results of database reports).
- Fulfillment services that send welcome letters and goods such as CDs.
- Invoice processing, printing, and mailing companies.
- Offsite customer service organizations.

About the Data Displayed in BRM Client Applications

You can change some information displayed in client applications to support your business needs. For example:

- Change the list of plans or edit your plan lists.
- Change the predefined reasons that describe why a change was made to an account.
- Add a payment method.
- Change the list of currencies you support.

About BRM Defaults

Almost all business policies have a default implementation. For example, the default billing cycle is one month. You can change the default implementation by editing configuration files, or customizing policy code.

When Is Programming Required?

Most BRM features can be customized without any programming. For example, most price list implementations, CSR-based registration, and billing setup requires no programming to customize. The following customizations require some programming:

- **Creating services.** If you offer a service that is not supported by default in BRM, such as a fax service, you must create a custom service to capture and rate fax-related events.
- **Modifying some BRM defaults.** In some cases, you must modify policy source code to change BRM behavior. For example, to rate Internet access by the number of bytes downloaded rather than by connection time, you must edit source code.

- **Creating custom applications.** Some BRM implementations can be greatly enhanced by creating simple applications that manipulate data in the BRM database. For example, you can write applications to do the following tasks:
 - Alert customers that their free hours are almost used up.
 - Search for customers who have late payments.
- **Customizing applications.** To add entirely new tabs or make major changes to functionality, you must use the Customer Center SDK.
- **Supporting a legacy system.** If you already have data stored in a database, you can write a Data Manager (DM) to provide an interface to that database.
- **Customizing event notification.** You use event notification to set up processes to run automatically. For example, you can customize BRM to send an email message to a customer automatically when a credit limit is nearly expired.
- **Implementing advanced pricing features.** Some pricing features (for example, product-level provisioning) require some programming for the initial setup.

Summary of Key Terms

Whatever you do with BRM, you should know the following terms:

Event

An online action that can be recognized by BRM and recorded in the BRM database. For example:

- Connecting to the Internet.
- Placing an IP phone call.
- Changing a password.
- Creating an account.

Billable event

An event that you charge customers for, such as an Internet connection or an IP phone call.

Rating

A BRM process that determines how much to charge customers for the use of your services.

Rating is performed on events. To rate an event, BRM does the following:

1. Measures the event.
2. Applies a charge to the resulting measurement.
3. Adds the charge to the customer's account balance.

Balance impact

The effect that rating an event has on a customer's account balance.

- A credit balance impact adds to the balance; the customer owes you more.
- A debit balance impact subtracts from the balance; you owe the customer.

Resource

An asset of economic value, such as US dollars, or a non-currency asset, such as Internet access hours.

Price list

A set of plans that specifies when, how, and how much to charge customers.

Billing

A BRM process, typically run daily, that compiles the fees incurred by customers into bill objects stored in the BRM database.

Accounting cycle

A one-month period during which all of a customer's charges are calculated, but not billed. Charges are recorded in the customer's balance. The timing of the accounting cycle for a customer is based on the customer's billing day of month.

Billing cycle

The time period during which charges accumulate in an account before a bill object is created and a request for payment is generated. One billing cycle can contain one or more accounting cycles.

Account

A collection of data stored in the BRM database about a specific customer. Account information includes:

- Information about the customer (for example, name and address).
- Which services the customer can use.
- Billing information, such as method of payment and billing day of month.

Service

A commodity sold by your company and that your customers can purchase and use, such as an email address or an Internet connection. When customers use a service, they generate billable events, which you use to bill customers.

A customer account can have an unlimited number of services.

Object

A record in the BRM database. All data is stored as objects. For example:

- The account object includes fields that contain the customer's first name, last name, and account status.
- Event objects include fields that record the start and stop times of the event, the account that generated the event, and the balance impact of the event.

Using BRM with Oracle Application Integration Architecture

This chapter is an overview of how to set up your Oracle Communications Billing and Revenue Management (BRM) system to integrate with external applications using Oracle Application Integration Architecture (Oracle AIA).

You can integrate BRM with applications such as customer relationship management (CRM) and financial management software.

About Oracle Application Integration Architecture

Oracle AIA is a set of products that enable you to set up and orchestrate cross-application business processes so that multiple applications can work together. Oracle AIA runs on top of Oracle Fusion Middleware.

Oracle AIA for Communications pre-built integrations are pre-built packaged process integrations between specific Oracle applications, including Siebel CRM and BRM, based on Oracle AIA. For example, you can use the pre-built integrations to do the following:

- Create products, discounts, and sponsorships in BRM and synchronize them with Siebel CRM, where they can be packaged and purchased by customers.
- Create BRM accounts from accounts created in Siebel CRM.
- Access billing information from BRM to display in Siebel CRM.
- Export general ledger (G/L) data from BRM for import into Oracle Financials.

For more information, see Oracle AIA and the AIA for Communications pre-built integration documentation on the Oracle Technology Network Web site:

<http://www.oracle.com/technology/>

Setting Up BRM to Work with Oracle Application Integration Architecture

To set up BRM to work with Oracle AIA:

- Install and configure the required BRM components. See "[Installing and Configuring the Required BRM Components](#)".
- Integrate BRM features with your external CRM application. See "[Integrating BRM Features with External CRM Applications](#)".

Installing and Configuring the Required BRM Components

To integrate BRM with Oracle AIA:

- Install, deploy, and configure JCA Resource Adapter in a J2EE application server. The adapter is the point of connection between BRM and external applications. Requests for information come to the adapter, which then calls BRM opcodes and returns data to the external application.
- Install and configure Synchronization Queue Data Manager (DM). This DM enables you to synchronize pricing data changes in the BRM database with external applications. For example, when a product is created or modified, Synchronization Queue DM sends the data to a database queue. The data in the queue can then be retrieved by an external application.
- Run the **pin_ledger_report** utility to export G/L data to XML files. This data can then be imported into financial management software such as Oracle Financials.

Integrating BRM Features with External CRM Applications

You can integrate the following BRM features with your external CRM applications:

- Collections. See "[Integrating Collections with External CRM Applications](#)".
- Friends and family promotions. See "[Integrating Friends and Family Promotions with External CRM Applications](#)".
- Invoicing. See "[Displaying Siebel CRM Promotion Names on Invoices](#)".

Integrating Collections with External CRM Applications

External CRM applications can track and manage collections activities, such as sending dunning letters, in BRM through the Oracle AIA architecture. Collections data is synchronized between the external CRM application and BRM as follows:

- The external CRM application updates the status of a collections action in BRM by calling the Collections Manager API through Oracle AIA and JCA Resource Adapter.
- Collections Manager notifies the external CRM application when a collections activity occurs, such as an account entering collections, by using the event notification system.
- Oracle AIA retrieves data from Collections Manager by reading views on the BRM collections tables.

