

Oracle® Communications Billing and Revenue Management

Patch Set Release Notes



Release 12.0

F10044-16

April 2024

The Oracle logo, consisting of the word "ORACLE" in white, uppercase, sans-serif font, centered within a solid red square.

ORACLE®

Oracle Communications Billing and Revenue Management Patch Set Release Notes, Release 12.0

F10044-16

Copyright © 2018, 2024, Oracle and/or its affiliates.

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, software documentation, data (as defined in the Federal Acquisition Regulation), 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 embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. 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®, Java, MySQL and NetSuite are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside 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, Epyc, and the AMD 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

Audience	xv
Documentation Accessibility	xv
Diversity and Inclusion	xv

1 Summary of Customer-Reported Fixes

Customer-Reported Fixes in BRM	1-1
Customer-Reported Fixes in BRM 12.0 Patch Set 8	1-1
Customer-Reported Fixes in BRM 12.0 Patch Set 7	1-5
Customer-Reported Fixes in BRM 12.0 Patch Set 6	1-7
Customer-Reported Fixes in BRM 12.0 Patch Set 5	1-10
Customer-Reported Fixes in BRM 12.0 Patch Set 4	1-14
Customer-Reported Fixes in BRM 12.0 Patch Set 3	1-30
Customer-Reported Fixes in BRM 12.0 Patch Set 2	1-37
Customer-Reported Fixes in BRM 12.0 Patch Set 1	1-46
Customer-Reported Fixes in ECE	1-54
Customer-Reported Fixes in ECE 12.0 Patch Set 8	1-55
Customer-Reported Fixes in ECE 12.0 Patch Set 7	1-62
Customer-Reported Fixes in ECE 12.0 Patch Set 6	1-67
Customer-Reported Fixes in ECE 12.0 Patch Set 5	1-69
Customer-Reported Fixes in ECE 12.0 Patch Set 4	1-70
Customer-Reported Fixes in ECE 12.0 Patch Set 3	1-75
Customer-Reported Fixes in ECE 12.0 Patch Set 2	1-76
Customer-Reported Fixes in PDC	1-79
Customer-Reported Fixes in PDC 12.0 Patch Set 8	1-79
Customer-Reported Fixes in PDC 12.0 Patch Set 7	1-80
Customer-Reported Fixes in PDC 12.0 Patch Set 6	1-80
Customer-Reported Fixes in PDC 12.0 Patch Set 5	1-81
Customer-Reported Fixes in PDC 12.0 Patch Set 4	1-82
Customer-Reported Fixes in PDC 12.0 Patch Set 3	1-84
Customer-Reported Fixes in PDC 12.0 Patch Set 2	1-85
Customer-Reported Fixes in PDC 12.0 Patch Set 1	1-85

Customer-Reported Fixes in Billing Care	1-87
Customer-Reported Fixes in Billing Care 12.0 Patch Set 8	1-87
Customer-Reported Fixes in Billing Care 12.0 Patch Set 7	1-89
Customer-Reported Fixes in Billing Care 12.0 Patch Set 6	1-90
Customer-Reported Fixes in Billing Care 12.0 Patch Set 5	1-90
Customer-Reported Fixes in Billing Care 12.0 Patch Set 4	1-91
Customer-Reported Fixes in Billing Care 12.0 Patch Set 3	1-92
Customer-Reported Fixes in Billing Care 12.0 Patch Set 2	1-93
Customer-Reported Fixes in Billing Care 12.0 Patch Set 1	1-97
Customer-Reported Fixes in Business Operations Center	1-98
Customer-Reported Fixes in BRM Thick Clients	1-100

2 New Features in BRM

New Features in BRM 12.0 Patch Set 8	2-1
BRM Supports Flexible Proration Options	2-1
Support for Conversion to Wholesale Hierarchy	2-3
Sharing Groups Now Support Multiple Schemas	2-3
Discount Offer Stacking Support	2-3
Pre-Expiry/Post-Expiry Notification Support	2-4
Kafka DM Enhancements	2-4
Real-time Balances from ECE Available As a SOAP Web Service	2-4
Price-tag-based Price Override Support in REST Services Manager	2-5
Two-phase Billing Support in REST Services Manager	2-5
BRM Supports Two-Way SSL Authentication	2-5
Rated Event Manager Support for CDR Streaming	2-5
New Features in BRM 12.0 Patch Set 7	2-5
Promise-to-Pay Enhancements	2-6
Balance Impact Stores Current Balance of Resources	2-6
Support for Setting Credit Floor Dynamically	2-6
First Usage Activation Enhancements in BRM	2-6
Balance Monitoring Enhancements	2-7
Support for Add-On Products in Deals	2-7
Sharing Group Improvements	2-8
Notifications Can Be Triggered After Event Occurrence	2-8
Proration Can Be Configured at the Product Level	2-8
Improvements to Tax Handling for Bill Disputes	2-8
Secure Communication to the BRM Database	2-9
Running Stored Procedures in BRM	2-9
New Features in BRM 12.0 Patch Set 6	2-9
Support for PINless Debit Payment Processing	2-9

Kafka DM Enhancements	2-9
BRM Notification Enhancements	2-10
BRM Supports Temporary Credit Limits	2-10
Customer Deposits	2-11
Loan Management Enhancements	2-11
BRM Can Now Track Failed Operations	2-11
Cloud Native Support for SSO	2-12
Exposing Charge and Tax Details for Payments and Adjustments	2-12
Support for More Granular Tax Code Application	2-12
New Features in BRM 12.0 Patch Set 5	2-12
Top-Up Enhancements	2-13
Loan Management	2-13
New Bill States	2-14
Sample Prepaid Service Life Cycle	2-14
XML Price List Enhancements for BRM	2-14
Rerating Noncurrency Resources When Products are Canceled	2-15
Options for Exceeding Credit Limits	2-16
New Residency Type to Support Distinct Searches	2-16
Customizing V\$SESSION Table Details for dm_oracle and dm_ifw_sync	2-16
BRM Now Supports Dynamic Charging	2-17
BRM Now Supports Dynamic Taxation	2-17
BRM Now Supports Promotions Based on Special Dates, Events, and Actions	2-18
Sending Messages to Customers through External Notification Applications	2-18
pin_inv_doc_gen Now Uses JDBC	2-18
BRM Cloud Native Enhancements in Patch Set 5	2-19
New Features Supported by BRM REST Services Manager	2-19
New Method for Loading Rated Events from ECE into BRM	2-20
New Features in BRM 12.0 Patch Set 4	2-20
BRM Cloud Native Enhancements in Patch Set 4	2-21
Monitoring BRM Components	2-22
BRM Now Supports ASC 606/IFRS 15 Revenue Recognition	2-23
Deferring Corrective Bills and Invoices	2-23
BRM Supports Integration with Apache Kafka Servers	2-24
Integrating BRM with External Care Applications	2-26
Running Business Operations and BRM Applications from Custom Clients	2-26
Using Paymentech Card Type Indicator	2-27
XML Product Offering Enhancements	2-27
BRM Supports Additional Cipher Suites	2-28
Web Services Manager Now Supports Oracle Access Manager 12.2.1.3.0	2-28
Support for TM Forum REST APIs	2-28
New Features in BRM 12.0 Patch Set 3	2-28

BRM Cloud Native Deployment Enhancements	2-29
BRM Server, Thick Clients, and PCC Now Localized	2-30
Revenue Assurance Supports Custom Applications	2-30
Pipeline Manager Can Now Support 28-Digit Decimal	2-30
New Timeout Parameter for Account Synchronization DM	2-30
Support Added for AES Encryption	2-31
Enable SSL/TLS between Client and the CM Only	2-31
Adjusting Taxes by a Specified Amount	2-31
New Features in BRM 12.0 Patch Set 2	2-31
Deploying BRM Services on a Cloud Native Environment	2-32
Wholesale Billing Enhancements	2-32
Specifying How to Calculate Deferred Taxes for Wholesale Billing	2-33
Additional Card Security Presence Values Supported for Card Validation or Authorization	2-34
Card Security Code Is Now Masked in Logs	2-35
Display Bills Generated Before Moving the Account to a Hierarchy	2-35
Managing Discount Validity Starting or Ending in Future Cycles	2-35
Wildcard in Item Type Selectors	2-36
BRM Supports POID Generation in ECE	2-36
Enabling Prepaid-Event Partitions	2-37
About Creating Custom Classes	2-38
Support for Stored-Credential Transactions for Payments	2-39
Storing Card Credentials for Future Transactions	2-40
Purging Card Credentials	2-41
Migrating Legacy Payment Information	2-42
Support for Migration of Legacy Data into BRM and ECE in Real Time	2-42
Importing Data	2-43
Deploying Data in BRM and ECE	2-45
Merging Legacy Balances	2-45
Support for Migrating Hierarchical Accounts Using Same Input File	2-46
Event Rounding Rules Can Be Used for Adjustments	2-46
New Staging Directory for Exported G/L reports	2-47
BRM Client Applications Supported from 12.0 Patch Set 2	2-47
New Features in BRM 12.0 Patch Set 1	2-48
Improved Performance for Large Accounts	2-48
Enabling Wholesale Billing	2-49
Creating Wholesale Accounts and Bill Unit Hierarchies	2-49
Configuring Wholesale Business Profile	2-50
Setting Up a Wholesale Parent	2-51
Rolling Charges Up to the Wholesale Parent	2-51
Rolling A/R Actions Up to the Wholesale Parent	2-52

Rolling Journals Up to the Wholesale Parent	2-52
Running Wholesale Billing	2-53
Configuring Billing Delay for Wholesale Hierarchies	2-54
Setting Up Billing-Time Discounts for Wholesale Hierarchies	2-54
Suppressing Bills for Wholesale Hierarchies	2-54
Trial Billing for Wholesale Hierarchies	2-54
Support for A/R Activities	2-55
Specifying Search Criteria for Retrieving Items, Events, and Bills	2-55
Moving Bill Units into or out of Wholesale Hierarchies	2-56
Configuration Changes	2-56
Opcode Changes	2-56
Schema Changes	2-57
Storable Class Changes	2-57
Utility Changes	2-57
pin_update_journals	2-58
pin_roll_up_ar_items	2-58
Delay Interval Can be Configured for Resolving Failed Payments	2-59
Enhanced Data Protection	2-60
Specifying Retention Period for Closed Accounts	2-61
Deleting Closed Accounts	2-61
pin_del_closed_accts	2-62
Enhanced Security for Root Wallet	2-64
Support for Rolling Back the BRM Patch Set	2-64
BRM 12.0 Is Now Certified with Mozilla Firefox 58.0	2-64
BRM 12.0 Is Now Certified with Perl 5.28.0	2-64
BRM 12.0 Is Now Certified with Paymentech 120 Byte Batch Version 3.0.0 R 12.4 and Online Authorization Version 7.4 R12.4	2-64
BRM 12.0 Is Now Certified with Tomcat 8.5.32	2-64

3 New Features in ECE

New Features in ECE 12.0 Patch Set 8	3-1
Extensions for Midsession-Rated Events	3-2
Reasons Now Included for Midsession-Rated Events	3-2
Midsession-Rated Events Created When USU Block Missing	3-2
Extensions for Sy Request-Response Flow	3-3
Active-Active Mode Can Now Process Requests in Local Sites	3-3
ECE Enhancements for Processing Free Allowances	3-3
Support for Tax Selectors and Tax Exemption Selectors	3-4
ECE Supports Flexible Proration Settings	3-4
ECE Cloud Native Contains New ConfigLoader Pod	3-4
Support for Prepaid Usage Overage	3-4

Rated Event Publisher Support for CDR Streaming	3-5
New Features in ECE 12.0 Patch Set 7	3-5
Time-Based Price Overrides and Dynamic Usage Rating	3-5
Bundle Renewal Triggered from ECE	3-6
Non-Linear Rating Configurable by Event Type	3-6
Balance Query Returns Lifecycle Information	3-6
Current Balance Now Returned with Balance Impact	3-6
First Usage Activation Enhancements	3-6
Network Signaling Optimization	3-6
Balance Monitoring and Group Credit Limits	3-6
Product Sharing	3-7
Logging of Usage Rating Failures	3-7
Robust Handling of Configuration Data	3-7
Quorum Policy Handling in ECE	3-7
Rolling Upgrade Enhancements	3-7
Access to Bundle Information in ECE	3-8
Ability to Call a BRM Opcode from ECE	3-8
CDR Generator Enhancements	3-8
Policy-Based Charging Notification Enhancements	3-9
Usage Consumption Management Report Certification	3-9
New Features in ECE 12.0 Patch Set 6	3-9
ECE SDK Includes New Rated Events Custom Plugin	3-9
ECE Now Supports Temporary Credit Limits	3-9
ECE Now Supports Loan Management	3-10
ECE Can Now Record Failed Usage Requests	3-11
HTTP Gateway Supports Charging Type-Based Configuration for CDR Generation	3-11
ECE Supports 5G SA Nchf_OfflineOnlyCharging Service	3-11
Enhancements for HTTP Gateway Registration with an NRF	3-11
ECE REST API Supports New CHF Operation Types	3-11
HTTP Gateway Now Determines Charging Type	3-12
New Features in ECE 12.0 Patch Set 5	3-12
ECE Now Generates CDRs for Unrated 5G Usage Events	3-12
SSL Now Between ECE and Persistence Database	3-12
ECE Cloud Native Supports Subscriber Tracing	3-13
Rated Event Formatter Changes for Rated Event Manager	3-13
New Features in ECE 12.0 Patch Set 4	3-13
New Features in ECE 12.0 Patch Set 3	3-14
Apache Kafka Now Supported for Notification Handling	3-14
Support for Configuring Diameter Gateway to Bypass Rating During ECE Downtime	3-15
Diameter Gateway Now Supports the 5G Non-Standalone Architecture	3-15
ECE Now Supports 5G CHF Protocol	3-15

New HTTP Gateway for 5G Networks	3-15
Active-Hot Standby and Segmented Active-Active DR Support Persistence-Enabled Deployments	3-16
New Features in ECE 12.0 Patch Set 2	3-16
New Tools to Monitor ECE	3-16
Running ECE Monitoring Tools	3-18
Customizing JMX-Based Tool Reports	3-19
Monitoring ECE Cluster	3-19
Monitoring ECE Using Grafana and Prometheus	3-19
Starting and Stopping ECE Metrics Collector	3-20
Collecting ECE Metrics Data	3-21
Setting Log Levels by Using Scripts	3-36
Using cohql_file_processor for Query	3-36
Duplicate Check Enhancement in ECE	3-36
Charging Operation Type Can be Configured for Expired Active Session Cleanup	3-37
Configuring Expired Objects Cleanup	3-37
ECE Now Generates POID for Events	3-37
Configuring Cluster ID	3-39
Connecting Rated Event Formatter Instances	3-39
Enabling Prepaid-Event Partitions	3-40
New Tool for Querying ECE Cache Data	3-40
ECE Now Supports Wildcard in Item Type Selectors	3-41
Support for Persisting BRS Configuration Data	3-41
Rated Event Partition in the ECE Persistent Database	3-43
New Features in ECE 12.0/12.0 Patch Set 1	3-44

4 New Features in PDC

New Features in PDC 12.0 Patch Set 8	4-1
PDC Supports Flexible Proration Options	4-1
PDC Enhancement to Consume Allowances before Charging	4-2
PDC UI Support for Tax Selectors and Tax Exemption Selectors	4-3
PDC UI Support for Automated Product and Discount Sharing Groups	4-3
PDC UI Support for 30-Day Proration	4-3
PDC UI Support for Add-On Charge Offers in Bundles	4-4
PDC Cloud Native Support for SSO	4-4
Support for ProductOffering and ProductOfferingPrice GET APIs	4-5
PDC Support for Discount Offer Stacking	4-5
New Features in PDC 12.0 Patch Set 7	4-5
GUI Support for Credit Limit Enforcement Enhancements	4-5
First Usage Activation Enhancements in PDC	4-5
Design-Time GUI Improvements	4-6

Improved Handling of Balance Elements	4-6
Proration Can Be Configured at the Offer Level	4-6
Support for Add-On Charge Offers in Bundles	4-6
New Features in PDC 12.0 Patch Set 6	4-7
New Features Supported by PDC REST Services Manager	4-7
New XML Options for Exceeding Credit Limits	4-7
Consuming Noncurrency Resources for Discounts in Order of Expiration	4-8
Tax Codes Are Now Created in PDC	4-8
PDC Support for More Granular Tax Code Application	4-9
PDC XML Tags	4-9
PDC Now Supports Single Sign-On	4-9
New Features in PDC 12.0 Patch Set 5	4-9
PDC Supports Event Notifications	4-10
Charges in PDC Support Dynamic Taxation	4-10
Moving Tax Codes to PDC	4-10
Distributing Noncurrency Balances in Incremental Buckets	4-15
Now Specify Whether to Continue Charging for SuspendedActive Accounts	4-16
Support for Purchasing the Same Offer Multiple Times	4-16
Prorating Billing DOM Changes, Package Transitions, and Bundle Transitions	4-16
Setting Charge Offer Cycle Alignment for Reactivated Bundles	4-17
PDC UI Supports Recurring Charges on Specific DOM	4-17
PDC UI Supports Versioning with Date Ranges	4-18
PDC UI Supports Options for Exceeding Credit Limits	4-18
PDC UI Supports Product Specification Attributes	4-19
PDC REST Services Manager Supports OAuth with Oracle Access Management	4-19
PDC REST Services Manager Supports Synchronizing New Elements	4-19
New Features in PDC 12.0 Patch Set 4	4-19
Creating Subscription Terms	4-20
Creating Deliverables	4-20
XML Product Offering Enhancements for PDC	4-20
Integrating PDC with External Enterprise Product Catalogs	4-21
New Features in PDC 12.0 Patch Set 3	4-21
PDC Now Supports 5G Rating in ECE	4-22
PDC Now Localized	4-22
New Features in PDC 12.0 Patch Set 2	4-22
Support for Configuring Item Type Selectors in the PDC User Interface	4-22
PDC Web Services for Creating and Exporting Components	4-22
Creating or Modifying Components using PDC Web Services	4-23
Exporting Components using PDC Web Services	4-26
PDC Synchronizes Event Data Using Event Types	4-30
Synchronizing and Publishing Event Type to ECE	4-30

PDC Now Supports Wildcard in Item Type Selectors	4-31
Support for Enabling SSO Using SAML	4-31
New Features in PDC 12.0 Patch Set 1	4-31
Support for Rolling Back the PDC Patch Set	4-31
Support for Secure Communication With the PDC Database	4-31

5 New Features in Billing Care

New Features in Billing Care 12.0 Patch Set 8	5-1
New Features in Billing Care 12.0 Patch Set 7	5-2
New Features in Billing Care 12.0 Patch Set 6	5-3
Billing Care GUI Enhancements in Patch Set 6	5-3
Billing Care REST API Enhancements in Patch Set 6	5-3
Billing Care SDK Enhancements in Patch Set 6	5-3
New Features in Billing Care 12.0 Patch Set 5	5-3
New Features in Billing Care 12.0 Patch Set 4	5-4
Monitoring Billing Care and Billing Care REST API	5-4
Billing Care Application Enhancements in Patch Set 4	5-4
Billing Care SDK Enhancements in Patch Set 4	5-5
New Features in Billing Care 12.0 Patch Set 3	5-6
Hierarchy Enhancements	5-6
Billing Care Installation Enhancements	5-6
Billing Care SDK Enhancements	5-6
Enhanced Billing Care Security	5-7
New Features in Billing Care 12.0 Patch Set 2	5-7
SDK Enhancements for Customizing Billing Care	5-7
View Bills Generated Before Moving the Account to a Hierarchy	5-8
Support for Account Number and Bill Number in Billing Care Embeddable URLs	5-8
Embeddable URLs for Purchase Package and Purchase Bundle Screens	5-8
New Features in Billing Care 12.0 Patch Set 1	5-8
Additional Options for Searching Bills	5-9
Billing Care Is Now Certified with Mozilla Firefox 58.0	5-9
Billing Care is Now Certified with Oracle Identity and Access Management 12c Release 2	5-9
SDK for Customizing the Purchase Flow and Assets Display	5-9
Support for Backdated Account Creation and Product Purchase	5-9
Support for Creating Wholesale Accounts and Billing Hierarchies	5-9
Support for Customizing Product Information Post Account Creation	5-10
Support for Deferred Actions	5-10
Support for Multischema	5-10
Support for Undefined Payment Method	5-10

6 New Features in Business Operations Center

New Features in Business Operations Center 12.0 Patch Set 8	6-1
New Features in Business Operations Center 12.0 Patch Set 4	6-1
New Features in Business Operations Center 12.0 Patch Set 3	6-2
Business Operations Center Installation Enhancements	6-2
Business Operations Center Supports Custom Jobs	6-3
Business Operations Center Upgrades Now Supported	6-3
Support for Enabling SSO using SAML	6-3
New Features in Business Operations Center 12.0 Patch Set 2	6-3
Workflow Job to Automate Billing Process	6-3
New Features in Business Operations Center 12.0 Patch Set 1	6-5
Business Operations Center is Now Certified with Jersey Bundle 2.27 (Patch Set 1)	6-5
Business Operations Center is Now Certified with Jackson 2.9.6 (Patch Set 1)	6-5
Business Operations Center is Now Certified with Log4j2 2.11.1 (Patch Set 1)	6-5
Business Operations Center Is Now Certified with Mozilla Firefox 58.0 (Patch Set 1)	6-5
Business Operations Center is Now Certified with Oracle Identity and Access Management 12c Release 2 (Patch Set 1)	6-6
Business Operations Center is Now Certified with Quartz Scheduler 2.3.0 (Patch Set 1)	6-6

7 Opcode Changes

New Standard Opcodes	7-1
New Policy Opcodes	7-3
Changed Standard Opcodes	7-5
Changed Policy Opcodes	7-21

8 Storable Class Changes

New Storable Classes	8-1
Changed Storable Classes	8-5

9 Event Notification Changes

New Event Notifications	9-1
Changed Event Notifications	9-2

10 Business Parameter Changes

New Accounts Receivable Business Parameter Entries	10-1
New Billing Business Parameter Entries	10-2
New Customer Business Parameter Entries	10-3
New Invoicing Business Parameter Entries	10-3
New Multibalance Business Parameter Entries	10-4
New Notification Business Parameter Entries	10-4
New Rating Business Parameter Entries	10-4
New Subscription Business Parameter Entries	10-5
New System Business Parameter Entries	10-8

11 Schema and Index Changes

Patch Set 8 Schema and Index Changes	11-1
Patch Set 7 Schema and Index Changes	11-2
Patch Set 6 Schema and Index Changes	11-4
Patch Set 5 Schema and Index Changes	11-7
Patch Set 4 Schema and Index Changes	11-11
Patch Set 3 Schema and Index Changes	11-14
Patch Set 2 Schema and Index Changes	11-14
Patch Set 1 Schema and Index Changes	11-17

12 Utility Changes

New Utilities	12-1
Changed Utilities	12-2
Removed Utilities	12-5

13 Known Problems

Known Problems in BRM	13-1
A/R Actions between Billing and Invoicing Not Captured in Invoice	13-1
Bill Now Generates Two Audit Objects for Nonpaying Child Items	13-1
Cannot Install BRM Reports and BRM Invoices in the Same Home Directory as Oracle Analytics Server	13-1
BRM Server and Client OUI Installers Not Localized in Russian	13-2
Default Invoices Do Not Display Custom Events and Items	13-2
Error When Configuring or Modifying Security Policies for Web Services Manager for JAX-WS	13-2
Event Browser Displays Data Internal to BRM in the Event Description	13-3
Invoices Do Not Display Charge Sharing Details Separately	13-3

Invoices Might Display Wrong Discount Amount	13-3
pin_inv_accts Fails for Migrated Accounts if Business Intelligence Publisher Invoicing Integration Is Enabled	13-3
Refund Is Incorrectly Calculated When Remaining Charge Discount Canceled	13-4
RE Loader Daemon Fails to Process Files Under Load	13-4
Sample Prepaid Service Life Cycle Does Not Support Provisioning of Telco Services	13-4
The SLM Feature Does Not Support Account Billing Relationships	13-5
Settlement Taxes Might Be Incorrectly Calculated	13-5
stop_eai_js Does Not Work	13-5
Summary Invoices Do Not Show Real-Time Discount and Tax Details for Items	13-6
Tax Not Recalculated after Rerating	13-6
Error When Creating a New Data Sequence for BRM	13-6
Known Problems in Business Operations Center	13-7
Known Problems in Billing Care	13-7
Duplicate Login Request When Using Single Sign-On	13-7
Firefox and Internet Explorer Fail to Connect Securely to Billing Care	13-8
Known Problems in PDC	13-8
PDC Search Functionality Not Working As Expected	13-8
Migrating Pricing Data Fails in a PDC System with BRE and RRE	13-8
Adding Balance Groups and Setting Credit Limits in a Package Displays Exceptions	13-8
Migration Fails Even When the Migration Process is Enabled to Skip Errors	13-9
You Are Not Warned to Save Your Changes	13-9
Known Problems in PCC	13-9
Known Problems in BRM Cloud Native Deployment	13-9
Invoicing Formatter Uses Incorrect XSLT Processor	13-9
Missing Subscriber State Configuration Files	13-9
syncPDC May Fail After Upgrade to PDC Cloud Native 12.0 Patch Set 3	13-10
Error When Upgrading ECE from BRM 12.0 Patch Set 6 to Patch Set 8	13-10
Known Problems in BRM REST Services Manager	13-10
Must Use Payment Methods that Are Already Associated with Bill Units	13-11
Cannot Allocate Payments to Bills of Different Bill Units	13-12

Preface

This guide includes information about the features added to the Oracle Communications Billing and Revenue Management (BRM) 12.0 patch sets.

Audience

This document is intended for all BRM users.

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.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

1

Summary of Customer-Reported Fixes

The Oracle Communications Billing and Revenue Management (BRM) 12.0 Patch Sets include multiple customer-reported bug fixes.

Topics in this document:

- [Customer-Reported Fixes in BRM](#)
- [Customer-Reported Fixes in ECE](#)
- [Customer-Reported Fixes in PDC](#)
- [Customer-Reported Fixes in Billing Care](#)
- [Customer-Reported Fixes in Business Operations Center](#)
- [Customer-Reported Fixes in BRM Thick Clients](#)

Customer-Reported Fixes in BRM

See the following for a list of fixed customer-reported issues for Oracle Communications Billing and Revenue Management (BRM) for each patch set:

- [Customer-Reported Fixes in BRM 12.0 Patch Set 8](#)
- [Customer-Reported Fixes in BRM 12.0 Patch Set 7](#)
- [Customer-Reported Fixes in BRM 12.0 Patch Set 6](#)
- [Customer-Reported Fixes in BRM 12.0 Patch Set 5](#)
- [Customer-Reported Fixes in BRM 12.0 Patch Set 4](#)
- [Customer-Reported Fixes in BRM 12.0 Patch Set 3](#)
- [Customer-Reported Fixes in BRM 12.0 Patch Set 2](#)
- [Customer-Reported Fixes in BRM 12.0 Patch Set 1](#)

Customer-Reported Fixes in BRM 12.0 Patch Set 8

[Table 1-1](#) lists the customer-reported issues that were resolved in BRM 12.0 Patch Set 8.

Table 1-1 Customer-Reported Fixes for BRM 12.0 Patch Set 8

Bug Number	Description
34909676	A user could not encrypt data using <code>pin_crypt_app</code> if the user was not defined in <code>/etc/passwd</code> . For a user authenticated by an external LDAP, there will be no entry in <code>/etc/passwd</code> . This was observed for apps linking to Oracle NZ ZT SDK. This has been fixed.

Table 1-1 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 8

Bug Number	Description
34595937	The pin_cycle_fees utility was fetching and updating canceled products, resulting in a deadlock and unnecessary performance issues. This has been fixed. The utility now prevents canceled products from being fetched, which improves performance.
34782991	If a closed bill returned an error, the cycle forward opcode was generating a core. This has been fixed. The code has been updated to exit gracefully without generating a core.
34909677	When an account or service was reactivated, all discounts were reactivated even if the discount was in inactive state before the account or service was reactivated. This has been fixed.
34770422	A recurring charge configured not to be charged when canceled was not being applied even if the service status was made inactive. If the status was not active, the product picked up for charging was later skipped and dropped before rating. This has been fixed.
34872438	When the pin_event_extract utility was run for backout and rerate, there was a failure while opening the pin_event_extract.cfg file. This has been fixed.
34504353	When there are multiple plans, during event creation, PIN_FLD_PROGRAM_NAME was not set due to which rerating was failing. This has been fixed.
34820222	When processing batch payments, for certain error conditions, the payment transaction was partially rolled back and an error buffer was cleared without logging the error. Because of this, though the payment failed event was recorded, the payment event itself was lost due to the partial roll back. This has been fixed. Now, errors are logged before clearing the error buffer.
34768047	In PCM_OP_SELECT_ITEMS, some fields were not getting copied over when the input flist was copied to an intermediate flist. This has been fixed.
34643637	When running trial billing for child accounts in a large hierarchy with -pay_type 10007 generates trial invoices for all child accounts in the system, it was observed that the database size was increasing exponentially. This was caused by memory leaks in invoicing. This has been fixed.
34872435	The pin_ledger_report utility was adding extra lines to the GL report. This has been fixed.
34763825	GL reports were generating empty reports for the secondary schema. This has been fixed.
34648175	Email notifications were not published to DM_AQ due to incorrect transaction handling. This has been fixed. Now, the transaction is committed for every email notification.

Table 1-1 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 8

Bug Number	Description
34574764	<p>The items created as part of migrating subordinate accounts in wholesale billing hierarchies were not having some of the fields populated. This caused subordinate items to not get rolled up to the parent account.</p> <p>To fix the issue, changes were made to populate PIN_FLD_AR_BILL_OBJ, PIN_FLD_AR_ITEM_OBJ, and PIN_FLD_ITEM_CLASS for subordinate items for wholesale billing hierarchies.</p>
34815112	<p>The pin_monitor_balance utility was reporting errors for write-off reversal events. It was observed that the PIN_FLD_SUB_BALANCES array was truncated in the input for write fields. This occurred because the balance_large column of EVENT_ESSENTIALS_T table was not being completely read by dm_oracle when there were a large number of balance arrays.</p> <p>This has been fixed.</p>
34699267	<p>Performance issues were observed with pin_cycle_fees when run in -verbose mode. This occurred because it attempted to collect a list of unique account IDs that were processed using a linear search on a large flist.</p> <p>This has been fixed. It now avoids the search for verbose mode, and only calls it for revenue assurance mode.</p>
34560096	<p>The config credit profile search was resulting in a "No memory" error in production environments, because the customer had a huge number of credit profiles configured.</p> <p>This has been fixed. The code was modified to search only required credit profiles and update it in the global flist, instead of fetching the whole object.</p>
34824478	<p>In the billing flow, a rounding issue was observed for the bill's service charges.</p> <p>This has been fixed. Rounding with correct precision occurs when the GenerateJournalEpsilon business parameter is enabled.</p>
34927430	<p>A performance issue was observed with the opcode PCM_OP_AR_RESOURCE_AGGREGATION.</p> <p>To fix the issue, the PCM_OP_AR_RESOURCE_AGGREGATION opcode flow was modified to use PCM_OP_READ_FLDS instead of PCM_OP_SERACH, which eventually improves the opcode performance.</p>
34459316	<p>It was observed that the tax created by adjustments done during the delay period was not included in the final bill.</p> <p>This has been fixed.</p>
34840219	<p>The PDC synchronization performance was very slow.</p> <p>This has been fixed. Some of the search queries and the pin_flist_get_next function were optimized. PIN_FLIST_ELEM_TAKE_PREV is introduced to optimize few of iterations during the set_price_list flow. All these changes are controlled using an Infranet.properties entries. To use the optimized code, set the property infranet.ondemandcaching=1 in the load_price_list Infranet.properties file. If the property is not set, set_price_list will perform as earlier.</p>
34932387	<p>With open item accounting, corrective bills were not updating the current total and total due in the bills after bill adjustments. This caused mismatches, and corrective bills still showed the original amounts.</p> <p>This has been fixed.</p>
34583876	<p>It was observed that the journal update was taking longer than expected.</p> <p>To fix the issue, the journal table was updated using a stored procedure instead of using BRM opcodes.</p>

Table 1-1 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 8

Bug Number	Description
34829048	The pin_recycle utility was forming a bad template and reporting an error. This was caused by an issue with string manipulation. This has been fixed.
34829050	When reactivation of closed accounts with stop_bill_closed_accounts was enabled, invoice items for new purchases were not displayed after reactivation. Due to performance reasons, invoicing searches for items within the last 60 days to reduce searching pruned partitions. When item_t was not partitioned and items were pre-created more than 60 days ago, a new invoicing business parameter DisableFilterItemByPruneTime was introduced to skip searches based on partition pruning time. The search now is done for all items.
34763827	It was observed that the parent CM process was doing the config refresh activity in another thread when the business parameter ConfigCacheRefreshInterval (2 minutes) was enabled. To fix the issue, the config refresh logic was moved from the multithreaded CM to separate child CM processes.
34824477	A fixed size was allocated for stored procedures in cache and was not configurable. This has been fixed. Changes were done to read the size from the CM pin.conf file so that users can configure the size according to business requirements.
34727941	The pin_wsdl_generator utility was throwing SAX parser exceptions as it was expecting to validate the XML file against a DTD file. Changes were made to set validation to false, so that the XML file is not validated against a DTD File.
34805598	The PIN_FLD_PROGRAM_NAME field was mandatory for the PCM_OP_CUST_CREATE_BAL_GRP opcode, but it was missing in the documentation. This has been fixed.
34703854	The pipeline was unable to start on Solaris if there was more than one iScript file in EvalScriptFiles . To fix the issue, a new input stream object was created for each file.
34629289	The PCM_OP_AR_GET_ACCT_BAL_SUMMARY opcode was throwing errors after deleting the last bill. To fix the issue, a check was introduced for last bill availability before fetching the same for processing.
34820221	The pipeline process was crashing when the customer login was found without having any valid service reference. To fix the issue, the error message "ERR_CUSTOMER_LOGIN_SERVICE_NOT_FOUND" was added. Also, a null check was added for the pointer variable.
34690037	Debug messages were logged incorrectly. This has been fixed. The validation message was moved from the discount trace to the normal level pipeline log when no usage map is found for the given event type.
34791706	The Balance Tab view for an account with multiple child accounts was taking approximately four minutes to load the page due to huge search results and corresponding processing time. This has been fixed. The search query now returns only the appropriate results.

Customer-Reported Fixes in BRM 12.0 Patch Set 7

Table 1-2 lists the customer-reported issues that were resolved in BRM 12.0 Patch Set 7.

Table 1-2 Customer-Reported Fixes for BRM 12.0 Patch Set 7

Bug Number	Description
34147509	The partition pruning condition was not being added for delayed events when the PCM_OP_SEARCH template contained a type-only POID and CREATED_T in its search arguments. The check for partitioned classes was skipped, causing the subsequent logic of adding the partition pruning condition not being run. This has been fixed.
33868116	When an environment variable was referenced in a pin.conf file in \${ENV} format and the environment variable was not defined, the Oracle DM process hung during startup. This has been fixed. The Oracle DM process now logs an error and exits gracefully.
34073234	The BRM cloud native CM service was not allowing external access using an IP address. Only one DNS name was supported in the TLS certificate. This has been fixed. You can now configure multiple DNS names and IP addresses in the TLS certificate.
34031791	The deployment of a large BRM storage model was failing because tablespace details were not getting updated correctly due to syntax issues while concatenating the Perl strings. This has been fixed.
34289048	The pin_multidb script was failing because internally PinResetSeq was failing. This was due to a CLASSPATH mismatch for ojdbc8.jar . This has been fixed. The correct CLASSPATH was updated for ojdbc8.jar and pin_multidb.pl to successfully complete the multischema configuration.
34043082	When 30-day proration was enabled, discount scale calculations were incorrect for sequential cycle discounts. This has been fixed.
34123644	The bill adjustment flow was not working when the BillPaymentDeallocation AR business parameter was enabled and the item array was sent in the input flist. This has been fixed. The opcode flow now unallocates the payment amount and performs the bill adjustment successfully.
34172078	Connection Manager (CM) was crashing while running the PCM_OP_AR_GET_BILL_ITEMS opcode because it was trying to read an object from an invalid output flist. Additional checks have been added in the PCM_OP_AR_GET_BILL_ITEMS opcode flow to report an error so that the CM will not crash.
34086305	The validity start and end dates for resources granted as part of purchase fees were calculated incorrectly. This has been fixed.
34246136	The Oracle DM process was running redundant queries, which degraded performance. This has been fixed.

Table 1-2 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 7

Bug Number	Description
34023158	When the PCM_OP_SUBSCRIPTION_SET_PRODINFO opcode was called, the start and end dates were updated incorrectly for deferred product purchases. This led to valid discounts not getting applied to products. This has been fixed.
34074210	When a charge offer included a maximum number of charge offer instances that a customer could own at one time, BRM included expired products in its count of active product instances. This has been fixed.
34043086	Cycle charges were not applied correctly when a product was activated and inactivated multiple times in a cycle. This has been fixed.
34391835	Multiple cycle arrears discounts were not applied correctly when they started and ended in the same cycle. This has been fixed.
34320482	When a service was suspended and unsuspended on the same day, the balance bucket for future months was not getting created even after running partial billing. The bucket was created only when the account was suspended and unsuspended again. This has been fixed.
34064240	Discounts were getting canceled completely even when only a partial quantity was canceled. This has been fixed so that the partial quantity is subtracted and the discount is not canceled.
34209613	The Paymentech DM was sending junk characters when sd_*_{dba,pdt,phone} was not configured. This has been fixed. The DM does not send junk characters in the M record and is also enhanced to send an empty M record so that defaults configured for Paymentech are used.
34125857	For payment inquiries, the debit number was not visible when the debit number was null in the BRM database for paid amount. This occurred because the PIN_FLD_DEBIT_NUM and PIN_FLD_CARD_TYPE fields were being dropped during calls to the PCM_OP_BILL_RCV_PAYMENT opcode. This has been fixed, and the logic is corrected to not drop these fields.
34175443	After account creation, duplicate packages were being displayed under subscriptions. This occurred because the grouping of purchased products was done using the plan POID and creation time. This has been fixed. The plan POID and package ID are now used for grouping.
34058072	Warning issues occurred when the declaration of variables was static in the header file and not used in the corresponding file where it was imported. This has been fixed by changing the location of variable declarations and definitions from the header file.
34052733	When the ApplyDiscountOnZeroCharge rating business parameter was disabled and the product scale was a non-zero value, a signal 11 error was generated during a call to PCM_OP_SUBSCRIPTION_SET_PRODINFO. This was caused by a memory release issue. This has been fixed.

Table 1-2 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 7

Bug Number	Description
34048119	A signal 11 error occurred in PCM_OP_CUST_COMMIT_CUSTOMER, because an flist was destroyed but the pointer was not assigned to null. This has been fixed.
34095404	When BRM opened a read-only transaction for an individual opcode that had only read operations, time-based consistency sometimes led to an ORA-01466 error. This has been fixed. If an ORA-01466 error occurs during read-only transactions, BRM now rolls back the transaction and retries the operation a configurable number of times.
34308528	Opcodes have been added to BRM Web Services. The following opcodes have been added in JCA15Adapter.rar , BrmWebServices.war , and infranetwebsvc.war : <ul style="list-style-type: none"> PCM_OP_SUBSCRIPTION_ORDERED_BALGRP PCM_OP_SUBSCRIPTION_ORDERED_BALGRP_BULK_MODIFY
34115373	While loading balances with credit profile fields in an input XML file, pin_cmt was throwing NullPointerException. This occurred because of a missing sequence in the REC_ID column of the CFG_CREDIT_PROFILE_T table. This has been fixed.
34052735	When the PIN_FLD_BUSINESS_PROFILE_OBJ input flist field in PCM_OP_CUST_COMMIT_CUSTOMER was set to a non-invoicing type, an iScript was not triggered. This has been fixed.
34223098	The cross-schema account hierarchy was failing because hierarchies can be built only with accounts residing in the same schema. The error message in the cm.pinlog file has been enhanced to include more meaningful error information.
33994664	When the ConfigCacheRefreshInterval system business parameter was enabled, the cache refresh for /config/provider_taxes happened every time taxation was requested in BRM. This has been fixed.

Customer-Reported Fixes in BRM 12.0 Patch Set 6

[Table 1-3](#) lists the customer-reported issues that were resolved in BRM 12.0 Patch Set 6.

Table 1-3 Customer-Reported Fixes for BRM 12.0 Patch Set 6

Bug Number	Description
33599255	For events having multiple tax codes, tax-only adjustments were reversing the wrong amount and an additional amount was getting reversed. This has been fixed.
33705754	When timestamp rounding was disabled, cycle fee end dates were still being rounded. This lead to incorrect scale calculation. This has been fixed.

Table 1-3 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 6

Bug Number	Description
33389642	<p>After the billing day of month (DOM) was changed within the delay period, the /schedule object's PIN_FLD_ACTG_FUTURE_T field was updated. When pin_bill_accts was run, the /schedule object's PIN_FLD_ACTG_FUTURE_T field was incorrectly being aligned to PIN_FLD_ACTG_NEXT_T.</p> <p>This has been fixed. Now, the /schedule object will not be created and instead the DOM will be updated in the /billinfo object, which will align with PIN_FLD_ACTG_FUTURE_T when billing is triggered.</p>
33150179	<p>Recurring fees were being applied when the subscriber was in a suspended state and the state was being changed from suspended to active.</p> <p>This has been fixed. PCM_OP_CUST_SET_STATUS passes a new flag to the PCM_OP_SUBSCRIPTION_SET_PRODUCT_STATUS opcode, which indicates whether to skip applying the cycle fee.</p>
33915097	<p>BRM cloud native encountered an invoice export issue in which the formatter threw an java.lang.NoClassDefFoundError:oracle/xml/parser/v2/DOMParser error due to missing XML parser JAR files.</p> <p>This has been fixed. In the formatter image, the xml.jar and xmlparserv2.jar files have been added to \$ORACLE_HOME/lib, and the orai18n-collation.jar file has been added to \$ORACLE_HOME/jlib.</p>
33803343	<p>BRM 12.0 introduced functionality that allowed accounts to be charged cycle fees when they were inactive or canceled. After upgrading from BRM 7.5 to BRM 12.0, cycle fees were applied to existing accounts that had inactive or canceled services.</p> <p>This has been fixed. A new business parameter (ApplyChargeOnInactiveOrCancelledProduct) controls whether to use the new functionality or keep the BRM 7.5 behavior.</p>
33899544	<p>After an account was written off, the payment reversal did not trigger a write-off or write-off reversal when the payment was made before the account write off. A write-off reversal was triggered for payment reversal only when the payment was made after the account write-off.</p> <p>This has been fixed. Payment reversals now perform a write-off reversal and then write off the remaining due.</p>
33830067	<p>The range in pin_flds was incorrect for some tax-related fields. The range from custom fields was getting applied to base fields.</p> <p>This has been fixed.</p>
33860907	<p>The /service object was NULL for account level-discounts. In this case, when setting end dates for a discount, BRM incorrectly picked up the root account.</p> <p>This has been fixed. The correct /account object is updated now.</p>
33910399	<p>A NULL POID for a custom item was causing an error in the read_flds operation for the item.</p> <p>This has been fixed. NULL checks were added to allow read_flds only when a valid item POID is available.</p>
33589197	<p>During rerating, new rerated events were not generated because the product was configured to not charge after cancellation (which is the default option).</p> <p>To fix this issue, the offer's default configuration has been changed to Never stop charging. This allows new events to be created and balances to be correctly populated during rerating.</p>

Table 1-3 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 6

Bug Number	Description
33935382	For resource buckets with in-advance billing, the validity of the resource bucket was inaccurate when the billing DOM was changed. This has been fixed so that validity dates are consistent even when the billing DOM is changed for a bucket with in-advance billing.
33910398	There was a requirement for charges to be applied to customers even when their service was inactive. A new business parameter, ApplyChargeOnInactiveOrCancelProduct , has been added so customers can decide whether to charge an inactive service or not.
33966621	Plan transitions were core dumping when external system operations resulted in an error during a custom call. This has been fixed by stopping the flist data from being destroyed.
33945645	When timestamp rounding was enabled, setting discount end dates was rounding the dates to midnight. This has been fixed.
33709249	Opening the Canceled Product tab in the Customized Plan table was resulting in a signal 11 error. This occurred because a NULL source string was copied to the destination string. This has been fixed. A NULL check was added to validate whether the source string is NULL. If NULL, the string copy function is not initiated.
33870612	The pin_rel utility generated an ORA-20003 error when dealing with sub-balance impacts with a 0 amount. This has been fixed.
33626806	An incorrect credit profile was being set for grants of daily bundles. This has been fixed. By default, the RE Loader procedure now sets the balance group's credit profile to 0 for currency resources and to 2 for non-currency resources.
33611926	While running the pin_ledger_report utility in -export mode, the default GL procedure was trying to insert duplicate rows in the LEDGER_REPORT_ACCTS_T table, resulting in a unique constraint violation error. This has been fixed. A new flist is used to create incremental objects.
33996955	Paymentech Data Manager (dm_fusa) was logging high watermark memory issues for large batches due to qm_shmsize 's maximum limit of 512 MB. To fix this, the maximum limit of qm_shmsize was increased to 1024 MB.
33968093	When there was no change in the usage event price, after rerating, tax charges were backed out but not reapplied for that event. This has been fixed.
33935381	After rerating, rollovers were incorrect because a new rollover event was not getting created when delayed billing was enabled. This has been fixed.
33705198	The pin_deposit utility was creating additional checkpoints even though a checkpoint already existed. This has been fixed.

Table 1-3 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 6

Bug Number	Description
33705205	An issue occurred during discount cancellation when PDC was configured to prorate charges and discounts upon cancellation. Discounts were canceled but the /purchased_discount object's PIN_FLD_CYCLE_END_T field was set incorrectly. This has been fixed.
33390957	Due to credit limit checking, rerating failed when publishing backout events. This has been fixed. Credit limit checks are now ignored for backout events during rerating.
33206140	The Java API XMLToFlist and BRMXMLToFlist classes were converting the string "dummy" to an empty string. This has been fixed.
33560838	An incorrect value was added to the /item object's PIN_FLD_AR_ITEM_OBJ field when the DeferredTaxJournaling business parameter was set to 2 (enabled). This has been fixed.
33752787	The PCM_OP_CUST_COMMIT_CUSTOMER opcode was resulting in a signal 11 error during account creation. This has been fixed.
33694272	When External Manager (EM) Gateway was restarted, the prepare phase of JCA Adapter was resulting in an exception. This has been fixed. A flag was added to the handleException method to ensure that the transaction status is not changed during the transaction prepare state.
33860010	Unused variable warnings were noted during code compilation. This has been fixed by removing all variables that were declared but not utilized in the code.
33637310	Bill units that were eligible for collections were not entering the collections process. This has been fixed. The search template was modified for custom profiles used for validation of scenarios in the fm_collections_pol module.
33852732	During the event adjustment process, the tax exemption amount was double the desired amount. This occurred because the exemption was not handled correctly, resulting in an overflow of values. This has been fixed.
30215740	Fortify issues were found in fix_data_dictionary.sql and create_amt_mv_pkb.sql . This has been fixed.

Customer-Reported Fixes in BRM 12.0 Patch Set 5

[Table 1-4](#) lists the customer-reported issues that were resolved in BRM 12.0 Patch Set 5.

Table 1-4 Customer-Reported Fixes for BRM 12.0 Patch Set 5

Bug Number	Description
33275372	During billing, the call to the update_transfer_event stored procedure was failing due to the length of event_poid_list , which had a variable size of 4000 but a database size of 32000. This has been fixed. The variable size has been updated to 32000.
33249053	The Java PCM returned an ERR_NO_SOCKET error when getting the output stream from the socket or when the socket channel received an IOException. ERR_NO_SOCKET was not being considered for reestablishing the socket connection. This has been fixed by introducing a STREAM_IO exception so that reconnect attempts will be done on the socket.
32992056	There were synchronization issues when the Java PCM connection pool was accessed by multiple concurrent threads. This has been fixed.
32788794	Files required for building the BRM Web Services Manager Docker image were not shipped with the BRM cloud native Dockerfiles package. This has been fixed. The required files are now included with the package.
33424947	The service monitor was getting created and pushed to Prometheus, resulting in an error. This has been fixed. The PodMonitor and ServiceMonitor will be created only when both Prometheus Operator is deployed and the corresponding BRM service is enabled.
33124301	Bill Now was not refunding discounts for cycle forward arrears products. When Bill Now was run, the charges were prorated when the apply_cycle_fees_for_bill_now business parameter was enabled. This has been fixed to pick the discount event for refund mode.
33296584	During billing, applying cycle forward charges failed when fetching rate plans. This has been fixed. Changes were done to skip fetching rate plans, which are not found in the database during billing.
33189662	When ECE granted noncurrency balances on first usage through a purchase event, the validity date of the granted balance was not timestamp rounded. For the same noncurrency balance, if consumption occurred on the same day through a purchase event, the validity dates were timestamp rounded. This caused a new bucket to be created rather than using the granted balance through first usage. This has been fixed. No timestamp rounding is done when consumption happens through a purchase event. The existing granted bucket resource will be used for the consumption instead of creating a new one.
33126712	When a recurring or one-time product was purchased with an end date, the sub_bal_validity end time of grants was being set as NEVER. This has been fixed. Now, the resource validity aligns to the product validity after enabling the business parameter RestrictResourceValidityToOffer under bus_params_multi_bal.xml.
33288689	After removal of the Xerces dependency, the JARs and CLASSPATH were not updated in BRM cloud native. This has been fixed. The XSLT engine parameter in the Infranet.properties file has been updated, and the Xerces JAR is removed from CLASSPATH.
33177179	If a product is configured with "stop charging only when cancelled", products were not being charged on service inactivation. This has been fixed. now and the product will be charges event after service inactivation

Table 1-4 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 5

Bug Number	Description
33198841	Only cycle arrears monthly events were being skipped during the cancellation of a product if it was already charged. This has been fixed.
32698080	A memory leak was reported in different FM components. This has been fixed.
32974824	There were performance issues reported with invoicing. When PIN_FLD_GL_ID was not found in PIN_FLD_INVOICE_DATE (which is called for each event), a new context was opened for each event in PCM_OP_INV_DECODE_INVOICE_DATA. This caused PIN_ERR_DM_CONNECT_FAILED and an error to be reported intermittently. This has been fixed. PCM_OP_ACT_POL_SPEC_EVENT_CACHE adds a default value for PIN_FLD_GL_ID if it is not passed in. If PIN_FLD_GL_ID is missing, one context is opened per invoice instead of per event.
33124299	Cycle_forward_arrear events were being assigned to the wrong items during the discounting flow. This has been fixed.
33126650	If fixed discounts that were not dependent on the product charge were configured, during product cancellation, corresponding discounts were getting refunded even though the product refund was zero. To fix this, use the new apply_discount_on_zero_charge business parameter. If this business parameter is disabled, discounting will not be called when the product scale after rating is zero.
33194360	When /event/billing/refund/cc was generated for failed payments, the PIN_FLD_ITEM_OBJ field was not being set. This caused the item object to not be selected. This has been fixed now to set the POID of PIN_FLD_ITEM_OBJ in /event/billing/refund/cc .
33036300	When the PCM_OP_SUBSCRIPTION_SET_PRODINFO opcode was called to modify a product before the product was activated and before a purchase fee was applied, the PIN_PROD_PURCHASE_CHARGED flag was set. This resulted in the purchase fee being skipped when the product was activated. This has been fixed by modifying the code to set PIN_PROD_PURCHASE_CHARGED only if the purchase fee is applied.
33008857	The Oracle DM was crashing when the PCM_OP_DELETE_FLDS opcode was called with PCM_OPFLG_LOCK_NONE. This has been fixed.
29283875	The BRM documentation mentioned that PCM_OP_SUBSCRIPTION_PURCHASE_PRODUCT takes absolute time for PIN_FLD_START_T and PIN_FLD_END_T. This implied that the date was an absolute date even though the input was always considered as relative to epoch time in terms of seconds. The documentation has been fixed to reflect the correct behavior.
32631247	The PCM_OP_COLLECTIONS_REPLACE_SCENARIO opcode was not canceling actions when called from custom code. This has been fixed.