To integrate collections with your external CRM:

1. Install the Agent-Assisted Billing Care Process Integration Pack.
2. Configure your external CRM application to send the status of collections actions to the `PCM_OP_COLLECTIONS_SET_ACTION_STATUS` opcode. The external CRM application sends information to the opcode through Oracle AIA and JCA Resource Adapter.

The attributes required to call the opcode are listed in the *BRM_Home/apps/brm_integrations/wsdls/BRMCollectionsServices.wsdl* file.

3. Configure BRM to publish **CollectionsAction** business events to the Oracle AQ database queue. See the discussion in *BRM Synchronization Queue Manager*.

4. Configure your external CRM application to retrieve the **CollectionsAction** business event from the Oracle AQ database queue. See the discussion in *BRM Synchronization Queue Manager*.
5. Configure the BRM **pin_collections_process** utility to publish **CollectionsInfoChange** business events to the Oracle AQ database queue:
 - a. Open the **pin_collections_process** configuration file (*BRM_Home/apps/pin_collections/pin.conf*) in a text editor.
 - b. Add the following entry to the file:


```
- pin_collections_process publish_run_details 1
```
 - c. Save and close the file.

See *BRM Collections Manager* for more information.

Integrating Friends and Family Promotions with External CRM Applications

External CRM applications can track and manage friends and family promotions in BRM through the Oracle AIA architecture.

When customers order a friends and family promotion, BRM stores information about the promotion in a product provisioning tag. During the pricing data synchronization process, the Synchronization Queue DM publishes the provisioning tag and its associated product in a **ProductInfoChange** business event to the Oracle AQ database queue. The external CRM application can then retrieve the **ProductInfoChange** business event from the database queue and update the information in its system.

To integrate friends and family promotions with your external CRM application:

- Configure BRM to publish **ProductInfoChange** business events to the Oracle AQ database queue.
- Configure your external CRM application to retrieve the **ProductInfoChange** business event from the Oracle AQ database queue.

For more information, see "About Sending Account Data to Pipeline Manager" in *BRM Synchronization Queue Manager*.

Displaying Siebel CRM Promotion Names on Invoices

You can set up your system to display Siebel CRM promotion names on customer invoices.

To include Siebel CRM promotion names on BRM invoices:

- Ensure that BRM is configured to display promotion details on invoices.
- Configure Siebel CRM to send information about the promotion to the **PCM_OP_SUBSCRIPTION_SET_BUNDLE** opcode. The application sends information to the BRM opcode through Oracle AIA and JCA Resource Adapter.

The attributes required to call the opcode are listed in the *BRM_Home/apps/brm_integrations/wsdl/BRMSubscriptionServices.wsdl* file.

- Use an invoice template that displays promotion names and details.

For more information, see "Adding Siebel CRM Promotion Names to Invoices" in *BRM Designing and Generating Invoices*.

Integrating BRM Features with External CRM Applications in a Multischema System

You can integrate the following BRM features with your external CRM application in a multischema system:

- Account Migration Manager (AMM). See "[Integrating Account Migrations with External Applications in a Multischema System](#)" for more information.
- Collections. See "[Integrating Collections with External Applications in a Multischema System](#)" for more information.

Integrating Account Migrations with External Applications in a Multischema System

Account migration is synchronized between the external CRM application and BRM in a multischema system as follows:

1. AMM populates the `MIGRATED_OBJECTS_T` cross-reference table in the primary BRM database with the batch ID and old and new POID values of all the objects that have successfully been migrated.
2. After successfully migrating a group of accounts from one schema to another, BRM generates **AccountInfoChange** business events. These events are sent to the Enterprise Application Integration Data Manager (EAI DM) by using an event notification message.
3. The EAI DM publishes the **AccountInfoChange** business events to the Oracle AQ database queue.
4. Oracle AIA retrieves the **AccountInfoChange** business events from the Oracle AQ database queue and updates the information in the AIA database.
5. Oracle AIA updates the AIA cross-reference table in the AIA database.
6. Oracle AIA deletes the entries in the `MIGRATED_OBJECTS_T` cross-reference table in the primary BRM database after reading the entries for a particular batch.

To integrate account migration with the external application in a multischema environment:

Important: Before running AMM, verify that all the collections related Oracle Data Integrator jobs have been completed successfully. For more information on the Oracle Data Integrator jobs, see the Oracle Data Integrator documentation.

1. Install the Oracle AIA for Communications pre-built integration that integrates BRM with your external application. For more information on the Oracle AIA for Communications pre-built integrations, see the Oracle AIA documentation.
2. Add the **publish_migrated_objects** entry to the AMM **Infranet.properties** file by doing the following:
 - a. Open the `BRM_Home/sys/amt/Infranet.properties` file in a text editor
 - b. Add the following entry:

```
publish_migrated_objects = value [, value2 ...]
```

where *value* is a comma-separated list of storable classes whose objects are stored in the `MIGRATED_OBJECTS_T` cross-reference table.

Note: If you do not set this entry, AMM will not integrate with an external application in a multischema environment.

- c. Save and close the file.
3. Configure BRM to generate **AccountInfoChange** business events to the Oracle AQ database queue. See the discussion on installing and configuring the Synchronization Queue DM in the BRM documentation.
4. Configure your external application to retrieve the **AccountInfoChange** business event from the Oracle AQ database queue. See the discussion on retrieving specific events from the Oracle AQ database queue in the BRM documentation.

Integrating Collections with External Applications in a Multischema System

To integrate collections with your external application using Oracle AIA in a multischema system, BRM populates the custom views in BRM collections tables with the `/collection_actions` POID schema number, hard-coded as 0.0.0.1 regardless of the schema where the account resides. For `/account` and `/billinfo` objects the external application reads the schema where the account is residing, whether it is a single-schema or a multischema environment.

See "[Integrating Collections with External CRM Applications](#)" for information on integrating collections with your external application.

Creating Products and Discounts for an External CRM

When you create products and discounts to be used in an external CRM, do not add them to deals and plans as described in the BRM documentation and in the Pricing Center Help. External CRMs, such as Siebel CRM, do not use BRM deals and plans. They create their own bundles with the BRM products and discounts.

You use the `pin_export_price` utility to export products and discounts from BRM to external CRMs.

Synchronizing Products That Have Multiple Rate Plans with External CRM Applications

BRM supports synchronizing a product that includes multiple rate plans configured using the rate plan selector in Pricing Center when integrated with external applications using Oracle AIA.

Additionally, the BRM customer, subscription, and rating components enable Oracle AIA order management to specify the price list name in the BRM product offerings that are used during the BRM rating process.

Validating Customer Contact Information

During account registration, BRM validates the format of customer contact information, such as phone numbers, before creating the account in the BRM database. For example, when a customer or customer service representative enters a phone number, BRM validates that the phone number includes the correct number of digits or contains parentheses around area codes.

By default, BRM accepts nine different telephone number formats. If your external CRM does not use one of the default formats, you must either:

- Add your telephone number format to the list of acceptable formats by using the Field Validation Editor.
- Configure BRM to accept any telephone number format. In the Field Validation Editor, set the telephone format mask to the following:

`^.*$`

Note: BRM stores customer contact data in the format passed in by the external CRM. BRM also uses this same format when creating customer invoices. You must configure your external application to pass in phone numbers in the format you would like displayed on customer invoices.

For information about setting the valid telephone number formats, see "Customizing Registration" in *BRM Managing Customers*.

BRM System Architecture

This chapter describes the Oracle Communications Billing and Revenue Management (BRM) system components.

Before reading this chapter, read "[Introducing BRM](#)".

About the Four-Tier Architecture

The BRM system uses a four-tier architecture consisting of the following tiers:

- [Application Tier](#)
- [Business Process Tier](#)
- [Data Management Tier](#)
- [Data Tier](#)

[Figure 3-1](#) shows the BRM system architecture:

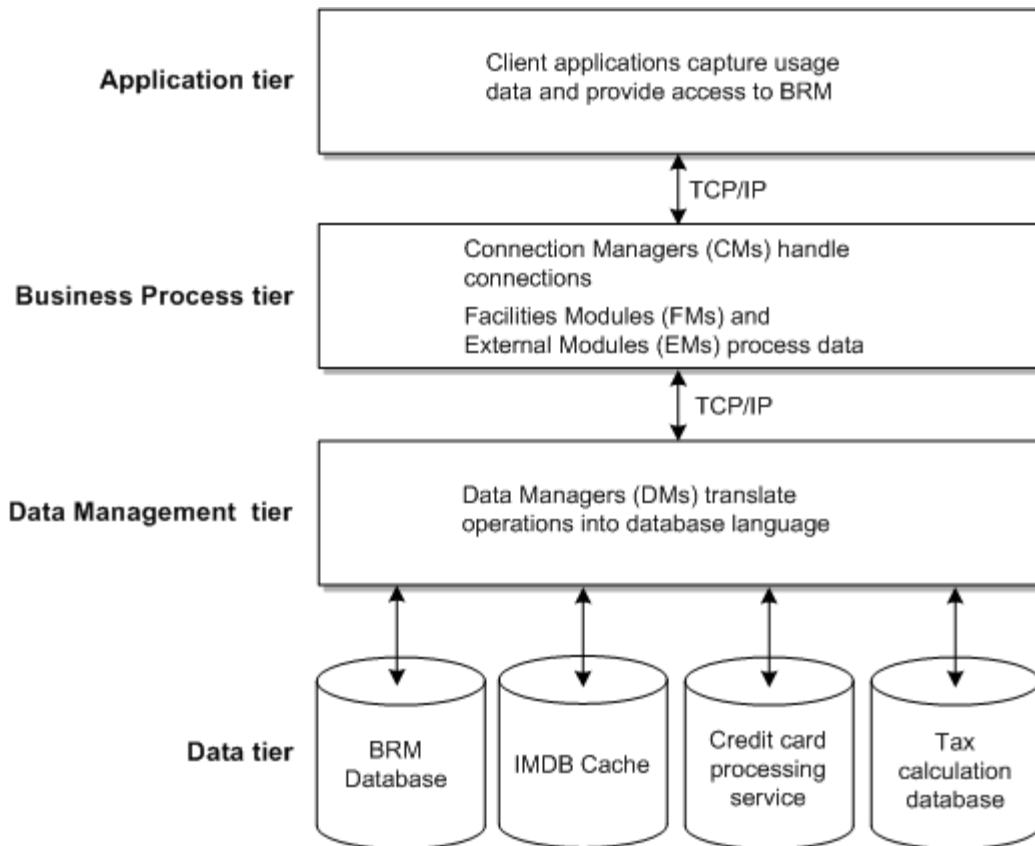
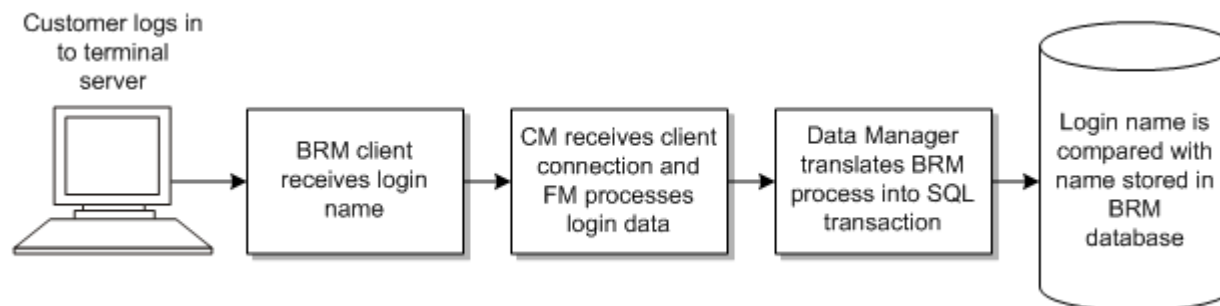
Figure 3–1 BRM System Architecture

Figure 3–2 shows how the BRM components process a customer login.

Figure 3–2 BRM Customer Login Process

Application Tier

The application tier consists of the following types of client applications:

- Client applications such as Customer Center and Pricing Center that you use to register and manage customers and their accounts
- Customer account management applications used by customer service representatives (CSRs), such as applications that capture data from customer service usage. For example, the RADIUS Manager captures information from your terminal server about a customer’s Internet session.

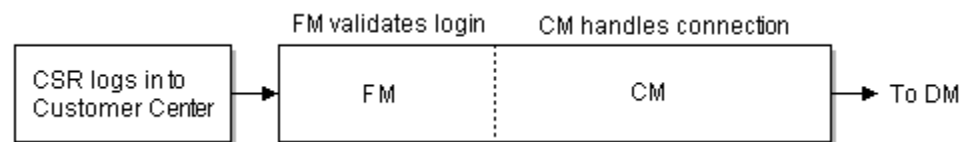
- Service integration applications such as Global System for Mobile Communications (GSM) Manager and General Packet Radio Service (GPRS) Manager
- Custom applications that you write to manage customers or support custom services and integrate your applications with BRM

The following example shows how applications connect to the BRM system.