Table 1-4 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 5

Bug Number	Description
32774746	The following opcodes were added to BRM Web Services and OracleBRMJCA15Adapter : <ul style="list-style-type: none"> PCM_OP_CUST_FIND_PROFILE PCM_OP_CUST_SET_TAXINFO PCM_OP_CUST_SET_BAL_GRP PCM_OP_CUST_CHANGE_BUSINESS_PROFILE
33380517	PCM_OP_CUST_DELETE_ACCT was deleting the /user_activity object. This has been fixed.
33321834	If the pcm_timeout_in_msecs parameter was set in the CM pin.conf file, the CM was not coming up. This has been fixed.
33510634	A memory leak was reported during a call to the PCM_OP_SUBSCRIPTION_CANCEL_PRODUCT opcode. This has been fixed.
33177178	The PIN_FLD_EVENT_POID_LIST data type was being changed to CLOB, and the DISTINCT search query was not accepting the CLOB data type. This has been fixed. The query was modified to a normal search query because POIDs in the item_t table are unique and duplicates are not possible.
33076623	When a refund was reversed, an /event/billing/refund/reversal event was generated without any balance impact. This has been fixed.
32554355	UEL considered any events with zero RUM quantity to be errors. This has been fixed. To suppress errors on ZERO_QUANTITY rating, enable the infranet.uel.ignore_zero_quantity_rating_status parameter in the Infranet.properties file.
33214373	Unexpected results occurred when rerating two accounts that were in a sharing group. This has been fixed. A new u option was added for rerating accounts in the sharing group properly.
33126690	Directly calling the PCM_OP_PRICE_SET_PRICE_LIST opcode failed when the discount trigger code or the discount rule code was longer than 10 characters. This has been fixed by extending the size of V_DSC_TRGR_CODE VARCHAR2 and V_DSC_RULE_CODE VARCHAR2 from 10 to 255 characters.
33184937	When UEL was checking for PIN_FLD_RATING_STATUS, the PCM_OP_ACT_USAGE opcode returned this field in all successful cases, resulting in failure. This has been fixed. UEL now ignores the PIN_FLD_RATING_STATUS output from PCM_OP_ACT_USAGE.
33076624	Some prepaid account balances were not synched from ECE to BRM. This has been fixed. An MTA application corrects the data in BRM by comparing its balances with the balances in ECE.
33314948	Transferring a service from one bill unit to another was failing in some cases because the flist was not initialized. Reading this flist data resulted in an error because it was pointing to an unknown memory location. This has been fixed.

Table 1-4 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 5

Bug Number	Description
33490938	The generate_hash function was throwing a “wallet does not contain the key” error because it couldn’t find ORACLE.CGBU.BRM.INVOICE_KEY. This has been fixed. A condition has been added to pin_config_editor to update the invoice key.

Customer-Reported Fixes in BRM 12.0 Patch Set 4

[Table 1-5](#) lists the customer-reported issues that were resolved in BRM 12.0 Patch Set 4.

Table 1-5 Customer-Reported Fixes for BRM 12.0 Patch Set 4

Bug	Description
31937598	When the PCM_OP_BULK_WRITE_FLDS opcode was invoked to change an alias, the call failed with a PIN_ERR_STORAGE error. The opcode was locking balance groups and retrieving too much data, which caused excessive memory consumption. This has been fixed.
31890124	When performing a full adjustment on an event including multiple tax balance impacts with negative tax amounts, the output tax jurisdiction was being calculated incorrectly. This has been fixed.
32023845	In Customer Center, loading a large number of events in the Event Browser was taking a long time. This was because the step size for the search was limited to 10 and time zone details were being read one at a time while retrieving individual records. This has been fixed. Now: <ul style="list-style-type: none"> The step size is 500 by default, and you can change it in the CustomerCenter.properties file. Time zone details are retrieved as part of the search rather than read individually.
32434963	In Billing Care, the history page for batch payments was displaying the same result multiple times. This has been fixed.
32039813	Creating an account with a package containing an account-level bundle was failing. This has been fixed.
31180010	In Payment Center, deleted suspended payments are displayed by default. You can now choose to hide deleted suspended payments by setting the paymentsearch.showremoved property in paymentcenter.properties to false .
32078942	In Billing Care, when raising an event dispute with an amount greater than 1000, the reason codes list of values were not loading. This has been fixed.

Table 1-5 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 4

Bug	Description
31452531	In Billing Care, searching for a specific bill for a specific account was taking a long time. This was because the search was retrieving all accounts rather than just the specified account. This has been fixed.
31037692	In Billing Care, processing a batch payment file was taking a long time and the page was not refreshing. This has been fixed.
32052513	In Billing Care, navigation was very slow and the default search was taking a long time to load the results. This has been fixed.
30528415	In Billing Care, you could not select a batch payment template due to a missing custom payment type schema. This has been fixed. Billing Care now generates a schema for the custom payment object that uses the default /payinfo object and events.
32607344	Rated Event Loader was not deleting TMP_PROF tables for files that initially failed to load with an UPDATE_ERROR, but were subsequently successfully loaded. This has been fixed.
32258047	Daily batch jobs, such as trial billing and billing, were resulting in frequent core dumps. This has been fixed.
32687843	Synchronizing users with DM_LDAP was failing because the PIN_FLD_DN returned by PCM_OP_SEARCH was incomplete. The service that copied the DN internally had the wrong destination buffer size. This has been fixed.
32983797	The load_config utility was loading empty balance element names as empty when they contained special characters. This has been fixed. The utility now accepts UTF-8 encoded special characters in the value field of the input XML file.
32483033	The load_pin_customer_segment utility skipped customer segment descriptions that contained Latin Bosnian characters, so that the customer segment entry created in /config/customer_segment had no description. This has been fixed. The utility now accepts UTF-8 encoded special characters in the value field of the input XML file.
31155909	Exporting invoices with a large number of events (more than 2000) was failing. This has been fixed. The maximum string length allowed has been increased to 32000.
32124017	Universal Event Loader was failing frequently with an invalid_conf error when infranet.connection was configured in Oracle Wallet. This has been fixed.
32832165	The database was using multiple execution plans instead of a single shared one. This was because the batch controller was using literals. This has been fixed. Now searches for overdue and failed batch controller use bind variables.

Table 1-5 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 4

Bug	Description
31037694	Some of the values in the eai_js and batch controller Infranet.properties files did not support environment variables. This has been fixed. Now the InfranetUtil.substituteEnvVariable() utility function replaces the environment variable with its value for any configuration parameters using an environment variable.
31577118	Converting a large XML payload containing custom fields to BRM flist format using the XmlToFlist Java API was taking too long. This has been fixed.
32372260	When processing multiple opcodes as part of the same XA transaction, if JCA Adapter received unexpected XA_END messages before XA_PREPARE and XA_COMMIT messages, the XA transaction ID was disassociated and could not be retrieved when the XA_PREPARE and XA_COMMIT messages arrived. This caused the opcodes to fail. This has been fixed. The transaction ID is only disassociated when JCA Adapter receives an XA_END message if the transaction is not ongoing.
31972993	While running many web services in parallel, the Data Manager (DM) that handles Connection Manager (CM) clients was not matching external correlation IDs to the correct web services. This has been fixed.
31224153	When the CM was refreshing /config objects, child CM processes were stuck in a thread mutex lock, causing application clients like pin_bill_accts to stop responding. This was because the refresh thread was opening a new context with every refresh. This has been fixed. Now the same initially-opened context is re-used in every refresh.
31851731	When ConfigCacheRefreshInterval was set to a non-zero value and a configuration was updated while updating the business parameters, CM was unresponsive when it tries to update the cache after the specified interval. This has been fixed.
30318315	The PCM_OP_TRANS_POL_COMMIT and PCM_OP_TRANS_POL_ABORT opcodes were receiving a NULL input flist, so any subsequent opcodes could not perform customizations. This has been fixed. The input flist for these opcodes now matches that of PCM_OP_TRANS_OPEN.
32333625	The pin_ctl stop all command was not stopping with a Can't create semaphore file error because the registry path was incorrectly set. This has been fixed.
32146320	The pin_rel utility was first checking for database connection data from Oracle Wallet, then if it wasn't there, checking Infranet.properties . This was the opposite of the intended behavior. This has been fixed. The utility now checks Infranet.properties first, then Oracle Wallet.
32823431	When running billing for accounts with a /billinfo status of 10103, the BILLING_UPDATES procedure generated an error due to an invalid pointer reference that populated PIN_FLD_NAME incorrectly in input flist. This has been fixed.

Table 1-5 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 4

Bug	Description
32451031	When using Bill Now between regular cycles, for any pre-created items, charges were not being tagged to the correct bill item. This has been fixed.
32088651	Some of the BRM policy opcode files were failing Fortify vulnerability scans and were incorrectly being reported as risks. This has been fixed.
31111615	Correlation IDs are were stored in HashMap for the life time of Java processes, which occupied significant memory for long-running programs. This has been fixed. Now the correlation IDs are stored in a ThreadLocal variable, which allows them to be cleared as part of the thread exit. Correlation IDs now also store incremental numbers for salt values instead of 0.
31030685	Vulnerabilities were reported in BRM code when running Fortify scan static source analyzer. This has been fixed.
31589840	In BRM Cloud Native deployments, the following of the subscriber state configuration files were missing from Connection Manager pod: <ul style="list-style-type: none"> • config_lifecycle_states.xsd • config_lifecycle_states.xml • config_service_state_map.xsd This has been fixed.
31701623	During sequential discounting, discounts were being calculated incorrectly when they were closed and then purchased again. This has been fixed.
30771215	While configuring high availability for the connection between dm_oracle and RAC, after service relocation dm_oracle was being reconnected to a new instance with a double number of sessions. This has been fixed.
31029474	There was excessive logging in dm_oracle.log when debug logging was disabled, due to unconditional logging being set for stored procedures. This has been fixed.
31785123	Creating a relationship between two accounts was resulting in the following error: <pre>PIN_ERR_UNKNOWN_EXCEPTION: force due_t = bulk_write_flds problem</pre> This has been fixed. Now the error buffer in dm_oracle is cleared when a balance lock query returns no rows to the PCM_OP_BULK_WRITE_FLDS or PCM_OP_BULK_DELETE_OBJ opcodes.
31500865	When a while card (%) was used in PCM_OP_SEARCH, the opcode was returning a BAD_ARG error and extra conditions were added in the where clause for the partitioned classes. This has been fixed.

Table 1-5 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 4

Bug	Description
31939059	Running pin_crypt_upgrade_keys for a second time re-encrypts already upgraded data keys in the cryptkey_t table. This has been fixed. Now, a new upgraded column in cryptkey_t indicates if the keys have been upgraded, and pin_crypt_upgrade_keys upgrades only the keys with 0 in the upgraded column. dm_oracle inserts the value in the upgraded column when new cryptkey_t records are inserted.
31486823	pin_crypt_app did not support -update_dm_conf and -update_fusa_conf for AES encryption. This has been fixed.
30756021	The Infranet.properties files for the UEL and EAI JS processes did not accept environment variables. This has been fixed. For example, you can use <code>infranet.log.file=\${custom_tracing_logdir}/test.log</code> in Infranet.properties , and the custom_tracing_logdir environment variable will be used to locate the log file.
31420362	The Infranet.properties files for the Rated Event Loader processes did not accept environment variables. This has been fixed. For example, you can use <code>infranet.log.file=\${custom_tracing_logdir}/test.log</code> in Infranet.properties , and the custom_tracing_logdir environment variable will be used to locate the log file.
30891415	In the BRM 12.0 Patch Set 2 SDK package, the fm_bill_pol_pre_fold.c : file, which is no longer used, caused compilation errors. The file has been removed from packaging and installation.
31974952	Running Valgrind against BRM 12.0 Patch Set 3 generated Invalid Read errors due to a memory issue. This has been fixed.
31372474	When running pin_deferred_act , a MTA application that invokes PCM_OP_ACT_SCHEDULE_EXECUTE to run scheduled actions was not reporting error details when the opcode failed. This has been fixed.
31811046	When running pin_deferred_act with the verbose parameter, a MTA application was displaying the wrong number in the error record count. This has been fixed. The MTA framework now does not increase the error count when the exit flag is set due to the maximum number of errors being reached.
32029981	Communication performance issues were occurring with local Unix ports between CM and DM. This has been fixed by disabling SSL hostname verification in the SSL handshake for AF_UNIX domain and the local UNIX port setup.
32779799	Syntax issues in the PL/SQL package declaration were occurring due to CREATE OR REPLACE PACKAGE BODY pin_get_fields database calls. This has been fixed.

Table 1-5 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 4

Bug	Description
32929448	When a SQL session's NLS_DATE_LANGUAGE parameter is set with any European language other than American English, the Date_to_Poid() function of the partition_utils SQL script fails. This has been fixed by replacing the date format in to_date() function from "English 01-JAN-1970" to "01011970".
32550611	In ECE, if auditing is disabled for /service fields, because ECE reads /au_service when restarting processes, ECE fails to start up. This has been is fixed by enabling audit as default for the following fields: <ul style="list-style-type: none"> • /service: <ul style="list-style-type: none"> – LOGIN – ALIAS_LIST – ALIAS_LIST.NAME • /account: <ul style="list-style-type: none"> – ACCOUNT_NO
31864990	In BRM High Availability configuration, running threads were hanging when a host was down. This has been fixed with a connect timeout setting.
32618809	In Billing Care, validity dates appear incorrectly. Instead of displaying the valid to end date, the current date was shown as default. This has been fixed.
32230733	In Billing Care, on large databases, the assets page and services were not loading due to Billing Care retrieving all /schedule objects. This has been fixed. The query now only retrieve the selected /schedule objects.
31082441	In Billing Care, you can now add additional attributes to application cookies by using SDK configurations. For example, to add a new attribute, samesite , to the application-level cookies, you use the Billing Care SDK to add it as a value for the appCookieAttribute configuration key in CustomConfigurations.xml : <pre><keyvals> <key>appCookieAttribute</key> <value>samesite=none; secure;</value> <desc>Semicolon seperated attributes to be added to all applicationcookies</desc> </keyvals></pre> This key indicates that third party cookies are allowed on a secure channel.
32668401	By default, for payment failures, the BRM refund process merges refund items into one. This has been fixed. Refund items are not merged when the payment status is available and the value is failed .
32453170	When processing writeoffs, PCM_OP_AR_BILLINFO_WRITEOFF included overpayment and unallocated credit adjustments. This has been fixed. These are no longer included.

Table 1-5 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 4

Bug	Description
32955726	<p>When processing a top-up, the BILL_DEBIT operation was making a credit limit check and the Connection Manager was throwing a ERR_CREDIT_LIMIT_EXCEEDED error. For top-ups, ECE does a credit limit check, so the check from BILL_DEBIT was a duplicate.</p> <p>This has been fixed. BILL_DEBIT no longer makes credit limit checks for top-ups from ECE.</p>
32501964	<p>If there is no THRESHOLDS field in the input flist for the PCM_OP_BILL_SET_LIMIT_AND_CR opcode, BRM adds the PIN_FLD_CREDIT_THRESHOLDS_FIXED field and calculates the value from existing values from other fields in the database, like PIN_FLD_CREDIT_LIMIT and PIN_FLD_CREDIT_FLOOR.</p> <p>Now you can use the PIN_FLD_FLAGS field in the PCM_OP_BILL_SET_LIMIT_AND_CR input flist to configure whether BRM uses the existing values from the database or sets PIN_FLD_CREDIT_THRESHOLDS_FIXED to NULL.</p> <p>To set it to NULL when no THRESHOLD field is sent, set PIN_FLD_FLAGS to 0x01. To use existing database values, do not set PIN_FLD_FLAGS.</p>
32668400	<p>In Business Operations Center, running PIN_STATE_CHANGE as a custom job fails with an incorrect parameters error. Business Operations Center did not support running PIN_STATE_CHANGE.</p> <p>This has been fixed.</p>
32298772	<p>In Customer Center, successfully run deferred collections actions were not displayed. This was because the corresponding /schedule objects were being deleted after successful completion of the pin_collections utility.</p> <p>This has been fixed. Now rather than deleting the /schedule objects, their status is updated and they are displayed in Customer Center.</p>
32484030	<p>The PCM_OP_COLLECTIONS_ADD_ACTION opcode was not performing customizations. This was because the opcode's process flow did not include the PCM_OP_COLLECTIONS_POL_CALC_DUE_DATE policy opcode.</p> <p>This has been fixed.</p>
32416993	<p>Suspend and terminate actions performed by the PIN_COLLECTIONS_PROCESS utility were failing with an ERR_NOT_FOUND error. This was because the action to close a service in a /billinfo was attempting to delete a field that was not present in the input flist.</p> <p>This has been fixed. The utility now checks that the field is present before processing the deletion.</p>
32680325	<p>Adding multiple subordinates to an account in multiple threads was taking a long time. This was because the INC_FLDS operation was running against the PIN_FLD_PARENT_FLAGS field of the parent account's /billinfo object.</p> <p>This has been fixed. The value of PIN_FLD_PARENT_FLAGS is no longer incremented with each subordinate. Now the value can be either 1 (if the account has subordinates) or 0 (if the account has no subordinates).</p>
32812435	<p>The PCM_OP_SUBSCRIPTION_PURCHASE_FEES opcode was resulting in a deadlock error. This was because item-type products were being cleaned up improperly.</p> <p>This has been fixed.</p>

Table 1-5 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 4

Bug	Description
32950646	Exporting G/L reports into file with the pin_ledger_report utility was taking a long time and causing memory issues and errors. This was because the utility was reading all reports into memory before processing. This has been fixed. The incremental report and XML file generation logic has been moved to the database, and data is fetched from the memory in steps. For example, fetch 10000 rows at a time and generate the G/L report XML file for the fetched data.
32992054	Invoice processing was failing when INVOICE_DATA stored in the /event table contained special characters like <, >, , even if a backslash was used as an escape character. This has been fixed. The PCM_OP_INV_DECODE_INVOICE_DATA opcode now escapes these characters.
31745265	When a payments input flist has a PIN_FLD_TYPE_STR field with a value of CTI , the BRM server and dm_fusa now send Card Type Indicator (CTI) / Product Card Type Indicator (PCTI) record to Paymentech or the Paymentech simulator. Paymentech and the Paymentech simulator include the same field in the output flist response.
30909463	The PIN_SEPA utility now generates XML files in accordance with the latest EPC schema versions (Rulebook v7) for sepa_dd and sepa_ct requests.
31756692	When a payments input flist has a PIN_FLD_TYPE_STR field with a value of CTI , the BRM server and dm_fusa now send Card Type Indicator (CTI) / Product Card Type Indicator (PCTI) record to Paymentech or the Paymentech simulator. Paymentech and the Paymentech simulator include the same field in the output flist response.
32915574	The PIN_RECOVER utility now passes PIN_FLD_ACH and PIN_FLD_CHANNEL_ID to payment recovery so that payment items are populated with the proper values.
32680327	When the PCM_OP_PRICE_SET_PRICE_LIST opcode was updating /data/price_list_changed objects, row lock contention was occurring. This has been fixed. The /data/price_list_changed update now happens at the end of the transaction to decrease the lock interval.
32146557	While publishing a price plan modify event, multilevel rate plan selectors are now handled properly to add rate plans during the event publication.
31582718	BRM no longer requires a CM restart when adding a new balance element ID (BEID) in a PDC-enabled or non-PDC enabled environment. The default PDC check has been removed from the /config/beid refresh process flow.
32366586	When using multiple rate plan selectors, the rerating process was failing for cycle_event . This was because by incorrect event details were being sent to fetch the rate plan required to rate the event. This has been fixed. Now only required event details are sent for rating.
31831938	When a payments input flist has a PIN_FLD_TYPE_STR field with a value of CTI , the BRM server and dm_fusa now send Card Type Indicator (CTI) / Product Card Type Indicator (PCTI) record to Paymentech or the Paymentech simulator. Paymentech and the Paymentech simulator include the same field in the output flist response.

Table 1-5 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 4

Bug	Description
32554792	The OP_RATE_GET_PRODUCT opcode was crashing with a Signal 11 error at fm_rate_cache_clean_product_data . This was because audit data was not being handled properly for some scenarios when cleaning rate plans. This has been fixed.
33030795	During rerating, multiple item_obj_id0 were created for usage events. This was because incorrect item object references were used at the event level when creating new item objects. This has been fixed.
32779665	Balances in ECE were incorrectly updated after rerating due to events published to ECE having the improper balance group revision values. This has been fixed. Now the adjustment events are published after the balance group revisions are updated.
32402233	When OP_SUBSCRIPTION_TRANSITION_PLAN is called twice within the same transaction, all products purchased within the two calls were given the same value for PIN_FLD_PACKAGE_ID field. This has been fixed. The package_id transaction cache is now refreshed at the end of plan_transition to ensure a new package_id is assigned for the next call.
32718280	The RestrictResourceValidityToOffer business parameter restricts resource validity end time to the end time of the product during first usage. This behavior has been extended to be applicable against purchased products with a set end date so that the corresponding sub_bal_validity end time for grants is set to the product end date.
32434961	After a service transfer, any operations that have rounded event times (like deferred purchase, set_proinfo, discount purchase or cancellation) failed to fetch the appropriate /balance_group . This was because the times were rounded to time values before the actual transfer time value. This has been fixed.
32381048	When canceling full discounts in previous cycles, purchase_end_t was not being set and the cancellation event was not generated. Instead, the end date was set to actg_next_t and the discount modify event was generated, as is the expected behavior for discounts being canceled in current or future cycles. This has been fixed. If discount_end is less than now_t , the full discount is canceled.
32078939	Sub-balance validity was not set while setting set_proinfo . Discount cancellation was not the using the PIN_RATE_FLG_CUT flag passed in input. This has been fixed. The CANCEL_DISCOUNT, SET_DISCOUNTINFO, and SET_DISCOUNT_STATUS opcodes and the pin_discount_cleanup utility accept the PIN_RATE_FLG_CUT and -rate_flag_cut flags and set the sub-balance bucket validity end dates.
32146555	If the dates in /au_billinfo were incorrect, PCM_OP_SUBSCRIPTION_GET_ACTG_DATES was going in an infinite loop. This has been fixed. An error now appears if the dates in /au_billinfo are wrong.

Table 1-5 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 4

Bug	Description
32950651	<p>The PCM_OP_CUST_SET_BILLINFO opcode was taking a long time to change an attribute in a parent's /billinfo. This was because a recursive call was made to all subordinate /billinfo objects, regardless of whether changes were made in the subordinate.</p> <p>This has been fixed. If there are no changed fields in the subordinates, the recursive call is skipped.</p>
32846749	<p>When discount with a set end date is purchased and backdated to previous accounting cycle after running billing, the discount validation fails and the discount is not considered for refunds.</p> <p>This has been fixed.</p>
32751700	<p>In the dm_vertex output flist, when setting the value of the PIN_FLD_RESULT field, the LocSetNameCriteria function call to Vertex STQ was returning a value in a variable that was used to take decision about pass or fail of jurisdiction validation.</p> <p>This has been fixed.</p>
31811039	<p>When an adjustment is done on a bill, its corrective bill and invoice should be generated immediately to avoid inconsistencies in the invoice. Generating a corrective bill and invoice after the next regular bill was causing the adjustment to appear in the regular bill and the corrective invoice.</p> <p>This has been fixed. Now you can choose to drop adjustment items from an invoice if they are allocated to other bills, like corrective bills, by setting the ARItemsInCorrectiveInvoice parameter in the bus_params_Invoicing.xml file. Setting this parameter will only show the adjustment item on the corrective bill.</p>
30996236	<p>Noncurrency event adjustment is not happening when the event has non-zero discount balance impact and zero amount charge balance impact.</p> <p>This has been fixed. Noncurrency discount balance impacts are now processed even if the event is not a discount-only event.</p>
31356891	<p>For most A/R action validation errors, the ebufp error buffer was not filled, but a description field was set with the error message in the return flist.</p> <p>This has been fixed. Now you can set the value of PIN_AR_SET_EBUF_ON_ERROR to 0x20 for A/R actions so that validation errors appear in the error buffer.</p>
31831937	<p>/item/payment were being dropped in next regular invoice when Accounts Receivable (AR) items in corrective invoices was enabled.</p> <p>This has been fixed. Only adjustment items allocated to other bills are dropped during corrective invoice generation.</p>
32015905	<p>Errors were being logged by the Connection Manager at PIN_ERR_LEVEL_ERROR instead of PIN_ERR_LEVEL_DEBUG.</p> <p>This has been fixed.</p>
32175575	<p>During wholesale billing, billing of parent account fails as virtual column was not handled.</p> <p>This has been fixed.</p>
32095034	<p>Memory leaks were observed in fm_bill_reverse.c and fm_pymt_item_search.c.</p> <p>This has been fixed.</p>

Table 1-5 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 4

Bug	Description
31103984	pin_collections_process was having performance issue. This was because some SQL search statements were using literals instead of bind variables. This has been fixed.
31043161	The CMT Infranet.properties file did not accept environment variables. This has been fixed.
31074575	When loading a custom XML with pin_cmt , the RefObj of type billinfo was giving an incorrect POID. This has been fixed. The proper /billinfo POID is selected from the cmt_reference_t table.
32019959	The PCM_OP_PRICE_COMMIT_DISCOUNT opcode was creating hash tables for all discounts in the database, which would take longer the more discounts were in database. This has been fixed. You can now optionally disable hashing for discounts.
32058272	If no AAC_ACCESS information was passed in the input flists, the PCM_OP_CUST_MODIFY_CUSTOMER and PCM_OP_CUST_COMMIT_CUSTOMER opcodes copied it from PIN_FLD_ACCTINFO. This has been fixed. You can now pass AAC information in the PIN_FLD_SERVICES field. This lets you map the AAC_ACCESS field to populate the extended attributes field for ProductImpl in the ECE Customer Cache. If there are no AAC fields in PIN_FLD_SERVICES, the values in PIN_FLD_ACCTINFO are used instead.
32008567	When committing discounts, a row lock contention occurred on the IFW_DISCOUNTMASTER table. This was because the set_discount_master procedure was trying update the database table even when the discount master name value in the database and input was NULL. This has been fixed.
31366932	Unlike other standard database tables, such as POID_TYPE or PROFILE BILLING_INFO_T, you could not create indexes for ACCOUNT_EXEMPTIONS_T to improve query performance. This has been fixed. You can now create indexes for ACCOUNT_EXEMPTIONS_T.
32105392	When performing a top-up for an active service with the PCM_OP_BILL_DEBIT opcode and when a service state moved from RECHARGE to ACTIVE, the service expiration date was not being extended. This has been fixed.
31791924	After upgrading BRM 7.5 PS16 to 7.5 PS23, the invoicing process was producing an error due to an issue in dm_oracle. dm_oracle was crashing at op_search_and_bulk_act due to a non-existent class being passed in the input flist. This has been fixed.
31881022	Generating reports using pin_ledger_report in export mode causing memory issues. This was because the utility was reading the detailed report into memory even if it was not being used to generate the XML report. This has been fixed. Now the utility reads the detailed report only when generating the XML report.

Table 1-5 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 4

Bug	Description
31880134	<p>Generating incremental reports using pin_ledger_report in export mode causing memory issues. This was because the utility was reading the detailed report from the database in a flist, then reading the previous report in the flist, and comparing them to generate the new incremental flist.</p> <p>This has been fixed. Now the incremental report generation logic is handled by the database.</p> <p>To implement this fix, after applying the patch, load one of the following files from the patch into the database, depending on your database character-set configuration:</p> <ul style="list-style-type: none"> • create_procedures_UTF8.plb • create_procedures_AL32UTF8.plb
31826416	<p>After upgrading from BRM 7.5 Patch Set 21 to Patch Set 23, the pin_inv_doc_gen utility for email was not updating invoice_status to 1. This has been fixed. For pin_inv_doc_gen, the OUTPUT_NAME field in the burst query now contains BILL_NO as a substring.</p>
31060274	<p>The dm_fusa data manager was crashing when the MT record in the response was not handled.</p> <p>This has been fixed. Now dm_fusa interprets MT records only if they are received. The simulator also simulates this scenario.</p>
31024364	<p>The the pin_mass_refund utility was failing with a PIN_ERR_DEADLOCK error. This was because the utility was locking balance groups with the PCM_OP_BILL_ITEM_REFUND opcode and accounts with the PCM_OP_BILL_ITEM_TRANSFER.</p> <p>This has been fixed. Now the accounts are also locked by PCM_OP_BILL_ITEM_REFUND and the balance groups are all locked in order.</p>
31948875	<p>Generic searches for /event/billing/charge/% events were failing. This was because the PIN_FLD_TRANSACTIONS field was being read in all searches, but this field only applies to /event/billing/charge/cc events, so any search results that were not /event/billing/charge/cc events failed.</p> <p>This has been fixed. Now PIN_FLD_TRANSACTIONS is read for /event/billing/charge/cc events only.</p>
31143570	<p>When the CM_OP_PYMT_COLLECT opcode was called by a custom wrapper opcode to create a customer, authorization was failing with a Signal 11 error, with a possible reason as a memory leak. The Connection Manager for fm_pymt_pol_apply_fee_parse_customer_segment was crashing.</p> <p>This has been fixed.</p>
31634436	<p>While creating a customer with a SEPA payment mode, IBAN validation was failing even for valid IBAN values.</p> <p>This has been fixed. IBAN validation is now performed according to ISO standards.</p>
31757793	<p>For SEPA payment types, banks were rejecting SDD files for transactions with zero charges.</p> <p>This has been fixed. Now SDD requests are suppressed when the amount is zero.</p>
32146559	<p>When committing discounts, if the discount trigger and master were reused, row lock occurred on the IFW_DISCOUNTMASTER table.</p> <p>This has been fixed.</p>

Table 1-5 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 4

Bug	Description
32164299	When committing discounts with the PCM_OP_PRICE_COMMIT_DISCOUNT opcode, row lock contention was occurring on the DATA_T table. This was because the commit was occurring during long-running open transactions. This has been fixed. Now the commit is deferred until the transaction is committed.
30944081	A sponsored account was unable to rate traffic when its own sub-balance was positive. Rating was not adhering to the configured credit limit. This was because of an error in the credit limit check when charge offers had an impact on both the sponsored account and the owning account. This has been fixed. The credit limit is no longer checked on an account's sponsored balances.
31643234	The data auditing and rating processes were behaving inconsistently. This was because BRM was not publishing updated effective_t of products to ECE during the set_proinfo flow. This has been fixed.
31960547	The wrong tax was applied on discount purchases. This was because the PIN_RATE_BAL_FLG_TAXABLE_OVERRIDDEN field was being set to the wrong value, and the amount_deferred for the discount impact was always being reset to zero, regardless of the value for the field. This has been fixed.
30963854	When rerating an already rerated account after running billing, the impact_type was not populating correctly for backout events generated after rerating after running billing. This has been fixed. The discount impact type is now reset during rerating.
31456478	The balance group ID and customer ID in ECE were mismatched. This was because the cookie used while fetching each balance group was corrupt. This has been fixed.
31218440	After the second rerating for a canceled product, the arrear_cycle event for last cycle was not generated. This has been fixed. The condition is now based on product status and events are generated for all cycles.
32112279	Rerating a conditional RUM impacted event was granting the conditional RUM balance impact again, but no debit occurred for the original conditional RUM balance impact. This was because the backout event was not being created, so balances were not updated properly. This has been fixed. During rerating, a backout event is now created when the session event's type is initiate or update .
31630568	During rerating, products were not returned based on priority. This has been fixed. Products are now returned in order from highest to lowest priority.
30883875	The PCM_OP_SUBSCRIPTION_SET_PRODINFO opcode was creating triple reversals. This was because multiple refund events were being generated for same product if its end dates were set and then canceled in separate transactions. This has been fixed.

Table 1-5 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 4

Bug	Description
32212986	<p>After a service transfer, while setting CYCLE_START_T, the PCM_OP_SUBSCRIPTION_SET_PRODINFO opcode was rounding the event_end_time to midnight before the transfer or cancellation time. This caused balances to be impacted on old balance group.</p> <p>This has been fixed. The transfer time is now used for event_end_t.</p> <p>Cancellation of cycle_arrears charges with zero scale was not being performed correctly. This was because the purchase, cycle, and usage end dates were not passed to apply_rate.</p> <p>This has been fixed.</p>
31326055	<p>Deleting a schedule object by calling the PCM_OP_ACT_SCHEDULE_DELETE opcode from custom code was taking a very long time. This was because the BAL_GRP_T table and the BRM root account were being locked.</p> <p>This has been fixed. Balance group locking is now skipped for root account objects.</p>
31709680	<p>Mismatched balances and issues applying taxes were occurring in bills. This was because discounts purchased in a delay period were being tagged to next bill, even if the original product charge event was tagged to current bill.</p> <p>This has been fixed.</p>
31906427	<p>In complex backdated order scenarios, discount claw back was not working properly when setting the end date of discount.</p> <p>This has been fixed.</p>
30935307	<p>With prorated discounts, discounts were counted inaccurately for the full cycle.</p> <p>This has been fixed. The proper end time is now read for canceled discounts. In sequential cycle flows for refunds, the PCM_OP_SUBSCRIPTION_CYCLE_FORWARD opcode is called with the modified end time so that it can provide appropriate refunds.</p>
31884566	<p>Discounts were not being applied if the purchased discount was backdated to the previous billing cycle.</p> <p>This has been fixed.</p>
31384943	<p>When backdating products with multiple rate tiers, discounts were being applied twice.</p> <p>This has been fixed.</p>
31589007	<p>In Subscription Management, enrichment from custom code in PCM_OP_CUST_POL_PREP_NAMEINFO was not being made for from some fields, such as PIN_FLD_EMAIL_ADDR and PIN_FLD_ZIP. The fields were still appearing as they were in the input list of PCM_OP_CUST_CREATE_CUSTOMER, before customization.</p> <p>This has been fixed.</p>
32324208	<p>An incorrect refund was being created when purchasing a second instance of the same discount.</p> <p>This has been fixed.</p>
31461126	<p>Timestamps for services in STATE_EXPIRATION_T were not being rounded when the pin_state_change utility changed life cycle states, even when rounding was enabled.</p> <p>This has been fixed.</p>

Table 1-5 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 4

Bug	Description
31797332	Transitioning plans with the PCM_OP_SUBSCRIPTION_TRANSITION_PLAN opcode was failing and the discount end dates were not being set properly. This has been fixed.
31510306	The product and discount purchase and cancellation events sent from BRM to ECE did not contain the exact POID revision, which caused ECE to work incorrectly. This has been fixed.
32078940	During service cancellation done on the same day as a service transfer, the PCM_OP_SUBSCRIPTION_SET_PRODINFO opcode was rounding end time, which caused invalid balance groups to be affected. This has been fixed. Rounding is no longer performed for cancellations.
32164301	During service cancellation done on the same day as a service transfer, the PCM_OP_SUBSCRIPTION_SET_PRODINFO, an error occurred when setting end dates for the product instance on the account receiving the transfer. This has been fixed.
32266270	Line-based taxes were being calculated incorrectly by dm_cvertex . This has been fixed.
31626614	<p>When using a tax engine other than Vertex CTQ, you can use the following / config/business_params entry for billing:</p> <pre> 0 PIN_FLD_PARAMS ARRAY [50] allocated 4, used 4 1 PIN_FLD_DESCR STR [0] "Specifies whether tax percentage should be added for zero tax amount. 0 - percentage is added for zero tax amount (default). 1 - do not add the percentage. The parameter is for cases where PIN_FLD_TAXPKG_TYPE = 0." 1 PIN_FLD_PARAM_NAME STR [0] "exclude_percent_for_zero_tax" 1 PIN_FLD_PARAM_TYPE INT [0] 1 1 PIN_FLD_PARAM_VALUE STR [0] "0" </pre> <p>When the entry is enabled, PIN_FLD_TAXPKG_TYPE is set to 0, and the tax amount is 0, percentage will set to 0 in the PIN_FLD_BAL_IMPACTS field. The default behavior is to show percentage when the tax amount is 0.</p>
31644822	During rating, BRM was using the first tax supplier on a purchase for all charge offers or products when a bundle contained multiple charge offers or a purchase contained multiple products, and the charge offers or products were assigned different tax suppliers. This has been fixed.
31757400	Zonemap values are case sensitive and we are supporting '#' as a valid character in a Zonemap string.
32246853	While performing a 100% event adjustment on a tier pricing charge event when there were multiple charge balance impacts, the discount was being applied multiple times. This has been fixed.

Table 1-5 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 4

Bug	Description
32595176	BRM was allowing a second request for full tax adjustment on an event that already had a full tax adjustment performed against it, without generating a proper error message. This generated and recorded a 0\$ adjustment event in BRM. This has been fixed.
31690657	When contributor string was passed in the input list of the PCM_OP_BILL_DEBIT opcode, a NULL value was set in the PIN_FLD_CONTRIBUTOR_STR field in the BAL_GRP_SUB_BALANCES_T table. This has been fixed. Now PCM_OP_BILL_DEBIT is checked for a contributor string, and the value of the string is used.
31168922	In Billing Care, opening a billing account with a large number of service accounts failed. This was because the data manager did not support a search query with more than 33 arguments. This has been fixed.
31859893	After performing BILL_NOW, the cycle tax was not adjusted for event level adjustments of a nonpaying child accounts. This was because the cycle tax event was generated for the parent account. This has been fixed. The BILL_NOW flow now checks the cycle tax on the parent account and considers whether the GenerateBillForChildBillnow parameter is enabled or disabled.
31445652	During product purchase, split charge packet range issues were causing errors in the discounting pipeline. This has been fixed.
32501968	After performing BILL_NOW, the cycle tax was not adjusted for event level adjustments of a nonpaying child accounts. This was because the cycle tax event was generated for the parent account. This has been fixed. The BILL_NOW flow now checks the cycle tax on the parent account and considers whether the GenerateBillForChildBillnow parameter is enabled or disabled.
32501970	After performing BILL_NOW, the cycle tax was not adjusted for event level adjustments of a nonpaying child accounts. This was because the cycle tax event was generated for the parent account. This has been fixed. The BILL_NOW flow now checks the cycle tax on the parent account and considers whether the GenerateBillForChildBillnow parameter is enabled or disabled.
31632773	The following Perl scripts were missing the use strict and use warnings Perl pragma compiler flags: <ul style="list-style-type: none"> • partition_utils • pin_bus_params • pin_close_items • pin_wSDL_generator This has been fixed.
32252778	The discounting pipeline was not starting after upgrade to BRM 12.0. This was because of a invalid index access of a string exception while processing a regular expression FSM. This has been fixed.

Table 1-5 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 4

Bug	Description
31356888	When a batch of events were submitted from Event Browser, the discounted events were being over-adjusted. This has been fixed.
31761529	The pin_deposit utility now considers existing checkpoints so that no more than one checkpoint is created.
31761429	Discounts are now considered while performing event-level accounts receivable actions.
31545961	Fixed plan transitions and their discount end dates are now set properly.
31660414	A check was added so that stored credential processing is not performed when the /payinfo object is not present.
31576857	The plan transition opcode was expecting certain fields even though they were optional. This has been fixed.
31595861	A check was added so that stored credential processing is not performed when the /payinfo object is not present.
31525930	When a charge is forced into a long cycle by the PCM_OP_SUBSCRIPTION_POL_SPEC_CYCLE_FEE_INTERVAL policy opcode, the cycle is honored by merging the cycles into a long cycle depending on the flag value in output flist of the opcode.
31525906	The PIN_FLD_NAMEINFO field was deleted from any event (except /event/customer/nameinfo) before it was recorded in the BRM database. This has been fixed.
31505709	The Connection Manager (CM) no longer encounters errors when the ConfigCacheRefreshInterval parameter is set to a non-zero value.
31494552	The PIN_FLD_NAMEINFO field returned from the PCM_OP_CUST_POL_PREP_NAMEINFO policy opcode is now correctly appended to the return flist of PCM_OP_CUST_CREATE_CUSTOMER .
31225903	A Connection Manager (CM) child process Mutex lock issue occurred during a configuration object refresh. This has been fixed.
31225878	A memory issue that occurred when generating CorrelationId in PortalContext of pcm.jar and there was an issue in the salt counter value of the CorrelationId string. These have been fixed.
31225827	The Paymentech data manager (dm_fusa) and simulator now interpret the MT record only if it is received.
31110284	The impact type is now set properly in the backout event during the rerating process.
31761680	A core dump in the fm_pymt_pol.so library file has been fixed.

Customer-Reported Fixes in BRM 12.0 Patch Set 3

[Table 1-6](#) lists the customer-reported issues that were resolved in BRM 12.0 Patch Set 3.

Table 1-6 Customer-Reported Fixes for BRM 12.0 Patch Set 3

Bug Number	Description
29548099	The BRM 12.0 SDK installation uses the source and PortalDevKit directories. The makefile expects the FM policy codes to be in PortalDevKit , but the installation actually copies them into the source directory. Because of this, makefile references were not resolving. This is now fixed. The include directory path is updated by creating a soft link as part of the core installation.
29760523	The number of CM connections in the environment was observed to be increasing to more than 40. This was because Java client applications connected to CM were idling beyond cm_timeout time, causing the CM to shut down. However, the socket connection remains in CLOSE_WAIT status. When the Java application becomes active again, it establishes a new socket connection and old socket still stays in CLOSE_WAIT even though the respective CM had shut down. This is now fixed; the CLOSE_WAIT connections are cleaned up.
29826061	GL.IS_REVENUE and GL.IS_AR functions in the GL Summary queries prevent parallelism. Including these function calls in the SQL for G/L summary causes the Oracle query optimizer to serialize execution of the queries. Changes are now made to enable parallelism for the functions IS_REVENUE and IS_AR and the query for unbilled_unearned report generation is altered to improve the performance.
30006029	In PCM_OP_SUBSCRIPTION_SET_DISCOUNTINFO and PCM_OP_SUBSCRIPTION_SET_PRODINFO, the Purchase Start Time was not getting rounded in the set_discountinfo call, even if timestamp rounding was enabled. Because of this, an error was reported if Purchase Start Time was greater than Cycle Start Time. The rounding logic is now added to fix this issue.
30063992	In a scenario where CIT/MIT is disabled but the transaction type customized, resubmitting a payment resulted in an otherwise blank PMT001 record to be sent. This caused Paymentech to respond with code 279 for the invalid record content. The resubmit scenario is now fixed and the PCM_OP_PYMT_COLLECT opcode will drop empty fields when CIT/MIT is disabled.
30072009	While validating mutually exclusive deals, in one case a function was not returning the correct value and validation was failing. This is now fixed and the function now returns the correct value.
30072015	For a plan with First Usage enabled, rerating was creating an unbilled cycle forward event. This is now fixed and appropriate flags are added in the rating call so that it will create an ADJUSTMENT/SHADOW event as applicable.
30078626	The script alter_column_dd_objects_table.source did not update some of the event subclasses in the system. This script is expected to update dd_objects_t.EVENT_TYPE with appropriate values for all events, but a few were missed. This is now fixed and missing event subclasses are added.
30093269	Billing Care was failing to fetch bills for a selected bill unit when there was a very large number of items present for a particular item subtype. This is now fixed so that the argument length for the query fetching data for the input item_sub_type will not exceed 4000 characters.

Table 1-6 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 3

Bug Number	Description
30108687	<p>If a cycle product was moved to inactive state just before the product end date was reached, perhaps because the account was suspended, the product will never be canceled on its end date because pin cycle fees - cancel was not considering products in an inactive state.</p> <p>In ECE, there is a configuration to remove expired products from ECE cache. Since the above product had an end date, it will be removed from ECE cache.</p> <p>Now when the account is activated again, BRM will synchronize all active and inactive products with ECE. The inactive cycle product will fail in sync because the product is not available in ECE cache since it had already expired and was removed.</p> <p>Now an additional option, pin_cycle_fees - cancel -inactive, is introduced to also cancel inactive products, so that expired products will not be synced to ECE upon reactivating a customer account.</p> <p>It was not possible to reactive an inactive account because the products associated to the account were also inactive and were removed from the ECE cache. This is now fixed.</p>
30120444	<p>Product-based rerating was not working for One-Time Charge products using the -p option. This is now fixed so that when purchase fee event needs to be rerated, the corresponding action purchase event is added to events that needs to be rerated.</p>
30120448	<p>There was a performance issue while processing real-time usage for certain system-level usage products. The issue is resolved by caching system products and discounts during the CM startup instead of caching every time a child is spawned. This behavior is controlled via the business parameter RatePreCacheProductAndDisc.</p>
30125969	<p>Rerating was doubling the rollover amount, because the next billing cycle data was not being correctly returned and the rollover event was not getting generated. The logic related to getting the next billing cycle data during rerating was modified so that rollover event will now be generated.</p>
30133930	<p>Partial billing was failing for a few accounts that previously belonged to a discount sharing group, but were no longer part of the sharing group. During billing, the old discount sharing groups were being fetched. This is now fixed. The unwanted error is skipped and the behavior is changed to exit gracefully when BRM fetches an expired discount sharing group during billing, resulting in a failure.</p>
30133932	<p>Purchasing a discount was failing after enabling the sequential_cycle_discounting business parameter. This was happening because the calc_only rating was failing while fetching the rate plan if the cycle beginning was before product commit date. This is now fixed.</p>
30146962	<p>The bill adjustment amount was not allocated properly if any item in the bill had real-time tax and BILL_DUE or that ITEM_DUE is zero. The AR flow is now modified so that it calculates the tax of the item or bill only if the item or bill has some amount due, and accordingly picking that item for adjustment.</p>
30158957	<p>If one transaction failed when running pin_deposit, subsequent transactions were also not processing properly. This is now fixed. The pin_deposit flow is modified to clear the error buffer before processing subsequent transactions so that the program does not exit abruptly.</p>
30187912	<p>After the migration of XSLT transformation from Xalan to Oracle XDK-XSLT library, date formatting via Java's overloaded method was not working. As a result, the emailed invoice was unable to display the date/time in the user's expected format. New static methods are now added to format the date in customer's expected format using Java's SimpleDateFormat specification.</p>

Table 1-6 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 3

Bug Number	Description
30187917	Extra tax amount was applied in case of event adjustment when the cancel event was sent in the input flist. The calculated tax amount was not rounded properly. This is now fixed to get the correct tax breakdown and the corresponding tax balance impact in the adjustment event.
30190209	The Rated Event Loader (RE Loader) process stopped abruptly with an out of memory issue. This was because of RE Loader daemon native memory growth, caused by a memory leak in JNI call. This is now fixed.
30205140	These business operations procedures were missing in the BRM 12.0 package and are now included: create_analytics_procedures_UTF8.plb and create_analytics_procedures_AI32UTF8.plb .
30211263	While using the load_config_item_types utility to import config_item_types.xml , item descriptions with special characters (like "ç", "ã") were not loaded into the database. This is now fixed and load_config_item_types now accepts European and special characters with UTF-8 encoding in the input XML file.
30228177	Messages were not committed to the DB when the flag to indicate that the external transaction is opened was set, but there was no external transaction and the publishing DM was waiting for TRANS_COMMIT which did not arrive because no external transaction was open. This has been fixed by modifying the code to properly identify if the PUBLISH_GEN_PAYLOAD is called with or without an external transaction.
30228179	Memory leaks were observed while deleting accounts with pin_del_closed_accts . This is now fixed.
30233959	Paymentech integration (dm_fusa) is now enhanced to support real-time update and China UnionPay cards.
30249952	By setting log_refresh_product and refresh_product_interval in the CM pin.conf , it is expected that the product cache would refresh regardless of the log level set. But the product cache was not refreshing for any log level. This is now fixed.
30249962	When the SOAP-based web services are invoked with invalid content in the XML input, it was observed that they print the XML parsing error message in the log file but continue further and call the opcode. This is now fixed. Change validations are now included to log the error and stop proceeding further.
30252742	Even when the CM received PIN_ERR_TIMEOUT during its startup, the CM continued to start. Appropriate checks are introduced before clearing the ebuf, so that any error occurred is stored and the CM does not start if PIN_ERR_TIMEOUT occurs.
30258068	A bill adjustment on a bill item containing a settled dispute is not working. This was because the settlement item transfer was fetching wrong session objects. This is now fixed.
30276798	While adjusting an event in a billinfo associated with a service-level balance group, the adjustment was getting created and the item being mapped under the default account level balance group. This impacted all AR transactions. This is now fixed. The balance group associated with the event is now being used.
30276799	To migrate /group/sharing/discounts data through the input XML file, the CMT XSD did not have option to provide the offering object for each discount separately. It had an option to specify offering_obj at the group level, so all the discounts under the discount sharing group would have same offering object. The CMT XSD is now modified to support offering objects per discount as well.

Table 1-6 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 3

Bug Number	Description
30287778	While processing refunds for cycle fees, the cycle fee opcode is called with modified end_t so it can provide appropriate refunds. This modified end_t skipped discounts in identify_discounts if they were canceled. However, this caused skipping of discounts even if it was applicable for the period. This is now fixed so that any discounts will be applied.
30287781	Charges were going to the wrong bill if there was a cancellation after running Bill Now in the delay period. Bill Now bills /item and changes its status. Then the cancellation charges were assigned to same /item even if it was closed, which caused a loss of revenue. This is now fixed and /item is considered only if it is in PENDING state.
30292968	pin_discount_cleanup was not considering inactive discounts for cleanup. This is now fixed to consider inactive discounts as well using the mode close_inactive .
30292978	You can now generate a notification event when a balance is transferred between two different balance groups using the opcode PCM_OP_BILL_ITEM_TRANSFER. This is now included when you use the /event/notification/billing/item/transfer notification event.
30327229	When an event adjustment is performed after a tax-only adjustment on the same event, some of the fields in the tax jurisdiction array were not correct. The issue happens because tax processing logic was exiting after processing the first tax jurisdiction array. The tax processing logic is now modified to consider all the taxes in the tax jurisdiction array present in the event.
30344746	The cycle fee event end date did not align to the next accounting cycle start date. There is a new optional registry entry GrantTimeZoneAlignmentControl which is now added to the FCT_Discount plugin. When GrantTimeZoneAlignmentControl registry entry is set to TRUE and the end date is configured as Relative to billing/account cycle for discount balance impact Resource Validity Period configuration, the end date will be aligned to corresponding accounting/billing cycle date.
30382891	Paymentech Real-Time Account Updater (RAU) support in dm_fusa was generating unwanted debug messages. This is now fixed and such messages are now removed.
30385891	The pin_rerate utility was behaving inconsistently for multiple scenarios. The cycle date calculation was wrong when rerating a quarterly cycle events. Fixed the issue while aligning cycle event dates.
30392335	When there are two cycle events for the same product/period due to a discount purchase, the rerated cycle event was not able to match the previous cycle event. As a result, the rerated cycle event was not populated with session_obj . This is now handled correctly.
30400747	A purchase fee was applied twice, even when the purchase was done only once. This has been fixed.
30406785	Real-time rerating was failing for a charge offer when the PIN_FLD_RATING_PRODUCTS information was truncated due to a string length limit. This has been fixed by increasing the string length.
30406793	During billing, incorrect products were selected for applying folds. This has been fixed.
30411712	The Account Synchronization DM (dm_ifw_sync) was not supporting timeout for its database connection. This has now been added.