Example: Customer Center

When a CSR logs in to the Customer Center application, a connection is made to a Connection Manager (CM). All customer management activities that the CSR performs with Customer Center are processed by the Facilities Modules (FMs) that are linked to the CM. [Figure 3–3](#) illustrates the process flow of a CSR login to BRM.

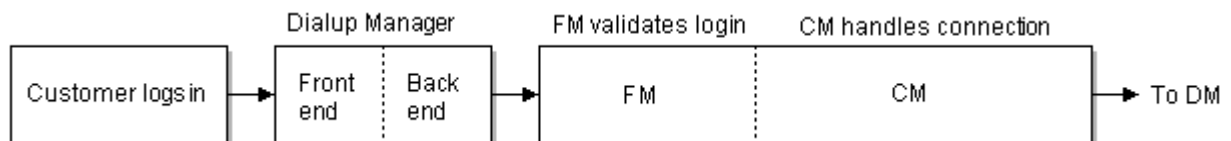
Figure 3–3 CSR Connection Process



Example: RADIUS Manager

When a customer connects to the Internet, a RADIUS Manager front-end process accepts the dialup connection. A RADIUS Manager back-end process serves as a BRM client application and sends the login information to the CM. [Figure 3–4](#) illustrates the RADIUS Manager example.

Figure 3–4 RADIUS Manager Connection Process



Note: If you have client applications running on the same server as the CM or DM, you still must use Transmission Control Protocol (TCP)/Internet Protocol (IP) to connect to the client.

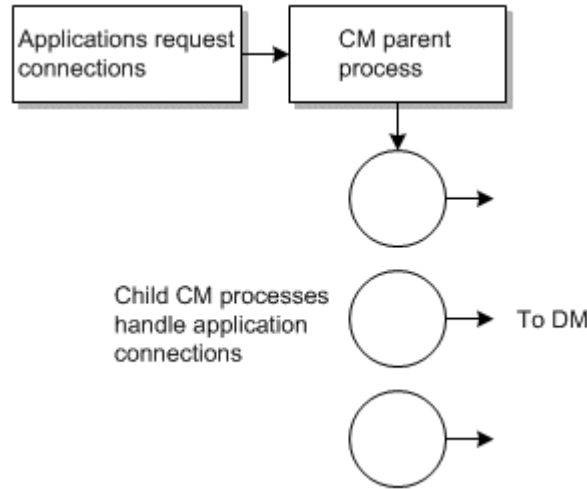
Business Process Tier

The business process tier consists of the following components:

- **Connection Managers (CMs):** CMs provide an interface between clients and the rest of the system. All client applications connect to the BRM system through a CM.
- **Facilities Modules (FMs):** CMs include FMs that process the data captured by the client. For example, when a user logs in, FMs process the login name and password. See "[About Facilities Modules \(FMs\)](#)".
- **External Modules (EMs):** EMs serve similar functions to FMs but must be started separately as a service or process. See "[About External Modules \(EMs\)](#)".

CMs run as daemons. When a client application requests a connection, the parent CM process spawns a child process to handle the connection. At that point, the application no longer communicates with the parent CM; all communication flows from the application to the child CM as depicted in [Figure 3-5](#).

Figure 3-5 Client Connections to Child CM



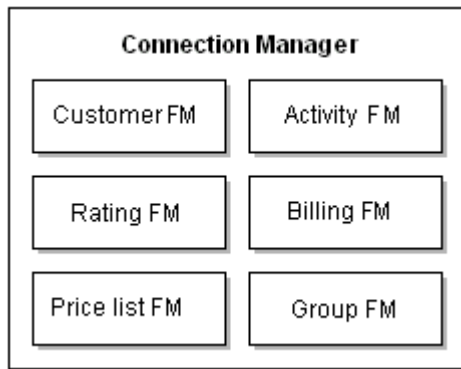
About Facilities Modules (FMs)

When an application connects to a CM, the data sent by the application is processed first by the FMs that are included in the CM.

There are separate FMs to handle different types of tasks. For example, if a CSR changes a customer’s password, the Customer FM validates that the new password has the correct number of characters.

FMs manage BRM activity and ensure that data is processed correctly. When you configure a CM, you specify the FMs that are linked to the CM. Typically, you use the default set of required FMs for each CM. [Figure 3-6](#) depicts a CM configured with several FMs.

Figure 3-6 Facilities Modules for CM



Each FM is dynamically linked to a CM when the CM starts. You can link optional or custom FMs to any CM. For example, if you configure the terminal server to connect to a CM, you can link the RADIUS Manager FMs to that CM.

Because FMs are linked to CMs, the FMs can get configuration information from the CM configuration files. Therefore, one of the ways you can customize BRM policies is by editing the CM configuration file. For more information, see ["Ways to Use and Customize BRM"](#).

You can customize policy FMs or add custom FMs to a CM. See *BRM Developer's Guide*.

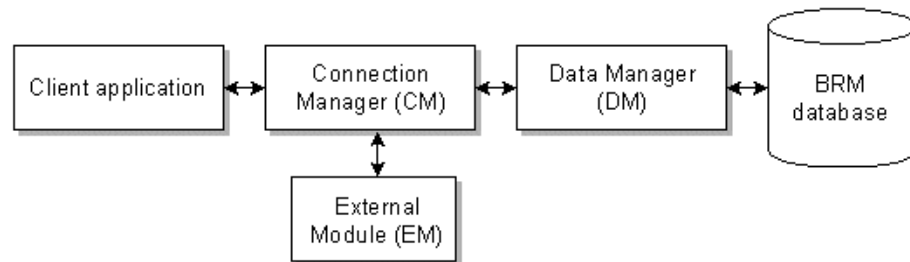
About External Modules (EMs)

An External Module (EM) is similar to an FM; it is a set of opcodes that perform BRM functions. However, it is not linked to a CM in the same way that an FM is. Instead, it runs as a separate process that you must start and stop.

The Payload Generator is an EM that processes events that are exported to external systems by using the optional Enterprise Application Integration (EAI) framework.

[Figure 3-7](#) shows where an EM fits in the system architecture:

Figure 3-7 External Module in BRM System Architecture



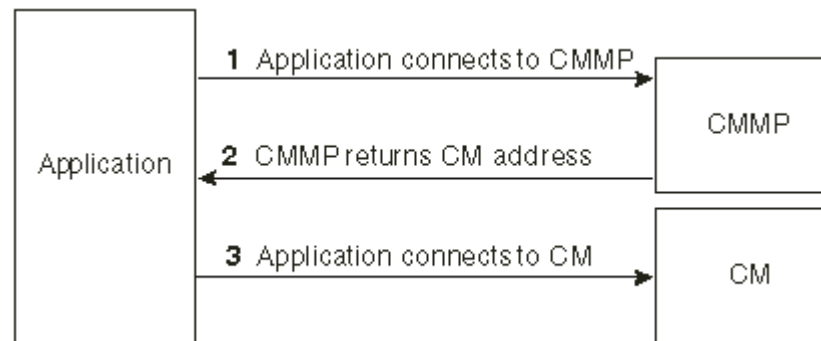
About Connection Manager Master Processes (CMMPs)

Instead of configuring an application to connect directly to a CM, you can configure it to connect to a Connection Manager Master Process (CMMP).

When an application connects to a CMMP, the CMMP selects a CM from a list you provide and gives the application the machine name and port of that CM. The application then uses that address to connect to the CM.

[Figure 3-8](#) shows how an application connects to a CM using a CMMP:

Figure 3-8 Using a CMMP to Connect to a CM



For more information, see the information about improving performance in *BRM System Administrator's Guide*.

Data Management Tier

The data management tier consists of the following components:

- **Data Managers (DMs)**, which translate requests from CMs into a language that the database can understand. For the BRM database, the language is SQL. There are also DMs for credit card processors, Vertex, and so forth. See "[About Data Managers \(DMs\)](#)".
- **Oracle In-Memory Database (IMDB) Cache Data Manager (DM)**, which translates requests from CMs into a language that Oracle IMDB Cache and the BRM database can understand. See "[About IMDB Cache DM](#)".

About Data Managers (DMs)

Data Managers (DMs) translate BRM operations into a form that can be understood by the data access system.

BRM includes the following DMs:

- The Oracle DM (**dm_oracle**) provides an interface to an Oracle BRM database.
- The Paymentech DM (**dm_fusa**) provides an interface to the Paymentech credit-card payment processor.
- The Vertex DM (**dm_vertex**) provides an interface to the Vertex Sales Tax Q Series and Communications Tax Q Series database.

Child processes are not spawned when a DM receives a connection request. Instead, child processes are spawned when a DM starts. You define the number of child processes in the DM configuration file.

When a DM receives a connection from a CM, the parent process or thread assigns a child process or thread to the connection. At that point, the client no longer communicates with the parent DM, and all communication flows from the CM to the child DM.

DMs run as a set of processes. Each process handles a single database transaction.

About Data Manager (DM) Back Ends

Each DM has a set of front-end processes that receive connection requests and send them to a shared-memory queue for processing by back-end processes. The back-end processes translate the transaction into the database language.

DM back ends are single-threaded. Therefore, a back end is dedicated to a single transaction for the duration of the transaction.

From the BRM database perspective, the back ends communicate to database back-end (shadow) processes. One database back-end process is started for each DM back end. See your database documentation for information on shadow processes.

You can write a new DM back end to integrate BRM with any relational, object-oriented, or legacy database management system. You can also layer BRM on top of basic file managers. For more information, see the documentation about writing a custom Data Manager in *BRM Developer's Guide*.

About IMDB Cache DM

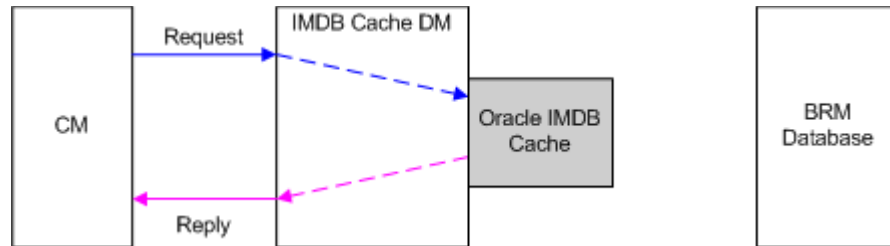
IMDB Cache DM provides an interface between the CM and Oracle IMDB Cache and between the CM and the BRM database. When IMDB Cache DM receives a request for

data from the CM, IMDB Cache DM determines which database to route the request to based on where the data resides:

- **In IMDB Cache:** IMDB Cache DM uses TimesTen Cache libraries to retrieve data from IMDB Cache.

Figure 3–9 shows the request and response flow for data that resides in IMDB Cache. When IMDB Cache DM receives authorization and reauthorization requests, it searches Oracle IMDB Cache for the required information and sends the response to the CM.

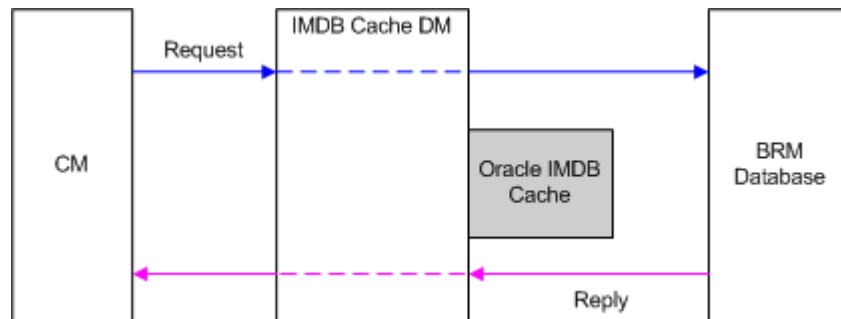
Figure 3–9 CM Interaction with Data in the Oracle IMDB Cache



- **In the BRM database:** IMDB Cache DM passes the request directly to the BRM database.

Figure 3–10 shows the request and response flow for objects that reside in the BRM database. IMDB Cache DM forwards database object requests to the BRM database and then forwards the response from the BRM database to the CM.

Figure 3–10 CM Interaction with Data in the BRM Database Not in Cache



Note: IMDB Cache DM is supported on 64-bit operating systems only.

Data Tier

The data tier consists of the BRM database and other data access systems, such as the Paymentech credit-card processing service.

About the BRM Database

The BRM database stores all of your business information and data, such as information about customer accounts, your price list, and the services you provide. All information stored in the BRM database is contained in objects.

An *object* is equivalent to a database record or a set of database records. Each type of object contains a set of related information. For example, an account object includes the customer's name and address.

Related database objects are linked to each other. For example, an account object is linked to the payment information object that stores a customer's credit card number.

To see examples of objects, use Event Browser in Customer Center or use Object Browser.

Note: The term *object* does not refer to an object request broker (ORB) or to the object programming environment. Instead, *object* refers to the BRM object data model. For more information about the storable object model, see *BRM Developer's Guide*.

About the Residency Type

RESIDENCY_TYPE is an attribute of storable objects and defines where the object resides in the BRM system. The residency type values are predefined in the data dictionary in the BRM database.

Table 3–1 describes where an object resides based on its residency type value.