Table 1-6 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 3

Bug Number	Description
30418717	<p>The condition expression for evaluating discounts was a 15-digit decimal by default. The condition evaluation was incorrect when this limit was exceeded.</p> <p>This has been fixed. You can now optionally set the limit to 28 digits with the new MAX28DIGIT_DECIMAL_SUPPORT environment variable before the Pipeline Manager is started.</p>
30457056 30661357	<p>During service transfer from one account to another, the bill unit of the to-account was wrongly updated as self-paying.</p> <p>This has been fixed.</p>
30457057	<p>The Conversion Manager was reported to have performance issues in a deployment that includes Elastic Charging Engine.</p> <p>This has been fixed.</p>
30457059	<p>The Rated Event Loader daemon was not shutting down gracefully while processing rated events from ECE. In addition, when restarting, the daemon was not picking up pre-existing *.bc files.</p> <p>This has been fixed.</p>
30462364	<p>The PCM_OP_SUBSCRIPTION_SET_PRODINFO opcode called PCM_OP_SUBSCRIPTION_FEES even when PIN_FLD_PURCHASE_START_T was not passed. The calc_cycle_from_cycle_start_t entry in the CM configuration file affected the generation of the cycle arrears event.</p> <p>This has been fixed.</p>
30462367	<p>When you created two discount sharing groups, one with the current datedEFFECTIVE_T and the other for a backdated EFFECTIVE_T, the backdated discount sharing group was not considered for valid backdated usage.</p> <p>This has been fixed.</p>
30499867	<p>During rerating, the event search workflow was not considering the configured event_start_timestamp parameter in the pin_event_extract.cfg file, but instead it was taking the event END_T.</p> <p>This has been fixed.</p>
30505484	<p>The Paymentech Data Manager was sending empty PMT001 records to Paymentech when Customer/Merchant Initiated Transactions (CIT/MIT) were disabled.</p> <p>This has been fixed.</p>
30510695	<p>Backdate validation failed when timestamp rounding was disabled and a full discount was purchased. The full discount rounded the discount's START_T to ACTG_LAST_T. When the discount's START_T was earlier than the timestamp of the last status change, validation failed.</p> <p>This has been fixed.</p>
30516311	<p>For an account, after a payment was allocated to cycle tax item of a bill, any bill level adjustment on the remaining amount was unsuccessful with an incorrect calculation of amount due.</p> <p>This has been fixed, to calculate the remaining cycle tax portion correctly soon after the payment allocation, so that during adjustment the remaining charge is calculated correctly.</p>
30521545	<p>The Paymentech Data Manager (dm_fusa) back end was reaching 100% CPU utilization when a verification request was sent and not returning from the call.</p> <p>This has been fixed.</p>

Table 1-6 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 3

Bug Number	Description
30521547	The Discover Card BIN ranges were not in line with the official Discover Card ranges. This has been fixed.
30521553	BRM now supports extended ranges for newly added cards for China UnionPay BIN.
30537768	There were issues reported by Fortify scans in the following SDK source code: fm_cust_pol_valid_all.c , sample_who.c , and BRMAdapterServletClient.java . This has been fixed.
30543119	The advanced discounting pipeline was returning PIN_ERR_BAD_VALUE when it was called from the pin_cycle_fees utility. This has been fixed.
30548436	A discount of 100% on a noncurrency charge failed when multiple charge packets of multiple noncurrency resources had both positive and negative charges. This has been fixed by properly segregating the charges for each resource and calculating the total for each resource to apply discounts.
30548438	Paymentech Manager has been updated to support the real-time account updater and China UnionPay card BIN ranges.
30580542	The account deletion opcode was not deleting an account in ECE, if there were open sessions for the account. This has been fixed, allowing forced deletion of accounts in ECE, even if open sessions are found.
30588746 30684160	For an account, after a backdated product activation was done, the cycle events were assigned to incorrect items. This has been fixed.
30612469	With thirty day proration enabled, and sequential discounting disabled, there were cases of incorrectly calculated discounts for the month of February having less than 30 days. This has been fixed.
30617487	Rerating was not considering the actual start and end dates of discounts to compare with the period for which the rerating was being done, leading to incorrect charges. This has been fixed.
30623602	Support for the legacy AES encryption scheme (&aes ... format data) was removed in BRM 12.0 release. It has now been added.
30633364	Conversion Manager was not able to do balance migration due to issues with validity dates. This has been fixed, by using the right validity dates from the balance buckets.
30661356	During rerating, a one time charge was not backed out for item-type products with delayed activation. This has been fixed. Also, BRM will no longer publish item type product and discount details to ECE with the ReratingComplete event.
30691495	The Paymentech simulator was returning a non-empty CIT/MIT transaction ID for MasterCard.
30700120	Discount validity rules were not getting set properly if the end details were given during discount purchase itself and purchase and end timestamp fell on different accounting cycles. This has been fixed, by setting discount validity rules correctly.
30702208	For some opcodes, specifications in XML and XSD were missing. These have been added.

Table 1-6 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 3

Bug Number	Description
30710324	When a cycle product was inactivated and then activated during the same accounting cycle, the proration flags were not set correctly, resulting in incorrect cycle charges. This has been fixed.
30756024	When bill-time discounting used current accounting dates, the rerating process was calculating discounts incorrectly for older cycles. This has been fixed.
30760716	BRM was not allowing amount-based tax only adjustments. The product has been enhanced to support amount-based tax only adjustments for both credit and debit amounts.
30761676	The offering object information was not passed into the PCM_OP_SUBSCRIPTION_POL_SPEC_CANCEL policy opcode, which was a limitation for extending the functionality around subscriptions. With this fix, PIN_FLD_OFFERING_OBJ is added to the input of the policy opcode.
30765776	With the sequential discounting enabled, there were a few issues: 1. The discount purchased on a modified product, canceled mid-cycle after purchase for full cycle, was applying the discount for full cycle. 2. The refund and reapply operations were going wrong if there were non-overlapping discounts in an accounting cycle. These have been fixed.
30806854	While rerating the recurring charge events for canceled products, the rerate object link was missing in the adjustment event. This caused issues during invoicing, as there was no link between the backed-out event and the rerated event. This has been fixed so that link is established between the backed-out and new event.
30806855	In the rerating flow, the offering object was not updated for a one time charging product with delayed activation, resulting in incorrect results. This has been fixed.

Customer-Reported Fixes in BRM 12.0 Patch Set 2

[Table 1-7](#) lists the customer-reported issues that were resolved in BRM 12.0 Patch Set 2.

Table 1-7 Customer-Reported Fixes for BRM 12.0 Patch Set 2

Bug Number	Description
26618794	In order to avoid duplicate tax calculation and incorrect tax adjustments, the is_tax_calculated flag was introduced earlier for pipeline rated events. But this caused an issue in the tax calculation during the event adjustment, when the event included had more than one tax balance impact. This has been fixed.
27123464	It was not possible to set the promise-to-pay date as the system date. This has been fixed. Now, additional validations are added to ensure that the promise-to-pay date is greater than the system date.
27632734	Paymentech Data Manager (DM) now supports Customer-Initiated Transaction (CIT) and Merchant-Initiated Transaction (MIT) for processing VISA credit cards. See "Support for Stored-Credential Transactions for Payments" .

Table 1-7 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 2

Bug Number	Description
27993355	The POID_ID required an increase in the number of bits from its current value of 44 bits because the allocated range of Portal object IDs (POID) were not sufficient. This has been now increased to 51 bits.
28342613	A payment of zero amount was failing in BRM with an error because there was PIN_FLD_PAYMENT value was not set in the response. This has been fixed.
28387497	BRM was not providing flexibility to set the transaction type for payments processed through Paymentech DM in the PCM_OP_PYMT_POL_PRE_COLLECT policy opcode. This has been fixed.
28444532	Oracle Data Manager was incorrectly logging an error message when insufficient back ends were configured. This has been fixed and now a warning message is displayed.
28466602	Indexes were rebuilt every time the audit tables were purged. This has been fixed. The purge_audit_tables.pl script has been modified and the indexes are not rebuilt if the \$rebuild_index entry in the <i>BRM_home/sys/archive/oracle/purge_audit_tables.conf</i> file is set to no , where <i>BRM_home</i> is the directory in which you installed BRM components. If this entry is not set or set to any other value, the indexes are rebuilt every time the audit tables are purged.
28490340	During the billing process, the end date in a cycle arrear event was set incorrectly when the product end date was the same as the actg_next_t value. This caused the resources to have incorrect validity dates. This has been fixed.
28521329	Paymentech DM now supports CIT and MIT for processing Mastercard, Japan Credit Bureau (JCB), and Diners Club credit cards. This has been fixed.
28543518	During subscription transfer, Adaptec AdvancedRAID Controller (AAC) details (for example, PIN_FLD_AAC_PROMO_CODE) of the service from the source account were not copied to the new service created at the destination account. This has been fixed.
28549500	The pin_recover utility was failing while searching for payments to be recovered. This has been fixed.
28552543	Web Services Manager was always sending the date-time stamp information in Universal Time Coordinated (UTC) as part of the response without considering the actual timezone. This has been fixed to send the date-time stamp as per the time zone configuration in the <i>BRM_home/deploy/web_services/Infranet.properties</i> file.
28556710	The rerate process did not update the original event. This has been fixed and some additional criteria has been added to identify and update the corresponding original event during rerating.
28588801	During the sequential discounting, non-applicable events were also considered while calculating the periods. This has been fixed.
28596194	The fm_pymt module was not sending transaction type to Paymentech DM. This has been fixed. Now, transaction type is used by Paymentech DM to identify the type of payment, for example, recurring, installment, and so on.

Table 1-7 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 2

Bug Number	Description
28603124	The scripts in <i>BRM_home/bin</i> was using two different versions of HTTPClient libraries, 4.4 and 4.5.3. This has been fixed to use only 4.5.3 version.
28607064	In a BRM deployment with Oracle Communications Elastic Charging Engine (ECE) as charging engine, when the BRM Oracle Data Manager commit failed, the BRM data was not in synchronization with the ECE cache data. This has been fixed by triggering a commit to ECE only after the BRM Oracle Data Manager commit is successful.
28631061	In the subscription opcode PCM_OP_SUBSCRIPTION_SET_PRODINFO, a memory leak was observed. This has been fixed.
28638444	In the subscription module, during the product cancellation, an incorrect scale for refund was used. This has been fixed.
28671913	The EVENT_POID_LIST required to store longer strings than it was allowed (4000 characters) in the database. This has been fixed.
28678759	A memory leak was observed in flist to string conversion routine. This has been fixed.
28689682	During rerating, the impact type for PIN_IMPACT_TYPE_TAX_RERATED was not being set to PIN_IMPACT_TYPE_TAX, and this was leading to incorrect tax amounts in the invoices. This has been fixed.
28700838	Backdated adjustments were not displayed in the corrective invoices. This has been fixed.
28774361	The PaymentDeallocation feature allows bill adjustment on a zero due bill or a partially paid bill. The same behavior has been now implemented for item and event-level adjustments. If BillPaymentDeallocation is enabled, do the following to allow the item and event-level adjustments: <ol style="list-style-type: none"> 1. Unallocate the payment applied on the corresponding bill item. 2. Do the item or event-level adjustment. 3. Reallocate the payment back to the bill item. If the payment amount is more than the due amount after the adjustment is done, then some unallocated amount is stored in /item/payment . This unallocated amount is displayed in the Adjustment/Payments not applied tab of Customer Center.
28774366	If an error is thrown in ECE, the error was always set to PIN_ERR_BAD_VALUE. Hence, it was not possible to identify the actual error details. This has been fixed. In case the output flist of the PUBLISH_ECE_EVENT operation contains ERROR_DESCR set to Credit Ceiling Breached , the ebuf is set to PIN_ERR_CREDIT_LIMIT_BREACHED instead of PIN_ERR_BAD_VALUE.
28780635	There was an issue of wrong selection of item type when dealing with in-advance billing events. /item/adjustment was created instead of /ite/mcycle_forward_arrear for a cycle_forward_arrear event when billing in-advance was set. This has been fixed.

Table 1-7 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 2

Bug Number	Description
28786829	The PCM_OP_BILL_MAKE_BILL_NOW opcode was resetting the error buffer set by a policy customization in the workflow. This has been fixed.
28794484	In the subscription module, if END_T passed was earlier than the cycle/purchase/update end dates, then the cycle start and end dates were set incorrectly leading to validation error. This has been fixed.
28812794	The Conversion Manager failed to load data into uniqueness table due to the presence of double quotes. This has been fixed to parse and omit double quotes in the uniqueness-related functions.
28812798	The PCM_OP_AR_BILL_ADJUSTMENT opcode had memory leaks. This has been fixed.
28824835	The Connection Manager module had some memory leaks. This has been fixed.
28837038	In the subscription module, for a backdated purchase done on the billing day of the month, the end date for the event was incorrectly set and this was causing wrong validity periods for the resources both in BRM and Pipeline Manager. This has been fixed.
28843066	The Load Price List utility was failing with core dump while committing or retrieving price list. This has been fixed.
28848746	The Billing Care user interface was not displaying the details of a bill that was generated before the account was moved into a hierarchy. This has been fixed.
28848748	The transaction opening workflow was not returning an error, while trying to open a transaction on a non-existent object. This has been fixed with proper checks for existence of an object before the object is tried for a lock to open the transaction on it.
28871778	The file name format of the General Ledger (G/L) reports generated in XML were different. This has been fixed to restore the earlier naming format.
28873152	Any auditable updates to the account creates an entry in the audit table for au_account_t . In addition, for a transaction that requires data from the audit table, searching in au_account_t was taking considerable time. In such cases, the updates had to fetch effective_t from account_t if au_account_t returned null. This has been fixed.
28905266	When an account is deleted using a wrapper opcode that invokes PCM_OP_CUST_DELETE_ACCT opcode, an error in the workflow did not roll back the changes to /uniqueness object. This has been fixed.
28923379	The PCM_OP_BILL_MAKE_BILL opcode finalized the bill when the workflow was invoked by a process other than pin_bill_accts indicated by the program name field. This has been fixed to finalize the bill only through the pin_bill_accts utility.
28944041	For PCI compliance, it was required to mask the PIN_FLD_SECURITY_ID field in the card attributes structure for enhanced card security. See " Card Security Code Is Now Masked in Logs ".

Table 1-7 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 2

Bug Number	Description
28944043	Paymentech DM has been enhanced to support Card Security Presence and to send Fraud Format Indicator (FR) record when this field is present. See " Additional Card Security Presence Values Supported for Card Validation or Authorization ".
28944047	Paymentech returned duplicate CIT and MIT transaction IDs. This has been fixed and transaction IDs for CIT and MIT records are generated in a random manner.
28977386	Oracle Data Manager was not responding for a long time when PIN_ERR_NO_MEM was returned from the flist copying method. This has been fixed.
28997123	In the subscription module, when multiple discounts were purchased with different validity dates in the same deal, the discounts rules were not set correctly for the individual discounts. The discount validity dates of the first discount was used for all other discounts in the deal. This has been fixed.
29017375	The Connection Manager was dumping core when running pin_deferred_act and pin_bill_day utilities due to memory management issues. This has been fixed.
29023518	Due to the scope of the header file, the customer management module was throwing a symbol look up error. This has been fixed.
29036175	In the subscription module, during the product cancellation operation, the refund amount was calculated incorrectly if it includes in-advance charges. This was due to considering event end date instead of cycle end date for calculating the scale for refund. This has been fixed.
29042246	During the billing delay period for a deployment with the billing delay configuration, any discount cancellation operation was not applying refunds. Only during the delay period this inconsistency was observed as the accounting dates available in the bill unit would not be updated and still reflect the previous cycle information. This has been fixed with proper checks to handle such events in the delay period.
29042248	A memory leak was observed in the customer registration module. This has been fixed.
29042250	In some cases, PCM_OP_EXEC_PROC was not returning the correct response. This has been fixed.
29120662	A memory leak was observed in the Paymentech DM. This has been fixed.
29120695	When creating /payinfo/cc objects, any custom fields passing into the workflow were not dropped, which caused a schema error. This has been fixed.
29127140	In the subscription module, the PCM_OP_SUBSCRIPTION_CANCEL_DEAL opcode caused segmentation fault core dump while processing discounts. This has been fixed.
29141791	When rerating was done before billing, the invoice data received from ECE was not present in the rerated event. Instead, original invoice data was populated in the rerated event. This has been fixed.

Table 1-7 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 2

Bug Number	Description
29146203	The modify customer operation was not applying the cycle discount information passed in the input due to missing Portal object ID (POID) information in PIN_FLD_DEAL_INFO. This has been fixed.
29151194	In the subscription module, canceling a purchased product with the cycle start date in the future and the purchase and usage start dates in the past, updated the cycle, purchase, and usage start dates. This has been fixed to update only the cycle start date.
29151195	For a purchased product, if either purchase, usage, or cycle end date was changed, the end date for purchase, usage, and cycle were updated. This has been fixed to update only the date that is changed.
29174549	In Collections Manager, the PCM_OP_COLLECTIONS_GROUP_CREATE opcode had memory leaks. This has been fixed.
29178997	Event adjustment was not working as expected if the amount or percentage was not provided in the input. The expectation was to do 100% event adjustment if the amount or percentage value was not provided. This has been fixed.
29204382	In an Application Integration Architecture (AIA) deployed environment, during any transaction where the request is sent from BRM to ECE, if an error occurred in ECE, then AIA was not able to capture the error from BRM. This has been fixed.
29221143	The discount validity rules were not set for the discounts starting or ending in a future cycle as the accounting dates were prone to change. This has been fixed to identify the discounts during billing that have start or end dates in the next billing cycle and set the validity rules appropriately.
29265113	In case of a discount sharing scenario, when multiple reauthorization requests are created, the reservation quantity was not updated and it stayed at sum of the quantity of the first authorization and last reauthorization. In this case, it was possible to reauthorize sessions indefinitely and certain limits were also not enforced. This was leading to revenue leakage. This has been fixed.
29278528	In the subscription module, the plan transition opcode resulted in segmentation fault dumping core. This has been fixed.
29342746	While rating with sponsored product, the credit limit check was done both on the sponsored product and the user product. Due to this, a credit limit breach was reported when the user product sub-balance was positive. This has been fixed by removing the credit limit check on the user product while rating with the sponsored product.
29342749	CPU usage for the Connection Manager module reached and stayed at 100%. This has been fixed.
29370866	In the subscription module, the PCM_OP_SUBSCRIPTION_SET_DISCOUNTINFO opcode failed for an account-level discount. This has been fixed.
29376633	In the Pipeline Manager Framework, a memory leak was reported while converting decimal to string. This has been fixed.

Table 1-7 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 2

Bug Number	Description
29395724	In a Service Lifecycle Management (SLM)-enabled deployment, when a deal was purchased with balance as zero, the service state was immediately set to recharge_only . This caused usage processing issues in ECE. This has been fixed. Now, status is not changed to recharge_only as soon as the purchase is completed.
29409245	In the subscription module, during the product cancellation operation, the validity end dates for noncurrency resources of an in-advance cycle was wrongly set. This has been fixed.
29428268	During the pin_collect process for payments, an error indicating PIN_ERR_PARTIAL was thrown, especially with CIT and MIT transactions. This has been fixed.
29441735	During the billing process, few items were incorrectly filtered during final billing, and it was due to item pruning in cases of billing suppression, suspension, or resume billing. This has been fixed.
29441740	When a back-dated account with a multi-month cycle was moved into a hierarchy, the child account's PIN_FLD_ACTG_NEXT_T did not align with that of the parent account. This has been fixed. Changes have been made to ensure that the dates are aligned.
29448150	Event adjustment was not working properly for tax-only adjustments. This has been fixed.
29448154	The PCM_OP_COLLECTIONS_GET_AGENTS_ACTIONS output flist did not return the PIN_FLD_COUNT field. As a result, the custom logic failed. This has been fixed.
29448156	In the subscription module, the product cancellation operation did not report missing validity information. This has been fixed.
29454752	Non-CIT and MIT were failing in an attempt to access the unavailable data. This has been fixed with appropriate null checks and error handling.
29460620	In the subscription module, incorrect discount amount was calculated when a discount was applied on a back-dated cycle forward monthly charge. This has been fixed.
29463336	In the subscription module, the pin_cycle_fees utility was not selecting the tailor made products when run in the Cancel mode. This has been fixed.
29480201	The AAA Services Framework, PCM_OP_TCF_AAA_AUTHORIZE opcode, had a memory leak. This has been fixed.
29496597	When a discount was purchased in deferred mode, and it was canceled subsequently before the actual start of the discount, the cycle, purchase, and usage start dates were not updated to align with the end dates. This has been fixed.
29514020	During billing, price-overridden products were not considered properly for rating. This has been fixed.
29514025	For a discount-sharing group, when balances were concurrently updated using the multithreaded application (MTA) framework, the balance updates were not made correctly obtaining necessary lock to the balance group. This has been fixed.

Table 1-7 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 2

Bug Number	Description
29526005	It was not possible to run multiple Rated Event (RE) Loader processes in parallel. When multiple RE Loader processes were run in parallel, the row lock contentions rendered the processes non-responsive. This has been fixed.
29533564	The PCM_OP_PYMT_COLLECT opcode was not working for transaction ID with a length more than 16 characters. This has been increased to 22 characters to be consistent with the Paymentech (fusa) specifications.
29533569	Wrong quantity for the event was picked up for rating. The event matching logic was considering the events based on the order in which they appeared. This has been changed to the longest match event and now that is considered first, overriding the order of events.
29533572	Tax percentage was not added even if the tax calculation was not using any third-party tax engines. This has been fixed. Changes are done to check tax_pkg_type for zero-tax amount cases.
29557884	In Oracle Communications Billing Care, it was not possible to track the user making any changes or updates in the News Feed section. This has been fixed. The external user data is persisted as PIN_FLD_EXTERNAL_USER in the /newsfeed storable class.
29557888	For an offer, if an usage event was rated that had 0 (zero) charge and then the offer was changed and rerating was 1 (one), there was no adjustment event created nor a new usage event was created with 0 charge. This has been fixed. Now, a new event with 0 charge is created and the general ledger ID (G/L ID) corresponding to the offer is impacted.
29584658	If the price of an one-time charge product is changed post billing and rerating and corrective invoicing is done, the rerated charge was not reflecting in the invoice. This has been fixed.
29590523	The RE Loader was failing with the error "too many open files" and required an update to handle maximum number of files. This has been fixed.
29596720	The pin_deferred_act utility was generating an error message in certain cases while dropping array elements, such as PIN_FLD_TELCO_FEATURES. This has been fixed.
29622932	A bill was not including the charges during service activation from deferred actions on the bill date. A new business parameter, BillDeferredActions , has been introduced in the Billing instance of the /config/business_params object. By default, the BillDeferredActions parameter is disabled and the charges are added to the next bill. If you set the BillDeferredActions parameter to enabled , the charges are included in the current month bill.
29634985	Even when a migration failed, the Conversion Manager was reporting it as success. This has been fixed.

Table 1-7 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 2

Bug Number	Description
29673013	<p>It was not possible to synchronize the account data loaded into BRM with ECE by using the Conversion Manager. By default, the BRM data is published to ECE asynchronously, which could potentially lead to inconsistent data between BRM and ECE, if there are any synchronization failures.</p> <p>Also, it was not possible to migrate the hierarchical accounts into BRM if the parent and child accounts are in the same files and the parent accounts were not loaded first.</p> <p>This has been fixed. You can now perform the following:</p> <ul style="list-style-type: none"> • Loading the migrated account data in BRM into ECE synchronously (in real-time). • Migrate parent and child accounts into BRM using the same input file.
29681830	<p>In Billing Care, it took longer time to display the Account page. It was due to the PCM_OP_AR_GET_ACCT_ACTION_ITEMS opcode, delaying loading of accounts to the Account page.</p> <p>This has been fixed.</p>
29714064	<p>When a product was canceled, refund was not going through if the product was purchased on Day 28 through Day 31 of a month with 31-day billing enabled and the next billing date did not exist in the subsequent month.</p> <p>This has been fixed.</p>
29795746	<p>BRM triggered ECE_POST_COMMIT in ECE even when the transaction was cancelled.</p> <p>This has been fixed.</p>
29808545	<p>Any operation that included the PROFILE_SUBSCRIBER_PREFS_T table was causing a full table scan.</p> <p>This has been fixed by introducing an index on the table.</p>
29820123	<p>For the pin_mass_refund utility, performance issues were observed.</p> <p>This has been fixed.</p>
29858023	<p>In the plan transition flow, a new notification event /event/notification/plan/transition/post_deal_cancel was required to be sent soon after cancellation.</p> <p>This has been added.</p>
29867410	<p>During the cancellation of a cycle-forward event, the subscription module did not fetch all the related events for reversal of charges.</p> <p>This has been fixed.</p>
29871639	<p>In the discounting module of Pipeline Manager, after unloading existing configuration, a memory leak was observed while loading the new configuration, which was causing the growth of the used memory.</p> <p>This has been fixed.</p>
29900322	<p>In the adjustment workflows, the decimal rounding precision was different from the configured precision for usage events.</p> <p>This has been fixed.</p>
29920007	<p>In Conversion Manager, data errors were reported while migrating balances. These errors were due to unreleased database connections from the previous tasks.</p> <p>This has been fixed.</p>
29937126	<p>During the billing process, it was observed that the next item POID list of the service object was corrupted.</p> <p>This has been fixed.</p>
29960514	<p>While running the pin_del_closed_accts utility, NO_MEM error occurred. In such cases, memory was not cleared.</p> <p>This has been fixed.</p>

Table 1-7 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 2

Bug Number	Description
29966023	There were memory management issues in the Oracle Data Manager when search operation was performed using the PCM_OP_BILL_ITEM_ADJUSTMENT opcode. This was because the search returned a huge number of events crossing the high watermark for the Data Manager process. This search template has been fixed to retrieve the calculated sum from the database instead of retrieving huge number of records and calculating the sum.
29966026	The Paymentech DM was not supporting Zero Value Account Verification (ZVAV) for American Express cards. This has been fixed.
30031382	The RE Loader Daemon could not process certain files resulting in unprocessed files pending in the input directory. This has been fixed.
30095369	The RE Loader process witnessed an unaccounted growth in memory due to memory leak. This has been fixed.
30156863	The create_analytics_procedures_UTF8.plb is now included.
28653606 29428272	The Daylight Saving Time calculation was incorrect for Brazil. This has been fixed.
28923384 28975424	The RE Loader process was not gracefully shutdown, when it was still processing files. This has been fixed. Now, the RE Loader continues to process the pending files and then shuts down. A new configuration entry, batch.shutdown.interval , is introduced in the <i>BRM_home/apps/pin_rel/Infranet.properties</i> file to set the shutdown interval in seconds. By default, this entry is set to 2.

Customer-Reported Fixes in BRM 12.0 Patch Set 1

[Table 1-8](#) lists the customer-reported issues that were resolved in BRM 12.0 Patch Set 1.

Table 1-8 Customer-Reported Fixes for BRM 12.0 Patch Set 1

Bug Number	Description
26810954	Web Services Manager was not fully compliant to SOAP 1.2 format. This has been fixed.
26810956 27241793	When the tiered taxation was applied, the tax information was consolidated instead of keeping it separate for each tier. This has been fixed.
26913721	After a service transfer from one account to another using the PCM_OP_SUBSCRIPTION_SERVICE_BALGRP_TRANSFER opcode, the UNIQUENESS_T table was not updated with the correct account POID causing a mismatch with the correct account POID in the SERVICE_T table. This has been fixed.
26954698	For a quarterly billing cycle, the accounting dates were calculated incorrectly and this resulted in incorrect accounting periods. This has been fixed.

Table 1-8 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 1

Bug Number	Description
26961858	Refunding a fixed discount was incorrectly causing the refund of extra charges or other charges. This has been fixed.
26968649	Conversion Manager was throwing an error when the input XML contains a comma (,) in it. This has been fixed.
26980059	During product cancellation, while setting the product's charged period dates based on proration settings, the in-advance bill period was also being considered and this was causing inconsistencies. This has been fixed.
26990857 27577925	Authorization was incorrectly granting more than the available shared resource. This has been fixed.
27001622	In Collections Manager, when a scenario is replaced with another, the due date calculated for collection actions was incorrect. This has been fixed.
27001628	In Collections Manager, there was a deadlock situation during the collections process. This has been fixed.
27001631	When getting the aggregated information about a resource from BRM, the process was failing if the events for those resources were purged in the system. This has been fixed.
27016775	After the backdated modification of the discount end time (END_T), the charges were calculated incorrectly if sequential discounting was enabled. This has been fixed.
27023170	The PIN_ERR_LOG_FLIST function was generating core dump when any field in the flist logged was null. This has been fixed.
27049618	The 32-bit BRM client applications were not working properly with large files on Linux platforms. This has been fixed.
27060891	When the pin_collections_process utility was run, if the event POID was sent as null in the input flist while calling the PCM_OP_AR_RESOURCE_AGGREGATION opcode, it was dumping core instead of returning a proper error message.
27067419	When using Conversion Manager for loading data, the parent account and account receivable details could not be provided at bill-unit level. This was causing issues if the account had multiple bill units. This has been fixed.
27067422	The discount purchase was failing when the discount validity was set to full or none. This has been fixed.
27116849	The tax percentages were not properly getting accumulated in the balance impact of the rated event. This has been fixed.
27130470	There was an issue with state enumerations that was causing a core failure in BRM. This has been fixed.

Table 1-8 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 1

Bug Number	Description
27159336	The audit data retrieved from the database was not in the correct order. As a result, the rerating process was not creating charging events. This has been fixed.
27159380	When setting the discount validity dates on cycle boundaries, the discount validity rules were not considered for calculation and this resulted in incorrect discounts. This has been fixed.
27159381	There was a need to reassign the range for custom fields. This has been fixed.
27178506	Rerating was failing for usage events created on the billing day. This has been fixed.
27181445	For hierarchical accounts, bill-level adjustments were not getting synchronized with ECE properly. This has been fixed.
27199237	When applying cycle discounts, full discounts could not be applied due to incorrect discount start dates set by the set_discountinfo opcode. This has been fixed.
27202235	When accounts had hierarchical bill units, account deletion was failing. This has been fixed.
27213533	When the fetch size reached the maximum limit, the pin_bill_accts utility reported a performance issue. This has been fixed.
27219879	The pin_recover utility in the -rfr mode was not updating the result to zero in the charge event. This has been fixed.
27223026	The subscription transfer workflow was not allowing to use the custom logic to control the rerate requests through policy. This has been fixed.
27226303	In BRM, it was possible to allocate the same adjustment to a bill twice. This has been fixed.
27229728	While loading offer profiles into BRM, the "Double Free Or Corruption" error was displayed. This has been fixed.
27236217	Rated Event Loader Daemon did not process backlog files and went into sleep mode. This has been fixed.
27244352	There was no option to disable journal creation for zero (0) balance records. This has been fixed.
27263078	Duplicate events were created for the same period for cycle-forward products and the events did not have any balance impacts associated with them. This has been fixed.
27266272	When PDC was used for configuring pricing objects, there was some issue in the Item Type Selector configuration due to format mismatch. This has been fixed.

Table 1-8 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 1

Bug Number	Description
27269943	The host name was not validated properly during the BRM installation in silent mode. This has been fixed.
27274074	The discount purchase operation was not considering the appropriate rate plan resulting in incorrect calculation of the discount period. This has been fixed.
27274076	For time-based full discounts, BRM was incorrectly calculating the discount amount. This has been fixed.
27277544	When a service configured to have only discounts was terminated, an error was thrown. This has been fixed.
27288519	For accounts with multiple bill units, write off reversal was failing as the reversal items pointed to one bill unit. This has been fixed.
27304029	Even if the log level was set to ERROR, some debug messages were getting recorded in the log files. This has been fixed.
27304030	The value for the StopBillClosedAccounts business parameter was always getting set to 1 irrespective of the value specified. This has been fixed.
27309569	The input data for the payment policy opcode, PCM_OP_PYMT_POL_PRE_REFUND, did not contain all the required information for policy processing. This has been fixed.
27309573	The pin_deposit utility was not collecting pre-authorized charges. This has been fixed.
27337657	When making adjustments, if the adjustment amount was more than the actual due amount, an inconsistent behavior was observed. This has been fixed.
27362564	In Customer Center, when the permission to access account receivables (/accounttool/araccess) was set to read-only, the customer service representative (CSR) was not able to view the adjustment details from the Bill Details or Item Details page. This has been fixed.
27369059	System discounts for cycle arrear events were not applied properly. This has been fixed.
27389701	Web Services Manager was returning the Cannot initialize error when triggering the opcodes. This has been fixed.
27415116	Rerating events after changing the accounting cycle day of month (DOM) was incorrectly impacting the balances. This has been fixed.

Table 1-8 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 1

Bug Number	Description
27418546	The PCM_OP_COLLECTIONS_CALC_AGING_BUCKETS opcode was not considering the latest or current hierarchy to calculate the amount due from aging buckets. This has been fixed.
27425461	When installing BRM in silent mode, the response file required the database system user password which was a potential security risk. This has been fixed.
27448146	When the billing delay was configured, auto-triggered billing from ECE was generating a final bill instead of a partial bill. This has been fixed.
27463239	Launching Suspense Center with Java WebStart had issues due to the version mismatch between the shipped version (4.5.3) and the version specified (4.4) in the orasoft.jnlp file. This has been fixed.
27463244	The General Ledger job was taking more time than expected and required a performance update. This has been fixed.
27507372	When a product configured with the cycle arrear event was canceled with a backdated date, the refund was not applied properly. This has been fixed.
27523584	Rerating after purchasing a discount was not working as expected. This has been fixed.
27523592	When a child account was moved under a parent account, the usage charges were not included in the correct bill. This has been fixed.
27539381	In some cases, while disconnecting a service, the PCM_OP_CUST_SET_STATUS opcode was entering a deadlock. This has been fixed.
27539388	If sequential discounting was enabled, the subscription discounting was getting applied on top of the system-level discount. This was causing errors in some scenarios. This has been fixed.
27552474	During rerating, additional tax balance impacts were created. As a result, the adjustment calculated was incorrect. This has been fixed.
27584537	Conversion Manager was not handling migration of accounts with quarterly bill cycles properly. This has been fixed.
27591136 27738257	When event adjustment was done after tax-only adjustment, the tax jurisdiction information was not updated properly in the balance impact of the event adjustment. This has been fixed.
27613611 27613617 28149096	Rated Event Loader (pin_rel) was exiting due to the out of memory error. This has been fixed.

Table 1-8 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 1

Bug Number	Description
27613636	During cancellation of a product configured with in-advance billing, incorrect START_T and END_T were set in the events. This has been fixed.
27621461	In Collections Manager, when an account had two bill units, a paying bill unit and a non-paying bill unit, the aging due was calculated incorrectly. This has been fixed.
27667959	While recovering the failed payment, duplicate transaction IDs were observed and this was due to running two applications to recover the same failed payment, one was pin_recover and the other was a custom application, in parallel. This has been fixed.
27667961	The General Ledger job was taking more time than expected and required a performance update. This has been fixed.
27684532	In Collections Manager, if the PCM_OP_COLLECTIONS_GET_ACTION_HISTORY opcode was triggered with a transaction having a database number different from that of the collection event user, the opcode was failing. This has been fixed.
27691744	For an account with multiple bill units, a payment reversal for a backdated account level payment was having effective date as the current system date instead of the back date. This has been fixed.
27745481	The Oracle Data Manager process was failing with dumping core when processing the Account Receivables workflow. This has been fixed.
27745484	BRM was not providing options to control or customize workflow over rerate requests which were created as part of subscription transfer operation. This has been fixed.
27745489	When the EventAdjustmentDuringcancellation flag was enabled, the flist data corruption issue occurred intermittently, which was dumping core. This has been fixed.
27751025	The PCM_OP_AR_RESOURCE_AGGREGATION opcode was returning incorrect aggregated amount. This has been fixed.
27771916	The cancel discount operation was not setting the validity dates properly for discounts for which the end dates were in previous cycles. This has been fixed.
27795345	If sequential discounting was enabled, the PCM_OP_SUBSCRIPTION_SET_DISCOUNTINFO opcode was not working properly for in-advance billing periods. This has been fixed.
27816387	Even after paying the bill amount, the bill amount from the last bill was displayed as balance brought forward. This has been fixed.

Table 1-8 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 1

Bug Number	Description
27881741	When an event was rerated twice, the tax jurisdiction was not handled properly for adjustment events. This has been fixed.
27881743	When the same product was purchased twice, canceling one of the product deleted the provisioning tag from both the products. This has been fixed.
27881746	For billing cycles which spans across multiple months, newsfeed was getting created for the /event/notification/service_item/make_bill event for each accounting cycle. This has been fixed.
27891324	In a prepaid environment, when the subscriber's balance was lower than the required balance for a purchase and if a deal was purchased, BRM displayed the credit limit exceeded error. In addition, signal 11 was observed in Connection Manager. This has been fixed.
27894048	The PCM_OP_SUBSCRIPTION_TRANSITION_PLAN opcode was not canceling the deals from the existing plan while moving to the new plan. This has been fixed.
27894050	During item adjustment, the search operation got into a infinite loop. This has been fixed.
27938633	The dm_fusa front end was not receiving data after the connection manager established connection with it. This has been fixed.
27938637	It was not possible to locate the Secure Sockets Layer (SSL) client wallet file due to the use of back slash (\) in the resource path instead of a forward slash (/). This has been fixed.
27938640	The PCM_OP_BILL_GET_ITEM_EVENT_CHARGE_DISCOUNT opcode was not allowing to add or modify the event data in its workflow through the PCM_OP_BILL_POL_GET_EVENT_SPECIFIC_DETAILS policy opcode. This has been fixed.
27956401	In Collections Manager, the collection write-off was failing for the bill unit in collection. This has been fixed.
27979521	The PCM_OP_SEARCH base opcode was failing when the template included a nested select query with the grouping by clause. This has been fixed.
27993358	During billing, the grantor information in the rollover bucket was incorrectly set to null. This has been fixed.
27993360	In LDAP Manager, the entry update operation was failing. This has been fixed.
28007850	When rating an event, sponsored products were ignored. This has been fixed.
28020470	If a service had two products and when one product was canceled, the EFFECTIVE_T value of the service was also updated. This was not allowing the processing of usage events for the other product. This has been fixed.

Table 1-8 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 1

Bug Number	Description
28089237	Quarterly cycles were getting merged which caused issues in applying charges. This has been fixed.
28089239	The date format in Web Services Manager could not be configured based on the Java's SimpleDateFormat specification. This has been fixed.
28119224	When in-advance billing was enabled, purchasing discount with a future date to become valid on ACTG_NEXT_T resulted in validation errors. This has been fixed.
28123147	Conversion Manager was not providing an option to capture bill_accounting_cycles_left and next_bill_t information for migration. This has been fixed.
28141241	The dm_ldap process was failing to start when SSL was enabled. This has been fixed.
28141243	After loading the rated events from Pipeline Manager, it was found that MOD_T of the corresponding items and journals were less than the CREATED_T. This has been fixed.
28159376	When an adjustment was applied in BRM, the revision number for the balance group was set incorrectly to zero causing data inconsistency between BRM and ECE. This has been fixed.
28200056	After purchasing the monthly cycle forward product, the newsfeed for the same did not include the bill unit information. This has been fixed.
28220691	When there were multiple bill adjustments, reallocation of the payment was done incorrectly. This has been fixed.
28247287	The bill generated using Bill Now was not including all items due for billing. This has been fixed.
28255178	Even if there was no amount left for adjustment, BRM was allowing adjustment up to the full adjustment amount. This has been fixed.
28259108	When an APN Map was reloaded using a semaphore, the rating functionality did not consider the new APN Map configuration. This has been fixed.
28261961	BRM was considering the regular monthly cycle forward event as backdated if the billing was run after the specified in-advance billing days in the same month. This has been fixed.
28261967	For cycle forward events, having discounts the newsfeed functionality was not working. This has been fixed.
28278327	Setting purchase start dates for first usage products was not allowed by using the PCM_OP_SUBSCRIPTION_SET_PRODINFO opcode. This has been fixed.

Table 1-8 (Cont.) Customer-Reported Fixes for BRM 12.0 Patch Set 1

Bug Number	Description
28281697	When BRM was processing orders by using JCA Adapter, if the CM instance goes down, the JCA Adapter was unable to restore the transaction state. As a result, the subsequent orders could not be processed without restart. This has been fixed.
28334999	In case of sequential discounting, expired or ended discounts were also applied when new discounts were purchased. This has been fixed.
28342603	When an account was created, the ACCOUNT_NO value was not updated correctly in the UNIQUE_ACCT_NO_T table. This has been fixed.
28342608	The event adjustment for the product configured for deferred taxes was not updating the tax jurisdiction information in the balance impacts correctly. This has been fixed.
28366495	There were memory leaks observed in the PCM_OP_TCF_AAA_STOP_ACCOUNTING opcode workflow. This has been fixed.
28366497	There was a memory leaks observed in the op_tcf_aaa_search_session workflow. This has been fixed.
28379547	In a BRM system integrated with ECE, External Manager (EM) Gateway was causing errors which led to data inconsistency. This has been fixed.
28379549	When the updated pricing data was loaded using semaphore, the charges were calculated incorrectly. This has been fixed.
28379551	With in-advance billing, the scale was calculated incorrectly as zero for a valid purchased product having an end date set. This has been fixed.
28415082	The PCM_OP_SUBSCRIPTION_TRANSITION_PLAN opcode was not canceling the deals from the existing plan while moving to the new plan. This has been fixed.
28463456	The rerating flow was not updating the session_obj and rerated_obj information correctly. This has been fixed.
28466613	When doing a plan transition, it was not possible to override the resource balance granted or add any custom workflow. This has been fixed.
28359386 28359381	In Collections Manager, if schedule object was either completed or canceled, it was effective only for collection group owners, but not for members. This has been fixed.

Customer-Reported Fixes in ECE

See the following for a list of customer-reported issues that were fixed in each Oracle Communications Elastic Charging Engine (ECE) patch set:

- [Customer-Reported Fixes in ECE 12.0 Patch Set 8](#)
- [Customer-Reported Fixes in ECE 12.0 Patch Set 7](#)
- [Customer-Reported Fixes in ECE 12.0 Patch Set 6](#)
- [Customer-Reported Fixes in ECE 12.0 Patch Set 5](#)
- [Customer-Reported Fixes in ECE 12.0 Patch Set 4](#)
- [Customer-Reported Fixes in ECE 12.0 Patch Set 3](#)
- [Customer-Reported Fixes in ECE 12.0 Patch Set 2](#)

Customer-Reported Fixes in ECE 12.0 Patch Set 8

[Table 1-9](#) lists the customer-reported issues that were resolved in ECE 12.0 Patch Set 8.

Table 1-9 Customer-Reported Fixes for ECE 12.0 Patch Set 8

Bug Number	Description
34141235 34258242	In previous releases, it was reported that additional, unnecessary SNRs were generated from ECE. This has been fixed. ECE generates the next SNR only after receiving an acknowledgment for the previous SNR with no connection loss.
34096234	In previous releases, the PARTITION_TIME column was not included in InvalidRatedEvent objects. This has been fixed.
33873681	In previous releases, the HTTP Gateway monitoring metrics did not include data for assessing the success or failure of charging requests. This has been fixed.
34136907	In the SLR flow, a NumberFormatException was reported for fields related to PendingPolicyCounter information. This has been fixed.
34017996	In previous releases, the 5G usage payload was failing when the time zone information included daylight saving time (DST) offset information in HTTP Gateway. This has been fixed.
34019605	The thread configuration for 5G unrated CDRs in the CDR Formatter required updates to improve performance. This has been fixed.
33941503	In previous releases, it was reported that some SNR notifications included expired balances. This has been fixed.
33997433	In previous releases, unrated CDR file names included only starting and ending timestamp information. This has been fixed. Unrated CDR file names now also include the cluster name and sequence number.
34058036 34312618	Automatic top-up functionality was enabled through non-counter grants during an on-going session. This is now similar to recurring bundle grants, so that it is available for all other parallel sessions immediately after the grant.

Table 1-9 (Cont.) Customer-Reported Fixes for ECE 12.0 Patch Set 8

Bug Number	Description
34073280	In previous releases, used service units rating was considering an additional 1 byte while processing the request. This has been fixed.
34021930	In previous releases, in a multi-RUM scenario, a TTC was generated incorrectly when just one of the RUMs was used. This has been fixed.
34012910	To improve CDR Formatter performance, the ECE 5G CDR session store has been enhanced with updated table definitions and additional index definitions.
34064745	In previous releases, performance issues were reported for 5G usage processing. This has been fixed with additional index definitions.
34121619	In previous releases, the realm and host information was not configurable for a multi-site deployment of ECE Diameter Gateway. This has been fixed.
34139881	In previous releases, breach notifications were not replicated to all members in a discount sharing group. This has been fixed.
34136087	In previous releases, the FINAL trigger was not supported at the rating group level. This has been fixed.
34145186	In previous releases, in some instances pending policy status calculation was done using incorrect balance information. This triggered unnecessary notifications. This has been fixed.
34090990	In previous releases, for 5G offline requests, it was required that the subscriber account be present in the BRM system. However, this is not true for an in-roaming subscriber. This has been fixed.
34103112	In previous releases, 5G offline requests were processing within the ECE charging grid, even if only unrated CDRs were to be generated for such requests. This has been fixed.
33878964	In previous releases, OCOMC ECE DC was not able to register or initialize due to missing monitor framework updates. This has been fixed.
34341927	In previous releases, when HTTP Gateway was processing 5G usage requests, it sometimes missed the generation of unrated CDRs or generated duplicate CDRs for concurrent requests. This has been fixed.
33062444	In previous releases, when the ECE persistence database was down, the usage charging flow involving conditional RUMs was delayed or there was no response in the charging flow back to the CTF. This has been fixed.
34400697	Thread leaks were reported in the purge rated events workflow. This has been fixed.

Table 1-9 (Cont.) Customer-Reported Fixes for ECE 12.0 Patch Set 8

Bug Number	Description
34380199	In previous releases, when creating new AVPs from the extensions API, the AVPs of type Enum were getting created as DiameterInteger32 instead of DiameterEnumerated. This has been fixed.
34000658	In previous releases, unrated CHF CDR generation was not supporting failover scenarios for a multi-site deployment. This has been fixed.
33972900 34746451	In previous releases, the validity time was not aligned with GSU for dynamic quota configuration. This has been fixed.
34401823	In previous releases, on multi-site active-active deployments, during customer loading from the BRM database to the ECE cache through Customer Loader or Customer Updater, the customers were not segregated into groups. This has been fixed.
34469111	In previous releases, in the charging flow, impacting for the consumed reservation was incorrectly done if the consumption was from a topped-up balance. This has been fixed.
34317216	In previous releases, for an ECE cloud native multi-site active-active deployment, both Pricing Updater and Customer Updater pods were started only in the primary site and not started in the remaining sites. This has been fixed.
34217563	In previous releases, when a session was in progress, balance updates were incorrect when a member was added or removed from a discount sharing group (DSG). This has been fixed.
34402458	In previous releases, some deployments of Coherence federation reported a destination error. This was due to thread leaks and fixed through Bug 34400697.
34211050	In previous releases, after a member was added to a discount sharing group, an SNR was not generated for any policy counter updates in member bundles. This has been fixed.
34205430	In previous releases, the same policy counter was being returned multiple times in an SLA. This has been fixed.
34262280	In previous releases, when multiple SNRs were sent for a policy counter during an ongoing session, IO timeout errors were reported. This has been fixed.
34310172	In previous releases, when a plan was about to expire, ECE was sending an incorrect validity time, without considering the end time of all applicable offers to evaluate the validity time for the request. This has been fixed.
34325037	In previous releases, 5G CHF (HTTP Gateway) was sending RARs using the Notify URI instead of the SCP URL configured in scpAuthorities. This has been fixed. Additionally, the notify URI is set in the header as "3gpp-Sbi-Target-apiRoot" and "user-agent" = "CHF".

Table 1-9 (Cont.) Customer-Reported Fixes for ECE 12.0 Patch Set 8

Bug Number	Description
34800211	In previous releases, the Tariff Time Change (TTC) was not reported in UTC. This has been fixed.
34777745	In previous releases, when a connectivity issue was experienced between 5G CHF (HTTP Gateway) and NRF, HTTP Gateway was retrying NRF Heartbeat for a preconfigured number of times. But, if the number of instances was just one, an option was required to retry until NRF was operational again. HTTP Gateway has been enhanced to support this now.
33909430	In previous releases, due to a mismatch in the currency balance element range between ECE and BRM, ECE was reporting a version mismatch. This has been fixed.
34698228	In previous releases, for an ECE active-active deployment, the Rated Event Formatter (REF) was failing on site 2 (secondary). This has been fixed.
34638932	In previous releases, 5G CHF (HTTP Gateway) was failing when the charging request included a null USU after the previous request included a closure trigger. This was also failing during the unrated CDR generation. This has been fixed.
34488077	In previous releases, in the rerating flow, impact category information was incorrectly populated as null. This has been fixed.
34049315	In previous releases, 5G unrated CDR generation did not have configuration for the Kafka topic name, number of partitions, SSL security, and compression methods. CDR generation has been enhanced to support this functionality. The configuration now also includes the wallet location and the TrustStore location.
34766318 34723470	In previous releases, for an active-active mode of ECE, the customer group list was not added during the deployment. This has been fixed.
34617995	In previous releases, during the charging flow, the UNIX epoch time-to-date conversion was incorrectly done for earliest-expiry-time. This has been fixed.
34693587	In previous releases, while processing charging requests, the routing data was set incorrectly and hence HTTP Gateway was unable to consume SNRs from ECS. This has been fixed.
34692273	In previous releases, first usage notifications were generated twice for the same subscriber. This has been fixed.
34483124	In previous releases, SNRs were not generated for parent accounts in a discount sharing group. This has been fixed.
34692304	In previous releases, the GSU calculation logic was incorrect. This has been fixed.

Table 1-9 (Cont.) Customer-Reported Fixes for ECE 12.0 Patch Set 8

Bug Number	Description
34627786	In previous releases, 5G CHF (HTTP Gateway) was not retrying NRF Heart Beat requests in case of any network exception. This has been fixed to retry for a configured number of times based on nrfHeartBeatRetryCount .
34542607	In previous releases, SNRs were not getting generated for a member of a discount sharing group when the member was deleted. This has been fixed.
34235570	In previous releases, the Tariff Time Change scenario was missing a reservation on the current bill cycle. This has been fixed.
34633958	In previous releases, 5G charging requests were failing with incorrect validation of volumeLimit trigger. This has been fixed.
34658482	In previous releases, 5G charging requests were failing at CDR Gateway for a Release request without Used Service Units information. This has been fixed.
34637428	In previous releases, it was observed that whenever a CCR-Update Request having USU=0 was received, ECE was adding an extra 10 bytes to the next Granted Service Units (GSU). This issue was observed in roaming zones, and when the rateZeroQuantity parameter was set to true in charging-settings.xml . This has been fixed.
34505169 34664433	In previous releases, the new automatic top-up bucket was not created when usage occurred during the last minute of the current bucket's validity. This has been fixed.
34247534	In previous releases, there was a discrepancy between the amount of reserved units and the Granted Service Units (GSU). This has been fixed.
34577587	In previous releases, for a multi-site active-active deployment of ECE, when a usage charging session was initiated in site 2, an SNR was generated and sent to the Kafka partition assigned to Diameter Gateway in site 2. However, it was observed that the SNR was picked up by Diameter Gateway in site 1. This has been fixed.
34527516	In previous releases, you started a session and triggered SLR through PCRF in Site1 to Diameter Gateway in site 1, and after some updates, you triggered STR through PCRF in Site 2 to Diameter Gateway in site 1. Then, Diameter Gateway in site 1 was responding with an STA to Site 1 instead of site 2, because Diameter Gateway was fetching the PCRF host from the session instead of the STR input. This has been fixed.
34569302	In previous releases, the Rated Event Formatter (REF) was not starting due to the incorrect mapping of the persisted_field and persisted_field_5g fields in the request specification. This has been fixed.

Table 1-9 (Cont.) Customer-Reported Fixes for ECE 12.0 Patch Set 8

Bug Number	Description
34589041	In previous releases, when nonlinear rating was configured, the Eval function defined in the pre-rating extension was not evaluated correctly. This was because any change due to the extension logic was not overriding the charging request's payload. This has been fixed.
34589097	In previous releases, when zeroRatedQty was enabled and incorrectly a midsession request for a termination came in, the request was failing. This is fixed now, with a check for not creating a mid session for terminate request.
34497871	In previous releases, when 5G CHF (HTTP Gateway) was processing multiple parallel charging requests, there was inconsistency in CDR generation. it was observed that multiple CDRs were of the same content. This has been fixed.
34614000	In previous releases, for a charging session with multiple rating groups, while generating a midsession CDR for a CCR update request with just one rating group, the midsession CDR was generated with just one rating group instead of all rating groups impacted by the session. This has been fixed.
34589076	In previous releases, while consuming data fully for a resource, reverse rating was failing with a null pointer exception. This has been fixed.
34567566	In previous releases, the consumed reservation calculation was incorrect for an update request due to rounding of used service units to the beat. This has been fixed.
34717616	In previous releases, for an ECE deployment with cache persistence configuration enabled, if the tax selectors were updated when the ECE database was down, usage processing was failing. This has been fixed.
34624902	In previous releases, Diameter Gateway was not consuming RAR notifications from Kafka topics. This has been fixed.
34697193	In previous releases, when ECE rated usage, subbalance impacts were created even for zero currency impacts for loan balances. This has been fixed.
34673867	ECE has been enhanced to provide an API to access the debit refund session list that can be used in the extension hooks for customizations.
34613541	The balance output map was having loan and current amounts as null. This has been fixed.
34602911	The balance query through Diameter Gateway was returning incorrect information when the balance item subtype was not set. This has been fixed.
34606521	For a CCR request with TTC enabled, the granted service units were incorrectly calculated. This has been fixed.

Table 1-9 (Cont.) Customer-Reported Fixes for ECE 12.0 Patch Set 8

Bug Number	Description
34652879	In previous releases, there was an issue with dynamic credit floors. This has been fixed to consider the sum of original granted balances at a given time as the dynamic floor.
34236341	In previous releases, there was no DSL support for lifecycle expiration dates. This has been fixed.
34700667	In previous releases, ECE was creating subbalance buckets with zero amounts. This has been fixed.
34760612	In a multischema deployment, when ECE was processing ProductInfoChange events through Customer Updater, the request was timing out in ECE. This has been fixed.
34810673	In previous releases, when processing a usage request, the used service units quantity was not getting rounded. This has been fixed.
34779306	It was observed that the error bit was not set in the Diameter CCA header for all protocol errors. This has been fixed.
34818739	In previous releases, for a tariff time change (TTC) scenario, the information for pre-TTC was not completely provided. This has been fixed.
34236998	In previous releases, when a Diameter client was sending a request for CER, the Diameter Gateway pod was not sending any response when an incorrect realm and origin host was used. This has been fixed. A message will now be logged when a correct realm and origin host are not used.
34796483	When processing delayed usage, the balance subtype set was incorrectly in rated events for loan balances. This has been fixed.
34711900	In ECE cloud native deployments, while processing incremental loading, the ece-preselect container was failing with an error as it could not process JDBC URL. This has been fixed.
34711877	In ECE cloud native deployments, loadCrossRefData logic was not included in the Customer Loader template. This has been fixed.
34862738	In previous releases, if the usage session arrived soon after the cycle boundary, and if the cycle fee processing was not complete in BRM, the charging workflow was not able to reserve from the correct balance bucket. This led to errors in resource quota management and consumption. This has been fixed by triggering the cycle fees from ECE and using the latest cycle grants for quota.
34397942	In previous releases, ECE did not cache the bundle/deal name and ID information, and then make it available to enrich rated events or notifications. This has been fixed. Now, the bundle/deal name and ID are accessible from the extension layer.

Table 1-9 (Cont.) Customer-Reported Fixes for ECE 12.0 Patch Set 8

Bug Number	Description
34184725	In previous releases, Diameter Peer Requests (DPRs) were not consistently answered by Diameter Gateway. This has been fixed.
34483649	In previous releases, the opcode call to PCM_OP_BAL_ECE_GET_BALANCES was not including loan balance information in the output. This has been fixed.
34543881	A null pointer exception was reported in IsBalanceUpdateRarGenerationQueryImpl, which was not handled. This has been fixed.

Customer-Reported Fixes in ECE 12.0 Patch Set 7

[Table 1-10](#) lists the customer-reported issues that were resolved in ECE 12.0 Patch Set 7.

Table 1-10 Customer-Reported Fixes for ECE 12.0 Patch Set 7

Bun Number	Description
33831830	In previous releases, you could not extend first-usage notifications in the post-charging workflow. ECE has been enhanced to support this.
33844645	CustomerPartitionSchemaMapping was updated even for single-schema deployments, which was not necessary. This has been fixed. A check for multischema has been added in the update flow for handling the CustomerPartitionSchemaMapping cache priming.
33629694	When there were multiple rating groups in a 5G online charging request, the quota reservation was not done correctly for granting units. This has been fixed.
33819713	When there were multiple used units in an offline-only charging request, HTTP Gateway was not mapping to the correct used units in ECE. This has been fixed.
33813619	In an active-active deployment, both primary and secondary instances of Rated Event Formatter were incorrectly configured to run on the same site. This has been fixed.
33769884	The 5G CHF CDRs generated by the CHF were missing certain attributes and the values of a few attributes were incorrect. This has been fixed.
33713970	Whenever the balance validity was updated in BRM, it was not synchronized correctly with ECE through EM Gateway. This has been fixed.
33773457	Diameter Gateway was not sending the URL redirection until a sufficient resource balance existed. This has been fixed.

Table 1-10 (Cont.) Customer-Reported Fixes for ECE 12.0 Patch Set 7

Bun Number	Description
33731809	The granted service units was incorrect when the reservation was from a single balance bucket. This has been fixed.
33770240	For 5G charging, multiple usage units were incorrectly considered mandatory. This has been fixed.
33745974	When a non-counter grant was done during discounting, the grant was not visible across sessions. This has been fixed.
33730620	For 5G charging, the policyCounterIds were kept incorrectly mandatory. This has been fixed.
33729723	For 5G charging, only one CHF instance was allowed to register with an NRF. Now, multiple CHFs can register with the same NRF.
33729402	For 5G charging, the multipleUnitUsage.usedUnitContainer.serviceld field was incorrectly considered mandatory. This has been fixed.
33704854	When a balance update was done on the BRM side for a customer, the balance was not synchronized to ECE and a Coherence error occurred. This has been fixed.
33697580	For 5G charging, the quotaManagementIndicator data type was incorrect. This has been fixed.
33606758	In an ECE active-active deployment, threshold breaches on an active site were not federated in decorative mode to the second site. This has been fixed.
33563419	During the charging process, ECE was not supporting and evaluating TotalC in Threshold_To expressions. This has been fixed.
33730943	In an ECE cloud native deployment, the PodHealthCheck process was creating orphan processes. This has been fixed.
33731972	In an ECE cloud native deployment, the cluster configuration was assuming a hard-coded name and was not allowing users to change it as required by the deployment. This has been fixed.
33734441	The ECE cloud native deployment was failing in interim patch 3 due to missing JAR files. This has been fixed.
33696433	The CHF was registering only for the Nchf_ConvergedCharging service with the NRF instance, and the Nchf_SpendingLimitControl service was omitted in the list of configured services. This has been fixed.
33814780	For 5G charging, ECE was not supporting tariff time change triggers in charging requests. This has been fixed.

Table 1-10 (Cont.) Customer-Reported Fixes for ECE 12.0 Patch Set 7

Bun Number	Description
33629355	For 5G charging, ECE was not using the quota management indicator to determine if a converged charging request contained online charging, offline charging, or both. This has been fixed.
33967432	HTTP Gateway was having issues in thread handling that was generating a huge number of threads. This has been fixed.
33892939	CHF CDRs generated by ECE had format-related issues specified by the 3GPP specifications. This has been fixed.
33646227 31338683	In ECE during usage processing, when a generic selector was not evaluating to any rule, a success was returned instead of a failure with the reason as NO_RATING_GRAPH_CONFIGURED. This has been fixed.
33969575	Automatic top-up grants were not handled correctly when there were tariff-time changes. This has been fixed.
33966053 33831757	Non-counter grants with an automatic top-up was not immediately recognized across all parallel on-going sessions. This has been fixed.
33902616	Threshold breach notifications were not replicated for child accounts in a discount sharing group. This has been fixed.
33963284	The PUID information was not properly updated in SNRs. This has been fixed.
33951010	Top-up bundle purchases were failing when the subscriber had both primary and secondary subscriptions. This has been fixed.
33873603	For 5G deployments, you could not register multiple HTTP Gateway instances with an NRF. This has been fixed.
33883486 33872083	HTTP Gateway was sending null-valued attributes for NRF registration, which caused a registration error. This has been fixed.
33935343	When a first-usage validity notification was generated, it was not replicated to all members in the subscription group. This has been fixed.
33764005	In ECE, the redirection DSL was not reading the correct data in case of an override in the payload. This has been fixed.
33267357	The RUM quantities were incorrectly updated when non-linear rating was enabled and when evaluating beat scenarios. This has been fixed.

Table 1-10 (Cont.) Customer-Reported Fixes for ECE 12.0 Patch Set 7

Bun Number	Description
33964143	The active session object cleanup parameter was missing from the cloud native deployment templates for ECE. This has been fixed.
33955909	ECE was allowed to generate custom SNRs in the usage flow extension points. This has been fixed.
33812526	In ECE, a NullPointerException was thrown when a tariff-time change was being processed with non-linear rating enabled. This has been fixed.
33929321	For a subscriber having a plan that was about to expire, any new session that was started or any update to an ongoing session just before the expiration, the validity time that was set in the response was later than the plan expiration time. This has been fixed.
33936076 32258304	Value maps were not accessible in pre-rating extensions, and DSL optimization was required during selector evaluation. This has been fixed.
34245333	In rated events, the sub-type values for loan balances and main balances were incorrect. This has been fixed.
33864314 33613568	ECE was generating RARs with Service ID and Rating Groups, which was having issues when there were active sessions with multiple different Rating Groups. This has been fixed. Now this has been fixed to generate an RAR without Service ID and Rating Groups so that it is honored by all active sessions for all Rating Groups.
34301863	Rated events were not including a customer's main balance and loan balance information. This has been fixed.
34185865	In ECE, bundles that expired out of billing cycles were not getting their grants replenished. With this release, ECE triggers cycle-forward fees calculation for grants through the offers that do not coincide with the billing cycle date.
34136283	The service expiration time was not synchronized from BRM to ECE through Customer Updater and EM Gateway. This has been fixed.
34185917	ECE was allowing the configuration of non-linear rating at the product-type level. This has been fixed.
34185968	The ECE customer cache was not including additional data, such as the deal ID, bundle ID, and name as part of purchased offerings. Due to this, it was not possible to include these details in rated events or notifications through extensions. This has been fixed.
34194932	In ECE, loan balances were not considered for consumption prior to main balance consumption. This has been fixed.

Table 1-10 (Cont.) Customer-Reported Fixes for ECE 12.0 Patch Set 7

Bun Number	Description
34174335	As part of balance synchronization with BRM, if the balance item type was not synchronized, the balance query and other balance-related operations were incorrect. This has been fixed. ECE now considers the balance item type as the main balance if it is not explicitly indicated with the balance type attribute.
33570517	The balance query operation was not returning life-cycle state information, such as the service expiration time. This has been fixed.
34323608	In ECE Patch Set 6, HTTP Gateway had a memory management issue. This has been fixed.
34141235	ECE was sending the next SNR before the response for the previous SNR was received from the network, which is contrary to 3GPP specifications. This has been fixed. ECE now waits for the previous SNR response before sending the next SNR.
34096234	The PARTITION_TIME column was not available in the InvalidRatedEvent table definition. This has been fixed.
33873681	In previous releases, ECE was not providing HTTP Gateway-related metrics. This has been fixed.
34136907	During the Sy SLR flow, ECE was generating a NumberFormatException for using an incorrect data type for an attribute. This has been fixed.
34017996	During 5G usage processing, a failure was reported when a request included a date field with a time zone having day light savings. This has been fixed.
34019605	5G CHF CDR Formatter was having some performance issues and required an optimized thread management. This has been fixed.
33941503	SNR notifications were generated based on available balances that incorrectly included expired balances. This has been fixed.
33997433	The 5G CHF CDR file naming syntax required an update. The file name syntax now includes the cluster name, starting time stamp, ending time stamp, and sequence number (<i>ClusterName-StartTimeStamp-EndTimeStamp-SequenceNumber.Extension</i>). For example: BRM-1654514133000-1654514134000-1.out.
34058036	Automatic top-up grants were not handled correctly for accounts configured with discount sharing group. This has been fixed.
34073280	While calculating the used and consumed units, a difference of one byte was encountered. This has been fixed.

Table 1-10 (Cont.) Customer-Reported Fixes for ECE 12.0 Patch Set 7

Bun Number	Description
34021930	Tariffs were evaluated to true even when the daily bundle charge rate plan was not applicable for a given usage request. This was also leading to creating multiple time points for TTC that were not actually applicable. This has been fixed.
34012910 34064745	The 5G CHF CDR Generator required performance improvements with database indexes.
34121619	In an active-active set up of Diameter Gateway, the realm and host configurations were incorrect. This has been fixed.
34139881	Threshold breach notifications were not triggered for all member accounts in a discount sharing group. This has been fixed.
34136087	For 5G usage rating requests that included multiple rating groups, the FINAL trigger was not supported. Instead, the granted service units was provided in the response. This has been fixed.
34145186	The pending policy counter was incorrect for SNRs that were triggered after purchasing certain bundles. This has been fixed.
34090990	In ECE, 5G offline charging was failing for usage requests for in-roaming subscribers, causing CDRs to not be generated. This has been fixed.
34103112	In ECE, every request was sent to charging irrespective of whether it required online or offline charging. This has been fixed. You can now configure whether to rate offline requests in ECE.
33878964	There was an initialization error for ECE Distribution Cartridge (DC) to work with ECE due to missing monitor framework dependency. This has been fixed.

Customer-Reported Fixes in ECE 12.0 Patch Set 6

[Table 1-11](#) lists the customer-reported issues that were resolved in ECE 12.0 Patch Set 6.

Table 1-11 Customer-Reported Fixes for ECE 12.0 Patch Set 6

Bug Number	Description
33829559	<p>The following issues were reported while generating a CHF CDR:</p> <ol style="list-style-type: none"> 1. The Record Type and Rating Indicator tags were not populated. 2. A negative duration was reported in the CHF CDR. 3. The record sequence number and local sequence number values were incorrect in CHF CDRs. 4. The totalVolume value was populated incorrectly. <p>These have been fixed.</p>
33985970	<p>The following issues were reported while generating a CHF CDR:</p> <ol style="list-style-type: none"> 1. The sequence number fields (LocalRecordSequenceNumber, LocalSequenceNumber, and RecordSequenceNumber) were not incrementing correctly as defined in the 3GPP specifications. 2. External trigger details for some CDRs were not populated even though causeOfRecordClosing was correctly mentioned. <p>These have been fixed.</p>
33809523	<p>Reservations were made incorrectly when a 5G HTTP request for converged charging included multiple rating groups.</p> <p>This has been fixed.</p>
33985962	<p>When an account purchased a charge offer that included usage charges for which charging is based on a generic selector, Diameter Gateway usage processing returned an incorrect status.</p> <p>This has been fixed.</p>
33834862	<p>It was not possible to get a first usage notification service context in the post charging extension.</p> <p>This has been fixed.</p>
33828497	<p>Defunct processes from the ECE Pod health check were blocking worker nodes, because the parent process did not correctly handle child process termination.</p> <p>This has been fixed.</p>
33794316	<p>In an active-active DR setup, threshold breach and aggregated threshold breach events were not getting federated across sites.</p> <p>This has been fixed.</p>
33880713	<p>In an ECE setup with non-linear rating enabled, errors occurred while processing usage requests. When the period field in a balance item impact was NULL, ECE threw a NULL pointer exception.</p> <p>This has been fixed.</p>
33724587	<p>After deploying ECE in a cloud native environment, ECE was not connecting to JConsole.</p> <p>This has been fixed.</p>
33828481	<p>The installation of ECE 12.0 Patch Set 4 Interim Patch 3 in a cloud native environment was failing due to missing JAR files.</p> <p>This has been fixed.</p>

Table 1-11 (Cont.) Customer-Reported Fixes for ECE 12.0 Patch Set 6

Bug Number	Description
33742497	When a balance update was published to ECE from BRM, ECE was returning a Coherence error. This has been fixed.

Customer-Reported Fixes in ECE 12.0 Patch Set 5

[Table 1-12](#) lists the customer-reported issues that were resolved in ECE 12.0 Patch Set 5.

Table 1-12 Customer-Reported Fixes for ECE 12.0 Patch Set 5

Bug Number	Description
33661199	The following issues were reported on 5G CHF HTTP Gateway functionality: <ol style="list-style-type: none"> 1. The HTTP Gateway was checking if CHF was already registered before reregistering, which was an unnecessary step. 2. The HTTP Gateway was not sending a correct header for the Nchf_ConvergedCharging_Create operation. 3. If an attribute in the JSON payload had a NULL string, NRF did not process the request. 4. The data type for pDUSessionChargingInformation.chargingId was incorrectly set to integer instead of unsigned integer. 5. If a request had multiple rating groups, the HTTP Gateway was not reserving quota correctly. These issues have been fixed.
32311282	Performance issues occurred during the clean up of active session objects. This has been fixed. Also, a validation on the product ID was introduced.
33294593	There was no service to connect to JConsole from an ECE cloud native deployment. This has been fixed.
32665703	RADIUS Gateway was not handling access requests that required the PBKDF2 algorithm. This has been fixed.
32647547	While transferring a service from one account to another, the item POID for the service from the old account was also transferred to the new account. This caused the usage for the new account to be incorrectly recorded for the old account. This has been fixed.
33291753	After a service was transferred to a new billing account, usage charges were still rolling up to the old billing account. This has been fixed.
33189250	There were cluster timeouts in an active-active setup configuration followed by ConcurrentModificationException. This has been fixed.
33436905	The CustomerUpdater was crashing due to data inconsistencies even when the continueCustomerLoaderOnError flag was enabled. This has been fixed.

Table 1-12 (Cont.) Customer-Reported Fixes for ECE 12.0 Patch Set 5

Bug Number	Description
33249239	The subscriber tracing functionality was not working during usage processing even after enabling it. This has been fixed.
33249243	The group notifications feature was not working properly due to thread deadlocks. This has been fixed.
33249242	Readiness probes were not available for EM Gateway, RADIUS Gateway, and Diameter Gateway services. This has been fixed. Readiness probes have been added to these gateways. The probes indicate that a Pod is ready when its ports are open and ready to accept connections.
33189231	First usage validity was not working because the calculated validity was incorrect. This has been fixed.
33331310	For GPRS events, both INVOICE_DATA and RUM_NAME were populated incorrectly when it was set in the post rating extension hooks. This has been fixed.
33372715	During rerating, a rating error occurred for an account with grants and usages. This has been fixed.
33373301	While rating the last chunk of used units, Diameter Gateway generated a DIAMETER_CREDIT_LIMIT_REACHED error if there was no balance available to determine the requested units. This has been fixed.
33459324	When a SIM was changed, there was an issue fetching the product information that caused a DIAMETER_RATING_FAILED error. This has been fixed.
33114726	In ECE cloud native deployments, a SSL handshaking issue with BRM server occurred. This has been fixed.
33658703	When a balance was exhausted and a new usage request arrived, Diameter Gateway was generating a DIAMETER_CREDIT_LIMIT_REACHED (4012) error rather than generating a Final Unit Indicator (FUI). This has been fixed.

Customer-Reported Fixes in ECE 12.0 Patch Set 4

Table 1-13 lists the customer-reported issues that were resolved in ECE 12.0 Patch Set 4.

Table 1-13 Customer-Reported Fixes for ECE 12.0 Patch Set 4

Bug Number	Description
31837847	When diameter gateways were shut down after a failure of an Elastic Charging Server, the gateways could not be restarted until the Elastic Charging Server and processes were started. This has been fixed.

Table 1-13 (Cont.) Customer-Reported Fixes for ECE 12.0 Patch Set 4

Bug Number	Description
32085459	After deploying ECE patch 11.3.0.8.10, duplicate usage requests were failing. This has been fixed.
31984386	After upgrading to ECE patch 11.3.0.9.10, invoice generation was failing. This was because the net_quantity field was not being updated when usage events were generated. This has been fixed.
31843322	While while synchronizing data from BRM to ECE for creating customers or billing and while members or a sharing group generated usage at the same time, transaction locks were resulting in 5012 errors, and services could not be used. This has been fixed. When an impacted account is in a transaction lock and then placed in the resubmit queue, a sleep is introduced. When the resubmit times out, a response with error code 4010 is returned.
32960759	Diameter gateway was timing out after 10 second when performing duplicate checks. This was because of Elastic Charging Server failures during the duplicate check. This has been fixed.
32647547	During usage rating, services that had been transferred from one account the another were still tagged with the old account. This was because IPL and NIPL were not fetched during the transfer, so charges were routed to the old items. This has been fixed.
32258295	When increase balance impacts were added to a counter with no credit limit or monetary balances changed, the SafetyNet based on the return value from <code>getAvailableBalance</code> method was not working. This has been fixed. A new <code>getAvailableBalanceUpToCreditLimit</code> method is now accessible through the <code>getBalance()</code> interface in PostRatingExtension .
33022624	For online usage UPDATE requests, when data bucket is completely consumed, <code>QuotaHoldingTime (QHT)</code> is being reported even though <code>GrantedServiceUnit (GSU)</code> is not set. This has been fixed.
32996820	When a new customer was created with a recycled MSISDN and a new family sharing plan was purchased in the same transaction, the sharing group creation and family sharing plan purchase was failing. This was because the <code>AlterationSharingAgreement</code> was updated in the old account and not the new account. This has been fixed
31571808	When requests to ECE from the Diameter Gateway timed out at the Diameter Gateway but continued to be processed in ECE, the Diameter Gateway was sending a response to the user or pricing gateway, even after the timeout, and the Diameter Gateway response was rejected because of the timeout. This has been fixed. If the transaction times out in the Diameter Gateway, the corresponding transaction in ECE is also cleared.
32470519	The HTTP gateway was not coming up. This was because of a the missing <code>jackson-datatype-jsr310-2.11.2.jar</code> file. This has been fixed.

Table 1-13 (Cont.) Customer-Reported Fixes for ECE 12.0 Patch Set 4

Bug Number	Description
32519770	The PCM_OP_CUST_DELETE_ACCT opcode was taking a long time to delete accounts. This was because of a slow flow, gateway latency spikes, and CPU usage spikes. This has been fixed.
32538295	In ECE UpdateExtension, calls to extensionContext.getExtensionsDataRepository() were returning nothing instead of the Extensions Data Repository View. This has been fixed.
31880868	When a terminate request was coming after session expiration, positive balances were being issue to prepaid subscribers. This was because ECE was sending a validity time of 0 for valid grant service units, and when validity time is 0, the network only sends CCR-U if the reservation is fully used or there is a CCR-T. This has been fixed.
31630743	When an SNA failed to process with an invalid AVP error, subsequent RAR/SNR notifications were getting stuck and the Diameter Gateway would stop processing RAR and SNR messages. This has been fixed.
31902139	For one time bundles, tariff time change (TTC) was not working when balance existed in the current month but not the next month balance and the system attempted to reserve a balance shortly before the balance expiry. This has been fixed.
31692737	The Diameter Gateway needed to be restarted because of increasing active sessions and heap size. This has been fixed. Stale sessions are now cleaned up.
32842304	When a product status was changed, active session cleanup was failing due to a no matching product error. This has been fixed.
31983454	Subscriber Preferences and Service Lifecycle were not included in CCA for EVENT_REQUEST (BALANCE_QUERY) even though they were enabled for use in extensions. This has been fixed.
31832442	ECE was failing to process RAR when there were multiple rating groups associated with the same session. This has been fixed.
32567193	The Diameter Gateway was not processing RAR notifications, which was increasing the pending RAR queue size. This has been fixed.
31559378	Restarting the Diameter Gateway during a rolling upgrade procedure was generating a TransportsMessageBundle-0903:Unable to create session error multiple times. This error occurs if durable subscription is being used with WebLogic Topic, and the same clientID and subscription name is in use. This has been fixed.

Table 1-13 (Cont.) Customer-Reported Fixes for ECE 12.0 Patch Set 4

Bug Number	Description
32007118	When a session has multiple updates, balance impacts were being applied for each CCR instead of treating it as a single event at the aggregate level. This has been fixed.
32013271	The CCA was returning incorrect GSUs and consumed reservations were not updated correctly because of how unratedNetQty was evaluated. This has been fixed.
31958554	Primary member data balances were not accessible for processing data usage CCR sessions that were dependent on the owner balance within a shared group. This has been fixed. Owner balances are now accessible in both post and prerating extensions.
32046630	Publishing customer updates requests to ECE was failing and ECE nodes were disjoining from the cluster and becoming inaccessible. This was because of improper initialization of the CQC key cache for public user identity (PUI). This has been fixed.
32002289	Subscriber tracing was not capturing SNR/RAR notification messages as expected. This has been fixed.
32003506	Incremental loading of customers was failing. This was because a SharingCustomerLock that existed when a customer was in a transaction was not removed automatically when the serving node went down. This has been fixed.
31897119	The replication of customer accounts between active and passive sites was showing a difference in relative account size between the primary and secondary ECE sites. This has been fixed.
31878546	The customerLoader process was running a query with a hard-coded value causing the generation of new SQL IDs and a new plan each time the query was run. This was impacting the customer migration process from legacy systems to BRM. This has been fixed. The query now uses bind variables instead of literals.
31630751	After changing a product's rate plan, ongoing data usage transactions were resulting in DIAMETER_UNABLE_TO_COMPLY messages instead of continuing with rating and committing the ongoing usage. This has been fixed.
31550821	When a recurring bundle was purchased or canceled during a session, ECE was rating the subsequent CCR as though the TTC condition were set to true, even if there was no real TTC. This has been fixed.
31535141	When Tariff-Time-Change AVP was enabled and triggered, Granted Service Units (GSU) of 0 was sent in response to UPDATE requests. This has been fixed.
31395496	In an ongoing sessions CCA was not being sent by ECE for a CCR-U request. This was causing the network to send a terminate request as a next request, abruptly ending the session that was in progress. This has been fixed.

Table 1-13 (Cont.) Customer-Reported Fixes for ECE 12.0 Patch Set 4

Bug Number	Description
31692727	The Diameter Gateway was not returning GSU in CCR-U replies, causing a mismatch of balances between the policy and charging rules function and ECE on the network side. This has been fixed.
31785258	The ActiveSession cache was not being replicated properly after switching the primary role from one ECE site to another. This has been fixed.
31769583	Changing the credit floor limit was not being updated in ECE correctly, which was impacting the subsequent rating and reservation behavior. This has been fixed.
31630756	ECE was failing to reserve remaining balance after CCR-I initial grant, affecting the processing of subsequent CCR requests. After receiving the first CCR-U, the USU was recorded correctly but the ActiveReservation in BalanceCache was incorrectly set to zero, which affected the filling of GSU in the CCA reply. This has been fixed.
31630747, 33108039	After a service transfer from one account to another, usage occurring after the transfer was incorrectly rolling up to the old account. This has been fixed.
31605626	Running the post_install.pl script after installing ECE 11.3 patch set 9 was giving the following error: <code>ORA-24006: cannot create QUEUE, PIN_2.ECE_SUSPENSE_QUEUE already exists.</code> This has been fixed.
31785256	An exception occurred during cache data federation. This has been fixed.
31666068	During the billing process, during diameter usage processing, the Diameter Gateway was throwing an <code>Unable to Comply</code> error. This has been fixed.
31824183	The ECE customer query had hard-coded literals instead of a bind variable, which caused issues while migrating customers into ECE. This has been fixed.
31501025	Reauthorization requests (RARs) were failing when there were multiple rating groups in the same session. This has been fixed.
31413905 31719731	The tariff time change was not working if the next month's balance was not available. This has been fixed.
31232536 31690795	Even after a billing account was changed for a non-paying child, usage charges were still being rolled up to the earlier billing account. This has been fixed.
31899413	With Final Unit Action as Final Unit Indicator (FUI)/REDIRECT, the Quota Holding Time (QHT) Attribute-Value Pair was missing in the CCA response. This has been fixed.

Table 1-13 (Cont.) Customer-Reported Fixes for ECE 12.0 Patch Set 4

Bug Number	Description
31296934 31782712	The selective subscriber tracing functionality was not capturing Sy SNR and Gy RAR messages. This has been fixed.
31872639	Conditional policy impacts were routed to the user balance instead of the sharer balance, even when the sharer balance was configured to be used. This has been fixed.
32092505	Account creation was failing because ECE was parsing the customer POID as an integer instead of a Long. This has been fixed.
32130386	Incremental synchronization of accounts was not working after an upgrade from ECE 11.3 Patch Set 9 Interim Patch 12 to ECE 12.0 Patch Set 3. This has been fixed. During incremental loading, you no longer have to provide -DpreDistributedWorkItems .
32184754	In ECE, the active session objects cleanup feature was causing performance degradation. This has been fixed.
32602507	When the database is down, ECE caches rated events in the Coherence cache, then pushes them to the database when it comes back up. Rated Event Formatter (REF) was skipping some cached events, which were eventually purged without being added to the BRM database. This has been fixed. Now, after the database comes back up, REF pauses processing for a new configurable time period, giving all of the cached events time to be pushed to the database. See "Configuring Rated Event Formatter" in <i>ECE Implementing Charging</i> for information about the parameter that controls this time period.
32551846	Active-active systems required a single REF instance on a single site that was used by all sites. This represented a potential single point of failure. This has been fixed. Now there is an REF instance at every site to format rated events locally, then disperse them to other sites. If an instance fails at one site, formatting fails over to a different REF site. See "Configuring an Active-Active System" in <i>BRM System Administrator's Guide</i> for information about active-active ECE systems.
N/A	Purchase or cycle pre-rated events passed from external systems into ECE for loading into BRM with Rated Even Loader (REL) include additional details like effective date-times and G/L data. Previously, only usage events were loaded by REL, so these details were ignored by ECE and not passed on to BRM. This has been fixed.

Customer-Reported Fixes in ECE 12.0 Patch Set 3

Table 1-14 lists the customer-reported issues that were resolved in ECE 12.0 Patch Set 3.

Table 1-14 Customer-Reported Fixes for ECE 12.0 Patch Set 3

Bug Number	Description
29960660	The Final Unit Indicator (FUI) was not triggered for the successive update requests into Diameter Gateway, after the FUI was sent once in the response for an earlier request. This has been fixed.
29990703	The Pricing Updater was throwing an exception when processing offer profile information. This has been fixed.
30013014	When BRM Gateway and BRM reconnection was interrupted, the BRM Gateway suspense queue was blocked. This has been fixed.
30018071	The same Public User Identity (PUI) was appearing multiple times in notifications. The data structure has been updated to include the PUI only once in notifications.
30018075	ECE notifications did not include the event time stamp, which is required for BRM opcode calls. This has been included now.
30085070	When ECE evaluated an item selector, the results were incorrect. This has been fixed.
30120617	While creating accounts in BRM, there were error messages in EM Gateway. Post-commit validations were introduced in ECE to make sure of data integrity on both BRM and ECE systems.
30175643	It was noticed that both Validity Time and Granted Units were zero for some usage responses. This has been fixed.
30191582	In ECE, customer data map entries were not properly serialized, leading to a NullPointerException. This has been fixed.
30293167 30715647	ECE did not trigger SNR notifications when balances or counters were updated on the core BRM server for billing or product purchase. This has been fixed.
30293175	For a sharing hierarchy, AR actions (payments, adjustments) on payee accounts were not updating the balances correctly in ECE.
30293184	The time zone information in usage records was not considered in Diameter Gateway. This affected the time model evaluation during the rating phase in ECE, resulting in incorrect charging.
30378999	The consumption rules information was not handled in ECE. This has been fixed.
30560539	BRM Gateway requests from ECE to BRM were defaulted to BRM schema 1. This has been fixed to use the correct schema using account POID as the routing POID.
30577374	When a subscriber was having active sessions, the delete account operation was failing. This has been fixed.
30596817	In ECE, EM Gateway was throwing timeout error, during updates from BRM to ECE.
30599578	In ECE, when more than one discount was available for rating, reverse rating was not working properly. This has been fixed.
30681648	In Diameter Gateway, SLR messages were not retrieving the user identity based on the multiple options available in AVP. This has been fixed.
31007224	During rating, a new method to fetch the product details based on effective time of call from audit was not available. This has been fixed.

Customer-Reported Fixes in ECE 12.0 Patch Set 2

Table 1-15 lists the customer-reported issues that were resolved in ECE 12.0 Patch Set 2.

Table 1-15 Customer-Reported Fixes for ECE 12.0 Patch Set 2

Bug Number	Description
28499167	If there were multiple threshold events, then the custom data map was incorrect. This has been fixed.
29252596	With the discount sharing enabled in the system, the customer eviction was not working properly. This has been fixed.
29269723	The values stored for session initiate request of a session were not available during the update request. This has been fixed.
29297126	The latency information within External Manager (EM) Gateway were displayed incorrectly in the log files. This has been fixed.
29352741	For a long running session with split scenario, there were rounding issues with different scaling on charging and discounting. This has been fixed.
29352848	There were threading issues that were generating errors in the usage or update processing flows. This has been fixed.
29361881	The update processing flow was throwing a "no valid product" exception. This has been fixed.
29383215	Modifications to subbalance validity was not handled properly. This has been fixed.
29415796	Diameter Gateway was losing precision while processing integers that were big. This has been fixed.
29422243	If there was any change in the impact category, such as the user moving from one country to another in the same zone, the daily bundle was granted even if it was already granted. This has been fixed.
29428394	For usages of 1KB, the rounding was not proper due to precision and scaling issues. This has been fixed.
29479716	The threshold breach notification was not including all Public User Identities (PUI) from the service alias list. This has been fixed.
29531402	During the usage processing, reverse rating was not working correctly when multiple discounts were evaluated. This has been fixed.
29543203	During the usage processing, discount evaluation was incorrect when multiple discounts were involved. This has been fixed.
29634128	Data in ECE cache was lost due to the growth of memory with unnecessary replication of Subscribe-Notifications-Requests (SNR). This has been fixed.
29646046	When processing usage, if there were multiple threshold breaches in the same session, the aggregated information of those breaches were not populated properly in the notification. This has been fixed.

Table 1-15 (Cont.) Customer-Reported Fixes for ECE 12.0 Patch Set 2

Bug Number	Description
29647134	The closed user group (CUG) evaluation of a customer who was earlier in a CUG was incorrectly done. This has been fixed.
29669600	When processing update requests, the rating profile was corrupted. This has been fixed.
29674445	When service balance group transfer was done within the same account, the item portal object IDs (POID) list and the next item POID list were not synchronized properly. This has been fixed.
29763898	Diameter Gateway was expecting proxyable bit in the command flag in credit control requests and this was causing Diameter Gateway to respond with an error DIAMETER_INVALID_HDR_BITS (3008) when the command flags were simply 0x80 for the request. This has been fixed and now Diameter Gateway does not validate for the proxyable bit in the command flags when processing credit control requests.
29764019	When a balance impact was null for a rated event, the database was generating a unique constraint violation. This has been fixed by marking the rated event as invalid.
29782159	In an ECE deployment with beat configured, the reverse rating was not considering the splits. This has been fixed.
29808711	A balance update request was failing with an exception during Subscribe-Notifications-Request (SNR) generation. This has been fixed.
29823002	During the rolling upgrade, the tax code configuration data updated through Java Management Extensions (JMX) was lost. This was because only the charging-setting.xml was updated, but the updates were not reflected in the appconfiguration cache. This has been fixed.
29843327	The DiameterGy extension was throwing an exception on a top-up request. This has been fixed by adding sufficient checks.
29960659	During usage processing, a post File Update Information (FUI) update was trying to reserve further units causing DIAMETER_CREDIT_LIMIT_REACHED status to be returned to the network. This has been fixed.
30013017	Any updates to Oracle Communications Billing and Revenue Management (BRM) such as Life Cycle notifications from ECE to BRM were stuck in the Java Message Service (JMS) queue. The BRM Gateway was not able to send them to BRM due to connection pool issues. The BRM connections were not released back to the pool leading to non-availability of connections for any further requests. This has been fixed.
30175640	During usage processing, both validity time and granted service units were derived from adjusted rateable usage metric (RUM) quantity instead of the range-end. This has been fixed.
30177988	During usage processing, threshold notification was sent out based on reservation amount instead of actual consumed amount. This has been fixed.

Customer-Reported Fixes in PDC

See the following for a list of customer-reported issues that have been fixed in each Oracle Communications Pricing Design Center (PDC) patch set:

- [Customer-Reported Fixes in PDC 12.0 Patch Set 8](#)
- [Customer-Reported Fixes in PDC 12.0 Patch Set 7](#)
- [Customer-Reported Fixes in PDC 12.0 Patch Set 6](#)
- [Customer-Reported Fixes in PDC 12.0 Patch Set 5](#)
- [Customer-Reported Fixes in PDC 12.0 Patch Set 4](#)
- [Customer-Reported Fixes in PDC 12.0 Patch Set 3](#)
- [Customer-Reported Fixes in PDC 12.0 Patch Set 2](#)
- [Customer-Reported Fixes in PDC 12.0 Patch Set 1](#)

Customer-Reported Fixes in PDC 12.0 Patch Set 8

[Table 1-16](#) lists the customer-reported issues that were resolved in PDC 12.0 Patch Set 8.

Table 1-16 Customer-Reported Fixes for PDC 12.0 Patch Set 8

Bug Number	Description
34787291	It was observed that when both the base and the derived event were modified in the same changeset, if a new field is added into the base event, the field is not being propagated into the supportedConfig object for the derived event. This has been fixed. When creating the supportedConfig object for a derived event, changes are now made to get the base event data from the same changeset before retrieving them from the PDC Server cache. The newly added field will be propagated to the derived event.
34740335	When a PDC user with a Pricing Reviewer role logged in to PDC, the user was able to view the Create Terms option. This has been fixed.
34700672	When upgrading to Patch Set 7, the customer-defined tax codes were not getting created in PDC, resulting in a missing data error when exporting the configuration data. Also, when deleting tax codes using the PDC UI, the default tax codes were removed from PDC but not from BRM. This has been fixed. Tax code configurations are mastered in PDC. Users need to create a BccTaxCode configuration for each customized tax code in the PDC system before exporting configuration data. In the RRE Transformation Engine, remove the deleted PDC tax code from the XREF table as well as the input tax code XML file used for loading into the BRM system.
34679033	During export using the -expRefAll option, if the number of child objects to be fetched is more than 1000, it was causing an ora-01795 exception. This has been fixed. During the export operation, if the number of child objects is more than 1000, the list is broken into batches of size 1000 and fetched from database so that the IN clause size is within the permitted range.

Customer-Reported Fixes in PDC 12.0 Patch Set 7

Table 1-17 lists the customer-reported issues that were resolved in PDC 12.0 Patch Set 7.

Table 1-17 Customer-Reported Fixes for PDC 12.0 Patch Set 7

Bug Number	Description
34352501	After upgrading PDC from 7.5 to 12.0 Patch Set 5, some charge offers could not be updated. This occurred because Patch Set 5 added a new configuration for the prorate cycle but the XREF database upgrade script was populating the wrong value as the default value. This has been fixed.
34278066	After upgrading PDC from 7.5 to 12.0, an issue occurred with discount quantity expressions because not all expressions were supported. This has been fixed. PDC validation now restricts quantity expression options only for the BRE pipeline. Restrictions were removed for ECE.
34124976	Some PDC scripts still referenced the transPLData.jar file even though it was removed from the code in a previous patch set. This has been fixed.

Customer-Reported Fixes in PDC 12.0 Patch Set 6

Table 1-18 lists the customer-reported issues that were resolved in PDC 12.0 Patch Set 6.

Table 1-18 Customer-Reported Fixes for PDC 12.0 Patch Set 6

Bug Number	Description
33733603	While creating a USC Selector, the Usage Type column was not displayed when the Convergent Usage profile was selected. This has been fixed.
33381695	When a plan transition entry was removed from PDC, it was not getting deleted from the TRANSITION_T table in BRM. This has been fixed.
33617669	Intermittently, objects were out of order in the payload, which resulted in the model selector not getting populated. This has been fixed.
33439274	Adjustment items created during rerating were not allocated to any bill even when the AllocateReratingAdjustments business parameter was enabled. This occurred because, in the PDC create charge offer page, the default value in the Stop Charging list was When Inactive and When Cancelled . This has been fixed. The default value in the Stop Charging list was changed to Never .

Table 1-18 (Cont.) Customer-Reported Fixes for PDC 12.0 Patch Set 6

Bug Number	Description
33921641	When a user logged in to PDC with the Pricing Reviewer role, a NullPointerException occurred when trying to open promoted Package objects. This occurred because there is no active changeset in Reviewer mode, but PDC was trying to access the active changeset while opening Package objects. This has been fixed. Accessing the active changeset when running in Reviewer Mode is now avoided.
33931861	If an impact category was deleted and then re-created, it could not be used in a zone model. This issue was caused by PDC not fetching the latest zone result items. This has been fixed.

Customer-Reported Fixes in PDC 12.0 Patch Set 5

Table 1-19 lists the customer-reported issues that were resolved in PDC 12.0 Patch Set 5.

Table 1-19 Customer-Reported Fixes for PDC 12.0 Patch Set 5

Bug Number	Description
33374664	PDC was not accepting monthly cycle forward and cycle arrears charges in the same charge offer. This has been fixed.
33171060	The BRE transformation engine was creating duplicate DiscountMaster for reusable PDC chargeSelectorSpec. This has been fixed.
33552021	When the CleanXrefDiscountMaster.sql script was run to clean up duplicate discountMasters in the BRE Transformation XREF table, it took 14 hours and still had not completed. This occurred because there were 120,000 discountMasters in the XREF table, and one of the discountMasters had 116,000 duplicated discountMasters. There was not enough memory in the original implementation to maintain this list of duplicated discountMasters for reporting. The script has been fixed.
33434673	BRM balance elements supported seven rounding modes, but PDC balance elements supported only four of them. PDC did not support the following rounding modes: <ul style="list-style-type: none"> • ROUND_FLOOR • ROUND_FLOOR_ALT • ROUND_DOWN_ALT This has been fixed.
33185277	StepC and StepQ are the only valid options for a cascading discount. Validations to enforce these options have been implemented.
33205981	TotalC and TotalQ were not supported in QuantityRange expressions. This has been fixed.
33001081	In discount offers, a customer was able to configure only integer values and unable to configure decimal values to increment the step(beat) field. This has been fixed.

Table 1-19 (Cont.) Customer-Reported Fixes for PDC 12.0 Patch Set 5

Bug Number	Description
32968407	<p>For a balance impact, it is possible to specify whether an impact applies to the Sharer's or Owner's balance groups. The request was for a similar ability when selecting balances for use in triggers and rules. For example, to permit the use of the following functors:</p> <ul style="list-style-type: none"> • Balance[Resource]: For a shared discount, get the Sharer balance of a resource. • MemberBalance[Resource]: For a shared discount, get the Member balance of a resource. <p>By permitting the expression to directly access a Member resource, it would no longer be necessary to create multiple discount rules for each resource that might be used. This could save hundreds of rules.</p> <p>To implement this, PDC has implemented the below features:</p> <ol style="list-style-type: none"> 1. Member Balance: A new Boolean flag has been added to balance expressions that can be used to indicate to ECE that a member balance needs to be considered while evaluating the expression rather than the owner balance. 2. Max and Min Functors: Two new functors were added for finding the maximum and minimum of two different expressions.

Customer-Reported Fixes in PDC 12.0 Patch Set 4

[Table 1-20](#) lists the customer-reported issues that were resolved in PDC 12.0 Patch Set 4.

Table 1-20 Customer-Reported Fixes for PDC 12.0 Patch Set 4

Bug Number	Description
32931796	<p>The SyncPDC process was failing on the BRM side when there were multiple G/L segments involved.</p> <p>This has been fixed</p>
32708512	<p>Updating a bundle was failing. This was because validation rules expected a deliverable object.</p> <p>This has been fixed. The validation rules no longer expect a deliverable object if the existing charge offers in the bundle do not have one.</p>
32645457	<p>While doing package transitions in PDC, when a transition rule is deleted and saved, it was still appearing in the user interface, but when the changeset was submitted, the transition rule would disappear. This was because the content wasn't being refreshed after deleting the transition rule.</p> <p>This has been fixed. There is now a force refresh after a transition rule is deleted and the package is saved.</p>

Table 1-20 (Cont.) Customer-Reported Fixes for PDC 12.0 Patch Set 4

Bug Number	Description
32172885	<p>While creating pricing objects, the following errors occurred in the PDC server log file: Error when in XmlValidationErrorHandler cos-nonambig: subscriptionTerms and subscriptionTerms (or elements from their substitution group) violate "Unique Particle Attribution".</p> <p>Although this had no functional impact in creating or modifying pricing objects, the error occurred because the subscription terms definition was included twice in the PricingObjects.xsd.</p> <p>This has been fixed.</p>
32164981	<p>Importing a zone map was failing due to a mismatch between transformation and PDC. The map was being created in PDC but failing in Transformation, so the transaction was rolled back and the entries for the map were deleted. The next time the zone map was submitted, PDC sent it as a MODIFY object, and Transformation could not find the object in the database.</p> <p>This has been fixed. Transformation now does a lookup to search for the object. If the object isn't found, Transformation treats the MODIFY transaction as CREATE.</p>
31955647	<p>When running PDCBRMWalletUtil.sh to change the wallet password for BRM Integration pack, the script was failing with the following error: Error: Could not find or load main class jvmopts.</p> <p>This has been fixed . The script now uses the bin/bash UNIX shell instead of /bin/sh.</p>
31583707	<p>PDC did not support taxation at the balance impact level for charges.</p> <p>This has been fixed.</p>
31583703	<p>PDC did not support tax time in rating and billing at the balance impact level for charges.</p> <p>This has been fixed.</p>
31511626	<p>PDC migration was failing to migrate discount model configurations.</p> <p>This has been fixed.</p>
31452654	<p>PDC did not support tax codes at the balance impact level for discounts.</p> <p>This has been fixed.</p>
31385700	<p>PricingDesignCenter.sql was failing because of an issue with ProvisioningTagAttrSpecId.sql.</p> <p>This has been fixed.</p>
31332903	<p>It was not possible to update a zone model in PDC. This was because during PDC migration, multiple entries in the BRM zone model were created when multiple service codes were associated with a single BRM service.</p> <p>This has been fixed.</p>
31257734	<p>When adding a new GLID to BRM, SyncPDC reported a validation error for a GLID name change, stating that the name was already used by another GLID object in PDC. This is because PDC was using the description field for GLID, so if multiple GLIDs had same description, there would be duplicate GLID names.</p> <p>This has been fixed. Now migration/syncpdc uses the name that you enter for the GLID instead of copying it from the description.</p>

Table 1-20 (Cont.) Customer-Reported Fixes for PDC 12.0 Patch Set 4

Bug Number	Description
31223178	Modifying charges for products with rate plan selectors was resulting in errors. This was because for migrated charge selectors, PDC prorate values (PRORATE_CHANGE, PRORATE_FULL, PRORATE_NONE) were set instead of BRM prorate values (prorate, full, none). This has been fixed.
31213200	During PDC migration, charge offer creation was failing. This was because required RUMs were missing, and the BRM RUM name was used instead of the PDC RUM name. This has been fixed. The migration scripts now check for required RUM names and PDC RUM names are used.
31124410	XML import for products was failing if the external ID in the XML doesn't exist in target system. This has been fixed. If external IDs are missing during XML import, they are created and added into the ExternalID table.
29770708	Filters were missing on the zone model screen of the PDC user interface. There was a long list of zone rules in the zone model and sorting the list timed out. This has been fixed.
31513293	When updating the pricing-price model code, PDC was overriding code with the internal ID of the migrated objects. This has been fixed.
31670528	PDC displays an <code>ADF_FACES-60097 Error</code> when trying to display some entries in the service-event map. To prevent this, click Save after adding or deleting each object in the Service-Event Map user interface.

Customer-Reported Fixes in PDC 12.0 Patch Set 3

[Table 1-21](#) lists the customer-reported issues that were resolved in PDC 12.0 Patch Set 3.

Table 1-21 Customer-Reported Fixes for PDC 12.0 Patch Set 3

Bug Number	Description
30312020	When migrating batch rating events (BRE) to PDC, migration changed the SERVICE_CODE_USED and SERVICE_CLASS_USED BRE fields to the PDC SERVICE_CODE and SERVICE_CLASS fields. However, there was no mapping information being maintained in the PDC SupportedConfig. Because of this, the rating engines did not start. This has been fixed.
30920955	You could not sort zone models in the PDC GUI because, while sorting data, PDC always retrieved it from the database. This has been fixed. PDC now maintains data in memory on the first retrieval and then updates it during create/update.

Customer-Reported Fixes in PDC 12.0 Patch Set 2

Table 1-22 lists the customer-reported issues that were resolved in PDC 12.0 Patch Set 2.