Table 3–1 Residency Type Descriptions

Attribute Value	Object Type	Residency
0	Database object	Objects reside in the BRM database.
1	In-memory object	Objects reside in Oracle IMDB Cache.
2	Shared object	Objects reside in Oracle IMDB Cache and the BRM database.
3	Transient object	Objects reside in the Oracle IMDB Cache data file.
4	Deferred database object	Objects reside in the BRM database and are passed through the IMDB Cache DM.
5	Static reference object	Objects reside in the BRM database but are cached in Oracle IMDB Cache when they flow back from the database.
6	Volatile object	Objects reside only in the Oracle IMDB Cache data file.
7	Dynamic reference object	Reference objects that reside in the BRM database and are updated more frequently in Oracle IMDB Cache.
101	Routing database object	/uniqueness objects.
102	General routing database object	All global objects, except /uniqueness objects, stored in the BRM database.
303	IMDB Cache resident expanded object	Event objects stored in Oracle IMDB Cache in expanded format.
304	BRM resident expanded object	Event objects stored in the BRM database in expanded format.

If you create a custom object, you must set its residency type to the correct value. For more information, see "Assigning Custom Objects a Residency Value" in *BRM System Administrator's Guide*.

About the Multischema Architecture

In multischema systems, the database layer of your BRM system consists of one primary schema and one or more secondary schemas in a *single database*. A primary DM is connected to the primary schema and secondary DMs are connected to the secondary schemas. Data is separated between the schemas as follows:

- The *primary database schema* stores two types of global objects: objects that the secondary DMs can only read, such as configuration and pricing objects, and objects that the secondary DMs can both read and update, such as uniqueness objects. The primary schema also stores subscriber data.
- The *secondary database schemas* store subscriber data, such as event and account objects.

The primary DM updates global read-only objects in the primary schema, and the secondary DMs read the data from views in the secondary schema on tables from the primary schema.

The secondary DMs use schema qualifications to read and modify updatable global objects stored in the primary schema.

For information about how to set up a multischema system, see "Managing a Multischema System" in *BRM System Administrator's Guide*.

About Multidatabase Manager

Multidatabase Manager is an optional feature that enables you to have multiple BRM database schemas in a single installation. This option can support very large installations (more than a few million subscribers). It enables you to split the main database into multiple schemas.

About Oracle RAC for a High-Availability BRM System

For a high-availability system, BRM recommends Oracle Real Application Cluster (Oracle RAC), which requires a reliable and highly available storage system. For more information on a high-availability system, see *BRM System Administrator's Guide*.

About Oracle IMDB Cache

Oracle In-Memory Database (IMDB) Cache is an in-memory database that caches performance-critical subsets of the BRM database for improved response time. IMDB Cache includes the following functionality:

- Caches data from BRM database tables
- Replicates cached data for high-availability systems
- Stores transient data

BRM stores customer account data and information related to the services that you provide in the BRM database. It uses Oracle IMDB Cache to cache a subset of the BRM database tables that are frequently accessed or that are performance-critical and require fast transaction response. For example, `/account`, `/service`, and `/balance_group` tables.

IMDB Cache provides fast access to the cache data resulting in improved transaction response times and increased system throughput:

- Creating, modifying, deleting, and searching for objects in memory eliminates the network connections for communicating with the BRM database server, thus improving transaction response times.

- Caching database tables in IMDB Cache reduces the workload on the BRM database, thus improving the overall system throughput.

About Objects in an IMDB Cache-Enabled System

BRM storable objects in an IMDB Cache-enabled system are assigned one of the following categories:

- [Transient Objects](#)
- [Reference Objects](#)
- [Database Objects](#)

Transient Objects

Transient objects contain ongoing session data necessary for processing authorization and reauthorization requests that require low latency and high throughput. These objects are transactional and are stored *only* in the IMDB Cache data file. They are created and updated during a session when the system is running and deleted at the end of the session after the data is committed to the BRM database. These objects can be accessed only through simple queries.

Reference Objects

Reference objects contain data such as subscriber information and resource balances that are read often but not updated often. These objects require low latency for read-only access and are updated only at the end of the session. Reference objects are created, updated, and stored in Oracle IMDB Cache and replicated to the BRM database.

Note: Reference objects initially stored in the BRM database are preloaded into Oracle IMDB Cache.

Examples of reference objects include:

- Account objects (*/account*).
- Service objects (*/service*).
- Bill objects (*/bill*)
- Item objects (*/item*)
- Balance group objects (*/balance_group*).

Only BRM reference objects with residency type values 5 and 7 are valid for caching. See "[About the Object Residency Type](#)".

Database Objects

Database objects are of two types:

- *Database-only* objects are created, updated, and stored *only* in the BRM database and are not cache in Oracle IMDB Cache. Examples of these database objects include configuration objects, device objects, and pricing objects, such as products, plans, and deals.
- *Hybrid database objects* are created and updated in Oracle IMDB Cache and propagated to the BRM database. These objects are deleted from Oracle IMDB Cache using the least-recently-used (LRU) aging policy. Search and read

operations on these objects, however, are performed in the BRM database. An example of this database object includes event objects.

About Active Session Objects

BRM stores information about subscriber calls from the start of the call to the successful completion of the call in *session* objects in the BRM database. In an IMDB Cache-enabled system, information about currently active calls is stored in the *active session* object (**/active_session**) in the IMDB Cache data file. Active session objects contain information about the call from the time a subscriber *tries* a call until the call ends, successfully or not.

The *Active Session Manager* (ASM) FM processes the active session objects and manages active call state information for the duration of a call. After a call is completed, a session object is created from the active session object in the BRM database. The active session object is then optionally deleted from IMDB Cache.

About the Object Residency Type

Storable objects in an IMDB Cache-enabled system reside in different databases. For instance, reference objects reside in Oracle IMDB Cache, transient objects reside in the IMDB Cache data file, and database objects reside in the BRM database. The storable object attribute, **RESIDENCY_TYPE**, defines where the object resides in the BRM system. The residency type values are predefined in the data dictionary in the BRM database.

Table 3–2 describes where an object resides based on its residency type value.

Table 3–2 Residency Type Descriptions

Attribute Value	Object Type	Residency
0	Database objects	Objects reside in the BRM database.
1	In-memory objects	Objects reside in Oracle IMDB Cache.
2	Shared objects	Objects reside in Oracle IMDB Cache and the BRM database.
3	Transient objects	Objects reside in the IMDB Cache data file.
4	Deferred database objects	Objects reside in the BRM database and are passed through the IMDB Cache DM.
5	Static reference objects	Objects reside in the BRM database but are cached in Oracle IMDB Cache when they flow back from the database.
6	Volatile objects	Objects reside only in the IMDB Cache data file.
7	Dynamic reference objects	Reference objects that reside in the BRM database and are updated more frequently in Oracle IMDB Cache.
101	Routing database object	/uniqueness objects.
102	General routing database object	All global objects, except /uniqueness objects, stored in the BRM database.
303	IMDB Cache resident expanded object	Event objects stored in Oracle IMDB Cache in expanded form.
304	BRM resident expanded object	Event objects stored in the BRM database in expanded form.