Table 1-22 Customer-Reported Fixes for PDC 12.0 Patch Set 2

Bug Number	Description
29199299	If the startDate in the bundleProductOffering was immediate , the timestamp in the deal may be off by -1 hour. This caused failures while loading the deal into BRM. This has been fixed.
29526136	In Pricing Center, it was possible to configure pro-ration settings for each of the charges under a charge selector. This was missing in PDC. This has been fixed. Now, the Charge Selector Configuration menu is available with options for Proration Settings and Remove Charge Selector . To configure the proration details, click Proration Settings .
29590589	While using the function balance[Temporary Resource] , an error was displayed in the PDC user interface. This has been fixed.
29635057	There were multiple issues while migrating real-time rating engine (RRE) and batch rating engine (BRE) data of BRM to PDC. BRE used multiple delayed events for the same service. Multiple PDC discount models created the discount model used by the discount selector and issues related to Plan and Deal transitions. This has been fixed.
29661212	User was not able to delete a ratable usage metric (RUM) because it was referenced in the older versions of the pricing and config objects. This has been fixed. Now, it is possible to delete the older versions of the referenced objects also, along with the RUM.
29825919	Pipeline rating was going wrong for the noncurrency resources. Earlier, PDC was transforming the balance element's numeric code as the resource name to the BRM system, which was not the correct implementation. This has been fixed in migration and transformation to transform the balance element's code as the resource name instead of numeric code. This fix is relevant to anyone using PDC migration and transformation.
29867480	While configuring roll over charge offers using the PDC user interface, errors were displayed. This has been fixed.
30150356	If a zone model is referenced in a pricing object, such as charge offer, and a new zone model is created, the user is unable to obsolete or delete any zone model either through the user interface or ImportExportPricing utility. This has been fixed.

Customer-Reported Fixes in PDC 12.0 Patch Set 1

Table 1-23 lists the customer-reported issues that were resolved in PDC 12.0 Patch Set 1.

Table 1-23 Customer-Reported Fixes for PDC 12.0 Patch Set 1

Bug Number	Description
20881534	<p>When the Save as option was used to copy a component, it was not copying only the component instead it copied the internally referenced components along with the component.</p> <p>This has been fixed.</p>
21575971	<p>After importing setup components, there were no rows created in certain tables which was causing issues.</p> <p>This has been fixed.</p> <p>When you install BRM, its installation scripts populate the sample data, including setup and pricing components. When using PDC, this data should not be initiated in BRM, but it should be defined in PDC and published to the BRM database. To ensure that PDC and BRM are integrated properly, you must clean up the BRM sample data before you start using PDC. See the discussion about cleaning the sample data in <i>PDC Installation Guide</i>.</p>
22488798	<p>The ImportExportPricing utility with the -d (delete) parameter worked only for obsoleting a setup component and only if the component was not referenced by any other component. After the component was made obsolete, it could not be used by any other components.</p> <p>This has been fixed.</p>
25091591	<p>Import of metadata was failing with a constraint violation if PDC was integrated with ECE.</p> <p>Recreate the index without COMPRESS ADVANCE LOW to fix this issue.</p>
25932012	<p>SyncPDC was failing if /event/realtimeDiscount class existed in the BRM database.</p> <p>Delete this class from the BRM database to fix this issue.</p>
27040318	<p>Syncpdc was failing during the transformation of the event if the usage class length was more than 4K, the maximum size of the column.</p> <p>This has been fixed.</p>
27756978	<p>PDC Weblogic Server managed server was logging verbose Eclipse Link messages (up to the FINE level) in the Weblogic Server log (nohup.out). The FINE level was not appropriate for Production environments.</p> <p>This has been fixed.</p>
27840554	<p>The BRM Integration Pack installation was failing because of certain privilege issues. A SYSDBA user credential was required to proceed.</p> <p>This has been fixed. The PDC Installer now accepts any user with database privileges not restricting to the SYSDBA user in the System Database Credentials screen.</p>
27881855	<p>When a rateable usage metric (RUM) was disassociated from all services and events in the service-event map, importing the RUM into PDC caused NULL pointer exception in the real-time rating transform engine.</p> <p>This has been fixed.</p>
27925018	<p>For the same recurring charge product definition, the rate was generated as fixed amount in the case of pricing center but as scaled amount in PDC. PDC actually supports fixed amount for recurring charges.</p> <p>This has been fixed.</p>
27963192	<p>When SyncPDC was run after changing the length of a field in the storable class associated with a custom event in BRM, the change was not reflected in the event definition in the PDC database.</p> <p>This has been fixed.</p>

Table 1-23 (Cont.) Customer-Reported Fixes for PDC 12.0 Patch Set 1

Bug Number	Description
28094045	The high-volume PDC tables were not getting purged which was leading to performance issues. This has been fixed.
28171359	For chargeshares, the General Ledger ID was migrated as Undefined instead of Not Set. This has been fixed.
28262068	The USC_MAP was case sensitive whereas the ZONE_RESULT was case insensitive. This has been fixed. The USC_MAP is also case insensitive

Customer-Reported Fixes in Billing Care

See the following for a list of customer-reported issues that were fixed in each Oracle Communications Billing Care patch set:

- [Customer-Reported Fixes in Billing Care 12.0 Patch Set 8](#)
- [Customer-Reported Fixes in Billing Care 12.0 Patch Set 7](#)
- [Customer-Reported Fixes in Billing Care 12.0 Patch Set 6](#)
- [Customer-Reported Fixes in Billing Care 12.0 Patch Set 5](#)
- [Customer-Reported Fixes in Billing Care 12.0 Patch Set 4](#)
- [Customer-Reported Fixes in Billing Care 12.0 Patch Set 3](#)
- [Customer-Reported Fixes in Billing Care 12.0 Patch Set 2](#)
- [Customer-Reported Fixes in Billing Care 12.0 Patch Set 1](#)

Customer-Reported Fixes in Billing Care 12.0 Patch Set 8

[Table 1-24](#) lists the customer-reported issues that were resolved in Billing Care 12.0 Patch Set 8.

Table 1-24 Customer-Reported Fixes for Billing Care 12.0 Patch Set 8

Bug Number	Description
34921603	When accounts had multiple bill units, the Settled Dispute dialog box was displaying incorrectly in the Newsfeed. This issue has been fixed.
34836116	There was a requirement to validate the payment date before a payment reversal was performed. The PCM_OP_BILL_REVERSE opcode returns the result as success or fail. This has been fixed. The transaction is stopped when the opcode's result is Fail.
34815110	The Payment Details dialog box was not displaying the Move Payment to Suspense link when payments were re-allocated through Payment Suspense. This has been fixed.

Table 1-24 (Cont.) Customer-Reported Fixes for Billing Care 12.0 Patch Set 8

Bug Number	Description
34960177	Bills were incorrectly displaying A/R actions and applied payments. This has been fixed. Billing Care now identifies whether a refund is from a payment or an adjustment.
34782990	Opening an account resulted in a "Search template too big" error message due to a large search template being used in the Generate a Bill Unit Summary REST endpoint. This has been fixed.
34801290	Billing Care incorrectly displayed that some child accounts were paid by the partner and not the reseller. This has been fixed. It now shows the correct paid by account for child accounts in multilevel hierarchy flows.
34787371	CSRs with read-only permission were not able to view invoices in Billing Care. This has been fixed.
34658103	The Newsfeed displayed a blank page when you clicked Recurring Charges . This has been fixed.
34647372	The Billing Care events table was displaying time in the local timezone when the event object had no timezone offset value set. You can now configure the event table's display timezone by using the customConfiguration.xml file.
34608840	While viewing rate customizations, Billing Care displayed an "Invalid service information" error message. This occurred when PIN_FLD_RATE_TIER was present at ARRAY [1] instead of ARRAY [0] in the /rate_plan object of the purchased product. This has been fixed.
34537316	The Billing Care Unallocated Payments dialog box was displaying junk values. This has been fixed.
34594332	Accounts with a large number of children were not opening due to a "Search template too big" error while fetching bill units. This has been fixed. The search template and API calls that fetch bill unit data were optimized.
34574762	In the Bill Search results table, even when the balance due is 0, the days overdue is shown as 0. This has been fixed.
34543790	Refunds were shown as reverse payments in the Payment Details dialog box. This has been fixed. The Payment Details dialog box now shows the refunded amount and audit trails for refund actions.
34523780	When the service login details were updated, the details were not synchronizing with ECE even though it was synchronized to the database. The call to the PCM_OP_CUST_UPDATE_SERVICES opcode was modified to update any type of service that generates a login event, which helps the synchronization with ECE.
34658110	Attempting to click on a disabled negative balance was calling the Open Negative Transfer Balance dialog box. This has been fixed.

Table 1-24 (Cont.) Customer-Reported Fixes for Billing Care 12.0 Patch Set 8

Bug Number	Description
34417594	Batch payments were getting created with a payment channel ID of zero for each payment, regardless of what was passed in the PMT file. This has been fixed. A new Locale header has been added to the PMT file for fetching the payment channel ID from the correct locale. By default, it uses en_US . The channel name string should be passed under the Channel field for each payment so that it sets the corresponding channel ID while collecting payments.
34396022	A bill with zero balance was showing as overdue in the Newsfeed. This has been fixed. A new check verifies that only amounts greater than 0 are displayed as overdue in the Newsfeed.

Customer-Reported Fixes in Billing Care 12.0 Patch Set 7

[Table 1-25](#) lists the customer-reported issues that were resolved in Billing Care 12.0 Patch Set 7.

Table 1-25 Customer-Reported Fixes for Billing Care 12.0 Patch Set 7

Bug Number	Release Notes
34175445	In a multischema BRM deployment, Billing Care was not displaying all bill units in an organization hierarchy. This has been fixed to show all bill units from all database schemas in an organizational hierarchy.
34131767	In the Billing Care Batch Payment interface, the Active tab did not display processed records correctly. This has been fixed.
34119136	For child accounts, the Billing Care Bills tab displayed the parent account POID rather than the parent's account number. The paid by parent account number was displayed correctly for bills in progress, but all previous bills displayed the parent account POID. This has been fixed.
34109642	Issues occurred when creating an inactive service in an account. This has been fixed.
34109640	When the Billing Care REST API was used to create a service profile, the effective date was set to the created date. This has been fixed. The Billing Care REST API now supports profile creation with a specific effective date passed in the request payload.
34095400	When using the search template to retrieve specified account numbers, the template was returning all accounts from the BRM database. This has been fixed.
33981515	When clicked twice, the Billing Care Newsfeed link was re-directing to the Balances page. This has been fixed.

Customer-Reported Fixes in Billing Care 12.0 Patch Set 6

Table 1-26 lists the customer-reported issues that were resolved in Billing Care 12.0 Patch Set 6.

Table 1-26 Customer-Reported Fixes for Billing Care 12.0 Patch Set 6

BUG Number	Description
33894625	The Billing Care REST API could not add or delete multiple aliases for a service. This has been fixed. The Billing Care REST API has been enhanced to support the addition and deletion of aliases.
33726114	In Billing Care, percentage adjustments were not working with bill amounts over 1,000. This has been fixed to show the proper value when the adjustment amount is more than 999.
33726113	The Pay screen in the Billing Care account creation wizard was blank after navigating back to the Configuring screen. This has been fixed.

Customer-Reported Fixes in Billing Care 12.0 Patch Set 5

Table 1-27 lists the customer-reported issues that were resolved in Billing Care 12.0 Patch Set 5.

Table 1-27 Customer-Reported Fixes for Billing Care 12.0 Patch Set 5

Bug Number	Description
33570044	A customer was not able to customize the account creation flow in Billing Care. This has been fixed. A customization hook was introduced to enable customizing the account creation flow in Billing Care, including an SDK sample.
33535652	The account creation REST API did not include an option for setting the account number in the input JSON. This has been fixed. A customization hook has been provided to enable customers to customize the account creation flow in Billing Care.
33512579	In batch payment cards, the View File link was navigating to a 401 error instead of displaying the file contents. This has been fixed.
33497816	Localized strings for custom labels defined under the <code>custom_extensions</code> group in custom XLF files were not being picked up by the Billing Care English resource bundle. This has been fixed.
33488572	The Billing Care bulk refund template was not fetching the events across child accounts. This has been fixed. The template now supports the POID type as input for search criteria and for creating a proper search template input flist when join is not specified in the filter criteria of XML.
33459924	Custom fields were not being displayed in the Billing Care Payment Details screen. This has been fixed.

Table 1-27 (Cont.) Customer-Reported Fixes for Billing Care 12.0 Patch Set 5

Bug Number	Description
33438993	In the Billing Care Make Payment dialog box, custom date fields for custom payment methods were being displayed in UNIX timestamp format. This has been fixed. The UNIX timestamp is now converted to the proper date format for display in the Billing Care UI.
33358070	When you selected View Invoice PDF for an account stored in a secondary schema, the invoice displayed only hard-coded strings. This has been fixed.
33139786	Billing Care was not able to fetch purchased products in the Subscription section for accounts in the secondary schema. This issue has been fixed. Billing Care now calls the PCM_OP_GLOBAL_SEARCH opcode rather than the PCM_OP_SEARCH opcode for multischema systems.
33105909	In Billing Care, purchasing a deal that had a two-month validity period was not working because PIN_FLD_END_T in /purchased_product and /purchased_discount was getting set to NULL. This has been fixed.
33655028	Hierarchies were not showing all child accounts in multischema systems. This has been fixed.
33560836	The Billing Care REST API was reporting errors when creating and updating service profiles. This has been fixed.

Customer-Reported Fixes in Billing Care 12.0 Patch Set 4

[Table 1-28](#) lists the customer-reported issues that were resolved in Billing Care 12.0 Patch Set 4.

Table 1-28 Customer-Reported Fixes for Billing Care 12.0 Patch Set 4

Bug Number	Description
30528415	The DataModelGenerator was not generating the schema for a custom payment object for a custom batch template. This has been fixed. Billing Care now generates schemas for custom payment objects with standard /payinfo objects and events.
32434965	The Billing Care REST API returns the same generic error regardless of whether an internal server error or a business error occurred. This has been fixed. The Billing Care REST API now returns more descriptive error messages and response codes.
32434963	In the Billing Care user interface, the history page of batch payments displays multiple lines of duplicate results for the same processed file. This has been fixed.
32230733	The Billing Care user interface was loading the assets page and was getting services on large databases. This was because the query was retrieving all schedule objects. This has been fixed. The query now only retrieves selected schedule objects.

Table 1-28 (Cont.) Customer-Reported Fixes for Billing Care 12.0 Patch Set 4

Bug Number	Description
32078942	In the Billing Care user interface, when disputing an event with an amount greater than 1000, the dispute amount and reason codes were not displaying. This has been fixed.
32052513	In the Billing Care user interface, searching for a suspended payment, assigning the payment to an account, and adding comments were all slow when there were a large number of custom fields. This has been fixed.
32618809	In the Billing Care user interface, validity dates were incorrectly shown in the date pickers for noncurrency balances. Instead of displaying the actual valid-to end date, the current date was always shown as the default date. This has been fixed.
31452531	In the Billing Care user interface, searching by bill number and account number was not displaying the correct results or sometimes not displaying any results. This has been fixed.
31082441	Billing Care was not correctly displaying pages loaded in an iFrame using Chrome browser up to version 80. This has been fixed. Billing Care now supports adding additional attributes to application cookies in the SDK.
31037692	In the Billing Care user interface, when importing a payment file to perform a batch payment, the page was not refreshing automatically even when the batch payment file was processed successfully. The page had to be reloaded manually. This has been fixed.
31245611	In the Billing Care user interface, when trying to allocate a suspended payment to an account with multiple open bills, not all of the bills were displayed in the Allocate payment page. This has been fixed.

Customer-Reported Fixes in Billing Care 12.0 Patch Set 3

[Table 1-29](#) lists the customer-reported issues that were resolved in Billing Care 12.0 Patch Set 3.

Table 1-29 Customer-Reported Fixes for Billing Care 12.0 Patch Set 3

Bug Number	Description
29351464	There was a requirement to switch between a summary and detailed view of account balances in the Billing Care Home tab. This is now supported through the Billing Care SDK.
29833415	Billing Care now supports timezones that automatically adjust for Daylight Saving Time (DST) changes. To do so, use the brmserver.timezone registry entry in the Billing Care SDK Configurations.xml file. By default, the entry is empty and Billing Care uses the WebLogic server's timezone.
29837457	In the Billing Care Bills tab, the hyperlink from a child account to its parent account was not working. This has been fixed.

Table 1-29 (Cont.) Customer-Reported Fixes for Billing Care 12.0 Patch Set 3

Bug Number	Description
30292975	In Billing Care, the account screen did not load when an account had more than 200 bill units attached to it. This occurred because the search template exceeded the maximum character range. This has been fixed by optimizing the search template.
30332380	The Billing Care REST API did not return the required security headers as part of the server response. This has been fixed.
30385889	The Account Profile dialog box in Billing Care displayed an account's country as Afghanistan even though the country was set to Sweden. This has been fixed.
30418719	In Billing Care, it is now possible to: <ul style="list-style-type: none"> • Disable the backdated feature for the Account Creation, Purchase Package, and Purchase Bundle screens. • Perform a backdated product purchase in the Purchase Bundle screen.
30541405	It was not possible to view a bill-level adjustment done through a custom opcode in the Billing Care GUI. This has been fixed. The Billing Care GUI has been enhanced to show bill-level adjustments done through custom opcodes.
30555807	The Notes section in the Billing Care Adjustments screen displayed an incorrect date. This has been fixed.
30761675	When attempting to set up a custom payment method in Billing Care, an exception occurred and the payment method was not displayed in the Billing Care GUI. This has been fixed.
30825689	When .pmt files were imported for processing batch payments, Billing Care kept processing the file until the page was reloaded. Thus, the page had to be reloaded every time you wanted to process a new batch payment. This has been fixed.
30829223	Billing Care displayed an incorrect net amount when you selected or deselected a negative amount. When a positive and negative amount was involved, Billing Care summed the two amounts rather than taking the difference. This has been fixed.
31012975	After a Chrome browser was updated to v80, the Billing Care screens did not load due to SameSite cookie enablement. Billing Care has been enhanced to support adding attributes to application cookies by using the Billing Care SDK. For example, a customer can use this new SDK capability to add the SameSite attribute to application-level cookies.

Customer-Reported Fixes in Billing Care 12.0 Patch Set 2

[Table 1-30](#) lists the customer-reported issues that were resolved in Billing Care 12.0 Patch Set 2.

Table 1-30 Customer-Reported Fixes for Billing Care 12.0 Patch Set 2

Bug Number	Description
27764510	In the Allocate Adjustment and Payment Details dialog boxes, the columns were not resizable for simple and advanced views. This has been fixed and the columns can now be resized.
28466526	In News Feed, when the Show Payment in Suspense link was clicked for a suspended payment, a payment suspense search page opened in a new browser tab. This has been fixed and clicking the link now opens the suspended payment record.
28543516	Terminating or Inactivating discounts for a service was throwing an error. This has been fixed.
28624766	In Billing Care Asset view, clicking Show Rate Customization was throwing an error, if the number of products purchased was more than 150. This has been fixed.
28664596	There was an issue with amount formatting and rounding for Singapore dollar currency. This has been fixed.
28671904	Billing Care is now enhanced to support account number and bill number instead of the corresponding Portal object IDs (POID) in the flows where Billing Care URLs are embedded in the external applications. See " Support for Account Number and Bill Number in Billing Care Embeddable URLs " for more information.
28671910 28853673	It was not possible to log in to Billing Care from a new browser tab using a bookmarked URL, if there was a browser tab already logged in with Billing Care. This has been fixed.
28707428	The Today date filter was not working in the payment suspense flow. This has been fixed and the payment suspense search template has been corrected.
28764136	Billing Care is now enhanced so that the JavaScript files changes are not required in the account number sorting during account search implemented through software development kit (SDK).
28768197	There was an issue while displaying the suspended payment details when the Move, and View in Payment Suspense link was clicked in the payment details page. This has been fixed.
28806804	Posting multiple comments (notes) was causing connection issues. This has been fixed and the connection is released to the free pool on completion of note update.
28806805	Some of the dependent classes for generating reports were not shipped as part of the REST WAR files package. This has been fixed and the missing dependencies are now included.
28837041	The connection to BRM was lost after timeout exception was thrown for the operations wrapped under a transaction. This has been fixed.
28854728	Transfers between billable credit and debit items were treated as accounts receivable. This resulted in incorrect breakup of charges under the Bills tab. This has been fixed.
28905269	While navigating to accounts receivable action details, after selecting a refund type accounts receivable action, the Back button was not working and its label was missing. This has been fixed.
28905272	It was not possible to install Billing Care REST in silent mode. This has been fixed.

Table 1-30 (Cont.) Customer-Reported Fixes for Billing Care 12.0 Patch Set 2

Bug Number	Description
28912852	Billing Care now displays a tool-tip in the Bill ID field of the following dialog boxes: Payment Details, Payment Details Item Affected, Account Adjustment, and Account Adjustment Details .
28912854	The Cancel link was not working for the notes in the Event Adjustment Details dialog box. This has been fixed.
29048170	When switching between bills, there was an issue with the Bills tab Show/Hide link. This has been fixed.
29161373	The Purchase Package and Purchase Deal flows are now supported as embeddable URLs. See " Embeddable URLs for Purchase Package and Purchase Bundle Screens " for more information.
29161376	Billing Care is now enhanced to display bills that were generated before the account was moved under a hierarchy. See " View Bills Generated Before Moving the Account to a Hierarchy " for more information.
29161377	It was not possible to enter special characters for the Billing Care Wallet Password field. This has been fixed.
29178999	For the bills in progress, wrong dates were displayed in the Item Adjustment dialog box. This has been fixed and the current billing cycle dates are displayed.
29272365	The Refund Bill dialog box was displaying an error that the amount was greater than 999.99. This has been fixed.
29272368	There were performance issues with the bill details view. It was taking more time to load particularly for the flows involving Billing Care URLs embedded in other applications. This has been fixed.
29290979	The collection actions scheduled dates were one day older than the actual due date. This has been fixed to show the correct scheduled dates.
29290981	If an account is created using customer center with a country, for example USA, and the same account was opened in Billing Care, a different country was displayed in the account profile details. This has been fixed.
29323429	There were alignment issues with the labels of the Tax Setup dialog box under Financial Setup. This has been fixed.
29349021	Discounts which were overridden as part of product purchase were not displayed in the breakup of the Original Charge or Discount or Net details in the Events dialog box. This has been fixed.
29376636	It is now possible to display custom error messages on the Billing Care user interface when purchasing a bundle.
29376640	It was not possible to link purchase reason and notes to a particular add-on purchase event. This has been fixed. A new method has been introduced, which retrieves the list of all the purchase events that can be used in the Billing Care SDK to assign to the notes.

Table 1-30 (Cont.) Customer-Reported Fixes for Billing Care 12.0 Patch Set 2

Bug Number	Description
29395718	To improve the performance of the Assets user interface, lazy loading of the assets has been implemented. By default, limited service information is retrieved and when the customer service representatives expand the particular service card, the additional service and offer details are retrieved.
29428269	The Bill in progress text was displayed twice in the Events page header. This has been fixed.
29435523	After installation, Business Operation Center was opening with a blank screen. This has been fixed.
29441732	Overdue days were shown incorrectly as the last configured period of the aging buckets. This has been fixed to show the actual overdue days. A new popup has been added to list the age of the overdue balances as per the aging bucket configuration.
29441738	Reason code was missing in the make payment flow. This has been fixed and the reason code is added now.
29493386	The Event Adjustment dialog box can now be customized to separate both Adjust the Amount field and Adjust the Percent field. Also, there was an issue with tax only adjustment. This has been fixed.
29511807	There was a performance issue while retrieving users who had access to the Payment Suspense screen because all the configured users were queried. This has been fixed and it is now possible to query only users associated with a specific role who has the required screen access.
29520395	The Billing Care SDK is now enhanced to support implementing the logic to restrict the deal list based on the role of the customer service representative. See <i>Billing Care SDK Guide</i> .
29629576	The Billing Care SDK is now enhanced to support implementing the logic for filtering deals list to display manual discount deals only. See <i>Billing Care SDK Guide</i> .
29629577	Object serialization error was logged in the Oracle Entitlements Server client weblogic logs. This has been fixed.
29645139	The future dated add on purchase were purchased with wrong dates. This has been fixed.
29649643	Billing Care is now enhanced to display the external user names (login user) in News Feed to clearly highlight who has made changes on the account or have performed any account receivables actions. See <i>Billing Care SDK Guide</i> .
29705003	It is now possible to customize Billing Care to include start date and end date based filtering for add on deal purchase.
29759310	In the Billing Care Item Adjustment dialog box, it is possible to see the total amount for which adjustment is done. But it does not display how much other amount was adjusted against specific item. This has been fixed.
29759312	To improve performance, Billing Care has been enhanced to disable direct click on the Assets view from the Home tab through customization. It is now possible to search through service ID to go to the Assets page which display only services associated with the search criteria.
29789695	The Oracle Entitlements Server obligation check was throwing an exception. A null value check on adjustment amount when percentage only adjustment is done using customization was not introduced to avoid this exception. This has been fixed.

Customer-Reported Fixes in Billing Care 12.0 Patch Set 1

Table 1-31 lists the customer-reported issues that were resolved in Billing Care 12.0 Patch Set 1.

Table 1-31 Customer-Reported Fixes for Billing Care 12.0 Patch Set 1

Bug Number	Description
27375507	In Billing Care, it was not possible to save the notes in the Payment Details dialog box when the amount was more than 999 and when the payment was added by using an opcode. This has been fixed.
27423362	When an attempt was made to open details using the account number hyperlink in the Payment Suspense Page Audit trail, it was opening the incorrect Suspense Account page. This has been fixed.
27500263	When a payment was made and then reversed, if a comment was added under reversal section, it was not visible unless the Payment Details dialog box was reopened. This has been fixed.
27698947	In Billing Care, a performance issue was reported while listing the deals during a product purchase operation. This has been fixed.
27722599	In Billing Care, in the Payment Details dialog box, the payment allocation table was not aligned properly. This has been fixed.
27771469	If a bill contains charges from subordinate bill units, the items were getting over allocated. This has been fixed.
27823059	The first column was Name in the Allocate Adjustment screen instead of Account . This has been fixed.
27828889	The Payment Allocation screen displayed the account POID instead of the account number. This has been fixed.
27874943	When a payment was first applied to an account and then moved to suspense and then again allocated to another account and then moved to suspense, the Audit Trail was not updated correctly. It was still showing the first account. This has been fixed.
27874945	While purchasing product/discount, the dates were not displaying correctly if there were timezone differences. This has been fixed.
27924915	Using the Billing Care SDK, it was not possible to customize the purchase deal screen and add a few custom fields. Also it was not possible to display the new fields in the asset display screen, changing the start /end date and change the overridden amount. This has been fixed. See <i>Billing Care SDK Guide</i> .

Table 1-31 (Cont.) Customer-Reported Fixes for Billing Care 12.0 Patch Set 1

Bug Number	Description
27931964	If a bundle was purchased with an override amount, the override amount was not accepted if it included ", ". This has been fixed.
27938642	The date in the Payment Reversal page after posting a comment was not displayed correctly unless the page was refreshed. Also, the first comment was not visible. This has been fixed.
28206783	Whenever a customization was added to Billing Care, the cache was not getting refreshed automatically. The user had to clear the browser cache and relogin. This has been fixed.
28220688	The override amount was not being retained when the product status was changed to active. This has been fixed.
28261963	The search results in the Payment Suspense screen was not displaying all the records as per the set pagination limit. This has been fixed.
28431427	The Account Details page for an account with multiple bill units was taking more time to open which was leading to time out. This has been fixed.
28543514	When a credit adjustment was applied to a bill with no dues, the Event Adjustment column in the adjust event actions screen was showing only the tax part of the event adjustment instead of the full adjustment amount. This has been fixed.
28543965	The DataModelGenerator utility was not generating custom XSD for custom objects available in BRM. This has been fixed.
28543979	When Billing Care submitted the batch payments through Lockbox file, the entire file was rolled back if it included under payments. This has been fixed.
28543981	Account details were not being displayed in the Billing Care Collections Page after implementing custom collections scenario and customizing the policy opcode. This has been fixed.

Customer-Reported Fixes in Business Operations Center

Table 1-32 lists the customer-reported issues that were resolved in the Oracle Communications Business Operations Center 12.0 patch sets.

 **Note:**

No customer-reported issues were resolved in Business Operations Center 12.0 Patch Set 5, Patch Set 6, and Patch Set 7.

Table 1-32 Customer-Reported Fixes for Business Operations Center

Bug Number	Patch Set	Description
34853383	8	It was possible for users to see all jobs in the Jobs page even when the required access was not granted. This has been fixed. Users can now see only allowed categories in the Jobs page, Job Timeline view, Job History page, and Home page.
34504354	8	The View failure report link was enabled for custom jobs when the /process_audit/billing object was populated in BRM for that job. In other cases, Business Operations Center doesn't show the link for custom jobs.
34504352	8	The Job History page was not loading when a /process_audit/export_gl object was created. This has been fixed.
32981684	4	The Business Operations Center installer now supports a pre-created Business Operations Center user. With this enhancement, you can create a Business Operations Center database user and tablespace, and then use these details in the installer to create the schema.
31954955	4	When installing Business Operations Center on an environment using RAC Database 19c, the silent installation was failing while running the boc_schema.sql script. This is fixed and the installer has been enhanced to use a pre-created database user and setup the Business Operations Center schema on the same. With this enhancement, you can now create the Business Operations Center database user and tablespace and use the same details in the installer to setup the BOC schema.
28602921	2	The SERVICE_HOST parameter in Business Operations Center configuration properties file restricted login with IP address when the parameter was set as hostname, and restricted login with hostname when the parameter was set as IP address. This has been fixed. Dependency on the SERVICE_HOST parameter has been removed to construct REST URL.
29033987	2	The Business Operations Center silent installer has been enhanced to update wallet with Secure Shell user and BRM connection keys.
29290982	2	During validation, the Business Operations Center installer was not accepting special character for passwords in the Wallet field. This has been fixed. Now, special characters in addition to numbers and characters are accepted.
29699403	2	The Business Operations Center failure report was showing Account Portal object ID (POID) instead of the account number. This has been fixed and the report now shows the account number.

Customer-Reported Fixes in BRM Thick Clients

Table 1-33 lists the issues reported by external sources for Oracle Communications Billing and Revenue Management (BRM) thick clients and provides a brief description of the resolution.



Note:

No customer-reported BRM thick client issues were resolved in 12.0 Patch Set 4, Patch Set 5, Patch Set 6, and Patch Set 7.

Table 1-33 Customer-Reported Fixes for BRM Thick Clients

Bug Number	Patch Set	Description
34629290	8	The Event Browser Custom Search template was not working. The extra fields RateTimezonId and TimezonId were included in PCM_OP_GLOBAL_STEP_SEARCH for every search in Event Browser. Some custom template searches may not include these fields in the main class and thus the search was not working. This has been fixed. The two fields are included in search results only when a search is done on /event or /event/% .
34962905	8	Customer Center was displaying duplicate entries for disputes in the Unresolved Disputes page. The "wrong" entry did not contain details on the Dispute Details section. This has been fixed.
30439709	3	In the Customer Center Services tab, CSRs with read-only permission could not copy service IDs. This is now enabled through a customization option.
30727930	3	The Windows version of Payment Tool 12.0.0.2.0 threw an error message when users attempted to log in. This has been fixed.
30146959	3	In the Customer Center Payments tab, when searching for payments and selecting All in the Bill Unit list, Customer Center displayed only payments related to a specific bill unit. This has been fixed.
30562624	3	UEL templates could not be modified in Developer Center. This was caused by issues with the XML4J parser and was fixed by changing it to a JDK parser.

2

New Features in BRM

The Oracle Communications Billing and Revenue Management (BRM) 12.0 Patch Sets include many new features.

Topics in this document:

- [New Features in BRM 12.0 Patch Set 8](#)
- [New Features in BRM 12.0 Patch Set 7](#)
- [New Features in BRM 12.0 Patch Set 6](#)
- [New Features in BRM 12.0 Patch Set 5](#)
- [New Features in BRM 12.0 Patch Set 4](#)
- [New Features in BRM 12.0 Patch Set 3](#)
- [New Features in BRM 12.0 Patch Set 2](#)
- [New Features in BRM 12.0 Patch Set 1](#)

New Features in BRM 12.0 Patch Set 8

BRM 12.0 Patch Set 8 includes the following enhancements:

- [BRM Supports Flexible Proration Options](#)
- [Support for Conversion to Wholesale Hierarchy](#)
- [Sharing Groups Now Support Multiple Schemas](#)
- [Discount Offer Stacking Support](#)
- [Pre-Expiry/Post-Expiry Notification Support](#)
- [Kafka DM Enhancements](#)
- [Real-time Balances from ECE Available As a SOAP Web Service](#)
- [Price-tag-based Price Override Support in REST Services Manager](#)
- [Two-phase Billing Support in REST Services Manager](#)
- [BRM Supports Two-Way SSL Authentication](#)
- [Rated Event Manager Support for CDR Streaming](#)

BRM Supports Flexible Proration Options

In previous releases, you could configure time stamp rounding only at a systemwide level by using the **timestamp_rounding** entry in the CM **pin.conf** file. You could specify that all time stamps were either:

- **Rounded to midnight.** This meant that product validity periods started at midnight, even if it was purchased later in the day. In this case, purchases of recurring deals would be

valid starting at midnight, causing any delayed usage that occurred between midnight and the purchase time to be incorrectly consumed from the new grant.

- **Set to the purchase time.** This meant that charges had to be prorated for the first day of the billing cycle. For example, if a product was purchased at 15:00 for a 30-day billing cycle, the customer was charged for 29 days and 9 hours.

To make proration more flexible and to avoid any balance validity or charging issues, BRM now allows you to create a product with separate time stamp rounding values for the following:

- **The validity period:** You can specify that the validity period starts at the purchase time or at midnight of the day the product is purchased. Alternatively, you can specify to use the systemwide setting in the CM **pin.conf** file.
- **The charging scale:** You can specify whether to charge for a full day or a partial day for the first day of the billing cycle. That is, if a product is purchased at 15:00 on 5 May, the customer can be charged for a full day or 9 hours for 5 May. Alternatively, you can specify to use the validity period setting.

You can configure the proration settings by using the **loadpricelist** utility or the **PCM_OP_PRICE_SET_PRICE_LIST** opcode.

To configure proration settings using an XML file and the **loadpricelist** utility, set these new XML elements under the **<product>** element:

- **<offer_validity_rounding>**: Specifies whether to start the product's validity period at the purchase time or at midnight of the purchase day.
 - **OFF**: Starts at the time of purchase. This overrides the CM **pin.conf** setting at the product level.
 - **ON**: Starts at midnight (00:00:00) of the day that the product is purchased. This overrides the CM **pin.conf** setting at the product level.
 - **NOT_SET**: Uses the systemwide setting in the CM **pin.conf** file. This is the default.
- **<scale_rounding>**: Specifies whether to charge for a full day for the first day of the recurring cycle:
 - **OFF**: Calculate it based on the **<offer_validity_rounding>** setting.
 - **ON**: Calculate it based on full days.

To configure proration settings using the **PCM_OP_PRICE_SET_PRICE_LIST** opcode, set the following input field fields:

- **PIN_FLD_OFFER_VALIDITY_ROUNDING**: Specify whether to start the product's validity period at the purchase time or at midnight of the purchase day:
 - **0**: Not Set. Uses the systemwide setting in the CM **pin.conf** file. This is the default.
 - **1**: On. Starts at midnight (00:00:00) of the day that the product is purchased. This overrides the CM **pin.conf** setting at the product level.
 - **2**: Off. Starts at the time of purchase. This overrides the CM **pin.conf** setting at the product level.
- **PIN_FLD_SCALE_ROUNDING**: Specify whether to charge for a full day for the first day of the recurring cycle:

- **OFF**: Calculate it based on the `PIN_FLD_OFFER_VALIDITY_ROUNDING` setting.
- **ON**: Calculate it based on full days.

For more information, see "Setting Full Day Proration" in *BRM Pipeline Rating and Discounting*.

Support for Conversion to Wholesale Hierarchy

You can now convert an existing hierarchy (with a paying parent account and non-paying child accounts) into a wholesale hierarchy. As part of this feature, a new multithreaded application (`pin_cust_convert_wholesale_hierarchy`) and a new opcode (`PCM_OP_CUST_CONVERT_WHOLESALE_HIERARCHY`) are introduced to perform the conversion. See "Converting Existing Bill Unit Hierarchies to Wholesale Billing" in *Configuring and Running Billing* for more information.

Sharing Groups Now Support Multiple Schemas

By default, all members of a sharing group must reside in the same database schema. To include a member from a different schema, you must first migrate the member to the same schema as all other members of the sharing group.

You can now enable the following sharing groups to include members from multiple schemas:

- Charge sharing groups (`/group/sharing/charges`)
- Discount sharing groups (`/group/sharing/discounts`)
- Profile sharing groups (`/group/sharing/profiles`)
- Product sharing groups (`/group/sharing/products`)

To do so, enable the **CrossSchemaSharingGroup** business parameter in the **system** instance of the `/config/business_params` object. See "Enabling Group Members to Reside in Multiple Schemas" in *BRM Managing Customers* for more information.

Discount Offer Stacking Support

Previously, when a discount was purchased a second time, it was treated as separate from the original purchase. It is now possible to extend the validity of a discount subscription by purchasing the discount multiple times. You can also define a grace period for resubscribing to a discount offer.

You can change the handling for discount repurchases using the field `PIN_FLD_MODE` under `/deal/discounts`. The available values for this field are:

- `PIN_SUBS_PURCHASE_DEFAULT = 0` (This is the default value if not set.)
- `PIN_SUBS_PURCHASE_LONGEST_DATE = 1`
- `PIN_SUBS_PURCHASE_EXTEND = 2`
- `PIN_SUBS_PURCHASE_OVERWRITE = 3`

You can set the grace period using the new `PIN_FLD_GRACE_PERIOD` field. This field has the following attributes:

- `PIN_FLD_GRACE_PERIOD_UNIT`
- `PIN_FLD_GRACE_PERIOD_OFFSET`

If the re-purchase of the offer is done within the defined grace period and the `PIN_SUBS_PURCHASE_OVERWRITE` mode is not in effect, then the same `/purchased_discount` instance will be extended. Otherwise, the normal purchase flow is followed.

For more information, see "Purchasing the Same Product or Discount Multiple Times" in *BRM Configuring Pipeline Rating and Discounting*.

Pre-Expiry/Post-Expiry Notification Support

When setting up your system to send messages to customers through an external notification application when triggering events occur in BRM, you can now:

- Specify whether customers are automatically opted in to receive a specific notification message type.
- Configure BRM to generate notification events a set amount of time before or after a service lifecycle state changes.
- (For in-advance and post-expiration notifications only) Specify whether to aggregate multiple events for the same notification type into one notification message for the customer.

For information, see "Sending Messages to Customers through External Notification Applications" in *BRM Managing Customers*.

Kafka DM Enhancements

You can now run the Kafka DM in one of these modes:

- **Asynchronous mode:** The Kafka DM records in a log file all business events that fail to publish to the Kafka server. You configure the name and location of the log file using the `<KafkaAsyncMode>` element in the `BRM_home/sys/dm_kafka/log4j2.xml` file. Asynchronous mode is the default.
- **Synchronous mode:** When a business event fails to publish to the Kafka server, the Kafka DM rolls back the transaction and returns an error to BRM.

You define the Kafka DM mode using the `BRM_home/sys/dm_kafka/dm_kafka_config.xml` file. For more information, see "About the Kafka DM" and "Editing the `dm_kafka_config.xml` File" in *BRM Developer's Guide*.

You can now also configure BRM to replace dynamic keys in message payloads with a value you specify in the `PCM_OP_PUBLISH_POL_PREP_EVENT` policy opcode. For more information, see "Configuring the Dynamic Key Value" in *BRM Developer's Guide*.

Real-time Balances from ECE Available As a SOAP Web Service

The `PCM_OP_BAL_GET_ECE_BALANCES` opcode, which returns real-time balances for a service from ECE, is now exposed as a SOAP Web Service via BRM Web Services Manager.

For more information, see "About WSDL Files and BRM Opcodes" in *BRM Web Services Manager*.

Price-tag-based Price Override Support in REST Services Manager

BRM REST Services Manager now supports price-tag-based price overrides. Multiple date-range-based price overrides can be passed as part of the automatic offer and service bundle creation.

Two-phase Billing Support in REST Services Manager

BRM REST Services Manager now supports two-phase billing. Multiple date-range-based price overrides can be passed as part of atomic offer and service bundle creation. The initiate and fulfill actions can be sent as two separate orders. The service will be created as part of the initiate order and will be activated as part of the fulfill order.

BRM Supports Two-Way SSL Authentication

You can now set up two-way SSL authentication between the database and the on-premise and cloud native deployment versions of the following: BRM, BRM REST Services Manager, Pipeline Configuration Center (PCC), Pricing Design Center (PDC), Business Operations Center, and Billing Care.

Rated Event Manager Support for CDR Streaming

The Rated Event Manager now supports streaming events to and from Kafka. This enables near-real-time reporting of rated event data and makes rated event data available for consumption from Apache Kafka by third party systems as well as by BRM. See "Event Streaming Mode (Patch Set 8 or later)" in *Loading Rated Events* for more information.

New Features in BRM 12.0 Patch Set 7

BRM 12.0 Patch Set 7 includes the following enhancements:

- [Promise-to-Pay Enhancements](#)
- [Balance Impact Stores Current Balance of Resources](#)
- [Support for Setting Credit Floor Dynamically](#)
- [First Usage Activation Enhancements in BRM](#)
- [Balance Monitoring Enhancements](#)
- [Support for Add-On Products in Deals](#)
- [Sharing Group Improvements](#)
- [Notifications Can Be Triggered After Event Occurrence](#)
- [Proration Can Be Configured at the Product Level](#)
- [Improvements to Tax Handling for Bill Disputes](#)
- [Secure Communication to the BRM Database](#)
- [Running Stored Procedures in BRM](#)

Promise-to-Pay Enhancements

Customers can now pay off bills that are in collections through multiple promise-to-pay installments. The installment schedule and amount can be set up automatically by Collections Manager or manually according to the customer.

You can now also configure whether credit limits are automatically increased when an account enters a promise-to-pay agreement and then decreased when the amount due is paid off.

For information, see "Managing Promise-to-Pay Agreements" in *BRM Collections Manager*.

Balance Impact Stores Current Balance of Resources

BRM now stores the current balance and available loan balance in events. This allows CSRs to look up a customer's available balance at the time of a particular transaction.

The main balance and available loan balance are recorded for both currency and non-currency resources, and for both debit and credit balance impacts.

Support for Setting Credit Floor Dynamically

BRM now allows for setting the credit floor dynamically from the granted sub-balance amounts that are valid for the current cycle. This allows BRM to set the credit floor automatically when customers are granted limited-time resources.

For example, assume you grant a subscriber 10 GB of data using a cycle fee. When the subscriber purchases a booster pack of 5 GB that is valid only during the current cycle, BRM sets the credit floor to 15 GB for the current cycle and then reverts it to 10 GB at the start of the next cycle.

A new field, `PIN_FLD_DYNAMIC_CREDIT_FLOOR`, has been added to the `/config/credit_profile` and `/event/billing/limit` objects to support this functionality.

For more information, see "Setting a Dynamic Credit Floor" and "Enabling Dynamic Credit Floors in Plans" in *BRM Configuring Pipeline Rating and Discounting*.

First Usage Activation Enhancements in BRM

You can now configure a deal or a plan to start on first usage without having to configure each product and discount in the deal or plan to start on first usage. When you use this option, the first usage of any product will activate all of the products in the deal or plan.

You can implement the first usage activation feature by using an XML file and the `loadpricelist` utility. See "Activating Products in Plans and Deals on First Usage" in *BRM Configuring Pipeline Rating and Discounting* for more information.

**Note:**

You can also use PDC to configure first usage activation in bundles and packages. See "[First Usage Activation Enhancements in PDC](#)".

Balance Monitoring Enhancements

BRM now supports a truly real-time balance monitoring group type: payment responsibility real-time credit enforcement (PR_RTCE). Its members include nonpaying child accounts and their services. PR_RTCE also allows you to change the credit limit settings for the group's owner and members to do the following:

- Roll up the credit limits from one or more child bill units in the group to the owner's bill unit. For example, assume a group's owner has a \$1000 credit limit and the three child group members each have \$100 credit limits. If you specify to roll up the credit limits of all three child group members, the group owner's credit limit would change to \$1300.
- Nullify a child's credit limit after it is rolled up to the owner. If you specified to nullify the credit limits in the previous example, BRM would change the credit limit for all three child group members to NULL. If you specified to maintain the credit limits, BRM would keep the credit limit for all three child group members at \$100.

For more information, see "Managing Balance Monitoring Groups" in *BRM Managing Customers*.

Support for Add-On Products in Deals

When you create your price list using the XML Pricing Interface, you can now include add-on products in your deals. All products in deals are base products by default, which means they are automatically included when the customer purchases the deal. Add-on products can be purchased with the deal or later on.

When you create an add-on product, you also specify how to determine its validity start date. The add-on product's validity start date is the end date of a product that you specify. For example, assume product A has a validity period from June 1 through June 15. If you specify to align add-on product B's validity period with product A, product B's validity start date would be June 15.

You can specify that an add-on product's validity dates align with:

- The base product that you specify
- The active base product that expires first
- The active base product that expires last
- The active product that expires first
- The active product that expires last

For more information, see "Configuring Add-On Products in Deals" in *BRM Configuring Pipeline Rating and Discounting*.

Sharing Group Improvements

BRM now supports product sharing groups, which allow a group owner to share a package with all group members automatically. This allows you to change rates once at the group owner level, rather than having to change each of the subscriber accounts.

You can also automate discount sharing in a similar way. You can share discounts at the top level of the hierarchy, with multiple billing accounts at lower levels sharing in the discount. When members are added to the hierarchy, they can be automatically added to the parent group account's discount sharing group.

If the parent account is removed from the hierarchy, all of its member accounts are automatically removed from the hierarchy, and therefore from the discount sharing.

For more information, see "Managing Product Sharing Groups and Discount Sharing Groups" in *BRM Managing Customers*.

Notifications Can Be Triggered After Event Occurrence

BRM can now send notifications after the due date for the following business events:

- **PostBalanceExpiry**
- **PostBillDue**
- **PostProductExpiry**
- **PostSubscriptionRenewalDue**
- **PostCollectionsActionDue**
- **PostInstallmentDue**

This allows you to remind customers to, for example, renew an expired subscription or pay a bill that is one week past due.

For more information, see "Sending Messages to Customers through External Notification Applications" in *BRM Managing Customers*.

Proration Can Be Configured at the Product Level

Previously, you configured proration at the system-wide level. Now, you can also configure in deals whether a product is prorated based on 30 days or on the actual number of days in the month.

You implement proration at the product level by using an XML file and the **loadpricelist** utility. For more information, see "Setting Proration for Products in a Deal" in *BRM Configuring Pipeline Rating and Discounting*.

Improvements to Tax Handling for Bill Disputes

When a partial payment has been made on an account, and a dispute is raised for the remaining amount, it is now possible to avoid charging taxes on the disputed amount.

In the case of tax-only billing disputes, the dispute can be raised against a specific amount of tax, rather than needing to use a percentage.

Secure Communication to the BRM Database

TLS and SSL encryption is now supported between all BRM components and the BRM database, including all Data Managers, ECE, PDC, Business Operations Center, Pipeline Configuration Center, Rated Event Loader, and Rated Event Manager.

Running Stored Procedures in BRM

You can now run BRM stored procedures or your custom stored procedures against the BRM database.

For information, see "Running Stored Procedures" in *BRM System Administrator's Guide*.

New Features in BRM 12.0 Patch Set 6

BRM 12.0 Patch Set 6 includes the following enhancements:

- [Support for PINless Debit Payment Processing](#)
- [Kafka DM Enhancements](#)
- [BRM Notification Enhancements](#)
- [BRM Supports Temporary Credit Limits](#)
- [Customer Deposits](#)
- [Loan Management Enhancements](#)
- [BRM Can Now Track Failed Operations](#)
- [Cloud Native Support for SSO](#)
- [Exposing Charge and Tax Details for Payments and Adjustments](#)
- [Support for More Granular Tax Code Application](#)

Support for PINless Debit Payment Processing

BRM can now support PINless debit payment transactions. To enable this functionality, you set the new **PINlessDebitProcessing** business parameter.

For more information, see "Configuring PINless Debit Payment Processing" in *BRM Configuring and Collecting Payments*.

Kafka DM Enhancements

The Kafka DM includes the following enhancements:

- You can now enable secure communication between the Kafka DM and the Kafka Server.
- You can now customize the PCM_OP_PUBLISH_POL_PREP_EVENT policy opcode to look up or provide delivery identifiers for delivery methods, such as the email address for an email delivery method.
- For each Kafka topic that you define, you can now specify to:
 - Use one of two new styles for messages sent in XML format: **CamelCase** and **OC3CNotification**.

Note: The **Notification** XML style has been deprecated.

- Add headers to each message sent to a Kafka topic.
- Write business events in a separate message payload.
- Specify the key value passed in a message or payload.
- Override the default mapping between a BRM field and an XML or JSON element. This can be done at the topic level or the payload level.

To support this new functionality, the **dm_kafka_config.xml** configuration file has been updated to XML version 2.0. See "Mapping Business Events to Kafka Topics" in *BRM Developer's Guide*.

For more information about the Kafka DM, see "About Integrating BRM with an Apache Kafka Server" and "Configuring BRM to Publish Notifications to Kafka Servers" in *BRM Developer's Guide*.

BRM Notification Enhancements

When configuring your system to send messages to your customers through an external notification application, such as Oracle Communications Convergent Charging Controller, you can now:

- Send messages when the following A/R actions occur: a payment method changes, a payment is successfully processed, a refund is issued, an auto-payment fails, and a payment is reversed.
- Generate notifications in-advance for collections actions and installments.
- Create custom delivery methods.
- Prevent messages from being sent to your customers on certain days of the year (called silent days).
- Add rollover details to **/event/notification/subscription/renewal** notification events before they are sent to the Kafka DM.

In addition, you now specify how long after the delivery time that messages can still be delivered to your customers by using the **AcceptableDelayTime** business parameter. In previous releases, you set this value by using the **acceptable_delay_time** parameter in the CM **pin.conf** file and **pin_gen_notifications pin.conf** file.

For more information, see "Sending Messages to Customers through External Notification Applications" in *BRM Managing Customers*.

BRM Supports Temporary Credit Limits

When creating or modifying an account, you can now add temporary credit limits to a customer's balance of minutes, US dollars (USD), or so on. For example, you could add a temporary credit limit of 400 USD that is valid from 1 November 2022 through 1 January 2023.

If a customer has both a permanent credit limit and a temporary credit limit for the same balance element, the temporary credit limit amount is appended to the permanent credit limit amount. For example, if a customer has a permanent credit limit of \$100 and a temporary credit limit of \$200 from 1 June 2023 through 1 July 2023, the customer's credit limit would be \$300 from 1 June 2023 through 1 July 2023.

You implement temporary credit limits in your custom client applications by using BRM opcodes. For more information, see "Customizing Credit Limits and Sub-Balance Consumption Rules" in *BRM Opcode Guide*.

Customer Deposits

You can create and manage customer deposits for devices, services, packages, or accounts. As part of this feature, you can:

- Create deposit specification profiles and deposit specifications, which define the underlying rules and properties for customer deposits. You can create these in Billing Care or by using the PCM_OP_DEPOSIT_CREATE_SPECIFICATION_PROFILE and PCM_OP_DEPOSIT_CREATE_SPECIFICATION opcodes.
- Create deposits for individual customers using either Billing Care or the PCM_OP_DEPOSIT_PURCHASE_DEPOSIT opcode with a custom client application. Before they can make a deposit, customers purchase a package or bundle containing a charge offer that has a purchase deposit event mapped to the charging rule of the deposit.
- Update, transfer, reverse, release, and refund deposits.

For more information, see "Managing Deposits" in *BRM Managing Customers*.

Loan Management Enhancements

Loan management contains the following enhancements:

- Customers can choose to opt in to or out of receiving loans.
- You can offer loans to customers who have opted in when their balance crosses specified thresholds, rather than automatically granting loans to all customers.
- Customers can purchase entire packages using loans, without impacting their current account balance.
- You can configure:
 - Thresholds for offering loans at the package level for all customers, and at the individual customer level.
 - Due dates for loan repayment.
 - What happens when the loan isn't repaid in time.
 - What to do if a top-up doesn't cover the full amount due.
 - The maximum number of active loans a customer can have.
- Loan balances are now tracked and managed as separate sub-balances from other types of balances.

For more information, see "Configuring Loans" in *BRM Configuring and Collecting Payments* and "Loan Opcode Workflows" in *BRM Opcode Guide*.

BRM Can Now Track Failed Operations

BRM operations may occasionally fail to process completely. For example, a payment could fail due to an insufficient balance or an incorrect account address. You can now configure BRM to store information about multithreaded application (MTA) utilities and custom client applications that fail, so you can view them for analysis and reporting, or reprocess them at a

later time. For more information, see "About Tracking Failed BRM Operations" in *BRM System Administrator's Guide*.

To configure MTA utilities to record failed operations, see "Configuring MTA Utilities to Record Operation Failures" in *BRM System Administrator's Guide*.

To configure a custom client application to record and retrieve failed operations, see "Managing Operation Failure Records" in *BRM Opcode Guide*.

Cloud Native Support for SSO

You can now set up a single sign-on (SSO) login method using SAML 2.0 for the following applications in a BRM cloud native environment:

- Billing Care

For more information, see "Setting Up SSO for Billing Care" in *BRM Cloud Native Deployment Guide*.

- Business Operations Center

For more information, see "Setting Up SSO for Business Operations Center" in *BRM Cloud Native Deployment Guide*.

Exposing Charge and Tax Details for Payments and Adjustments

You can now configure BRM to track how much tax is being settled as a part of a payment. If you configure this feature, the *item* object contains the settled tax amount, the settled taxed amount, and the settled nontaxed amount. This feature supports using the tax percentage in the original event to calculate the tax amount during adjustments. See "Configuring Itemized Tax Information" in *BRM Calculating Taxes* for more information.

Support for More Granular Tax Code Application

You can now configure tax selectors and tax exemption selectors to apply taxes based on account, service, event, and profile attributes. This allows you to choose whether to use the direct tax code or to choose the tax code using the selectors in the rate plan while creating the product. See "About Calculating Taxes" in *Calculating Taxes* for more information.

New Features in BRM 12.0 Patch Set 5

BRM 12.0 Patch Set 5 includes the following enhancements:

- [Top-Up Enhancements](#)
- [Loan Management](#)
- [New Bill States](#)
- [Sample Prepaid Service Life Cycle](#)
- [XML Price List Enhancements for BRM](#)
- [Rerating Noncurrency Resources When Products are Canceled](#)
- [Options for Exceeding Credit Limits](#)

- [New Residency Type to Support Distinct Searches](#)
- [Customizing V\\$SESSION Table Details for dm_oracle and dm_ifw_sync](#)
- [BRM Now Supports Dynamic Charging](#)
- [BRM Now Supports Dynamic Taxation](#)
- [BRM Now Supports Promotions Based on Special Dates, Events, and Actions](#)
- [Sending Messages to Customers through External Notification Applications](#)
- [pin_inv_doc_gen Now Uses JDBC](#)
- [BRM Cloud Native Enhancements in Patch Set 5](#)
- [New Features Supported by BRM REST Services Manager](#)

Top-Up Enhancements

Top-up payments have been enhanced to support:

- Cash and check payment methods
- Recurring standard top-ups
- Topping up noncurrency resources, such as minutes, tokens, or Gigabytes

To be able to top-up noncurrency resources, you must do the following:

- Add the following line to your *BRM_home/sys/data/pricing/example/pin_rum* file and then load it into the database using the **load_pin_rum** utility:

```
/event/billing/topup : Topup_Charge :  
PIN_FLD_TOPUP_RESOURCE_INFO.PIN_FLD_TOPUP_AMT : none
```

- Add the following line to your *BRM_home/sys/data/pricing/example/pin_usage_map* file and then load it into the database using the **load_usage_map** utility:

```
/event/billing/topup : Topup : 0: 0: 0: 0: 0: 0: 0: 0: topup_charge
```

- Add the following line to your *BRM_home/sys/data/pricing/example/pin_event_map* file and then load it into the database using the **load_event_map** utility:

```
/account : /event/billing/topup : Topup_Charge Event
```

For more information, see "Configuring Top-Ups" in *BRM Configuring and Collecting Payments*.

You implement this new top-up functionality in your custom client applications by calling BRM opcodes. For more information, see "Managing Top-Ups" in *BRM Opcode Guide*.

Loan Management

You can grant loans to prepaid customers when their credit is insufficient for rating (dynamic loans), or when their balance falls below a configured threshold (automatic loans).

For more information, see "Configuring Loans" in *BRM Configuring and Collecting Payments*.

You implement loan functionality in your custom client applications by calling BRM opcodes. For more information, see "Loan Opcode Workflows" in *BRM Opcode Guide*.

New Bill States

All **/bill** objects now include the `PIN_FLD_STATE` field, with the following possible values:

- UNDEFINED
- INPROGRESS
- NEW
- PARTIALLYPAID
- SETTLED
- ONHOLD
- SENT
- VALIDATED

BRM automatically sets the bill state to `INPROGRESS`, `NEW`, `PARTIALLYPAID`, or `SETTLED` as part of the bill life cycle.

You can submit requests to the Update a Customer Bill BRM REST Service Manager API endpoint to update the bill state from `INPROGRESS` to `ONHOLD`, then from `ONHOLD` back to `INPROGRESS`.

All bills created before applying patch set 5 have the state set to `UNDEFINED`. You can optionally set them to the other applicable states by using a stored SQL procedure.

For more information, see:

- "About Bill States" in *BRM Concepts*
- "Setting the State for Preexisting Bills" in *BRM Patch Set Installation Guide*
- [REST Services Manager API for Billing and Revenue Management](#)

Sample Prepaid Service Life Cycle

The default `config_lifecycle_states.xml` file, which defines states for a sample prepaid service life cycle, contains a new lifecycle state: **109** (SuspendedActive).

Note:

If your existing state-to-status mapping uses the **109** lifecycle state, you will need to introduce a new macro and recompile the code.

For more information, see "About the Sample Prepaid Service Life Cycle" in *BRM Managing Customers*.

XML Price List Enhancements for BRM

BRM now supports the following enhancements when you define your price lists in XML and upload them using the `loadpricelist` utility:

- **Setting a grace period when purchasing the same product multiple times:** When creating products, you can specify what happens if customers purchase the same product more than once.
See "Purchasing the Same Product or Discount Multiple Times" in *BRM Configuring Pipeline Rating and Discounting*.
- **Granting noncurrency balances in increments:** When you credit noncurrency balances, such as free minutes, in a one-time allotment or in a recurring cycle, you can choose to grant the noncurrency balance in smaller portions on an incremental basis. For example, a grant of 30 Gigabytes with a one-month validity could be distributed as 1 Gigabyte per day.
See "Splitting Noncurrency Balances into Multiple Validity Periods" in *BRM Configuring Pipeline Rating and Discounting*.
- **Configuring how to apply cycle fees when customers transition from one plan or deal to another:** When customers transition from one plan or deal to another in the middle of their billing cycle, BRM, by default, prorates the cycle fees for both the original and new plan or deal. You can now configure BRM to instead apply only the cycle fee for the original plan or deal, or apply only the cycle fee for the new plan or deal.
See "Transitioning Plans and Deals" in *BRM Configuring Pipeline Rating and Discounting*.
- **Configuring how to apply cycle fees when customers change their billing day of month (DOM):** When customers change their billing DOM in the middle of their billing cycle, a partial bill is created. For example, if the billing DOM is changed in June from the 15th to the 30th, a partial bill is created for June 15 through June 29. By default, BRM applies a prorated cycle fee to this partial billing cycle. You can now configure BRM to instead apply the full cycle fee or no cycle fee to the partial billing cycle.
See "Prorating Fees for Billing DOM Changes" in *BRM Configuring Pipeline Rating and Discounting*.
- **Configuring cycle alignment for products in reactivated deals:** When customers suspend and then reactivate a deal, you can specify when the cycle should align for products in the deal. By default, the cycle aligns with either the billing date or the original purchase date, but you can specify that it instead aligns with the reactivation date.
See "Setting Product Cycle Alignment for Reactivated Deals" in *BRM Configuring Pipeline Rating and Discounting*.
- **Specifying whether to stop or continue rating inactive, canceled, or SuspendedActive accounts:** When you create a rate, you can specify whether BRM should continue charging or stop charging accounts that have an Inactive status, a Cancelled status, or a SuspendedActive custom life cycle state.
See "Stop Rating Inactive, Canceled, or SuspendedActive Accounts" in *BRM Configuring Pipeline Rating and Discounting*.

Rerating Noncurrency Resources When Products are Canceled

You can now configure BRM to automatically create rerate jobs when a product with noncurrency resources is canceled. To do so, you use the new **CreateRerateJobDuringCancel** business parameter.

For more information, see "Creating Rerate Jobs for Canceled Noncurrency Resources" in *BRM Rerating Events*.

Options for Exceeding Credit Limits

You can now specify what happens to subscriptions when customers exceed their credit limits.

You can prevent customers from exceeding their credit limit and either prorate the resources according to the available balance, or fail the subscription and notify an external system for further processing.

You can allow customers to exceed their credit limit and:

- Use all available balance and record the remaining amount as an outstanding amount.
- Use all available balance and grant a loan for the remaining amount.
- Leave the available balance and record the entire amount as an outstanding amount.
- Skip billing for this cycle.

You can configure this in the PDC UI or using the loadpricelist utility. For more information, see "Allowing Customers to Exceed Their Credit Limit" in *PDC Creating Product Offerings* and "Allowing Customers to Exceed Their Credit Limit" in *BRM Configuring Pipeline Rating and Discounting*.

New Residency Type to Support Distinct Searches

When searching for objects in the BRM database, because of the way row numbers are generated for complex searches, you cannot use pagination with the order by clause when PIN_FLD_FLAGS is set to perform a distinct search (256 or 768).

A new residency type (9 GLOBAL_DB_VIEW) lets you create custom storable classes and database views specifically for searching. These views create a flat structure for fields that are normally nested multiple levels within a storable class, without creating duplicate copies in the database. This allows you to search more effectively from only the required data, performing distinct searches with ordering and pagination.

See "About Performing Distinct Searches with Ordering and Pagination" in *BRM Developer's Guide*.

Customizing V\$SESSION Table Details for dm_oracle and dm_ifw_sync

You can now customize the Oracle Data Manager (DM) and Account Synchronization DM to populate unique data in the V\$SESSION table in the BRM database. This allows for unique classification and identification of BRM applications by properties set in their database sessions.

To customize the Oracle DM to populate data in the V\$SESSION table, add the following entries to your *BRM_home/sys/dm_oracle/pin.conf* file:

```
- dm vsession_module module
- dm vsession_action action
- dm vsession_client_info clientInfo
- dm vsession_client_id clientID
```

where:

- *module* is the name of the module to run for a specific database session.
- *action* is the name of the action to run for a specific database session.
- *clientInfo* is the information for a specific database session.
- *clientID* is the client identifier for a specific database session.

To customize the Account Synchronization DM to populate data in the V\$SESSION table, add the following entries to your *BRM_home/sys/dm_ifw_sync/pin.conf* file:

```
- dm_ifw_sync vsession_module module
- dm_ifw_sync vsession_action action
- dm_ifw_sync vsession_client_info clientInfo
- dm_ifw_sync vsession_client_id clientID
```

BRM Now Supports Dynamic Charging

You can now configure BRM to dynamically change the rate for one-time and recurring fees per customer based on the date.

For more information, see "Dynamically Changing One-Time and Recurring Fees Based on Date" in *BRM Configuring Pipeline Rating and Discounting*.

BRM Now Supports Dynamic Taxation

BRM now supports dynamic tax calculation, which defers tax calculation until the end of a billing cycle but calculates taxes using the tax rate at the time an event occurred. For example, if an account purchases a product on May 15 and has a billing date of May 30, BRM calculates the tax on May 30 using the tax rate from May 15. This provides the benefits of billing-time taxation while allowing you to change tax rates in the middle of a billing cycle. For more information, see "About Calculating Taxes" in *BRM Calculating Taxes*.

The way you configure tax codes has also changed in Patch Set 5. You now configure tax codes by doing the following:

1. Editing the new *BRM_home/sys/data/config/config_taxcodes_map.xml* file.
2. Running the **load_config** utility to load the tax codes into the */config/taxcodes_map* object in the BRM database.

See "About Creating Tax Codes (Patch Set 5)" in *BRM Calculating Taxes* for more information.

Caution:

In Patch Set 5, you cannot use **SyncPDC** to synchronize tax codes from BRM to PDC. You must instead manually re-create the BRM tax codes in a PDC-compliant XML file and then load them into the PDC database. See "[Moving Tax Codes to PDC](#)".

Previously, you configured tax codes by editing the *BRM_home/sys/cm/taxcodes_map* file. You did not load the file into the database. Instead, BRM stored the contents of the **taxcodes_map** file in its cache.

When you install BRM 12.0 Patch Set 5, it enables the dynamic tax calculation option and starts reading tax codes from the **`/config/taxcodes_map`** object. You can disable dynamic taxation and continue using the `BRM_home/sys/cm/taxcodes_map` file to create tax codes by doing the following:

1. Open the Connection Manager (CM) configuration file (`BRM_home/sys/cm/pin.conf`).
2. Set the value of the **`dynamic_taxation`** entry to **1**:

```
- fm_bill dynamic_taxation 1
```

The default value of **0** enables the dynamic tax calculation option and requires tax codes to be stored in the **`/config/taxcodes_map`** object.

3. Save and close the file.
4. Stop and restart the CM.

BRM Now Supports Promotions Based on Special Dates, Events, and Actions

You can now apply promotions to your customers' accounts based on a special date, a specific event, or a specific action. For example, you could grant 100 free minutes to your customers when they successfully top up their account balance or on their membership anniversary.

For more information, see "Working with Promotions" in *BRM Configuring Pipeline Rating and Discounting*.

Sending Messages to Customers through External Notification Applications

You can now configure your system to send messages to your end customers through an external notification application, such as Oracle Communications Convergent Charging Controller, when a triggering event occurs in BRM. See "About Sending Messages to Customers through External Notification Applications" in *BRM Managing Customers*.

pin_inv_doc_gen Now Uses JDBC

The **`pin_inv_doc_gen`** utility now uses JDBC rather than JNDI for connecting to the BI Publisher database.

In addition, the utility's **`Infranet.properties`** file (`BRM_home/apps/pin_inv_doc_gen/Infranet.properties`) includes the following new entries for configuring the connection pool:

- **`infranet.schedulerdb.url`**: Specifies the scheduler database URL in the following format:

```
jdbc:oracle:thin:hostname:port/service
```

where *hostname* is the hostname of the scheduler database, *port* is the port number for the scheduler database, and *service* is the service name of the scheduler database.

- **infranet.schedulerdb.user**: Specifies the user name for the scheduler database.
- **infranet.schedulerdb.credentials**: Specifies the security credentials for connecting to the scheduler database.
- **infranet.jdbcpool.size**: Specifies the initial number of connections maintained in the pool. The default is set to the same as **burst.threadpool.size**.
- **infranet.jdbcpool.maxsize**: Specifies the maximum number of connections that can be created. The default is set to the same as **burst.threadpool.maxsize**.

For more information, see "Configuring the pin_inv_doc_gen Utility" in *BRM Designing and Generating Invoices*.

BRM Cloud Native Enhancements in Patch Set 5

BRM cloud native deployments include the following enhancements in BRM 12.0 Patch Set 5:

- PDC cloud native services have moved to WebLogic Kubernetes Operator (WKO). This allows PDC to use WKO's features such as provisioning, lifecycle management, application versioning, product patching, scaling, and security. It also enables the use of tooling that is native to the WKO infrastructure for monitoring, logging, tracing, and security. For more information, see "Configuring Pricing Design Center" in *BRM Cloud Native Deployment Guide*.
- Business Operations Center cloud native services include a new **rcuArgs** key for specifying the additional arguments to pass to the Repository Creation Utility (RCU) during the create operation. By default, the value is an empty string, but you can pass flags such as "-honorOMF" when the connected database system uses OMF to manage tablespace files. For more information, see "Adding Business Operations Center Keys" in *BRM Cloud Native Deployment Guide*.
- To simplify the deployment of test and demonstration systems, BRM cloud native now includes a new uniform **role** key that allows you to set the DBA user role for all BRM services at once. For more information, see "Configuring Global Values" in *BRM Cloud Native Deployment Guide*.

New Features Supported by BRM REST Services Manager

You can now use BRM REST Services Manager to make additional TM Forum based REST requests.

BRM REST Services Manager supports the following capabilities of the TM Forum API specifications:

- **TMF 654: Prepay Balance Management API**:
 - Retrieving balance actions, top-up balances, and transfer balances.
 - Creating buckets (balance groups), top-up balances, and transfer balances.
 - Updating bucket thresholds for accounts.
 - Limiting and offsetting results when retrieving buckets
- **TMF 666: Account Management API**: Listener endpoints for creating and updating accounts based on notification events.
- **TMF 678: Customer Bill Management API**: Updating the bill state of a customer bill from inProgress to onHold and from onHold to inProgress.

BRM REST Services Manager now also supports:

- Using numbers in addition to BRM IDs for accounts, bills, bill item, adjustments, and disputes in certain endpoints.
- As an extension, including `originalBalance` values for accumulated balances and buckets.
- Monitoring for on-premises deployments of BRM REST Services Manager using Prometheus and Grafana. See "Monitoring BRM Components" in *BRM System Administrator's Guide*.

For more information, see *REST Services Manager API for Billing and Revenue Management*.

New Method for Loading Rated Events from ECE into BRM

The Rated Event Manager, a new method of loading rated events from Elastic Charging Engine (ECE) into BRM, has been added. This method includes four different operation modes. The primary reason for the new functionality is to allow rated events to be loaded directly into the BRM database without requiring an intermediate shared file store, although optimized modes using file stores are also included.

The modes of the new method are:

- **DIRECT**: This mode loads the data directly from ECE into the regular tables in the BRM database. This is the most direct method, but it uses more bandwidth than the `ZIP_DB` option.
- **ZIP_DB**: This mode creates a ZIP file and stores it in the `/batch/rel` object in the database, where a BRM process loads it into the regular tables.
- **ZIP_FILE**: This mode compresses the CDRs in a transaction into a single file before transferring it to an intermediate file system where it is picked up by BRM.
- **CDR**: This mode is similar to the original method. It transfers individual files to an intermediate file system, where they are picked up by BRM. You might choose this method if you want to read or perform some action on the files before they are picked up by BRM.

For more information about the methods of loading rated events, see "Methods of Transferring Rated Events to BRM" in *BRM Loading Rated Events*.

The RE Manager also includes a utility, `rel_manager`, some of which can be used even if you are not using the RE Manager. For more information about this utility, see "Rated Event Loader Manager Utility" in *BRM Loading Rated Events*.

For more information about configuring this feature, see "Configuring Rated Event Manager" in *BRM Loading Rated Events*.

New Features in BRM 12.0 Patch Set 4

BRM 12.0 Patch Set 4 includes the following enhancements:

- [BRM Cloud Native Deployment Enhancements](#)
- [Monitoring BRM Components](#)
- [BRM Now Supports ASC 606/IFRS 15 Revenue Recognition](#)

- [Deferring Corrective Bills and Invoices](#)
- [BRM Supports Integration with Apache Kafka Servers](#)
- [Integrating BRM with External Care Applications](#)
- [Running Business Operations and BRM Applications from Custom Clients](#)
- [Using Paymentech Card Type Indicator](#)
- [XML Product Offering Enhancements](#)
- [BRM Supports Additional Cipher Suites](#)
- [Web Services Manager Now Supports Oracle Access Manager 12.2.1.3.0](#)
- [Support for TM Forum REST APIs](#)

BRM Cloud Native Enhancements in Patch Set 4

BRM cloud native deployments include the following enhancements in BRM 12.0 Patch Set 4:

- It now supports integration with an external product catalog, such as Oracle Digital Experience for Communications Launch Experience, through the PDC REST Services Manager. See "Configuring PDC REST Services Manager" in *BRM Cloud Native Deployment Guide*.
- It now supports integration with external care applications through the new BRM REST Services Manager. See "Configuring BRM REST Services Manager" in *BRM Cloud Native Deployment Guide*.
- You can now set up the ECE image to persist its cache data in the Oracle database, creating a permanent backup of the cache in case a node fails, a partition is lost, or so on. ECE automatically recovers the cache data from the persistence database when it is needed. See "Enabling Persistence in ECE" in *BRM Cloud Native Deployment Guide*.
- You can now set up the ECE customerupdater Pod to load customer data incrementally into ECE cache. See "Incremental Customer Loading in ECE" in *BRM Cloud Native Deployment Guide*.
- To improve security, the Business Operations Center Pod no longer depends on a root user and passwordless SSH to run business operations jobs. Instead, the Pod now runs jobs by using JavaPCM APIs to run an opcode through the CM. See "Configuring Business Operations Center" in *BRM Cloud Native Deployment Guide*.
- You can now use the Kubernetes Horizontal Pod Autoscaler to automatically scale up or scale down the number of BRM Pod, BRM REST Services Manager, or ECE Pod replicas in your deployment based on a Pod's CPU or memory utilization. See "Setting Up Autoscaling of BRM Pods," "Setting Up Autoscaling in BRM REST Services Manager," and "Setting Up Autoscaling of ECE Pods" in *BRM Cloud Native System Administrator's Guide*.
- You can now use external applications such as Prometheus and Grafana to monitor the system processes of the following BRM cloud native services:
 - BRM. See "Monitoring BRM Cloud Native Services" in *BRM Cloud Native System Administrator's Guide*.
 - ECE. See "Monitoring ECE in a Cloud Native Environment" in *BRM Cloud Native System Administrator's Guide*.

- PDC. See "Monitoring PDC in a Cloud Native Environment" in *BRM Cloud Native System Administrator's Guide*.
 - Billing Care and Billing Care REST API. See "Monitoring and Autoscaling Billing Care Cloud Native" in *BRM Cloud Native System Administrator's Guide*.
 - Business Operations Center. See "Monitoring and Autoscaling Business Operations Center Cloud Native" in *BRM Cloud Native System Administrator's Guide*.
 - BRM REST Services Manager. See "Monitoring and Autoscaling BRM REST Services Manager Cloud Native" in *BRM Cloud Native System Administrator's Guide*.
- You can now rotate your BRM root key on a regular basis. See "Rotating the BRM Root Key" in *BRM Cloud Native System Administrator's Guide*.
 - BRM Server now supports connections to a BRM multischema database. See "Configuring BRM for a Multischema Database" in *BRM Cloud Native Deployment Guide*.
 - ECE Pods now support connections to a BRM multischema database. See "Configuring ECE for a Multischema BRM Environment" in *BRM Cloud Native Deployment Guide*.

To simplify the deployment of test and demonstration systems, BRM cloud native now includes:

- A new uniform password key, named **uniPass**, that allows you to set the password for all BRM services at once. See "Configuring Global Values" in *BRM Cloud Native Deployment Guide*.
- New **isEnabled** keys for enabling or disabling each BRM service in your cloud native deployment. See "Specifying the BRM Services to Deploy" in *BRM Cloud Native Deployment Guide*.

To improve system and application performance, BRM cloud native now allows you to:

- Deploy the Connection Manager (CM) and Oracle Data Manager (DM) containers into the same Pod. See "Deploying the CM and DM Containers in the Same Pod" in *BRM Cloud Native System Administrator's Guide*.
- Tune the number of threads that are available for an application's connection with the CM. See "Tuning Your Application Connection Pools" in *BRM Cloud Native System Administrator's Guide*.

To make it easier to navigate and find information in the BRM cloud native documentation, the *BRM Cloud Native Installation and Administration Guide* has been replaced with two separate books:

- *BRM Cloud Native Deployment Guide*: This book describes how to configure and deploy BRM cloud native.
- *BRM Cloud Native System Administrator's Guide*: This book describes how to perform common system administrator tasks such as running utilities and applications, rotating root passwords, autoscaling Pods, and tuning system performance.

Monitoring BRM Components

You can now monitor the following BRM components by using external applications, such as Prometheus and Grafana:

- Connection Manager (CM)
- Oracle Data Manager (DM)
- Account Synchronization DM
- Synchronization Queue DM
- BRM Java applications, such as Batch Controller and REL Daemon
- Web Services Manager

For more information, see "Monitoring BRM Components" in *BRM System Administrator's Guide*.

BRM Now Supports ASC 606/IFRS 15 Revenue Recognition

BRM now allows you to set up revenue recognition that complies with the Accounting Standards Codification (ASC) 606 and International Financial Reporting Standard (IFRS) 15 accounting standards. In BRM, this is called deliverable-based revenue recognition.

You set up deliverable-based revenue recognition by doing this:

1. Enabling deliverable-based revenue recognition in BRM.
2. Setting up your general ledger.
3. Creating deliverables, which define the goods and services that you sell to your customers.
4. Associating deliverables with your charge offers.
5. Creating subscription terms, which define the commitment periods, cancellation options, and renewal options for your customers' contracts.
6. Creating bundles and packages for the products and services that you sell.
7. Associating your subscription terms with the bundles and packages that you sell to your customers.

When customers purchase your packages and accept the terms, the subscription becomes a contract. You manage your customers' contracts, such as modifying its renewal options or canceling a contract early, by using Billing Care. When their contracts reach the end of the commitment period, the contracts are either canceled or automatically renewed when you run the **pin_contracts** utility.

For more information about:

- Deliverable-based revenue recognition, see "About Deliverable-Based Revenue Recognition" in *BRM Collecting General Ledger Data*.
- Generating reports that comply with deliverable-based revenue recognition, see "Generating General Ledger Reports" in *BRM Collecting General Ledger Data*.
- Managing your customers' contracts, see "Managing Customer Contracts" in *BRM Managing Contracts*.

Deferring Corrective Bills and Invoices

Whenever you make an adjustment to a bill, you also generate a corrective bill and a corrective invoice. You can now defer generating the corrective bill and corrective invoice until after the next regular billing and invoicing period. To do so, you enable the new **ARItemsInCorrectiveInvoice** business parameter.

For more information, see "About Deferred Corrective Invoices" in *BRM Designing and Generating Invoices*.

BRM Supports Integration with Apache Kafka Servers

You can now integrate BRM with Apache Kafka Servers. This allows you to keep data synchronized between BRM and your external applications that are connected to the Kafka server. To synchronize account, pricing, and other data, BRM takes data from internal notification events and constructs a business event that is published to a topic in your Kafka server. Your external applications can then retrieve and process the data from the Kafka topic.

You integrate BRM with a Kafka server and configure it to publish data to the Kafka server by using the Connection Manager (CM), the Enterprise Application Integration (EAI) framework, and the new Kafka Data Manager (DM).

For more information about this feature, see "About Integrating BRM with an Apache Kafka Server" in *BRM Developer's Guide*.

Upgrading from BRM 12.0 Patch Set 4

If you are an existing customer upgrading from BRM 12.0 Patch Set 4 to BRM 12.0 Patch Set 5, you install the Kafka DM on your system by following the instructions in "Installing the Patch Set" in *BRM Patch Set Installation Guide*.

Upgrading from BRM 12.0, BRM 12.0 Patch Set 1, BRM 12.0 Patch Set 2, or BRM 12.0 Patch Set 3

If you are an existing customer upgrading from BRM 12.0, BRM 12.0 Patch Set 1, BRM 12.0 Patch Set 2, or Patch Set 3 to BRM 12.0 Patch Set 4 or later, you install the Kafka DM on your system by doing the following:

1. Install the BRM 12.0 Patch Set 4 or BRM 12.0 Patch Set 5 overlay package (**brmserver_12.0.0.x.0_platform_generic.jar**) on top of your existing system by following the instructions in "Installing the Patch Set" in *BRM Patch Set Installation Guide*.
2. Install the latest version of Apache Kafka. For instructions on downloading and installing Kafka, see "[Apache Kafka Quickstart](#)" on the Apache Kafka website.
For the latest compatible software version, see "BRM Software Compatibility" in *BRM Compatibility Matrix*.

3. Set the KAFKA_HOME environment variable to the path in which the Kafka library JARs are installed.

```
setenv KAFKA_HOME Kafka_path
```

4. Open the *BRM_home/inventory/registry.xml* file in an editor.
5. Remove the existing **BRMServer** and **CORE** sections.

For example, if your existing system is BRM 12.0 Patch Set 3, you would remove this **BRMServer** section from the file:

```
<feature status="installed" name="BRMServer" version="12.0.0.3.0">
  <sessions>
    <session id="1"
date="2021-06-23T03:02:21.969-07:00"action="install"/>
  </sessions>
</components>
```

```

    <component status="installed"
name="BRMServer"version="12.0.0.3.0">
    <sessions>
    <session id="1"
date="2021-06-23T03:02:21.969-07:00"action="install"/>
    </sessions>
    <targets>
    <target qualifier="filegroup1.jar+BRM"source="filegroup1.jar"
symbol="brm.symbol" location="BRM"status="installed">
    <sessions>
    <session id="1"date="2021-06-23T03:02:21.969-07:00"
action="install"/>
    </sessions>
    </target>
    </targets>
    </component>
</components>
</feature>

```

You would also remove this **CORE** section from the file:

```

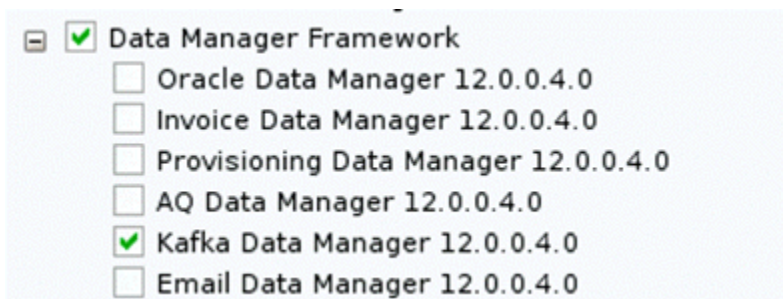
<feature status="installed" name="CORE" version="12.0.0.3.0">
    <sessions>
    <session id="1" date="2021-06-23T03:02:21.969-07:00"action="install"/>
    </sessions>
    <components>
    <component status="installed"
name="CORE"version="12.0.0.3.0">
    <sessions>
    <session id="1"
date="2021-06-23T03:02:21.969-07:00"action="install"/>
    </sessions>
    <targets>
    <target qualifier="filegroup1.jar+"source="filegroup1.jar"
symbol="root.symbol" location="" status="installed">
    <sessions>
    <session id="1"date="2021-06-23T03:02:21.969-07:00"
action="install"/>
    </sessions>
    </target>
    <target qualifier="filegroup2.jar+oui"source="filegroup2.jar"
symbol="oracle.nginst.core.symbol" location="oui"status="installed">
    <sessions>
    <session id="1"date="2021-06-23T03:02:21.969-07:00"
action="install"/>
    </sessions>
    </target>
    </targets>
    </component>
    </components>
</feature>

```

6. Save and close the file.
7. Follow the instructions in "Installing BRM" in *BRM Installation Guide* to install the BRM 12.0 Patch Set 4 or BRM 12.0 Patch Set 5 full build package (**brmserver_12.0.0.x.0_platform_generic_full.jar**), where *x* is **4** or **5**.

When the Installation Type screen appears during the installation process, select the **Custom** installation option to install individual components. In the Feature Sets Selection screen, select **Kafka Data Manager 12.0.0.x.0** under **Data Manager Framework**. [Figure 2-1](#) shows an example of the Feature Sets Selection screen for Patch Set 4.

Figure 2-1 Feature Sets Selection



For more information, see "Installing Individual BRM Components" in *BRM Installation Guide*.

Integrating BRM with External Care Applications

You can now integrate BRM with external care applications by using the new BRM REST Services Manager. This integration allows you to manage customer accounts, bills, payments, and so on in an external care application while using BRM to do the rating and billing.

BRM REST Services Manager includes an API and an SDK. The BRM REST Services Manager API supports the following TMF API specifications:

- TMF 666: Account Management API
- TMF 678: Customer Bill Management API
- TMF 676: Payment Management API
- TMF 654: Prepay Balance Management API
- TMF 635: Usage Management API

The BRM REST Services Manager SDK allows you to extend the framework to support additional API query parameters, payload extensions, and attribute extensions.

For information about:

- BRM REST Services Manager, see *BRM REST Services Manager API*.
- Installing BRM REST Services Manager, see "Installing BRM REST Services Manager" in *BRM Installation Guide*.
- Setting up OAuth authorization for BRM REST Services Manager, see "BRM REST Services Manager Security" in *BRM Security Guide*.

Running Business Operations and BRM Applications from Custom Clients

You can now set up your custom client applications to run business operation jobs, such as billing and invoicing, or other BRM applications such as **pin_virtual_time**.

For more information, see "Setting Up Custom Clients to Run Business Operations" in *BRM System Administrator's Guide*.

Using Paymentech Card Type Indicator

The Paymentech Data Manager (**dm_fusa**) can now request and receive Card Type Indicator (CTI) version 02 information from Paymentech during the authorization process. CTI specifies the type of payment card that is being used by the customer, such as a prepaid card, Visa credit card, or MasterCard credit card. **dm_fusa** automatically supports sending and receiving CTI information in online transactions, but you need to customize **dm_fusa** to support CTI in batch transactions.

For more information, see "Obtaining Card Type Indicator Information from Paymentech" in *BRM Configuring and Collecting Payments*.

XML Product Offering Enhancements

BRM now supports the following enhancements when you define your product offerings in XML and upload them using the **loadpricelist** utility:

- **Creating extended attributes:** You can create extended attributes for charge offers, discount offers, chargeshare offers, bundles, and packages. Extended attributes save extra information that may be useful to external applications. You create extended attributes while creating your pricing components. You can add custom validation using a policy opcode. Although BRM does not act on the extended attributes, the information is stored in the BRM database, and can be queried by external applications. See "Defining Extended Attributes for Pricing Components" in *Configuring Pipeline Rating and Discounting*.
- **Purchasing the same charge offer multiple times:** When creating bundles, you can specify what happens if customers purchase the same charge offer more than once. See "Purchasing the Same Product or Discount Multiple Times" in *Configuring Pipeline Rating and Discounting*.
- **Applying recurring charges on a specific day of the month:** When creating charge offers with hourly validity, you can apply recurring charges on a specific day of the month instead of the customer's billing date or the purchase date. See "Aligning Recurring Charges and Product Validity to a Specific Day of the Month" in *Configuring Pipeline Rating and Discounting*.
- **Using date ranges for versioning:** When creating charge offers, you can add charges with new date ranges to create new versions of the same charge offer and determine whether existing subscriptions move to the new charge or continue with the old charge. Charge offers created before BRM 12.0 Patch Set 4 automatically use the existing functionality, where existing subscriptions move to the new charge. You can update the value of the new **date_range_type** field for the charge offers to change to the new versioning options. See "Using Date Ranges for Versioning" in *Configuring Pipeline Rating and Discounting*.

Note:

PDC also supports these enhancements with the **ImportExportPricing** utility. See "XML Examples of Pricing Components" and "Configuring Extended Attributes for Pricing Components" in *PDC Creating Product Offerings*.

BRM Supports Additional Cipher Suites

BRM now supports additional cipher suites. For more information, see "BRM-Supported Cipher Suites" in *BRM System Administrator's Guide*.

Web Services Manager Now Supports Oracle Access Manager 12.2.1.3.0

Web Services Manager now supports OAuth 2.0 authentication through Oracle Access Manager 12.2.1.3.0. There are also new procedures for setting up OAuth authentication through Oracle Access Manager.

For more information, see "Securing Web Services Manager with OAuth 2.0" in *BRM Web Services Manager*.

Support for TM Forum REST APIs

You can now use BRM REST Services Manager to make TM Forum based REST requests to manage accounts, bills, payments, refunds, balances, and usage in BRM.

BRM REST Services Manager supports the following TM Forum API specifications:

- **TMF 635: Usage Management API**
Retrieves usage information.
- **TMF 654: Prepay Balance Management API**
Retrieves bucket, accumulated, disputed, and adjusted balances. Creates adjusted and disputed balances.
- **TMF 666: Account Management API**
Retrieves billing cycle specifications (bill units and bill unit infos in BRM).
- **TMF 670: Payment Methods API**
Creates, retrieves, and deletes payment methods.
- **TMF 678: Customer Bill Management API**
Retrieves customer bills, rates, and invoices. Creates bills on demand.
- **TMF 676: Payment Management API**
Creates payments, retrieves payment information, and allocates payments to a bill. Creates refunds and retrieves refund information.

BRM REST Services Manager also supports extending the TMF REST framework with Oracle-specific query, request, and response attributes.

For more information, see:

- *REST Services Manager API for Billing and Revenue Management*
- "Installing BRM REST Services Manager" in *BRM Installation Guide*

New Features in BRM 12.0 Patch Set 3

BRM 12.0 Patch Set 3 includes the following enhancements:

- [BRM Cloud Native Deployment Enhancements](#)

- [BRM Server, Thick Clients, and PCC Now Localized](#)
- [Revenue Assurance Supports Custom Applications](#)
- [Pipeline Manager Can Now Support 28-Digit Decimal](#)
- [New Timeout Parameter for Account Synchronization DM](#)
- [Support Added for AES Encryption](#)
- [Enable SSL/TLS between Client and the CM Only](#)
- [Adjusting Taxes by a Specified Amount](#)

BRM Cloud Native Deployment Enhancements

The BRM cloud native deployment now includes containers for the following BRM components:

- BRM Apps Job
- Billing Care REST API
- Configurator Job
- Database Upgrade
- Elastic Charging Engine
- Email Data Manager
- Pipeline Configuration Center
- Roaming Manager
- Web Services Manager

Billing Care and Business Operations Center now support high availability through WebLogic Kubernetes Operator.

The BRM cloud native deployment also includes enhancements that allow you to:

- Automatically pull images from a private Docker registry.
- Automatically roll deployments by using annotations.
- Expose a directory as a CM ConfigMap, so the BRM cloud native deployment can access your custom input files that are outside of a Pod.
- Integrate Business Intelligence (BI) Publisher with the BRM cloud native deployment, so you can generate invoices for your customers.
- Integrate BRM thick clients, such as Customer Center and Pricing Center, with the BRM cloud native deployment.
- Integrate the BRM JCA Resource Adapter with the BRM cloud native deployment, so you can run BRM opcodes from outside of a Pod.
- Make the CM the SSL endpoint for the BRM cloud native deployment. In this case, TLS can be enabled only between BRM client applications and the CM. TLS is disabled between CM and all downstream components such as DMs and EMs.
- Build and deploy Vertex Manager.
- Rotate the BRM root password regularly.

- Use custom TLS certificates to secure connections between the BRM cloud native deployment and external service providers, such as payment processors and tax calculators.
- Run BRM applications and utilities on demand without entering into a Pod.
- Configure and run PDC utilities, such as **SyncPDC** and **ImportExportPricing**, without entering into a Pod.
- Deploy Paymentech Data Manager (dm-fusa) in high-availability mode.
- Use a centralized logging system through an Elasticsearch, Kibana, and Fluentd image stack.
- Upgrade existing BRM cloud native deployments from 12.0 Patch Set 2 to 12.0 Patch Set 3.
- Upgrade existing PDC cloud native deployments from 12.0 Patch Set 2 to 12.0 Patch Set 3.

For more information, see *BRM Cloud Native Deployment Guide*.

BRM Server, Thick Clients, and PCC Now Localized

BRM server, BRM thick clients such as Customer Center and Pricing Center, and Pipeline Configuration Center (PCC) are now available in localized versions. The following languages are supported: French, Italian, Spanish, Japanese, Korean, Chinese Simplified, Chinese Traditional, Russian, and Portuguese Brazilian.

Localized versions of the software are available in the full installer JAR files. For example, to install a localized version of the BRM thick clients, use the **brmclients_all_12.0.0.3.0_generic.jar** file.

Revenue Assurance Supports Custom Applications

You can now configure BRM to collect revenue assurance data from the following when they are run as part of a custom business operations job:

- Custom MTA applications. See "Enabling Custom MTA Applications to Generate Revenue Assurance Data".
- Custom non-MTA applications. See "Configuring Custom Non-MTA Applications to Generate Revenue Assurance Data".

Pipeline Manager Can Now Support 28-Digit Decimal

You can now configure Pipeline Manager to support 28-digit decimal rather than the default of 15-digit decimal.

To configure Pipeline Manager to support up to 28-digit decimal, set the **MAX28DIGIT_DECIMAL_SUPPORT** environment variable to **Y** before you start Pipeline Manager and its services.

New Timeout Parameter for Account Synchronization DM

You can now specify the length of time, in milliseconds, that the Account Synchronization DM waits for a response from the database before timing out. To do so, you use the new **database_request_timeout_duration** parameter in the

BRM_Home/sys/dm_ifw_sync/pin.conf file. See "Configuring Account Synchronization DM Database Connection Attempts" in *BRM Installation Guide*.

Support Added for AES Encryption

BRM now supports the AES encryption method for systems that have upgraded from BRM 7.5 to BRM 12.0 Patch Set 3. See "About AES Encryption" in *BRM Developer's Guide*.



Note:

The AES encryption method is supported for backwards compatibility only. Oracle recommends using the more secure OZT encryption method instead.

Enable SSL/TLS between Client and the CM Only

You can now enable SSL/TLS between clients and the CM only, while disabling SSL/TLS between the CM and DM/EM. To do so, run this command from the *BRM_home/setup/scripts* directory:

```
perl sslConfig.pl 2
```

For more information, see "Enabling or Disabling SSL/TLS for BRM Components" in *BRM System Administrator's Guide*.

Adjusting Taxes by a Specified Amount

With this enhancement, you can adjust the tax levied on an event by a specified amount, such as by 5 US dollars, or by a percentage, such as by 5%. See "Calculating Taxes for Accounts Receivable Actions" in *BRM Calculating Taxes*.

In previous releases, you could adjust the tax by a percentage only.

New Features in BRM 12.0 Patch Set 2

BRM 12.0 Patch Set 2 includes the following enhancements:

- [Deploying BRM Services on a Cloud Native Environment](#)
- [Wholesale Billing Enhancements](#)
- [Additional Card Security Presence Values Supported for Card Validation or Authorization](#)
- [Card Security Code Is Now Masked in Logs](#)
- [Display Bills Generated Before Moving the Account to a Hierarchy](#)
- [Managing Discount Validity Starting or Ending in Future Cycles](#)
- [Wildcard in Item Type Selectors](#)
- [BRM Supports POID Generation in ECE](#)
- [Support for Stored-Credential Transactions for Payments](#)
- [Support for Migration of Legacy Data into BRM and ECE in Real Time](#)
- [Support for Migrating Hierarchical Accounts Using Same Input File](#)

- [Event Rounding Rules Can Be Used for Adjustments](#)
- [New Staging Directory for Exported G/L reports](#)
- [BRM Client Applications Supported from 12.0 Patch Set 2](#)

Deploying BRM Services on a Cloud Native Environment

BRM, Oracle Communications Pricing Design Center (PDC), Billing Care, and Business Operations Center now support their deployment on a cloud native environment, allowing you to harness the benefits of cloud with the services of BRM.

To deploy these BRM services, you use the new Oracle Communications BRM Cloud Native Deployment Option. This automates the deployment of BRM products and speeds up the process to get services up and running. Product deployments are preconfigured to communicate with each other through Helm charts.

For more information, see *BRM Cloud Native Deployment Guide*.

Wholesale Billing Enhancements

With this enhancement, the following are supported in wholesale billing:

- The usage charges calculated by ECE and Pipeline Manager are also considered for wholesale billing.

During billing, all the usage charges applied to the nonpaying child bill units (*wholesale child accounts*) are aggregated and rolled up to the paying parent bill unit (*wholesale parent account*). When rated events of the nonpaying child accounts are loaded into BRM, the `/tmp_journals_to_process` objects are created in BRM instead of the `/journal` objects. The `/tmp_journals_to_process` objects are created if deferred taxation is configured and the `CycleTaxInterval` business parameter is set to billing. See "[Specifying How to Calculate Deferred Taxes for Wholesale Billing](#)" for setting the `CycleTaxInterval` business parameter and the discussions about enabling and disabling taxation globally in *BRM Calculating Taxes* for configuring deferred taxation.

- You can move child bill units into a wholesale hierarchy or move them out of a wholesale hierarchy even if there are pending charges or due amounts. The due amount of the child bill unit or the bill generated before moving the child bill unit is moved along with the child bill unit. You can generate invoice and apply the payment for that due amount after moving the child bill unit.
- You can generate invoices for both paying parent bill unit and nonpaying child bill units.

In the Summary of Current Charges section in the parent bill unit's invoice, you can view the sum of the service-level charges, taxes, and surcharges rolled up from the nonpaying child bill units as account-level charges. The adjustment details for the child bill units are not displayed in the parent bill unit's invoice as they are performed at account-level. However, the adjustments in the parent bill unit's invoice includes the sum of all the adjustments performed for the nonpaying child bill units.

In the child bill unit's invoice, you can view the plan and account receivable (A/R) action details for that specific bill unit.

- The `/tmp_ar_item_to_roll_up` objects are enabled for partitioning automatically.

You can create the **/tmp_ar_item_to_roll_up** objects for wholesale billing in partitions and purge these objects from the database by using the **partition_utils** utility. When the **partition_utils** utility is run, the **/tmp_ar_item_to_roll_up** objects with the status as **1** (processed) are purged. You can run the **partition_utils** utility for the specific time interval.

Specifying How to Calculate Deferred Taxes for Wholesale Billing

You can specify how BRM calculates deferred taxes for wholesale billing by setting the **CycleTaxInterval** business parameter to **billing**.

When set to **billing**, the tax is forwarded from the child account to the parent account. BRM calculates taxes for the parent account only, but the single tax item on the parent account includes taxes from both the parent and child accounts.

To specify how to calculate deferred taxes for wholesale billing:

1. Go to the **BRM_home/sys/data/config** directory, where **BRM_home** is the directory where you installed BRM components.

2. Create an XML file from the **/config/business_params** object:

```
pin_bus_params -r BusParamsBilling bus_params_billing.xml
```

This command creates the XML file named **bus_params_billing.xml.out** in your working directory. To place this file in a different directory, specify the path as part of the file name.

3. Open the **bus_params_billing.xml.out** file.
4. Search for the following line:

```
<CycleTaxInterval>accounting</CycleTaxInterval>
```

5. Change **accounting** to **billing**.

```
<CycleTaxInterval>billing</CycleTaxInterval>
```

Note:

BRM uses the XML in this file to overwrite the existing billing instance of the **/config/business_params** object. If you delete or modify any other parameters in the file, these changes affect the associated aspects of the BRM subscription configurations.

6. Save this file as **bus_params_billing.xml**.
7. Load the XML file into the BRM database:

```
pin_bus_params bus_params_billing.xml
```
8. Stop and restart Connection Manager (CM).
9. (Multischema systems only) Run the **pin_multidb** script with the **-R CONFIG** parameter. For more information, see *BRM System Administrator's Guide*.

Additional Card Security Presence Values Supported for Card Validation or Authorization

For credit card validations or authorizations, Paymentech Data Manager (**dm_fusa**) sends the Fraud Format Indicator (FR) or Product Record (PFR) record with the card security presence value. When the card security code (such as VISA CVV2) is present in the PIN_FLD_SECURITY input flist field, Paymentech DM sets the card security presence value to 1 and sends the FR or PFR record for validation by default.

With this enhancement, you can customize BRM opcodes to send other card security presence values (such as 9 (No Value)) in the input flist to Paymentech DM. You can use the PIN_FLD_AVAILABLE flist field to provide the card security presence value.

Note:

The PIN_FLD_AVAILABLE field can be used for both online and batch transactions.

Following are the enumerated names and values supported in the PIN_FLD_AVAILABLE field:

```
typedef enum pin_pymt_card_secid_presence {
    PIN_PYMT_CSP_BLANK = 0,
    PIN_PYMT_CSP_AVAILABLE = 1,
    PIN_PYMT_CSP_ILLEGIBLE = 2,
    PIN_PYMT_CSP_NOT_PROVIDED = 5,
    PIN_PYMT_CSP_NO_CSV = 9,
} pin_pymt_card_secid_presence_t;
```

These names and values are defined in the *BRM_home/include/pin_pymt.h* file.

You can customize the PCM_OP_PYMT_POL_PRE_COLLECT opcode to send PIN_FLD_AVAILABLE and PIN_FLD_SECURITY_ID in the input flist.

For credit card transactions, Paymentech Data Manager now does the following:

- If only PIN_FLD_SECURITY_ID value is present in the input flist, Paymentech DM sends the FR or PFR record for validation with the card security presence value set to 1 (value present).
- If both PIN_FLD_SECURITY_ID and PIN_FLD_AVAILABLE values are present in the input flist, Paymentech DM sends the FR or PFR record for validation with the card security presence value set to the PIN_FLD_AVAILABLE value.
- If only the PIN_FLD_AVAILABLE value is present in the input flist, Paymentech DM sends the FR or PFR record for validation with the card security presence value set to the PIN_FLD_AVAILABLE value.
- If both PIN_FLD_SECURITY_ID and PIN_FLD_AVAILABLE values are not present in the input flist, FR or PFR record is not sent for validation.

Card Security Code Is Now Masked in Logs

BRM applications may log flists containing the sensitive customer data. In previous releases, the card security code, such as VISA CVV2 or American Express CID, passed in the PIN_FLD_SECURITY_ID input flist field was not masked and appeared as clear text in the logs.

With this enhancement, the PIN_FLD_SECURITY_ID value is masked during logging. The card security codes passed in this field appear as masked fields in logs.

Display Bills Generated Before Moving the Account to a Hierarchy

In previous releases, when an account was moved to a hierarchy as a child account, the bills generated for the account before moving it to the hierarchy were not displayed in the bills list in Billing Care.

BRM opcodes have been modified to retrieve and display all the bills for an account in hierarchy. This allows you to view the bills that are generated before and after moving the account to a hierarchy with the details (such as item list, event details, A/R actions, and payments) in Billing Care.

To support this feature, the following opcodes now include the new PIN_INCLUDE_CHILDREN_ALL value for the PIN_FLD_INCLUDE_CHILDREN parameter in the input flist:

- PCM_OP_AR_GET_BILLS
- PCM_OP_AR_GET_ACCT_BILLS
- PCM_OP_AR_GET_BAL_SUMMARY
- PCM_OP_AR_GET_ACCT_BAL_SUMMARY
- PCM_OP_AR_GET_ACTION_ITEMS
- PCM_OP_AR_GET_ACCT_ACTION_ITEMS
- PCM_OP_AR_GET_DISPUTES
- PCM_OP_AR_GET_DISPUTE_DETAILS
- PCM_OP_AR_GET_BILL_ITEMS

When you set the value of PIN_FLD_INCLUDE_CHILDREN to **3** (PIN_INCLUDE_CHILDREN_ALL), all the bills generated for the child account (before and after moving to the hierarchy) are retrieved.