IMDB Cache DM uses the residency type values to determine which database to send request operations to. Because IMDB Cache DM uses dual connections, if the data resides in IMDB Cache, IMDB Cache DM uses the IMDB Cache connection to retrieve the data. If the data resides in the BRM database, IMDB Cache DM uses the Oracle database connection to retrieve the data.

The object residency type value is also used by the `pin_tt_schema_gen` utility to determine whether the object must be cached. Only reference objects with residency type values of 1, 5, or 7 are valid for caching in memory.

If you create any custom objects, you must set the residency type of the custom object to the correct value. For more information, see "Assigning Custom Objects a Residency Value" in *BRM System Administrator's Guide*.

About the Pipeline Manager System Architecture

Pipeline Manager is used for rating and discounting events in batch and real-time.

The Pipeline Manager system architecture consists of:

- The pipeline framework that controls the Pipeline Manager system functions.
- The pipelines that the framework runs, which perform rating and discounting.
- The data pool that provides data in memory, used for rating and discounting.
- The Pipeline Manager database that stores data used for rating and discounting.

Figure 3–11 shows how a billable event is rated in batch by Pipeline Manager and recorded in the BRM database. In this case:

1. Pipeline Manager rates event data from CDR files.
2. Rated Event (RE) Loader loads rated events into the BRM database.
3. Account balances are updated.

Figure 3–11 Billable Event Rating by Pipeline Manager and Storage in BRM Database

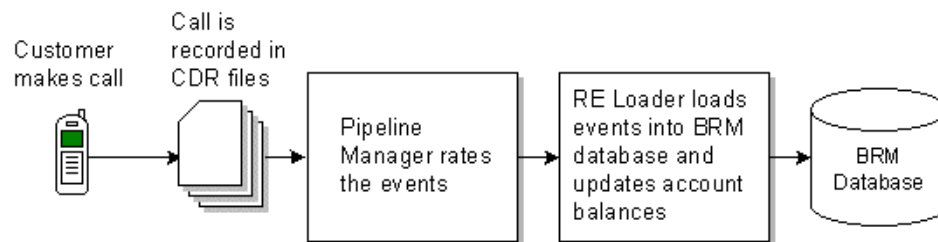
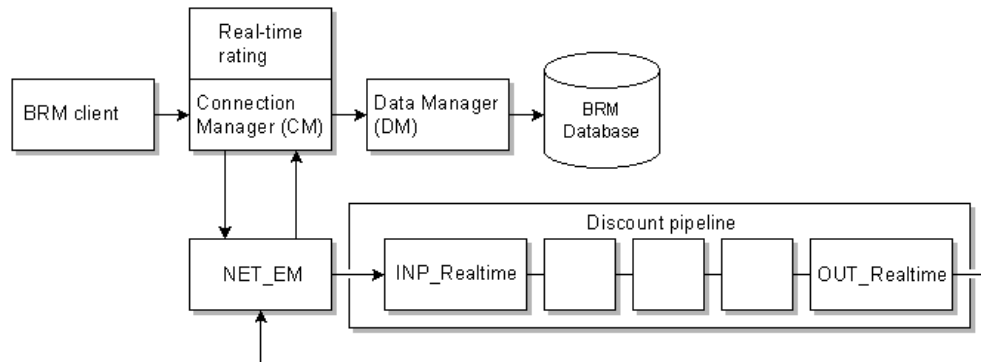


Figure 3–12 shows how real-time discounting works.

Figure 3–12 Real-Time Discounting

In this case:

1. BRM sends an event to the pipeline for real-time discounting.
2. The NET_EM module sends the event to the pipeline.
3. Pipeline Manager returns the discounted amount.
4. Account balances are updated in the BRM database.

About the Pipeline System Components

When you configure an instance of the Pipeline Manager, you configure a set of system components and one or more pipelines. The system components are:

- Controller. See "[About the Controller](#)".
- EDR Factory. See "[About the EDR Factory](#)".
- Transaction ID Controller. See "[About the Transaction ID Controller](#)".
- Sequencer. See "[About the Sequencer](#)".
- Event Handler. See "[About the Event Handler](#)".

About the Controller

The Controller manages and monitors the entire Pipeline Manager instance. The Controller performs these functions:

- Starts and stops a Pipeline Manager instance.
- Initiates and coordinates different threads.
- Checks for new semaphore file entries.
- Generates a log message table that is used by the LOG module to create the process log file, the pipeline log files, and the stream log file.

You configure the Controller by using the registry file. For information, see the Pipeline Manager documentation in *BRM System Administrator's Guide*.

About the EDR Factory

The EDR Factory is a mandatory pipeline component that generates and allocates memory to EDR containers in a single pipeline.

When a transaction starts, the EDR Factory:

1. Allocates memory for each container.

2. Generates an EDR container for each piece of the input stream, including one for the header, one for each EDR, and one for the trailer, by using the container description file.
3. After the pipeline writes information to the output file, the EDR Factory empties the container and releases the cache. The EDR Factory can then reuse the memory for new containers.

You configure the EDR Factory by using the **EDRFactory** section of the registry file.

About the Transaction ID Controller

The Transaction ID Controller generates unique IDs for all open transactions in your pipelines. An instance of Pipeline Manager contains only one Transaction ID Controller.

The Transaction ID Controller performs these functions:

- Stores blocks of transaction IDs in cache. The Transaction ID Controller issues IDs to TAMs directly from cache.
- Uses the transaction state file or table to track ID numbers.
- Assigns ID numbers to transactions.

You configure the Transaction ID Controller by using the **TransactionIDController** section of the registry file. For information, see the Pipeline Manager documentation in *BRM System Administrator's Guide*.

About the Sequencer

The BRM Sequencer is an optional Pipeline Manager component that performs one of these functions:

- *Sequence checking*, which ensures that a CDR file is not processed more than once by keeping track of each CDR file's unique sequence number. A sequence check also logs gaps in sequence numbers.
- *Sequence generation*, which generates sequence numbers for output files. This functionality is used when CDR input files do not have sequence numbers and when pipelines split CDR input files into multiple output files.

Note: Sequence generation is not required when there is a one-to-one correspondence between input and output files. In this case, sequence numbers can be passed through to the output file.

Each pipeline can be configured to use one or more Sequencers. You configure your Sequencers by using the **SequencerPool** registry entries, and you assign Sequencers to pipelines by using the **Output** registry entries.