Note:

When the PIN_FLD_INCLUDE_CHILDREN parameter is set to **3**, PIN_FLD_BILLINFO_OBJ is mandatory in the input flist.

Managing Discount Validity Starting or Ending in Future Cycles

In previous releases, the discount proration was not set properly if the discount validity was starting or ending in a future cycle.

With this enhancement, during billing, BRM identifies the discounts that starts or ends in the next billing cycle and sets the discount validity and proration appropriately. For example, if proration for a discount is set to **Full discount**, full discount is applied even if the discount validity ends in the middle of the next billing cycle.

For more information on discount validity, see the discussion about configuring discount validity in *PDC Creating Product Offerings* if you are using PDC or the Pricing Center Online Help if you are using Pricing Center.

Wildcard in Item Type Selectors

Oracle Communications Pricing Design Center (PDC) now supports wildcard (*) in item type selectors. By setting the **true** or **false** value for the **applicableToAllChildServices** and **applicableToAllChildEvents** elements in PDC, you can configure whether all the child services or events of a service or event must be considered for item assignment.

If set to **true**, the real-time rating engine and batch rating engine consider all child services or events for the specified item assignment. If set to **false**, the real-time rating engine and batch rating engine do not consider child services or events for the specified item assignment.

If you are using wildcard in item type selectors, you must set the **PDCEnable** entry to **true** in the **DAT_ItemAssign** module in the *Pipeline_home/conf/wireless.reg* file.

For more information on using wildcard in item type selectors, see "[PDC Now Supports Wildcard in Item Type Selectors](#)".

BRM Supports POID Generation in ECE

In the previous releases, ECE was using the Portal object IDs (POIDs) generated in BRM for tracking rated events and bill items created in ECE.

With this enhancement, POIDs can be generated in ECE. ECE uses Rated Event Formatter to generate the required POIDs. To support this feature, the following changes have been made in BRM:

- All the existing BRM storable class definitions have been modified to include the event type. The event type is used for creating separate partitions for different set of events. When you create new custom classes in BRM, you must now set the **Event Type** field. For more information, see "[About Creating Custom Classes](#)".
- The **PCM_OP_SDK_SET_DD**, **PCM_OP_SET_DD**, and **PCM_OP_GET_DD** opcodes have been modified to support the partitioning of prepaid events.
- The following have been introduced to enable partitions for prepaid events:
 - The **prepaid_partition_set** and **prepaid_partition_transition_mode** entries introduced in **dm_oracle**.
 - The **prepaidPartitionSet** parameter introduced in the **system** instance of the **/config/business_params** object.

For more information, see "[Enabling Prepaid-Event Partitions](#)".

- The **partition_utils** utility now supports the **-t prepaid** parameter. You can run the **partition_utils** utility with the **enable** operation and this parameter to create partitions for the prepaid events.

Enabling Prepaid-Event Partitions

If you are using ECE for usage rating, you must enable prepaid-event partition in BRM for generating the POIDs in ECE.

To enable prepaid-event partitioning:

Note:

In multischema systems, perform this task first on the primary BRM installation machine and then on the secondary BRM installation machines.

1. Open the `BRM_Home/sys/dm_oracle/pin.conf` file in a text editor.
2. Set the `prepaid_partition_set` entry to a numerical value between **2** and **7**. For example:

```
- dm prepaid_partition_set 2
```

If this entry is set to **0**, ECE uses the POIDs received from BRM for the prepaid events.

3. Set the `prepaid_partition_transition_mode` entry to **1**:

Note:

Setting this entry to **1** enables Data Manager to retrieve the partitions for the existing events. After retrieving all the partitions for the existing events (for example, after 90 days), set this entry to **0** to disable this mode.

```
- dm prepaid_partition_transition_mode 1
```

4. Save and close the file.
5. Create an editable XML file from the **system** instance of the `/config/business_params` object:

```
pin_bus_params -r BusParamsSystem bus_params_system.xml
```

6. Set the `prepaidPartitionSet` parameter to the value you specified in step 2. For example:

```
<prepaidPartitionSet>2</prepaidPartitionSet>
```

7. Save the file as `bus_params_system.xml`.
8. Load the XML file into the BRM database:

```
pin_bus_params bus_params_system.xml
```

9. Stop and restart CM.
10. (Multischema systems only) Run the `pin_multidb` script with the `-R CONFIG` parameter. For more information, see *BRM System Administrator's Guide*.
11. Go to the `BRM_home/apps/partition_utils` directory.

12. Create partitions for the prepaid events by running the following command:

```
partition_utils -o enable -t prepaid
```


For more information, see the discussion about partitioning database tables in *BRM System Administrator's Guide*.

For information on enabling POID generation and prepaid-event partitions for ECE, see the discussion about POID generation in *BRM Elastic Charging Engine Release Notes*.

About Creating Custom Classes

When you create new custom classes in BRM, you must now set the **Event Type** field. [Table 2-1](#) lists the event types in BRM. For instructions on how to create custom classes, see the discussion about creating custom classes in *BRM Developer's Guide*.

Note:

When you create a custom class, note the following:

- If the event type is set to NONE, the corresponding event is not synchronized with PDC.
- When you add a subclass, ensure that the event type matches the event type of the parent class if the event type is set to anything other than NONE.

Table 2-1 Event Types in BRM

Event Type	Description
USAGE_PREPAID	Specifies the prepaid events (real-time charging events) from ECE.
USAGE_POSTPAID	Specifies the delayed events in BRM and postpaid events (offline charging events) from ECE.
SUBSCRIPTION_RECURRING	Specifies the recurring subscription event.
SUBSCRIPTION_ROLLOVER	Specifies the subscription events generated for rollovers.
SUBSCRIPTION_BILL_TIME_DISCOUNT	(For internal use only) Specifies the subscription events generated for bill-time discounts.
SUBSCRIPTION_FOLD	(For internal use only) Specifies the subscription events generated for folds.
SUBSCRIPTION_ONE_TIME	(For internal use only) Specifies the one-time subscription events.
SUBSCRIPTION_REMITTANCE	(For internal use only) Specifies the subscription events generated for remittances.
NONE	Specifies that the event does not belong to any other event type.

After you create the custom class and synchronize it with PDC, it is not recommended to change the event type of the custom class. However, if you have set the incorrect event type, you can change the event type by updating it first in BRM and then in PDC.

For changing the event type in BRM by editing the custom class, see the discussion about creating, editing, and deleting fields and storable classes in *BRM Developer's Guide*.

For updating the event type in PDC and publishing it to ECE, see the discussion about synchronizing and publishing the event type in "[PDC Synchronizes Event Data Using Event Types](#)".

Support for Stored-Credential Transactions for Payments

Credit card networks, such as VISA, MasterCard, Diners, Discover, JCB, and American Express, support the stored credential framework. They allow the merchants to use stored credentials for transactions. A stored credential is a payment information (such as an account number or a payment token) of a card holder stored by a merchant or its agent, a payment facilitator, or a staged digital wallet operator to process future transactions for the card holder. The Paymentech card processors also support customer-initiated or merchant-initiated transactions with stored credentials.

With this enhancement, BRM supports payment transactions with stored credentials for VISA, MasterCard, Diners, Discover, JCB, and American Express cards. You can also customize BRM to support stored-credential transactions for other card networks. When VISA, MasterCard, Diners, Discover, JCB, and American Express cards are used for payments, the `PCM_OP_PYMT_COLLECT` opcode sends the following information required for card transactions to Paymentech Data Manager (**dm_fusa**) and stores the responses received from Paymentech DM for future transactions:

- Type of charge, such as recurring, one-time, and installment.
- Transaction type.
- Information on whether the card details are stored for future use.
- A unique ID (TXID) obtained from a previous verify/charge transaction of the same type

You can override this information sent to Paymentech DM based on your business requirements. If you do not want to store credentials for future transactions, you can remove this information from the input passed to the `PCM_OP_PYMT_COLLECT` opcode.

For more information on storing or purging card credentials, see the following:

- [Storing Card Credentials for Future Transactions](#)
- [Purging Card Credentials](#)

If you already have payment information stored in BRM for cards that support the stored credential framework, see "[Migrating Legacy Payment Information](#)".

To support the stored credential framework, the following changes have been made in BRM for this feature:

- A new array, `PIN_FLD_TRANSACTIONS`, has been introduced in the **lpayinfo/cc** storable class with the following optional fields to hold the Stored Credential Framework-specific information:
 - **PIN_FLD_BILLINFO_OBJ**. The bill unit (billinfo) for which payment is applied using the payment information (payinfo) in this array.

 **Note:**

You can associate one payment information with multiple bill units. Each bill unit has its own recurring cycle and each series of recurring cycles must have its own TXID.

- **PIN_FLD_MODE.** The message type with which the initial transaction was performed.
- **PIN_FLD_TRANS_ID.** The TXID received in the initial transaction response.
- A new array, **PIN_FLD_TRANSACTIONS**, has been introduced in the **/event/billing/charge/cc** and **/event/billing/validate/cc** storable class with the following optional fields to hold the Stored Credential Framework-specific information:
 - **PIN_FLD_TRANS_ID.** The TXID received in the initial transaction response.
 - **PIN_FLD_MODE.** The message type with which the transaction was performed.
 - **PIN_FLD_FLAGS.** The flag which indicates whether the credentials are stored in the file.
 - **PIN_FLD_TYPE.** The transaction type.
- The following opcodes have been modified to include the **PIN_FLD_TRANSACTIONS** array:
 - **PCM_OP_PYMT_COLLECT**
 - **PCM_OP_PYMT_POL_PRE_COLLECT**
 - **PCM_OP_PYMT_CHARGE**
 - **PCM_OP_PYMT_CHARGE_CC**
 - **PCM_OP_CUST_COMMIT_CUSTOMER**
 - **PCM_OP_CUST_CREATE_PAYINFO**
 - **PCM_OP_CUST_SET_PAYINFO**
 - **PCM_OP_CUST_POL_VALID_PAYINFO**
- The following tables have been added to store the information in the **PIN_FLD_TRANSACTIONS** array:
 - **EVT_BILLING_CHARGE_CC_TRANS_T**
 - **EVT_BILL_VLDT_CC_TRANS_T**
 - **PAYINFO_CC_TRANS_T**
- BRM payment collection utilities, such as **pin_collect** and **pin_deposit**, have been enhanced to support stored credentials.

For more information on the field definitions and valid values, see *BRM Opcode Flist Reference*.

Storing Card Credentials for Future Transactions

To store the credit card information for future transactions, use **PCM_OP_PYMT_COLLECT**.

This opcode does the following:

1. Receives the PIN_FLD_TRANSACTIONS array from the following opcodes if a VISA, MasterCard, Diners, Discover, JCB, or American Express card is registered for payment:
 - **PCM_OP_CUST_COMMIT_CUSTOMER.** This opcode passes the PIN_FLD_TRANSACTIONS array as input when you register a new customer with **Credit Card** as the default payment method and the cc_validate or cc_collect flag in the CM configuration file or credit card tokenization is enabled.
 - **PCM_OP_CUST_SET_PAYINFO.** This opcode passes the PIN_FLD_TRANSACTIONS array as input when you set **Credit Card** as the default payment method for the account.
 - **PCM_OP_CUST_CREATE_PAYINFO.** This opcode passes the PIN_FLD_TRANSACTIONS array as input when you add a new credit card and set that as the default payment method for the account.
2. Accepts the card information in the PIN_FLD_TRANSACTIONS array, adds missing information as required, and then passes the information as input to the PCM_OP_PYMT_POL_PRE_COLLECT policy opcode.

 **Note:**

You can add, update, or remove the card information in the PIN_FLD_TRANSACTIONS array by customizing the PCM_OP_PYMT_POL_PRE_COLLECT policy opcode.

3. Receives the updated card information and PIN_FLD_CHARGES from the PCM_OP_PYMT_POL_PRE_COLLECT policy opcode.
4. Sends the card information in the PIN_FLD_TRANSACTIONS array to Paymentech DM by calling the PCM_OP_PYMT_CHARGE or PCM_OP_PYMT_CHARGE_CC opcode.
Paymentech DM appends the required records based on the information received and sends the transactions to Paymentech. If the transaction is successful, Paymentech DM retrieves the TXID from the Paymentech response and passes it to the PCM_OP_PYMT_COLLECT opcode.
5. Accepts the TXID received from Paymentech DM and stores it in the PIN_FLD_TRANSACTIONS array in the **/payinfo/cc** object for future transactions.

Purging Card Credentials

When **Credit Card** is set as the default payment method, the PIN_FLD_TRANSACTIONS array with the card information for each bill unit is stored in the **/payinfo/cc** object. If the default payment method is changed to any other method or card, the PIN_FLD_TRANSACTIONS array in the **/payinfo/cc** object are automatically purged from the database if the payments for the accounts are not in cardholder-initiated installments. If the payments are in cardholder-initiated installments, the card holder must delete the PIN_FLD_TRANSACTIONS array manually after the payment is made for the last installment.

Migrating Legacy Payment Information

If there is legacy payment information in BRM for cards that support stored credential framework, it is automatically migrated to the new format, which includes the fields for storing credentials. BRM uses this information only for merchant-initiated transactions.

After migration, when the merchant initiates the first transaction with this payment information, the `PCM_OP_PYMT_COLLECT` opcode sets the TXID in the input as **EXISTING9999999** to indicate that this transaction is initiated with the legacy payment information. The opcode then stores the TXID received from Paymentech DM as an additional TXID.

Support for Migration of Legacy Data into BRM and ECE in Real Time

In previous releases, all the legacy data migrated to BRM were loaded into Oracle Communications Billing and Revenue Management Elastic Charging Engine (ECE) asynchronously using **CustomerLoader**. In case if the ECE update failed, the BRM and ECE data would be unsynchronized. If ECE uses the unsynchronized data to rate usage events, the events might be rated incorrectly.

Also, it was not possible to migrate legacy service and balance data incrementally into BRM and ECE. For example, when the legacy data was migrated into BRM in phases, if the account already migrated to BRM had some services associated with it, you could not migrate additional legacy services for the same account. As a result, the legacy service and balance updates could not be loaded into ECE.

With this enhancement, you can migrate the legacy data completely or incrementally into the BRM system by using the **pin_cmt** utility and also load all the migrated data into ECE synchronously (in real time). For example, after the initial migration of the legacy account and service data into BRM and ECE, you can migrate additional services and balances for the same migrated accounts into BRM and also synchronize them with ECE in real time.

To support this feature, the following changes have been made in BRM:

- The **CMT_Service.xsd** schema file has been introduced to support incremental migration of legacy service data.
- The **CMT_Balances.xsd** schema file has been modified to support incremental migration of legacy balance data.
- A new entry, **infranet.cmt.uselegacybalances**, has been introduced in the `BRM_home/apps/cmt/Infranet.properties` file to support incremental migration of legacy balance data.
- The **cmt_mta_cycle_fees** utility, which is run internally by the **pin_cmt** utility, has been modified and renamed as **cmt_mta_deploy**. Do not run this utility by itself.
- Following new parameters are introduced for migrating legacy data by using the **pin_cmt** utility:
 - **deploy_ece** specifies to deploy the migrated data in BRM and also load them into ECE synchronously. You can use this parameter for complete or incremental migration of legacy data into BRM and ECE synchronously.

 **Note:**

If ECE is not integrated with BRM, you need to use the **deploy** parameter for both complete and incremental migration of legacy data into BRM. For more information, see the discussion about loading legacy data into the BRM database in *BRM Migrating Accounts to the BRM Database*.

- **deploy_db** specifies to deploy the migrated data only in BRM. You can use this parameter if you want to load the legacy data into ECE asynchronously using **CustomerLoader**.

 **Note:**

After deploying the legacy data by using the **deploy_db** parameter, you need to load the data manually into ECE by using the **CustomerLoader** utility and then run the **pin_cmt** utility with the **apply_cycle_fees** parameter to apply the cycle fees for the migrated accounts.

For information on loading data using the **CustomerLoader** utility, see the discussion about using **CustomerLoader** in *ECE Implementing Charging*.

- **apply_cycle_fees** specifies to apply the cycle fees for the migrated accounts. You can use this parameter only when you load the legacy data into ECE asynchronously.

When you use the **deploy_ece** parameter, the **pin_cmt** utility deploys the migrated legacy data in BRM and also sends it to ECE in real time through External Manager (EM) Gateway. In this case, both the BRM database and the ECE cache updates occur in a single transaction. If the ECE cache update succeeds, the updates are saved to the BRM database. If the cache update fails, the database updates are rolled back. This ensures that the BRM and ECE data remain synchronized whether the cache update succeeds or fails. You can use the **cmt.pinlog** files in BRM and error logs in ECE to troubleshoot the errors. For more information on troubleshooting Conversion Manager, see *BRM Migrating Accounts to the BRM Database*. For more information on troubleshooting ECE, see *BRM System Administrator's Guide*.

To migrate the legacy account, service, and balance data completely or incrementally into BRM and ECE, do the following:

1. Import the legacy data into BRM. See "[Importing Data](#)".
2. Deploy the converted data. See "[Deploying Data in BRM and ECE](#)".

Importing Data

You import legacy data into the BRM database one file at a time.

Before importing the legacy data, do the following:

1. If you are migrating custom data, create new or extended storable classes for migration. See the discussion about migrating data by using new and extended storable classes in *BRM Migrating Accounts to the BRM Database*.
2. Create XML files with the legacy data that you want to import, which conform to the format detailed in the corresponding XSD files; such as **CMT_Account.xsd**,

CMT_Service.xsd, and **CMT_Balances.xsd** file. See the discussion about creating XML files in *BRM Migrating Accounts to the BRM Database*.

3. Open the *BRM_home/apps/cmt/Infranet.properties* file in a text editor.
4. Ensure that the database connection is specified.
5. If you are migrating legacy balances, to replace the existing balances in BRM with the legacy balances, set the **infranet.cmt.uselegacybalances** entry to **true**.

 **Note:**

This entry takes precedence over the **infranet.cmt.deleteexistingbalances** entry. If set to **true**, the **infranet.cmt.deleteexistingbalances** entry is not used.

 **Note:**

When the **infranet.cmt.uselegacybalances** entry is set to **true**, you must ensure the following:

- Check if the **VALID_FROM** date is specified for all the sub-balances to be migrated. The **VALID_FROM** date is required for identifying the sub-balances to be replaced.
- Import the legacy balances before deploying the account and service data to the BRM production area. You cannot replace the balances after you deploy the migrated account and service data.

6. Save and close the file.

To import the legacy data completely or incrementally into the BRM database:

1. Go to the *BRM_home/apps/cmt* directory.
2. Do one of the following:
 - If you have not created custom storable classes for migration, import the legacy data by running the following command:

```
pin_cmt -import -file XML_input_data_file stage_ID
```

where *XML_input_data_file* is the XML file with the legacy data, and *stage_ID* is the unique identity of the staging area.

- If you have created custom storable classes for migrating custom data, import the legacy data by running the following command:

```
pin_cmt -import_custom -file XML_input_data_file stage_ID
```

 **Note:**

Ensure that the *stage_ID* is unique for each migration.

Deploying Data in BRM and ECE

Before deploying the imported data in BRM and ECE, do the following:

- Ensure that all the pricing data is loaded into the ECE cache. See the discussion about verifying that pricing data is loaded into ECE in *ECE Implementing Charging*.
- Ensure that the real-time synchronization is enabled in ECE. See the discussion about enabling real-time synchronization of BRM and ECE customer data updates in *ECE Implementing Charging*.

To deploy the imported data to the BRM production area and load the data into ECE synchronously, run the following command:

Note:

If you are migrating the complete legacy data, you can import the legacy account, service, and balance data into the BRM database and then deploy all the legacy data together.

You must import the legacy balances before deploying the account and service data to the BRM production area. You cannot replace the balances after you deploy the migrated account and service data.

```
pin_cmt -deploy_ece DOM stage_ID
```

where *DOM* is the billing cycle's day of month.

The migrated legacy data deployed in BRM and are synchronized with ECE in real time.

If you are deploying the services, the **pin_cmt** utility does the following:

- Applies the cycle fees in BRM and synchronize them with ECE if the **infranet.cmt.deploy.opcode** entry is set to **true**. See the discussion about applying cycle fees to deployed accounts in *BRM Managing Customers*.
- Deploys the legacy balances if the **infranet.cmt.uselegacybalances** entry is set to **true**.

Merging Legacy Balances

BRM merges the migrated legacy balance data with the existing balance data in the BRM database. If the sub-balance already exists in the BRM database, it replaces the sub-balance with the corresponding legacy balance. If the sub-balance does not exist, it adds the legacy balance to the BRM database. BRM identifies the existing sub-balances in the BRM database by using the combination of the following data in **/cmt_balances** objects:

- resource ID
- POID of the granting product/charge offer or discount
- VALID_FROM date

In case if the VALID_FROM date is not specified, the first sub-balance available for the specified granting product or discount is replaced with the legacy balance. However, you must ensure that the VALID_FROM date is specified for all the legacy balances to be migrated. This ensures that the balances are replaced appropriately.

In case if the POID of the granting product or discount is not specified, a new sub-balance is created with the specified `VALID_FROM` and `VALID_TO` dates.

For more information on migrating data, see the discussion about loading legacy data into the BRM database in *BRM Migrating Accounts to the BRM Database*.

Support for Migrating Hierarchical Accounts Using Same Input File

In previous releases, you could migrate a legacy hierarchical account group to BRM only if the parent and child accounts belonging to that group were present in different XML input files.

With this enhancement, you can migrate the legacy hierarchical account groups to BRM even if the parent and child accounts in the groups are present in the same XML input file. When you run the `pin_cmt` utility, all the hierarchical account groups in the input file are loaded into the BRM database. However, if any group in the file contains a multi-level account hierarchy (for example, Account C's bill unit is paid by Account B, which is in turn paid by Account A), you cannot load the parent and child accounts in that group using the same input file. You must create separate input files for parent and child accounts and load the parent accounts into the BRM database before loading the child accounts.

Event Rounding Rules Can Be Used for Adjustments

You can now use the rounding rules configured for events in adjustments. For example, if you configured usage events with six decimal precision, the event-level adjustment on the usage fee is rounded to six decimal precision.

To enable this feature, run the `pin_bus_params` utility to change the `UseEventRoundingRulesForAdjustment` business parameter. For information about this utility, see *BRM Developer's Guide*.

To use the rounding rules of corresponding events for adjustments:

1. Go to the `BRM_home/sys/data/config` directory.
2. Create an XML file from the `/config/business_params` object:

```
pin_bus_params -r BusParamsAR bus_params_AR.xml
```

3. In the XML file, search for the following parameter and change **disabled** to **enabled**:

```
<UseEventRoundingRulesForAdjustment>enabled</  
UseEventRoundingRulesForAdjustment>
```

4. Save the file as `bus_params_AR.xml`.
5. Load the XML file into the BRM database:

```
pin_bus_params bus_params_AR.xml
```

6. Stop and restart CM.
7. (Multischema systems only) Run the `pin_multidb` script with the `-R CONFIG` parameter. For more information, see *BRM System Administrator's Guide*.

New Staging Directory for Exported G/L reports

In the previous releases, when the **pin_ledger_report** utility was run in the export mode, the utility created a staging directory named, **TEMP_XML_STAGING**, in the *OutputDirectory* and stored the output G/L report files temporarily in this directory before moving them to the *OutputDirectory*, where *OutputDirectory* is the directory on your system in which to create the output G/L report files.

With this enhancement, the **pin_ledger_report** utility creates a staging directory named, **TEMP_GL_REPORT_STAGING**, in the same location as the *OutputDirectory* and stores the output G/L report files temporarily in this directory before moving them to the *OutputDirectory*. For example, if the *OutputDirectory* is `pin/users/GL`, the G/L Report Files are stored in the new `pin/users/TEMP_GL_REPORT_STAGING` directory.

Note:

To enable the **pin_ledger_report** utility to store G/L report files in the new staging directory, do one of the following:

- Ensure that the utility has the permission to create a directory and file in the location in which the *OutputDirectory* is available.
- Manually create the **TEMP_GL_REPORT_STAGING** directory in the same location as the *OutputDirectory* and add the permission for the utility to create or store files in this directory.

BRM Client Applications Supported from 12.0 Patch Set 2

Table 2-2 lists the BRM client applications that are now released in BRM 12.0.

Table 2-2 Supported BRM Client Applications

Component	Description
Customer Center	BRM now supports Customer Center (including Customer Center SDK). You can now use Customer Center or Oracle Communications Billing Care for customer management tasks. For more information, see the Customer Center Online Help.
Collections Center	BRM now supports Collections Center. You can now use Collections Center or Billing Care for collections or debt management tasks. For more information, see the Collections Center Online Help.
Configuration Center	BRM now supports Configuration Center.
Field Validation Editor	BRM now supports Field Validation Editor to define how to validate customer data.
Payment Center	BRM now supports Payment Center. You can now use Payment Center or Billing Care instead for payment management tasks. For more information, see the Payment Center Online Help.
Payment Tool	BRM now supports Payment Tool. You can now use Payment Tool or Billing Care for processing payments. For more information, see the Payment Center Online Help.

Table 2-2 (Cont.) Supported BRM Client Applications

Component	Description
Pricing Center	BRM now supports Pricing Center. You can now use Pricing Center or Pricing Design Center for creating product offerings for BRM. For more information, see the Pricing Center Online Help.

New Features in BRM 12.0 Patch Set 1

BRM 12.0 Patch Set 1 includes the following enhancements:

- [Improved Performance for Large Accounts](#)
- [Delay Interval Can be Configured for Resolving Failed Payments](#)
- [Enhanced Data Protection](#)
- [Enhanced Security for Root Wallet](#)
- [Support for Rolling Back the BRM Patch Set](#)
- [BRM 12.0 Is Now Certified with Mozilla Firefox 58.0](#)
- [BRM 12.0 Is Now Certified with Perl 5.28.0](#)
- [BRM 12.0 Is Now Certified with Paymentech 120 Byte Batch Version 3.0.0 R 12.4 and Online Authorization Version 7.4 R12.4](#)
- [BRM 12.0 Is Now Certified with Tomcat 8.5.32](#)

Improved Performance for Large Accounts

Wholesale business accounts with large account hierarchies can have a large number of services each representing a subscription account. This can affect the billing and invoicing performance if the accounts had a large number of billing items to be processed.

With this enhancement, you can setup wholesale billing for handling large wholesale business accounts. In wholesale billing, you set up a bill unit hierarchy for account receivable (A/R) operations. In this hierarchy, the wholesale business account is the parent account with the paying parent bill unit and the services (subscriptions) in this account are child accounts with nonpaying child bill units. This enables BRM to consolidate the charges, discounts, A/R items, bill items, journals, and taxes across the services under the wholesale business account and perform the A/R operations, billing, and invoicing at the wholesale business account level. This improves the billing and invoicing performance for wholesale accounts with large hierarchies.

If you want to enable wholesale billing for all your accounts, you can enable system-wide wholesale billing by setting the **WholesaleBillingSystem** business parameter in the **billing** instance of the **/config/business_params** object. When this business parameter is enabled, you can create only wholesale accounts and bill unit hierarchies. For more information, see "[Enabling Wholesale Billing](#)" and "[Creating Wholesale Accounts and Bill Unit Hierarchies](#)".

If you want to enable wholesale billing only for specific accounts, you can set up an account with the paying parent bill unit as the wholesale parent and then create the wholesale bill unit hierarchy. You can create multiple wholesale bill unit hierarchies

based on your business requirements. You need not enable system-wide wholesale billing. For more information, see "[Creating Wholesale Accounts and Bill Unit Hierarchies](#)".

Enabling Wholesale Billing

To enable wholesale billing system-wide:

1. Go to `BRM_home/sys/data/config`.
2. Create an XML file from the `/config/business_params` object:

```
pin_bus_params -r BusParamsBilling bus_params_billing.xml
```
3. In the XML file, set the following entry to **enabled**:

```
<WholesaleBillingSystem>enabled</WholesaleBillingSystem>
```
4. Save the file as `bus_params_billing.xml`.
5. Load the file into the BRM database:

```
pin_bus_params bus_params_billing.xml
```
6. Stop and restart the Connection Manager (CM).
7. (Multischema systems only) Run the `pin_multidb` script with the `-R CONFIG` parameter. For more information, see *BRM System Administrator's Guide*.

Creating Wholesale Accounts and Bill Unit Hierarchies

You can create accounts and bill unit hierarchies for wholesale billing by using Billing Care or by using custom applications to call BRM opcodes. A wholesale bill unit hierarchy is headed by a paying parent bill unit with nonpaying child bill units beneath it. You can have only one paying bill unit in a wholesale bill unit hierarchy. And, you cannot add more than one bill unit to a wholesale parent account.

For more information on accounts and bill unit hierarchies, see the discussion about creating and managing account and bill unit hierarchies in *BRM Managing Accounts Receivable*.

Note:

You cannot convert an existing bill unit hierarchy in your system into a wholesale bill unit hierarchy. You must ensure the following for enabling wholesale billing:

- If you are enabling system-wide wholesale billing, you must set the **WholesaleBillingSystem** business parameter before creating wholesale accounts and bill unit hierarchies.
- If you are enabling wholesale billing for specific accounts, you must set up a wholesale parent before creating the wholesale bill unit hierarchy.

To create accounts and bill unit hierarchies and set up the wholesale parent account by:

- Using Billing Care, see the discussion about creating accounts and configuring billing hierarchies in the Billing Care Online Help.
- Using BRM opcodes, see the discussion about creating accounts in the *BRM Opcode Guide* and "[Setting Up a Wholesale Parent](#)".

 **Note:**

Before you set up the wholesale parent either by using Billing Care or BRM opcodes, you must configure the wholesale business profile. See "[Configuring Wholesale Business Profile](#)".

You can add any existing bill unit to a wholesale bill unit hierarchy or set up a new wholesale bill unit hierarchy by using the existing bill units in BRM. However, you must ensure the following:

- There are no pending items or payments in the bill unit that you are adding to the hierarchy.
- The parent bill unit is the paying bill unit and it is set as the wholesale parent for billing.
- The wholesale parent for the wholesale bill unit hierarchy is set before creating the hierarchy.

This ensures that the charges and other billing-related items of the nonpaying child bill units in the hierarchy are rolled up to the paying parent bill unit during billing.

Configuring Wholesale Business Profile

To configure the wholesale business profile:

1. Open the **pin_business_profile.xml** file in an XML editor or a text editor.
By default, this file is in the *BRM_home/sys/data/config* directory.
2. Search for the corporate wholesale business profile.
3. Set the **<WholesaleBilling>** entry to **yes**:

```
<WholesaleBilling>yes</WholesaleBilling>
```

4. Save and close the file.
5. Load the **pin_business_profile.xml** file by running the following command:

```
load_pin_business_profile pin_business_profile.xml
```

 **Note:**

- When you run the utility, the **pin_business_profile.xml** and **business_configuration.xsd** files must be in the same directory. By default, both files are in *BRM_home/sys/data/config*.
 - This utility needs a configuration (**pin.conf**) file in the directory from which you run the utility.
 - If you do not run the utility from the directory in which **pin_business_profile.xml** is located, include the complete path to the file.
6. (Multischema systems only) Run the **pin_multidb** script with the **-R CONFIG** parameter. For more information, see *BRM System Administrator's Guide*.

Setting Up a Wholesale Parent

To set up a parent bill unit for wholesale billing, you assign the bill unit that you want use as the wholesale parent to the wholesale business profile. To configure the wholesale business profile, see "[Configuring Wholesale Business Profile](#)".

You can assign the bill unit (**/config/business_profile** object) during or after account creation:

- To assign the bill unit to a business profile during account creation, see the discussion about assigning bill units to business profiles in *BRM Opcode Guide*.
- To assign the bill unit of an existing account to a business profile, see the discussion about changing a bill unit's business profile in *BRM Opcode Guide*.

Rolling Charges Up to the Wholesale Parent

Note:

In BRM 12.0 and BRM 12.0 Patch Set 1, the usage charges calculated by ECE and Pipeline Manager are not considered for wholesale billing; for example, the usage charges for telephony services.

During final billing, all the charges (such as recurring, purchase, and usage charges) applied to the nonpaying child bill units (*wholesale child accounts*) are aggregated based on the item-tag-to-item-type mapping (*item configuration*) and are rolled up to the corresponding bill items of the paying parent bill unit (*wholesale parent account*).

If the bill item for any item type does not exist for the paying parent bill unit, the bill item is created during billing and the charges are rolled up to that item. However, Oracle recommends to pre-create the bill items for the different item types by setting the **precreate** element to **true** in the *BRM_home/sys/data/pricing/example/config_item_types.xml* file. For more information, see the discussion about mapping item tags to item types in *BRM Configuring and Running Billing*.

The total and due amounts of the paying parent bill unit are updated to reflect the roll-up and the due amount of each nonpaying child bill unit is set to **0**. Payments are applied only to the paying parent bill unit. From BRM 12.0 Patch Set 2, invoicing of due amount of child bill unit is supported. See "[Wholesale Billing Enhancements](#)".

If you want the charges for different services to be rolled up to different bill items, you can assign different item types for different services. For example, for rolling up cycle forward fees for IP and GSM services, you can configure and assign the following items: **/item/ip/cycle_forward** for the IP service and **/item/gsm/cycle_forward** for the GSM service.

You can also assign a different item type (or a noncumulative bill item) to track charges specific to the paying parent bill unit.

For information on assigning items, see the discussion about using event and service combinations to assign bill items in *BRM Configuring and Running Billing*.

Rolling A/R Actions Up to the Wholesale Parent

To manage balances for the A/R actions, BRM uses A/R items. A/R items include adjustment, dispute, settlement, payment, refund, payment reversal, write-off, and write-off reversal items. For more information, see the discussion about A/R management in *BRM Concepts*.

To roll up A/R actions for wholesale billing, you use the **pin_roll_up_ar_items** utility. The **pin_roll_up_ar_items** utility processes all the temporary A/R items (*/tmp_ar_item_to_roll_up* object) of the nonpaying child bill units and rolls the balance impact up to the corresponding A/R items of the paying parent bill unit. For example, this utility rolls the adjustments allocated to the nonpaying child bill unit's **/item/cycle_forward** item up to the adjustment item associated with the **/item/adjustment** item of the paying parent bill unit.

You can run the **pin_roll_up_ar_items** utility on a daily basis to ensure that the A/R items of the paying parent bill unit are kept up to date. However, you must run this utility once before billing the paying parent bill unit. For more information on this utility, see "[pin_roll_up_ar_items](#)".

In addition, you can use the **pin_roll_up_ar_items** utility to roll up the adjustment items that are created as a result of rerating. During rerating, the temporary A/R items (*/tmp_ar_item_to_roll_up* object) are created for the nonpaying child bill units if the following conditions are met:

- The event has already been billed.
- The event occurred prior to general ledger posting.
- The event is unbilled but the automatic allocation of rerating adjustments is disabled.

If the event is unbilled and the automatic allocation of rerating adjustments is enabled, the rerating adjustment is allocated to the bill item of the nonpaying child bill unit.

Note:

Rerating adjustments rolled up to the paying parent bill unit are allocated to the corresponding A/R item only if it exists in the paying parent bill unit. If the A/R item does not exist, the rerating adjustments remain unallocated at the parent level.

Rolling Journals Up to the Wholesale Parent

For nonpaying child bill units, the **/tmp_journals_to_process** objects are created instead of the **/journal** objects at the time of rating. The **/tmp_journals_to_process** objects are created only if the **cycle_tax_interval** entry in the CM configuration file is set to **billing**. For more information, see the discussion about tax calculation for account groups in *BRM Calculating Taxes*.

The **/tmp_journals_to_process** objects contain revenue and tax data. For wholesale billing, BRM uses these objects primarily to track and consolidate taxes for billing-time taxation. To roll up journals for wholesale billing, you use the **pin_update_journals** utility.

 **Note:**

To roll up journals for wholesale billing, you must ensure that the general ledger reporting is enabled. For more information, see the discussion about general ledger reporting in *BRM Collecting General Ledger Data*.

The **pin_update_journals** utility processes all the **/tmp_journals_to_process** objects of the nonpaying child bill units and rolls them up to the corresponding **/journal** object of the paying parent bill unit.

You can run the **pin_update_journals** utility on a daily basis to ensure that the paying parent bill unit is kept up to date. However, you must run this utility once before billing the paying parent bill unit. For more information on this utility, see "[pin_update_journals](#)".

If deferred taxation is configured to consolidate taxes into a single item (if **cycle_tax_interval** is set to **billing**), the **pin_update_journals** utility enables you to roll the taxes up into a single item for both the paying parent and nonpaying child bill units. The total tax is calculated at the paying parent level for the entire hierarchy using the aggregated total due as the basis.

If deferred taxation is configured to calculate taxes separately for the parent and each nonpaying child bill unit (if **cycle_tax_interval** is set to **accounting**), the **/journal** objects are created for the nonpaying child bill units instead of **/tmp_journals_to_process** objects and the taxes are not rolled up to the paying parent bill unit.

Running Wholesale Billing

 **Note:**

The following features are not supported for wholesale billing:

- Bill Now
- On-purchase (on-demand) billing
- Skipped billing

If you are using bill suppression for billing wholesale accounts, you must run wholesale billing at the end of each accounting cycle.

To run wholesale billing:

1. Run the **pin_roll_up_ar_items** utility which rolls A/R actions up to the paying parent bill unit. See "[pin_roll_up_ar_items](#)".
2. Run the **pin_update_journal** utility which rolls journals up to the paying parent bill unit. See "[pin_update_journals](#)".
3. Run the **pin_bill_accts** utility which performs regular billing. See the discussion about the **pin_bill_accts** utility in *BRM Configuring and Running Billing*.

 **Note:**

After the nonpaying child bill units are billed, you cannot perform A/R activities (such as adjustments, disputes, and settlements) on the billed events of the nonpaying bill units until the paying parent bill unit is billed.

4. Run the **pin_inv_accts** utility for the wholesale parent account to generate invoices for bills.

Configuring Billing Delay for Wholesale Hierarchies

Delayed billing is supported for wholesale hierarchies. You must specify the billing delay even if it is not used. In this case, you can set the billing delay interval to **0**. See the discussion about configuring billing delay in *BRM Configuring and Running Billing*.

Setting Up Billing-Time Discounts for Wholesale Hierarchies

For wholesale bill unit hierarchies, you set up a billing-time discount as follows:

- Configure billing-time discount only for the paying parent bill unit in the hierarchy.
- Configure BRM to apply the billing-time discount at the end of the billing cycle instead of the accounting cycle.
- Configure non-billing-time discounts (usage discounts) for the nonpaying child bill units in the hierarchy. The usage discount increments the counter. For example, if the billing-time discount for the paying parent bill unit is based on total monthly charges, you can create a discount for a nonpaying child bill unit that increments the counter when charges are applied.

And, for rolling the discounts up to the paying parent bill unit at the time of billing, you customize the **PCM_OP_SUBSCRIPTION_POL_PRE_CYCLE_DISCOUNT** policy opcode to return the list of balance element/resource IDs of the counters (in the **PIN_FLD_BALANCES** output flist field) for which the balances to be rolled up to the paying parent bill unit in the hierarchy.

For more information on billing-time discounts, see the discussion about creating discount offers in *BRM Creating Product Offerings*.

Suppressing Bills for Wholesale Hierarchies

You can use bill suppression to postpone finalizing bills for wholesale accounts. When bill suppression is enabled, the charges applied to the nonpaying child bill units are rolled up to the paying parent bill unit at the end of the accounting cycle. Therefore, you must run the **pin_roll_up_ar_items**, **pin_update_journals**, and **pin_bill_accts** utilities in the same order at the end of each accounting cycle. See "[Running Wholesale Billing](#)".

Trial Billing for Wholesale Hierarchies

When you perform trial billing for wholesale bill unit hierarchies, you must run the billing for nonpaying child bill units (*wholesale child accounts*) before running the billing for the paying parent bill unit (*wholesale parent account*). You can perform this by running the **pin_trial_bill_accts** utility with the **-pay_type** parameter. For more

information, see the discussions about trial billing for bill unit hierarchies and sharing groups and running trial billing according to payment type in *BRM Configuring and Running Billing*.

Support for A/R Activities

For wholesale bill unit hierarchies, the support for A/R activities varies from level to level:

- **Account level.** Adjustments, disputes, and settlements can be performed only at the parent bill unit level after billing.
- **Bill level.** Adjustments, disputes, and settlements can be performed only after billing at the parent bill unit level.
- **Event level.** Adjustments, disputes, and settlements can be performed at the child bill unit level before and after billing.
- **Item level.** Adjustments and disputes can be performed at the child bill unit level only before billing. After billing, adjustments and disputes are allowed only at the parent bill unit level. However, settlements can be performed before and after billing at the child bill unit level.

Write-offs can be performed only at the parent bill unit level after billing.

Specifying Search Criteria for Retrieving Items, Events, and Bills

To retrieve a list of items, events, or bills, BRM uses the following A/R and payment opcodes:

- PCM_OP_AR_GET_ACTION_ITEMS
- PCM_OP_AR_GET_ACCT_ACTION_ITEMS
- PCM_OP_AR_GET_BAL_SUMMARY
- PCM_OP_AR_GET_ACCT_BAL_SUMMARY
- PCM_OP_AR_GET_BILL_ITEMS
- PCM_OP_AR_GET_BILLS
- PCM_OP_AR_GET_DISPUTE_DETAILS
- PCM_OP_AR_GET_DISPUTES
- PCM_OP_AR_GET_ACCT_BILLS
- PCM_OP_PYMT_ITEM_SEARCH
- PCM_OP_PYMT_MBI_ITEM_SEARCH
- PCM_OP_PYMT_SELECT_ITEMS

Based on the search criteria provided as input, these opcodes search all the bill units in a hierarchy. This can have an impact on the wholesale billing performance if you have large wholesale bill unit hierarchies.

To improve the wholesale billing performance, you must restrict the search to find items only for the specific bill units instead of searching all the bill units in a wholesale bill unit hierarchy.

Similarly, BRM uses the PCM_OP_AR_GET_ITEM_DETAIL and PCM_OP_AR_GET_ITEMS opcodes to retrieve the details of an A/R item or bill item for a bill unit. For wholesale hierarchies, these opcodes cannot retrieve all the details about the rolled-up items. For example, for the A/R items of the paying parent bill unit, these opcodes cannot retrieve the corresponding transfer events for the rolled-up disputes and settlements. Therefore, you must modify the search to retrieve only the details that are available for wholesale hierarchies.

For information on the search criteria for these opcodes, see *BRM Opcode Guide*.

Moving Bill Units into or out of Wholesale Hierarchies

You can move the nonpaying child bill units into or out of a wholesale bill unit hierarchy. You can also move them between wholesale bill unit hierarchies. Before moving a bill unit, ensure that there are no pending items or payments in that bill unit.

When you move the nonpaying child bill unit to another hierarchy, all the items in that bill unit are associated with the corresponding items of the new paying parent bill unit and the new parent is billed for them. If a bill item does not exist in the new parent, it is created and the charges are rolled up to that item.

Configuration Changes

You set the **StagedBillingFeeProcessing** business parameter to 4 to enforce cycle fee processing prior to billing and apply cycle forward fees in parallel by service with service charges aggregated to a single account item. With this enhancement, this option is no longer required. Now, the valid values for this parameter are only 0, 1, 2, and 3.

Opcode Changes

To support wholesale billing, the following opcode changes have been made:

- The following new public opcode has been introduced:
 - PCM_OP_SUBSCRIPTION_POL_PRE_CYCLE_DISCOUNT
- The following public opcodes have been modified:
 - PCM_OP_AR_ACCOUNT_ADJUSTMENT
 - PCM_OP_AR_ACCOUNT_WRITEOFF
 - PCM_OP_AR_BILL_ADJUSTMENT
 - PCM_OP_AR_BILL_DISPUTE
 - PCM_OP_AR_BILL_SETTLEMENT
 - PCM_OP_AR_BILL_WRITEOFF
 - PCM_OP_AR_BILLINFO_WRITEOFF
 - PCM_OP_AR_EVENT_ADJUSTMENT
 - PCM_OP_AR_EVENT_DISPUTE
 - PCM_OP_AR_EVENT_SETTLEMENT
 - PCM_OP_AR_ITEM_ADJUSTMENT
 - PCM_OP_AR_ITEM_DISPUTE
 - PCM_OP_AR_ITEM_SETTLEMENT
 - PCM_OP_AR_ITEM_WRITEOFF
 - PCM_OP_BILL_MAKE_BILL
 - PCM_OP_CUST_COMMIT_CUSTOMER
 - PCM_OP_CUST_CREATE_CUSTOMER

- PCM_OP_CUST_DELETE_ACCT
- PCM_OP_CUST_SET_BILLINFO
- PCM_OP_INV_MAKE_INVOICE
- PCM_OP_INV_POL_PREP_INVOICE
- PCM_OP_PYMT_COLLECT
- PCM_OP_PYMT_MBI_DISTRIBUTE
- PCM_OP_SUBSCRIPTION_RERATE_REBILL

For more information, see "[Opcode Changes](#)".

Schema Changes

To support wholesale billing, the following schema changes have been made:

- The following tables are added:
 - EVENT_ACT_ROLLUP_ITEMS_T
 - TMP_AR_ITEM_TO_ROLL_UP_T
- The following columns are added in the ITEM_T table:
 - ITEM_CLASS
 - AR_ITEM_OBJ
- The following column is added in the TMP_JOURNALS_TO_PROCESS_T table:
 - AR_BILLINFO_OBJ
- The following indexes are added:
 - I_TMP_AR_ITM_ROLLUP__ID
 - I_TMP_AR_ITM_ROLLUP__STATUS
 - I_ITEM_AR_ITEM_OBJ__ID
 - I_TMP_JOURNALS_TO_PROCESS__AR

Storable Class Changes

To support wholesale billing, the following changes have been made:

- The following new storable classes have been introduced:
 - **/event/activity/roll_up**
 - **/tmp_ar_item_to_roll_up**
- The following storable class has been modified:
 - **/tmp_journals_to_process**

For more information, see "[Storable Class Changes](#)".

Utility Changes

To support wholesale billing, the following new utilities have been introduced:

- [pin_roll_up_ar_items](#)

- [pin_update_journals](#). This utility replaces the `pin_update_items_journals` utility.

pin_update_journals

Use the `pin_update_journals` utility to process temporary journals (*/tmp_journals_to_process* object) of the nonpaying child bill units and roll them up to the paying parent bill unit. You must run this utility before billing the paying parent bill unit.

To connect to the BRM database, the `pin_update_journals` utility needs a configuration file in the directory from which you run the utility. See "Connecting BRM Utilities" in *BRM System Administrator's Guide*.

Location

`BRM_home/bin`

Syntax

```
pin_update_journals [-verbose] [-help]
```

Parameters

-verbose

Displays information about successful or failed processing as the utility runs.

-help

Displays the syntax and parameters for this utility.

Results

If the `pin_update_journals` utility does not notify you that it was successful, look in the utility log file (`default.pinlog`) to find any errors. The log file is either in the directory from which the utility was started or in a directory specified in the configuration file.

Error Handling

When the `pin_update_journals` utility encounters an error while processing the A/R items in the temporary tables, it sets the `PIN_FLD_BILLING_STATUS` billing status field of the paying parent bill unit (*/billinfo* object) to `PIN_BILL_ERROR`. In addition, it sets the appropriate bit of the `PIN_FLD_BILLING_STATUS_FLAGS` field of the */billinfo* object as `PIN_BILL_FLAGS_UPDATE_JOURNALS_ERROR` (bit value 0x2000).

After you have resolved the processing errors, you can reprocess the A/R items by running the `pin_update_journals` utility again.

pin_roll_up_ar_items

Use the `pin_roll_up_ar_items` utility to process temporary A/R items (*/tmp_ar_item_to_roll_up* object) of the nonpaying child bill units and roll the balance impact up to the corresponding A/R items in the paying parent bill unit. You can run multiple threads of `pin_roll_up_ar_items` to process A/R items for different paying parent bill units.

You must run this utility before billing the paying parent bill unit. For more information, see "[Rolling A/R Actions Up to the Wholesale Parent](#)".

To connect to the BRM database, the **pin_roll_up_ar_items** utility needs a configuration file in the directory from which you run the utility. See the discussion about connecting BRM utilities in *BRM System Administrator's Guide*.

Location

BRM_home/bin

Syntax

```
pin_roll_up_ar_items [-verbose] [-help]
```

Parameters

- **verbose**: Displays information about successful or failed processing as the utility runs.
- **help**: Displays the syntax and parameters for this utility.

Results

If the **pin_roll_up_ar_items** utility does not notify you that it was successful, look in the utility log file (**default.pinlog**) to find any errors. The log file is either in the directory from which the utility was started or in a directory specified in the configuration file.

Error Handling

When the **pin_roll_up_ar_items** utility encounters an error while processing the A/R items in the temporary tables, it sets the PIN_FLD_BILLING_STATUS billing status field of the paying parent bill unit (**/billinfo** object) to PIN_BILL_ERROR. In addition, it sets the appropriate bit of the PIN_FLD_BILLING_STATUS_FLAGS field of the **/billinfo** object as PIN_BILL_FLAGS_UPDATE_ITEMS_ERROR (bit value 0x4000).

After you have resolved the processing errors, you can reprocess the A/R items by running the **pin_roll_up_ar_items** utility again.

Delay Interval Can be Configured for Resolving Failed Payments

In previous releases, when the **pin_recover** utility and a custom application were run in parallel to resolve failed credit card or debit card payments, duplicate transaction IDs were created.

With this patch, a new entry, **event_search_delay**, has been introduced in the *BRM_home/apps/pin_billd/pin.conf* file to specify the delay interval for resolving failed payments. When set, the **pin_recover** utility processes only the events (**/event/billing/charge/cc**) that were created before the specified delay interval.

To configure the delay interval for resolving failed payments:

1. Open the billing utility configuration file (*BRM_home/apps/pin_billd/pin.conf*) in a text editor.
2. Search for the **event_search_delay** entry.
3. Specify the delay interval:

```
- pin_recover event_search_delay value
```

where *value* is the delay interval in seconds. By default, it is set to 0.

For example, setting the **event_search_delay** entry to 300 delays the event search for resolving failed payments by 5 minutes:

```
- pin_recover event_search_delay 300
```

4. Save and close the file.

Enhanced Data Protection

BRM now includes the following security enhancements to protect the subscriber's personal data:

- Deleting closed accounts and all the related objects (such as events, items, bills, invoices, journals, newsfeeds, and user activities) and the audit data automatically from BRM after the specified retention period.
- Purge deleted accounts and the associated customer data synchronously on BRM and Oracle Communications Elastic Charging Engine (ECE). For more information, see the discussion about purging accounts from the ECE cache in *BRM ECE Release Notes*.
- Securing communications between BRM applications and the database. See the discussion about configuring SSL for the BRM database in *BRM 12.0 Patch Set Installation Guide*.

To support the security enhancements, the following changes have been made in BRM:

- The **ClosedAcctsRetentionMonths** business parameter has been introduced to specify the retention period for the closed accounts. See "[Specifying Retention Period for Closed Accounts](#)".
- The **pin_del_closed_accts** utility has been introduced to delete closed accounts from BRM after the specified retention period. See "[pin_del_closed_accts](#)".
- The PCM_OP_CUST_DELETE_ACCT opcode has been modified to ensure that all the BRM objects and audit entries containing the subscriber's personal data are purged.