For more information about the Sequencer, see the Pipeline Manager documentation in *BRM System Administrator's Guide*.

About the Event Handler

The Event Handler is an optional pipeline framework component that starts external programs when triggered by internal events. For example, you can configure the Event Handler to launch a script that moves event data record (EDR) output files to a specific directory whenever the output module finishes processing them.

An instance of the Pipeline Manager uses only one Event Handler, which monitors the events for all pipelines in your system. Each registered module in your system automatically sends events to the Event Handler. You define which of these events trigger external programs by using the `ifw.EventHandler` section of the registry file.

When the Event Handler receives an event from a registered module, it:

1. Checks to see if the event is mapped to an action.
2. Performs one of the following:
 - Starts the associated program or script.
 - If no action is mapped, ignores the event.
3. Queues any events it receives while the external program is running.
4. Waits for the external program to terminate.

About the Data Pool

The data pool is a set of modules that store data used by all the pipelines in a single Pipeline Manager instance. Data modules are named with the prefix "DAT", for example, DAT_AccountBatch.

Data modules get their data from the Pipeline Manager database and from the BRM database at startup. As data changes in the BRM system, the data is updated in the data pool.

For more information, see the Pipeline Manager documentation in *BRM System Administrator's Guide*.

About Pipelines

A single Pipeline Manager instance runs one or more pipelines. Each pipeline includes the following components:

- The *Pipeline Controller*, which you use to manage the pipeline. See "[About the Pipeline Controller](#)".
- The *input module* reads data from the input stream, converts CDR files into the internal EDR input format, and performs error checking on the input stream.
- *Function modules* perform all rating tasks and EDR management tasks for a pipeline. Function modules process the data in the EDRs. Each function module performs a specific task, for example, checking for duplicate EDRs or calculating zones.

Function modules do not store any data; instead they get data from data modules. For example, to rate an event, the FCT_MainRating module gets pricing data from the DAT_PriceModel module.

Function modules have two dependencies:

- Some modules require previous processing by other modules.
- Some modules get data from data modules.
- The *output modules* convert internal EDRs to output format and write the data to the output streams.
- The *log module*, which you use to generate and manage your process, pipeline, and stream log files.

About Using Multiple Pipelines

You create multiple pipelines to do the following:

- Maximize performance and balance system loads. For example, you can create multiple pipelines to handle multiple input streams.
- Manage different types of processing. For example, you can create separate pipelines for zoning, rating, and preprocessing. In this case, you can use the output of one pipeline as the input for another pipeline, or pipelines can run in parallel. To improve performance, aggregation is typically performed in a separate pipeline.

When you create multiple pipelines, they run in parallel in a single Pipeline Manager instance. You configure all pipelines in the same registry file. Each pipeline has its own input and output configuration, EDR Factory, Transaction Manager, and set of function modules. However, all pipelines share the same set of data modules.

You can also use a pipeline to route EDRs to different Pipeline Manager instances. For example, when you use multiple database schemas, you use the FCT_AccountRouter module to send EDRs to separate instances of Pipeline Manager.

About the Pipeline Controller

The Pipeline Controller manages all processes for one pipeline.

The Pipeline Controller performs the following functions:

- Starts and stops the pipeline.
- Initiates and coordinates the pipeline's threads. See "[About Thread Handling](#)".
- Defines the valid country codes and international phone prefixes for the pipeline. The pipeline's function modules retrieve this information during processing.
- Manages pipeline input and output.

You configure the Pipeline Controller by using the **Pipelines** section of the registry file.

About Thread Handling

You can configure each pipeline to run with *multithreaded* processing or *single-threaded* processing. By default, each pipeline is configured for multithreaded processing.

You select single-threaded or multithreaded mode to optimize performance. For more information, see the documentation about tuning Pipeline Manager performance in *BRM System Administrator's Guide*.

About the Pipeline Manager Database

The Pipeline Manager database stores business configuration data, such as price models and rate plans. Pipeline Manager accesses this information when you first start Pipeline Manager or when you force a database reconnection. Pipeline Manager then stores a copy of your pricing and rating data in your data modules.

Pipeline Manager modules connect to the Pipeline Manager database through the Database Connect module (DBC).

About Configuring Pipeline Manager

To configure Pipeline Manager, you use the following files to manage the Controller:

- *Registry files*, which you use to configure a Pipeline Manager instance at system startup.
- *Semaphore files*, which you use to configure and control pipelines during run time.

For more information, see the documentation about configuring Pipeline Manager in *BRM System Administrator's Guide*.

You can also use the `pin_ctl` utility to start and stop Pipeline Manager.

Configuring the Four-Tier Architecture

All BRM processes can run on the same computer, or they can be distributed among different computers. Distributed processing provides a great deal of flexibility in configuring your system, especially if you have multiple machines running CMs and DMs. For example, you can do the following:

- Run redundant system processes (for example, Customer Center, billing applications, CMs, and DMs).
- Connect clients to specific CMs, or use a CMMP to route to any available CM.

For more information, see *BRM Installation Guide*.

Communication Between System Components

Communication between applications and CMs, and CMs and DMs, is accomplished by TCP/IP. You assign each component an IP port and host name. TCP/IP facilitates the use of firewalls, proxies, and filters.

Any kind of network connection that supports TCP/IP supports BRM (for example, local area network, virtual private network, and PPP).

Communication between DMs and data access systems uses the protocol required by the data access system. For example:

- DM-to-Oracle communication is carried out using Oracle SQL*Net, a product that lets distributed computers access a central Oracle database.
- DM-to-Paymentech communication uses the Paymentech protocol.

For information about the configuration file entries that enable communication between BRM components, see the documentation about configuration files in *BRM System Administrator's Guide*.

Four-Tier Architecture and Failure Recovery

The four-tier architecture provides high reliability in the event of system component failures:

- If the BRM database is offline, any interrupted transactions are rolled back when the database is restarted. The DM automatically attempts to connect to the database until a connection is made.
- If a DM is offline, any interrupted transactions are timed out and rolled back by the database. The CM automatically attempts to connect to the DM until a connection is made. You can provide each CM with a list of DMs. If a DM is unavailable, the CM can connect to a different one.

If a DM fails while there are transactions in its queue, the transactions must be resubmitted. The contents of the queue are lost when the DM fails.

- If a CM is offline, the database rolls back any interrupted transactions. You can use a CMMP to provide multiple CMs for clients to connect to. See "[About Connection Manager Master Processes \(CMMPs\)](#)".
- If a client fails, any interrupted transaction is timed out and rolled back by the database.

In all cases, errors are reported to the client application. Depending on its capabilities, the client can display an error, log the error, or retry the transaction.

- In all cases, broken transactions are rolled back by the database, so no partial transactions are recorded.