Note:

You can use the PCM_OP_CUST_DELETE_ACCT opcode to delete accounts in a production system, but ensure that you use this opcode with care.

For more information on the PCM_OP_CUST_DELETE_ACCT opcode, see the discussion about deleting accounts in *BRM Opcode Guide*.

 **Note:**

The PCM_OP_CUST_DELETE_ACCT opcode does not delete all the custom objects. You can write a custom logic to clean up the custom objects in BRM when the **/event/notification/account/pre_delete** and **/event/notification/account/delete events** are generated by the PCM_OP_CUST_DELETE_ACCT opcode.

Specifying Retention Period for Closed Accounts

You can specify the number of months the closed accounts must be retained in BRM by setting the **ClosedAcctsRetentionMonths** parameter in the **customer** instance of the **/config/business_params** object.

To specify the retention period for closed accounts:

1. Go to *BRM_home/sys/data/config*.
2. Create an XML file from the **/config/business_params** object:

```
pin_bus_params -r BusParamsCustomer bus_params_customer.xml
```
3. Set the **ClosedAcctsRetentionMonths** entry to the number of months that you want to retain the closed accounts:

```
<ClosedAcctsRetentionMonths>number_of_months</ClosedAcctsRetentionMonths>
```
4. Save the file as **bus_params_customer.xml**.
5. Load the XML file into the BRM database:

```
pin_bus_params bus_params_customer.xml
```
6. Stop and restart the CM.
7. (Multischema systems only) Run the **pin_multidb** script with the -R CONFIG parameter. For more information, see *BRM System Administrator's Guide*.

Deleting Closed Accounts

You can delete closed accounts in BRM after the retention period by using the **pin_del_closed_accts** utility.

To delete closed accounts:

1. Go to the *BRM_home/apps/pin_bildd* directory.
2. Do the following as appropriate:

 **Note:**

To delete all closed child accounts in a hierarchy and sharing groups, run these commands in the following order.

- To delete all closed nonpaying child accounts at different levels in a hierarchy:


```
pin_del_closed_accts -subord -leaf
pin_del_closed_accts -subord
```

- To delete member accounts in sharing groups:

```
pin_del_closed_accts -members_sharing
```

- To delete paying child accounts at different levels in a hierarchy:

```
pin_del_closed_accts -members_billing
```

Note:

Run this command for each paying account in a hierarchy. For example, if there are two paying accounts in a hierarchy, run this command twice to delete both paying accounts.

3. Run the following command, which deletes all remaining closed accounts, including the top-level parent account in the hierarchy:

```
pin_del_closed_accts
```

4. If you want to delete specific closed accounts by using a file, run the following command:

Note:

Run the **pin_del_closed_accts -file** command only if you want to delete specific accounts. Ensure you use this command with care.

```
pin_del_closed_accts -file file_name
```

For example:

```
pin_del_closed_accts -file closed_accts_list.txt
```

The utility deletes the accounts specified in the input file. You must provide the account details in list format. For example:

```
0 PIN_FLD_RESULTS      ARRAY [0] allocated 20, used 1
1   PIN_FLD_POID       POID [0] 0.0.0.1 /account 123 0
0 PIN_FLD_RESULTS      ARRAY [1] allocated 20, used 1
1   PIN_FLD_POID       POID [0] 0.0.0.1 /account 234 0
```

For more information about the utility and its parameters, see "[pin_del_closed_accts](#)".

pin_del_closed_accts

Use the **pin_del_closed_accts** utility to delete closed accounts from BRM. This utility calls the `PCM_OP_CUST_DELETE_ACCT` opcode to delete the closed accounts that are older than the specified retention period. For specifying the retention period, see "[Specifying Retention Period for Closed Accounts](#)".

To connect to the BRM database, the **pin_del_closed_accts** utility needs a configuration file in the directory from which you run the utility. See the discussion about connecting BRM utilities in *BRM System Administrator's Guide*.

Location

BRM_home/bin

Syntax

```
pin_del_closed_accts -subord [-leaf]
                    -members_sharing
                    -members_billing
                    -file file_name
                    [-verbose] [-help]
```

Parameters

- **subord [-leaf]**: Deletes the closed nonpaying child accounts at the bottom of the hierarchy.
- **subord**: Deletes the remaining closed nonpaying child accounts which are parents of other child accounts at the different levels of the hierarchy. Running the **pin_del_closed_accts** utility with this parameter does not delete the top-level parent account in the hierarchy.

You need to run the **pin_del_closed_accts** utility without any parameters after deleting all the paying and nonpaying child accounts at different levels in the hierarchy to delete the top-level parent account.
- **members_sharing**: Deletes the member accounts of the sharing groups; for example, discount and charge sharing groups.
- **members_billing**: Deletes the closed paying accounts in the hierarchy that are used for billing purposes.
- **file file_name**: Deletes the accounts specified in the input file. The *file_name* is the name and location of the file that contains the list of accounts for deletion. The account details in this file must be in the flist format.

Note:

Running the **pin_del_closed_accts** utility with this parameter deletes all the accounts specified in the input file even if the accounts are not older than the retention period. When you use this parameter, ensure that the input file contains only the closed accounts that need to be deleted.

- **verbose**: Displays information about successful or failed processing as the utility runs.
- **help**: Displays the syntax and parameters for this utility.

Results

The **pin_del_closed_accts** utility notifies you when it successfully deletes the closed accounts and the associated customer data.

If the **pin_del_closed_accts** utility does not notify you that it was successful, look in the utility log file (**default.pinlog**) to find any errors. The log file is either in the directory from which the utility was started or in a directory specified in the configuration file.

After you have resolved the errors, you can delete the closed accounts by running the **pin_del_closed_accts** utility again.

Enhanced Security for Root Wallet

When you run the **pin_crypt_app** utility with the **-genrootkey** parameter, BRM now prompts for the root wallet password. This ensures that the root wallet is secured.

For more information, see the discussion about modifying the root encryption key in *BRM Developer's Guide*.

Support for Rolling Back the BRM Patch Set

BRM now allows you to roll back a BRM patch set. For example, if you experience issues after installing BRM 12.0 Patch Set 1, you can roll back BRM to 12.0.

For more information, see the discussion about rolling back a patch set in *BRM 12.0 Patch Set Installation Guide*.

BRM 12.0 Is Now Certified with Mozilla Firefox 58.0

Currently, BRM 12.0 is certified with Mozilla Firefox 54.0.1. With this patch, BRM 12.0 is certified with Mozilla Firefox 58.0.

For more information, see *BRM Compatibility Matrix*.

BRM 12.0 Is Now Certified with Perl 5.28.0

Currently, BRM 12.0 is certified with Perl 5.24.0. With this patch, BRM 12.0 is certified with Perl 5.28.0.

For more information, see *BRM Compatibility Matrix*.

BRM 12.0 Is Now Certified with Paymentech 120 Byte Batch Version 3.0.0 R 12.4 and Online Authorization Version 7.4 R12.4

Currently, BRM 12.0 is certified with Paymentech 120 Byte Batch Version 3.0.0 R 4.2 and Paymentech Online Authorization Version 7.4 R5. With this patch, BRM 12.0 is certified with Paymentech 120 Byte Batch Version 3.0.0 R 12.4 and Paymentech Online Authorization Version 7.4 R12.4.

For more information, see *BRM Compatibility Matrix*.

BRM 12.0 Is Now Certified with Tomcat 8.5.32

Currently, BRM 12.0 is certified with Tomcat version 8.5.16. With this patch, BRM 12.0 is certified with Tomcat version 8.5.32.

For more information, see *BRM Compatibility Matrix*.

3

New Features in ECE

The Oracle Communications Elastic Charging Engine (ECE) 12.0 Patch Set releases include several new features.

Caution:

Deploying charging for 5G with HTTP Gateway (5G CHF) requires a cloud native deployment of ECE and BRM components. The HTTP Gateway can be used only on an ECE cloud native system.

Topics in this document:

- [New Features in ECE 12.0 Patch Set 8](#)
- [New Features in ECE 12.0 Patch Set 7](#)
- [New Features in ECE 12.0 Patch Set 6](#)
- [New Features in ECE 12.0 Patch Set 5](#)
- [New Features in ECE 12.0 Patch Set 4](#)
- [New Features in ECE 12.0 Patch Set 3](#)
- [New Features in ECE 12.0 Patch Set 2](#)
- [New Features in ECE 12.0/12.0 Patch Set 1](#)

New Features in ECE 12.0 Patch Set 8

ECE 12.0 Patch Set 8 includes the following enhancements:

- [Extensions for Midsession-Rated Events](#)
- [Reasons Now Included for Midsession-Rated Events](#)
- [Midsession-Rated Events Created When USU Block Missing](#)
- [Extensions for Sy Request-Response Flow](#)
- [Active-Active Mode Can Now Process Requests in Local Sites](#)
- [ECE Enhancements for Processing Free Allowances](#)
- [Support for Tax Selectors and Tax Exemption Selectors](#)
- [ECE Supports Flexible Proration Settings](#)
- [ECE Cloud Native Contains New ConfigLoader Pod](#)
- [Support for Prepaid Usage Overage](#)
- [Rated Event Publisher Support for CDR Streaming](#)

Extensions for Midsession-Rated Events

By default, ECE generates a rated event for a network session only when a Terminate operation ends the session. You can configure ECE to generate a rated event in the middle of an ongoing active session based on trigger criteria. In previous releases, the available trigger criteria for creating a midsession-rated event was limited to the following:

- Duration (for example, create a rated event every 4 hours that a session is active)
- Quantity (for example, create a rated event whenever downloaded data totals 70 MB or more)
- Time of day (for example, create a rated event daily at 23:00:00 during the life of the session)

You can also trigger a midsession-rated event based on custom criteria, such as tariff changes or specific network conditions. To do so, you extend ECE at the pre-rating or post-rating extension points.

For more information, see the following:

- These ECE SDK sample programs: **SamplePreRatingMidSessionExtension** and **SamplePostRatingMidSessionExtension**
- "Configuring ECE to Generate Midsession Rated Events" in *ECE Implementing Charging*

Reasons Now Included for Midsession-Rated Events

When creating midsession-rated events, ECE now adds why a rated event was triggered during an active session. For example, the reason could be that:

- The configured volume was reached
- The configured duration was reached
- The configured time of the day was crossed

In previous releases, ECE did not include the reason for creating the midsession-rated event.

For more information, see "Viewing Reasons for Midsession-Rated Events" in *ECE Implementing Charging*.

Midsession-Rated Events Created When USU Block Missing

When the network sends a Final Unit Indicator (FUI) followed by a top up, ECE generates a reauthorization request (RAR). The network then sends an update request, sometimes not including the Used Service Units (USU) block for the ongoing session. When this occurs, ECE can now generate a midsession-rated event. This ensures that any part of the reservation consumed by the account is reported, preventing revenue loss.

In previous releases, if the USU block was missing from the update request, ECE cleared the reservation and information about the amount of the reservation consumed, causing revenue exposure.

For information, see "Generating Midsession-Rated Events When USU Block Missing" in *ECE Implementing Charging*.

Extensions for Sy Request-Response Flow

The Diameter Sy request-response flow now includes extension hooks for customizing Sy processes. For example, you could customize ECE to suspend Sy signaling when a subscriber's account is suspended.

For information, see "Diameter-Request Processing Extension Points" in *ECE Implementing Charging*.

Active-Active Mode Can Now Process Requests in Local Sites

In previous releases, ECE active-active disaster recovery configurations always processed usage requests according to the preferred site assignments in the **customerGroup** list. For example, if subscriber A's preferred primary site was site 1, ECE would always process subscriber A's usage requests in site 1. If subscriber A's usage request was received by production site 2, it was always sent to production site 1 for processing.

Now, you can configure the ECE active-active mode to process usage requests in the site that received the request, regardless of the subscriber's preferred site. For example, if a subscriber's usage request is received by production site 1, it is processed in production site 1. Likewise, if the usage request is received by production site 2, it is processed in production site 2.



Note:

This configuration does not apply to usage charging requests for members of a sharing group. Usage requests for sharing group members continue to be processed in the same site as the sharing group parent.

For more information, see "Processing Usage Requests in the Site Received" in *BRM System Administrator's Guide*.

ECE Enhancements for Processing Free Allowances

ECE includes enhancements for processing free allowances, such as 100 free minutes or 5 free movie rentals.

In previous releases, ECE first applied charges for all usage and then applied a discount for the granted allowance. This ensured that the customer was not charged until the allowance was exhausted. For example, assume a customer owned a plan that granted 500 GB of free data per month and charged \$10 per 100 GB of data. If a customer used 600 GB of data in a month, ECE would first apply a \$60 charge for the 600 GB of data and then apply a \$50 discount for the 500 GB of free data. Applying charges first and then applying discounts was based on how PDC was being configured to address allowances.

PDC now allows you to create product offerings that consume granted allowances before applying any charges. To improve processing performance, ECE also supports this rating flow. ECE can now consume free allowances first and then charges for usage based on pricing configurations in PDC. Using the previous example, if the customer uses 600 GB of

data in a month, ECE now consumes the 500 GB of free data and then applies a \$10 charge for the remaining 100 GB of data usage.

For information, see "Processing Granted Allowances Before Applying Usage Charges" in *ECE Implementing Charging*.

Support for Tax Selectors and Tax Exemption Selectors

ECE supports the tax selectors and tax exemption selectors configured in PDC for taxation during charging.

Tax selectors and tax exemption selectors are used to apply taxes based on account, service, event, and profile attributes. This allows you to choose whether to use the direct tax code or to choose the tax code using the selectors in the rate plan while creating the product.

For more information, see "Configuring Taxation in ECE" in *ECE Implementing Charging*.

ECE Supports Flexible Proration Settings

ECE now supports flexible proration settings. When charging for usage, ECE now uses the charge-offer-level settings for proration that are configured in PDC. For more information, see "[PDC Supports Flexible Proration Options](#)".

ECE Cloud Native Contains New ConfigLoader Pod

In previous ECE cloud native releases, ConfigLoader was a Kubernetes job. This meant that if an ECE cluster went down, you had to manually restart the cluster.

In Patch Set 8, ConfigLoader has been changed to a Pod. This enables data to be automatically loaded into the ECS pods from the persistence database whenever the ECE cluster restarts.

For more information, see "Enabling Persistence in ECE" in *BRM Cloud Native System Administrator's Guide*.

Support for Prepaid Usage Overage

You can now configure ECE to capture any overage amounts by prepaid customers during an active session, which can help you prevent revenue leakage. If the network reports that the number of used units during a session is greater than a customer's available allowance, ECE charges the customer up to the available allowance. It then creates a CDR with information about the overage amount and sends it to a new ECE Overage Kafka topic.

To enable ECE to check for and capture any overage:

1. Access the ECE configuration MBeans in a JMX editor, such as JConsole. See "Accessing ECE Configuration MBeans" in *ECE Implementing Charging*.
2. Expand the **ECE Configuration** node.
3. Expand **charging.server**.
4. Expand **Attributes**.
5. Set the **checkReservationOverImpact** attribute to **true**.

For more information, see "Configuring ECE to Support Prepaid Usage Overage" in *ECE Implementing Charging*.

Rated Event Publisher Support for CDR Streaming

ECE's Rated Event Publisher now supports streaming events to and from Kafka. This enables near-real-time reporting of rated event data and makes rated event data available for consumption from Apache Kafka by third party systems as well as by BRM. See "Event Streaming Mode (Patch Set 8 or later)" in *Loading Rated Events* for more information.

New Features in ECE 12.0 Patch Set 7

ECE 12.0 Patch Set 7 includes the following enhancements:

- [Time-Based Price Overrides and Dynamic Usage Rating](#)
- [Bundle Renewal Triggered from ECE](#)
- [Non-Linear Rating Configurable by Event Type](#)
- [Balance Query Returns Lifecycle Information](#)
- [Current Balance Now Returned with Balance Impact](#)
- [First Usage Activation Enhancements](#)
- [Network Signaling Optimization](#)
- [Balance Monitoring and Group Credit Limits](#)
- [Product Sharing](#)
- [Logging of Usage Rating Failures](#)
- [Robust Handling of Configuration Data](#)
- [Quorum Policy Handling in ECE](#)
- [Rolling Upgrade Enhancements](#)
- [Access to Bundle Information in ECE](#)
- [Ability to Call a BRM Opcode from ECE](#)
- [CDR Generator Enhancements](#)
- [Policy-Based Charging Notification Enhancements](#)
- [Usage Consumption Management Report Certification](#)

Time-Based Price Overrides and Dynamic Usage Rating

Data related to time-based price overrides is now synchronized with ECE. Multiple overrides are available for different date ranges for both charge offers and discount offers.

ECE also now allows you to override currency or non-currency resources for a limited or unlimited period of time without customization. These overrides can be defined at the balance impact level for charge offers and discount offers.

For more information, see "Managing Dynamic Charging Overrides for Online Sessions" in *ECE Implementing Charging*.

Bundle Renewal Triggered from ECE

If an active session crosses a bundle validity boundary, ECE now requests a bundle renewal from BRM. BRM then renews the bundle and the updated grant is synchronized with ECE, to ensure that rating continues correctly.

Non-Linear Rating Configurable by Event Type

Previously, you could only configure non-linear rating at the product level. Now, you can configure non-linear rating for specific events configured for a product. This provides more flexibility when configuring products.

For more information, see "Configuring Incremental Rating for Midsession Rating Condition Changes" in *ECE Implementing Charging*.

Balance Query Returns Lifecycle Information

The balance query operation now returns the lifecycle state and lifecycle expiry information. The calling application can use this information to take appropriate action, such as disconnecting a voice call if the subscriber is in a suspended state.

Current Balance Now Returned with Balance Impact

You can now configure ECE to include the current balance and current loan balance as part of the balance impact in rated events. CSRs will then have this information available if a customer calls to inquire about a particular transaction.

First Usage Activation Enhancements

Data for configuring bundles or packages to start on first usage is now synchronized with ECE. When you use this option, the first usage of any charge offer or discount offer activates all of the offers in the bundle or package.

Network Signaling Optimization

Many times, bundles expire at midnight of a particular day. If your customers are using the bundles at that time, the renewal and request messages that are transmitted at midnight can cause an undue network load. To prevent this, ECE can now randomize the validity times for a service so that renewal requests do not occur simultaneously.

For more information, see "Optimizing Network Signaling" in *ECE Implementing Charging*.

Balance Monitoring and Group Credit Limits

ECE now joins BRM in supporting balance monitoring and monitor group shared balances. You can set up shared balances for specific services and ECE will update BRM with the balance impacts after rating.

Product Sharing

ECE, like BRM, has added support for product sharing. This allows you to change rates once at the group owner level, rather than having to change each of the subscriber accounts.

In support of this, ECE has added product sharing agreements. During usage rating, ECE will consider priority to determine whether to use the shared package.

Logging of Usage Rating Failures

A new logger has been created for failed usage rating requests. Failed usage requests now include the reason for the failure, along with the customer and session identifiers. For example, in the case of a Diameter Gateway usage rating failure with `NO_QUALIFIED_CHARGE_OFFERS`, you might see:

```
ERROR - - - - Failing Usage Request for subscriber ID : <<PUID>>, session
ID :<<Session ID>,
reasons : [NO_QUALIFIED_CHARGE_OFFERS, ZERO_RUM_QUANTITY]
```

This feature is configuration-driven and works even if debug mode is not enabled.

For more information, see "Troubleshooting Failed Usage Requests" in *BRM System Administrator's Guide*.

Robust Handling of Configuration Data

ECE uses the **charging-settings.xml** file to store configuration information. This file is loaded into the cache when ECE starts, and run-time changes can be made using JConsole. Starting with this release, if this configuration file is accidentally deleted or corrupted while ECE is running, you can rebuild the file from the cache. The rebuilt file will also contain any run-time configuration changes that were made using JConsole.

For more information, see "Troubleshooting a Corrupted ECE Configuration File" in *BRM System Administrator's Guide*.

Quorum Policy Handling in ECE

You can now set a timeout survivor quorum, which is the minimum number of cluster members that must remain running to avoid data loss when the cluster service terminates suspect members. For more information:

- For ECE cloud native environments, see "Adding Elastic Charging Engine Keys" in *BRM Cloud Native Deployment Guide*.
- For ECE on-premise systems, see "Configuring Cluster Quorum Policy" in *BRM System Administrator's Guide*.

Rolling Upgrade Enhancements

ECE previously had a single wait parameter (**rollingUpgradeGatewayReadinessWait** in the **ece.properties** file) to ensure that all of the gateways bootstrapped completely during a rolling upgrade. However, different gateways can take different amounts of time to bootstrap.

ECE now has the following separate **ece.properties** parameters for each gateway so that each gateway does not have to wait for the time required by the slowest gateway:

- **rollingUpgradeEmGatewayReadinessWait**
- **rollingUpgradeCdrGatewayReadinessWait**
- **rollingUpgradeDiameterGatewayReadinessWait**
- **rollingUpgradeRadiusGatewayReadinessWait**
- **rollingUpgradeHttpGatewayReadinessWait**

For more information, see "Starting the Rolling Upgrade Process" in *ECE Installation Guide*.

Access to Bundle Information in ECE

Bundle (deal) information is now cached in ECE, so that it can be sent with rated events. The deal POID, deal name, and purchased bundle name are available through APIs in extension points, so that you can use this information when you are extending the functionality.

Ability to Call a BRM Opcode from ECE

It is now possible to call a BRM opcode from ECE as a response to an event by extending BRM Gateway functionality. Any updates to data due to this opcode call in BRM are synchronized back into ECE by the update flow through External Manager (EM) Gateway.

For more information, see "BRM Gateway Request Processing Extension Points" in *ECE Implementing Charging*.

CDR Generator Enhancements

Several enhancements to the 5G Charging Function CDR Generator have been included in this release:

- **Scalability:** It is now possible to have more than one CDR Gateway and CDR Formatter per schema.
- **Disaster Recovery:** The system will now process any orphan records on a failed site as soon as the site becomes active, in parallel with live records. In addition, a configuration parameter has been added to set the duration after which, if no updates to the record occur, the record will be considered an orphan.
- **Performance:** Many performance improvements are provided in the release, including Oracle Database and NoSQL CDR data store improvements, optimized CPU utilization for object mappers, and improved metrics for CDR generation and the HTTP gateway.
- **CDR Files:** A new parameter has been introduced to limit the number of unrated CDRs included in a single file when CDR Formatter is writing out the CDRs.

Previously, only the timestamps were included in the file name (in the format *StartTimeStamp_EndTimeStamp.Extension*). The file name format has been updated to *ClusterName_StartTimeStamp_EndTimeStamp_SequenceNumber.Extension* in order to uniquely distinguish CDR files generated by different clusters, and also to allow for proper sequencing of files so that the downstream systems can process them in the correct order.

For more information, see "Generating CDRs for External Systems" in *ECE Implementing Charging*.

Policy-Based Charging Notification Enhancements

Previously, all notifications for policy label breaches were sent in parallel. Now, policy label breaches for the same session are sent sequentially, with the later message being sent after an acknowledgment is received for the earlier notification. Notifications for breaches in different sessions are still sent in parallel.

For more information, see "How ECE Processes Policy Requests for Online Network Mediation System" in *ECE Implementing Charging*.

Usage Consumption Management Report Certification

The "TMF677 - Usage Consumption Management" report, added in Patch Set 3, has now received Open API certification from TM Forum.

New Features in ECE 12.0 Patch Set 6

ECE 12.0 Patch Set 6 includes the following enhancements:

- [ECE SDK Includes New Rated Events Custom Plugin](#)
- [ECE Now Supports Temporary Credit Limits](#)
- [ECE Now Supports Loan Management](#)
- [ECE Can Now Record Failed Usage Requests](#)
- [HTTP Gateway Supports Charging Type-Based Configuration for CDR Generation](#)
- [ECE Supports 5G SA Nchf_OfflineOnlyCharging Service](#)
- [Enhancements for HTTP Gateway Registration with an NRF](#)
- [ECE REST API Supports New CHF Operation Types](#)
- [HTTP Gateway Now Determines Charging Type](#)

ECE SDK Includes New Rated Events Custom Plugin

The ECE SDK package includes the new **SampleRatedEventFormatterKafkaCustomPlugin.java** sample plug-in, which writes fully rated events into a JSON file that is published to Apache Kafka topics.

For more information, see "ECE Sample Programs" in *ECE Implementing Charging*.

ECE Now Supports Temporary Credit Limits

When creating or modifying an account in BRM, you can add temporary credit limits to a customer's balance of minutes, US dollars (USD), or so on. For example, you could add a temporary credit limit of 400 USD that is valid from 1 November 2022 through 1 January 2023. Account Synchronization Manager now includes this information about temporary credit limits when synchronizing account updates between BRM and ECE.

The temporary credit limit is used by:

- The ECE Balance Query Java API when querying subscriber balances.
- ECE when performing credit breach checks.

To synchronize temporary credit limits between BRM and ECE, ensure that:

- Your event notification file (**pin_notify_ifw_sync** or **pin_notify_kafka_sync**) includes the **/event/notification/billing/temp_limit** and **/event/billing/limit** events.

For more information, see the *ECE_home/occeserver/brm_config/pin_notify* file.

- Your payload configuration file (**payloadconfig_ifw_sync.xml** or **payloadconfig_kafka_sync.xml**) includes the **TemporaryLimit** and **TemporaryLimitExpiry** business events and their event definitions.

For more information, see the *ECE_home/occeserver/brm_config/payloadconfig_ece_sync.xml* file.

ECE Now Supports Loan Management

You can grant loans in BRM to prepaid customers when their credit is insufficient for rating (dynamic loans), or when their balance falls below a configured threshold (automatic loans). Account Synchronization Manager now includes information about any loan balance items when synchronizing customer data between BRM and ECE.

Loan balance items are used by:

- The ECE Balance Query Java API when returning user balances.
- ECE when creating loan threshold notifications during the rating process.

To synchronize loan balance items between BRM and ECE, ensure that:

- Your event notification file (**pin_notify_ifw_sync** or **pin_notify_kafka_sync**) includes these events:

- **/event/billing/loan_credit**
- **/event/billing/loan_debit**
- **/event/billing/loan_fee**
- **/event/billing/loan_grant**
- **/event/billing/loan_late_fee**
- **/event/billing/loan_pullback**
- **/event/billing/loan_recovery**
- **/event/billing/loan_recovery_reversal**

For more information, see the *ECE_home/occeserver/brm_config/pin_notify* file.

- Your payload configuration file (**payloadconfig_ifw_sync.xml** or **payloadconfig_kafka_sync.xml**) includes these business events and their event definitions:

- **LoanCredit**
- **LoanDebit**
- **LoanFee**

- **LoanGrant**
- **LoanLateFee**
- **LoanPullback**
- **LoanRecovery**
- **LoanRecoveryReversal**

For more information, see the *ECE_home/occeserver/brm_config/payloadconfig_ece_sync.xml* file.

ECE Can Now Record Failed Usage Requests

ECE may occasionally fail to process usage requests. For example, a data usage request could fail because a customer has insufficient funds. You can now configure ECE to publish details about failed usage requests, such as the user ID and request payload, to the ECE failure topic in your Kafka server. Later on, you can reprocess the usage requests or view the failure details for analysis and reporting. For more information, see "Recording Failed ECE Usage Requests" in *ECE Implementing Charging*.

HTTP Gateway Supports Charging Type-Based Configuration for CDR Generation

By default, HTTP Gateway sends all 5G usage requests to ECE Server for online and offline charging. You can now configure HTTP Gateway to convert some of the usage requests into Call Detail Records (CDRs) based on the charging type. You can specify whether to route online charging requests, offline charging requests, or both to ECE Server for charging or to the CDR Gateway for generating CDRs.

For more information, see "About Using the HTTP Gateway" in *ECE Implementing Charging*.

ECE Supports 5G SA Nchf_OfflineOnlyCharging Service

ECE HTTP Gateway supports Nchf_OfflineOnlyCharging service operations for offline-only charging requests in 5G standalone (SA) deployments.

For more information, see *ECE 5G CHF Protocol Implementation Conformance Statement*.

Enhancements for HTTP Gateway Registration with an NRF

HTTP Gateway includes the following enhancements for NRF registration:

- You can now register one HTTP Gateway server with multiple NRF endpoints.
- You can now register multiple HTTP Gateways with one NRF endpoint.

For more information, see "Configuring Registration Details for the HTTP Gateway Server" in *ECE Implementing Charging*.

ECE REST API Supports New CHF Operation Types

The ECE REST API now supports the following CHF operation types:

- Creating an initial request for an offline-only charging session

- Updating a request for an offline-only charging session
- Ending an offline-only charging session

In addition, all ECE REST API endpoints are now documented in this new book: *REST API for Elastic Charging Engine*.

HTTP Gateway Now Determines Charging Type

HTTP Gateway now determines whether a usage request requires online or offline charging. It does so as follows:

- For INITIATE requests, based on the **multipleUnitUsage** block. If the block is present, the request needs online charging. If the block is missing, the request needs offline charging.
- For UPDATE requests, based on the value of the **quotaManagementIndicator** field in the request. If the value is set to **ONLINE_CHARGING**, the request needs online charging. If the field is missing or the value is set to **OFFLINE_CHARGING**, the request needs offline charging.
- For TERMINATE requests, based on the value of the **quotaManagementIndicator** field in the request. If the value is set to **ONLINE_CHARGING**, the request needs online charging. If the field is missing or the value is set to **OFFLINE_CHARGING**, the request needs offline charging.

For more information, see "Connecting ECE to a 5G Client" in *ECE Implementing Charging*.

New Features in ECE 12.0 Patch Set 5

ECE 12.0 Patch Set 5 includes the following enhancements:

- [ECE Now Generates CDRs for Unrated 5G Usage Events](#)
- [SSL Now Between ECE and Persistence Database](#)
- [ECE Cloud Native Supports Subscriber Tracing](#)

ECE Now Generates CDRs for Unrated 5G Usage Events

ECE can now generate call detail records (CDRs) for unrated 5G usage events. You need to enable generating CDRs in the HTTP Gateway. You can configure ECE to publish CDRs as files or send them to a Kafka messaging service.

For more information, see "Generating CDRs" in *ECE Implementing Charging* and "CHF-CDR Format" in *ECE 5G CHF Protocol Implementation Conformance Statement*.

SSL Now Between ECE and Persistence Database

The ECE installer now enables SSL communication between ECE and the Oracle persistence database by default. Also, the installer's ECE Persistence Database Details screen includes new fields.

For more information, see "Installing ECE by Using the GUI Installation" in *ECE Installation Guide*.

ECE Cloud Native Supports Subscriber Tracing

In ECE cloud native, you can now selectively trace your subscribers' sessions based on one or more subscriber IDs. You can also specify to trace and log selective functions, such as alterations (discounts), charges, and distributions (charge sharing), for each subscriber.

This feature introduces the following keys to the `oc-cn-ece-helm-chart/values.yaml` file:

```
subscriberTrace:
  filePath: "/home/charging/config/subscriber-trace.xml"
  logMaxSubscribers: "100"
  logMaxSubscriberSessions: "24"
  logExpiryWaitTime: "1"
  logCleanupInterval: "2"
  logLevel: "DEBUG"
  subscriberList: ""
```

For more information, see "Configuring Subscriber-Based Tracing for ECE Services" in *BRM Cloud Native System Administrator's Guide*.

Rated Event Formatter Changes for Rated Event Manager

The Rated Event Formatter (RE Formatter) was changed to support the Rated Event Manager feature. A new RE Formatter plug-in, **RatedEventManagerCdrPlugin**, is now available for sending rated events to BRM. For more information about configuring RE Formatter for the new plug-in, see "Configuring RE Formatter" in *BRM Loading Rated Events*.

For more information about the Rated Event Manager feature, see "[New Method for Loading Rated Events from ECE into BRM](#)".

New Features in ECE 12.0 Patch Set 4

ECE 12.0 Patch Set 4 includes the following enhancements:

- It now supports the active-active disaster recovery configuration. In this configuration, ECE can be deployed with multiple clusters across multiple geographic sites and each site performs active rating simultaneously. All sites utilize resources and continue rating operations after a site failure.

For more information, see "Configuring ECE for Disaster Recovery" in *BRM System Administrator's Guide*.

- It can now spawn multiple BRM gateway processes for Kafka-enabled ECE notification queues, with each process catering to a fixed set of customer notification handling.

For more information, see "Configuring Multiple BRM Gateway Instances" in *BRM System Administrator's Guide*.

- ECE persistence databases can now support multischema systems.

For more information, see "Enabling Multischema Support in Persistence Databases" in *BRM System Administrator's Guide*.

- ECE now supports 5G Charging Function (CHF) communication through a Services Communication Proxy (SCP).

For more information, see "Configuring Communication through SCP" in *ECE Implementing Charging*.

- The HTTP Gateway now supports secure communication through SSL. You configure the HTTP Gateway to use SSL when you install ECE.
See "Installing ECE" and "Post-Installation Tasks for an ECE Integrated Installation" in *ECE Installation Guide*.
 - ECE now supports the latest 3GPP standards.
For more information, see *ECE 5G CHF Protocol Implementation Conformance Statement* and *ECE Diameter Gateway Protocol Implementation Conformance Statement*.
 - HTTP Gateway now supports four request processing extension points:
 - **RequestReceived** extension. The role of the RequestReceived extension is to manipulate the charging data before the usage request is processed by HTTP Gateway and to provide an immediate response that bypasses the online charging system (OCS) completely.
 - **PreOCS** extension. The role of the PreOCS extension is to manipulate the mapped ECE usage request payload to perform enrichments that are not possible in the RequestReceived extension.
 - **PostOCS** extension. The role of the PostOCS extension is to manipulate the ECE usage request before the HTTP Gateway response is returned to the 5G client.
 - **PostOCSBalanceQuery** extension. The role of the PostOCSBalanceQuery extension is to manipulate the ECE usage response before the HTTP Gateway response is returned to the 5G client.
- For more information, see "Customizing Rating" in *ECE Implementing Charging*.

New Features in ECE 12.0 Patch Set 3

ECE 12.0 Patch Set 3 includes the following enhancements:

- [Apache Kafka Now Supported for Notification Handling](#)
- [Support for Configuring Diameter Gateway to Bypass Rating During ECE Downtime](#)
- [Diameter Gateway Now Supports the 5G Non-Standalone Architecture](#)
- [ECE Now Supports 5G CHF Protocol](#)
- [New HTTP Gateway for 5G Networks](#)
- [Active-Hot Standby and Segmented Active-Active DR Support Persistence-Enabled Deployments](#)

Apache Kafka Now Supported for Notification Handling

In addition to using Oracle WebLogic queues, you can now use Apache Kafka topics for notification handling. The following gateways support Kafka topics:

- HTTP Gateway supports only Apache Kafka topics. See "Connecting ECE to a 5G Client" in *ECE Implementing Charging*.
- Diameter Gateway supports both WebLogic queues and Kafka topics. See "Connecting ECE to a Diameter Client" in *ECE Implementing Charging*.

- BRM Gateway supports both WebLogic queues and Kafka topics. See "Configuring BRM Gateway" in *BRM System Administrator's Guide*.

Support for Configuring Diameter Gateway to Bypass Rating During ECE Downtime

During a planned maintenance or an unplanned downtime of ECE nodes, you can configure Diameter Gateway to continue receiving CCRs and responding to the service network to ensure continued service delivery. When Diameter Gateway is configured to bypass rating, it persists the CCRs to the Oracle NoSQL database. Later, when ECE is restored, you can replay the persisted CCRs to the ECE charging servers for rating and updating balance impacts. With this functionality, services can be rendered to the subscribers without any interruption. For more details, see *ECE Implementing Charging*.

Diameter Gateway Now Supports the 5G Non-Standalone Architecture

You can now use Diameter Gateway to integrate ECE with 5G networks. For more information, see "Connecting ECE to a Diameter Client" in *ECE Implementing Charging*.

It also has been enhanced to support Extended Bandwidth AVPs and the new NR Secondary RAT AVPs. For more information, see "Diameter Gateway Support for 5G Non-Standalone Architecture" in *ECE Diameter Gateway Protocol Implementation Conformance Statement*.

ECE Now Supports 5G CHF Protocol

ECE now supports the 5G Charging Function (CHF) protocol. It supports Nchf interfaces for converged charging and spending limit control for policy. For more information, see *ECE 5G CHF Protocol Implementation Conformance Statement*.

New HTTP Gateway for 5G Networks

ECE now supports online charging for 5G networks through the new HTTP Gateway. HTTP Gateway integrates ECE with your 5G clients and performs the following functions:

- Receives ECE REST API requests from 5G clients and then translates them into batch request server (BRS) requests.
- Submits BRS requests to ECE servers.
- Receives responses from ECE servers and then translates them into REST API responses.
- Responds to the 5G clients.
- Consumes notifications from the ECE Notification topic and then notifies the 5G clients by making a REST call to the URL stored in the system.

After the HTTP Gateway is set up, your 5G clients submit requests to ECE by using these new ECE REST API endpoints:

- Create a Spending Limit Control Subscription Endpoint
- Update a Spending Limit Control Subscription Endpoint
- End a Spending Limit Control Subscription Endpoint
- Create an Nchf Converged Charging Service Endpoint

- Update an Nchf Converged Charging Service Endpoint
- End an Nchf Converged Charging Service Endpoint
- Get Balances Endpoint
- Get a Usage Consumption Report Endpoint

For more information, see "Connecting ECE to a 5G Client" in *ECE Implementing Charging*.

Active-Hot Standby and Segmented Active-Active DR Support Persistence-Enabled Deployments

In previous releases, active-hot standby and segmented active-active configurations were not supported when ECE persistence was enabled.

With this enhancement, you can now configure active-hot standby and segmented active-active disaster recovery (DR) configurations when ECE persistence is enabled. Earlier, the `RatedEventFormatter` instance was running in both primary and secondary sites, but now there is only one `RatedEventFormatter` instance running among the sites.

For more information, see "Configuring ECE for Disaster Recovery" in *BRM System Administrator's Guide*.

New Features in ECE 12.0 Patch Set 2

ECE 12.0 Patch Set 2 includes the following enhancements:

- [New Tools to Monitor ECE](#)
- [Duplicate Check Enhancement in ECE](#)
- [Charging Operation Type Can be Configured for Expired Active Session Cleanup](#)
- [ECE Now Generates POID for Events](#)
- [New Tool for Querying ECE Cache Data](#)
- [ECE Now Supports Wildcard in Item Type Selectors](#)
- [Support for Persisting BRS Configuration Data](#)
- [Rated Event Partition in the ECE Persistent Database](#)

New Tools to Monitor ECE

You can now use the following new monitoring tools to collect the ECE data for monitoring the ECE cache size, rating performance, latency, and so on:

- **`ece_cluster_overview`**. Collects the ECE cluster summary. Use this tool to display the summary in a text or comma-separated value (CSV) format in Linux.
- **`ece_full_gc`**. Collects the complete Garbage Collection (GC) debug logs for any ECE grid member or a specific java application by using the process ID.
- **`ece_log`**. Collects the specified log-level data. You can also use this script to set log levels. See "Modifying Log Levels by Using Scripts" for more information.

- **ece_metrics_collector.** Runs the ECE Metrics Collector tool. This tool starts a light-weight HTTP Server to collect the ECE metrics data and present the data to third-party monitoring tools. Use this tool to view the ECE metrics using open-source tools, such as Grafana and Prometheus. See "Monitoring ECE Using Grafana and Prometheus" for more information.
- **jvm_info.** Collects the JVM data on a periodic basis. The JVM data includes memory usage, central processing unit (CPU) usage, open files, and so on.
- **ece_brs_info.** Connects to a specific Java process in which the batch request service (BRS) is running and collects the data such as the number of requests processed and latency over time.
- **ece_queues.sh.** Collects Oracle Communications Pricing Design Center (PDC) metrics from WebLogic Server, such as the size of the pricing data, the number of messages in the queue, and so on.
- **cohql_file_processor.** Starts the query tool (**query.sh**) and allows it to remain connected to the ECE cluster. Handles the queries on ECE Coherence caches in the non-interactive or interactive mode. Use this script to query ECE data instead of running the **query.sh** script frequently.
- **count_ref_files.** Counts the number of Rated Event Formatter output files and their volume. You can use this script to estimate the volume of data that can be processed by Rated Event Formatter and the storage space required to store the rated event data in the event of failure.
- **get_log_slices.** Collects data from the specified set of ECE log files for the given time period to perform error analysis. You can use this script to collect the detailed information regarding a specific problem.
- **parse_ece_chronicler.** Parses the ECE batch request service (BRS) chronicler metrics to aggregate the data into configurable time periods. This script provides a general view of the ECE rating performance.
- **parse_ecedc.** Parses the Oracle Communications Offline Mediation Controller Elastic Charging Engine (ECE) Distribution Cartridge (DC) log files to collect the data for monitoring the rating performance and throughput of ECE DC deployed in Offline Mediation Controller; for example, batch processing time, data read time, batch submission time, timeouts, and so on.
- **parse_ece_pricing_load.** Parses the INFO-level log data in the Pricing Updater log files. This script provides the information required to monitor the Pricing Updater performance, such as the volume of data loaded, number of items processed, and the time taken to process the pricing data.
- **parse_ece_start.** Parses the INFO-level data in the ECE server log files generated during initial data loading. This script enables you to identify the performance problems in the initial data loading and startup process.

You can use these tools along with Coherence reports and ECE Monitoring Agent to monitor your ECE system. By default, the ECE monitoring tools are installed in the `ECE_home/tools` directory. If you change the location of these tools, run the following command in Bash shell, which sets the `ECE_TOOLS_DIR` and `PATH` environment variables to point to the new location:

```
export ECE_TOOLS_DIR=path_to_tools
export PATH=path_to_tools
```

For more information on these tools, see the following:

- [Running ECE Monitoring Tools](#)
- [Customizing JMX-Based Tool Reports](#)
- [Monitoring ECE Cluster](#)
- [Monitoring ECE Using Grafana and Prometheus](#)
- [Setting Log Levels by Using Scripts](#)
- [Using cohql_file_processor for Query](#)

Running ECE Monitoring Tools

You can run all these ECE monitoring tools with the default settings by using the following wrapper utilities:

Note:

You do not have to run each monitoring tool individually.

- **ece_util_launcher.sh**. Runs the following Groovy/JMX-based tools by using the default JMX credentials: **ece_cluster_overview**, **ece_metrics_collector**, **ece_brs_info**, **jvm_info**, **ece_log**, and **ece_full_gc**.
- **perl_util_launcher.sh**. Runs the following Perl-based tools: **cohql_file_processor**, **count_ref_files**, **get_log_slices**, **parse_ece_chronicler**, **parse_ececdc**, **parse_ece_pricing_load**, and **parse_ece_start**.

You can also create symbolic link between tools to simplify the start up process. You can create symbolic links by running the following command in the *ECE_home/tools* directory or the directory in which the tools are available:

```
./mklinks all
```

For example, this creates symbolic links from **ece_cluster_overview** and **start_ece_metrics_collector** to the **ece_util_launcher.sh** and **perl_util_launcher.sh** utilities. This enables **ece_cluster_overview** and **start_ece_metrics_collector** to run the **ece_util_launcher.sh** and **perl_util_launcher.sh** utilities internally without any intervention.

Later, if you want to remove the symbolic link, run the following command:

```
./mklinks -r all
```

For the syntax and parameters used with the ECE monitoring tools, run the help (-h) command. For example:

```
ece_cluster_overview.groovy -h
```

Following is an example for running **ece_cluster_overview** to collect the cluster summary:

```
./ece_cluster_overview -e /scratch/ri-user-1/opt/OracleCommunications/ECE/ECE/occeserver/config/eceTopology.conf -U controlRole -P 'R&D' -c
```

Customizing JMX-Based Tool Reports

The JMX-based tools generate reports in the tabular format. The output for these tools are defined in the *ECE_home/tools/TableDefinition.properties* file. You can customize the reports generated by these tools by updating this file or by using the filters when you run the tool command. For example, you can define the ECE attributes to be included in the reports and also change the width and format of the table based on your requirements.

For more information, see the *ECE_home/tools/TableDefinition.properties* file.

Monitoring ECE Cluster

The **ece_cluster_overview** tool connects to an ECE cluster through a JMX-enabled node and queries data about the cluster members and the state of ECE caches. This tool can be used to retrieve data on periodic basis in the text or comma-separated value (CSV) format.

Note:

When using this tool to query data from Coherence clusters, which have a large number of members, retrieving data from remote members frequently might affect overall cluster performance.

Oracle recommends to review the management refresh expiry (**tangosol.coherence.management.refresh.expiry** value in the *ECE_home/occeserver/config/ece.properties* file) for the cluster you are monitoring and adjust the interval to reduce the frequency of data retrieval from remote members. However, the data collected may not be completely accurate.

You can also use this tool to query batch request service (BRS) data for charging clients, such as Diameter Gateway and Oracle Communications Offline Mediation Controller, and retrieve general JVM information, such as heap usage and CPU usage.

You can set the parameters used for collecting ECE cluster details by running the **ece_cluster_overview.groovy** script. For information on the parameters, default values, and examples, run the help (**-h**) command.

Monitoring ECE Using Grafana and Prometheus

You use the ECE Metrics Collector (**ece_metrics_collector**) tool that runs a light-weight HTTP server to collect system metrics from ECE on demand and convert the data into a format that can be processed by third-party monitoring tools. This tool provides different metric groups to collect different ECE metrics and present them to the third-party monitoring tools at different intervals.

You can use ECE Metrics Collector with the following third-party monitoring tools for monitoring ECE:

- **Prometheus.** An open-source monitoring tool which:
 - Stores system metrics in a time-series database.
 - Provides flexible queries to retrieve data for analysis.
 - Supports real-time alerting.

- **Grafana.** An open-source visualization tool that presents graphical dashboards by retrieving data from Prometheus. This tool contains multiple presentation formats and also provides an SDK for customizing dashboards.

The ECE monitoring process that uses Grafana and Prometheus is as follows:

1. ECE Metrics Collector collects the ECE metrics data in its different groups and presents them in the text-based exposition format to Prometheus by using HTTP endpoints.

To configure, start, or stop ECE Metrics Collector, see "[Starting and Stopping ECE Metrics Collector](#)".

2. The Prometheus monitoring server connects to each HTTP endpoint in ECE Metrics Collector at different intervals periodically and queries the metrics data. For example, Prometheus collects the ECE cache metrics every 30 seconds and the usage rating metrics every 10 seconds.

To configure Prometheus for querying ECE metrics data, see the Prometheus documentation.

3. The Prometheus monitoring server stores the metrics data in a time-series database and runs rules on the metrics data to aggregate the data and generate alerts or notifications based on the configuration.

4. Grafana retrieves the ECE metrics data from the Prometheus monitoring server and presents different graphical dashboards for different users. The dashboard contains one or more panels which displays the metrics for ECE.

To configure Grafana for displaying the ECE metrics data, see the Grafana documentation.

For information on the metrics collected by ECE Metrics Collector, see "[Collecting ECE Metrics Data](#)".

Starting and Stopping ECE Metrics Collector

Before starting ECE Metrics Collector, you can set the ECE Metrics Collector parameters used for collecting ECE metrics by running the **ece_metrics_collector.groovy** script. For information on the parameters, default values, and examples, run the help (-h) command.

To start or stop ECE Metrics Collector:

1. On the driver machine, change to the *ECE_home\oceserver\bin* directory.

2. Start ECC:

```
./ecc
```

3. Do one of the following:

- To start ECE Metrics Collector, run the following command:

```
start_ece_metrics_collector
```

The HTTP server is started and it listens on the specified port. The default port is 5050.

- To stop ECE Metrics Collector, run the following command:

```
stop_ece_metrics_collector
```

The HTTP server is stopped.

You can verify if ECE Metrics Collector is running by connecting to the HTTP server using the following URL:

```
http://your_server:5050/metrics
```

Collecting ECE Metrics Data

ECE Metrics Collector collects metrics in different groups to produce data for monitoring ECE. See the following for more information:

- [ECE_GRID_MEMBER_METRICS](#)
- [BRS_CLIENT_METRICS](#)
- [CACHE_METRICS](#)
- [JVM_METRICS](#)
- [SESSION_AND_EVENT_METRICS](#)
- [SERVICE_METRICS](#)
- [FEDERATION_METRICS](#)

ECE_GRID_MEMBER_METRICS

The ECE_GRID_MEMBER_METRICS group contains the ece_grid_members metric which provides information about the running status of nodes in the topology. ECE Metrics Collector compares the information retrieved from the Coherence grid with the information in the ECE topology file to retrieve the list of all nodes in the grid with their status.

[Table 3-1](#) describes the metrics that ECE Metrics Collector retrieves for grid members.

Table 3-1 Grid Member Metrics

Metric Name	Type	Description
ece_grid_members	Gauge	Contains the state of all the members in the ECE grid. The metric shows 1 for nodes that are running and 0 for nodes that are down.

For example, if the diameterGateway1 and ratedEventFormatter1 nodes are running and the query and Customer Loader utilities are down, the ece_grid_members metric that is retrieved is as follows:

```
ece_grid_members{id="21", pid="2313", machine="abc.us.example.com",
nodeName="diameterGateway1", role="diameterGateway"} 1
ece_grid_members{id="22", pid="3239", machine="abc.us.example.com",
nodeName="ratedEventFormatter1", role="ratedEventFormatter"} 1
ece_grid_members{id="0", pid="0", machine="abc.us.example.com", nodeName="Cohq1Shell",
role="query"} 0
ece_grid_members{id="0", pid="0", machine="abc.us.example.com",
nodeName="customerLoader", role="customerLoader"} 0
```


BRS_CLIENT_METRICS

The BRS_CLIENT_METRICS group contains the metrics for tracking throughput and latency of the charging clients, which use batch request service (BRS), such as Diameter Gateway, Offline Mediation Controller, or any custom charging clients.

Table 3-2 describes the metrics that ECE Metrics Collector retrieves for ECE charging clients.

Table 3-2 ECE Charging Client Metrics

Metric Name	Type	Description
ece_brs_total_task_count	counter	Tracks the total number of requests processed by a charging client. This metric shows the number of requests accepted, processed, timed-out, or rejected by the ECE component. You can use this in Prometheus to track the approximate processing rate over time, aggregate over all client applications, and so on.
ece_brs_current_task_count	gauge	Tracks the number of requests processed by a charging client in the current scrape interval. This metric shows the number of requests accepted, processed, timed-out, or rejected by the ECE component since the last query. You can use this metric to determine the delta.
ece_brs_current_throughput_per_second	gauge	Tracks the throughput of a charging client for each charging operation type in the current query interval. For example, if the data is collected every 10 seconds, the current value is considered as the throughput for that specific operation type since the last query. This metric provides the throughput information for the following operation types: Initiate , Update , Terminate , Cancel , Price_Enquiry , Balance_Query , Debit_Amount , Debit_Unit , Refund_Amount , and Refund_Unit . You can use this metric to gauge the overall system throughput for each type of usage requests.
ece_brs_current_latency_milliseconds	gauge	Tracks the current operation latency(in milliseconds) for a charging client in the current scrape interval. This metric contains the BRS statistics tracked using the charging.brsConfigurations MBean attributes. This configuration tracks minimum, maximum, and average latency for an operation type since the last query. The maximum window size for the collecting this data is 30 seconds, so the query has to be run within every 30 seconds. This metric provides the latency information for the following operation types: Initiate , Update , Terminate , Cancel , Price_Enquiry , Balance_Query , Debit_Amount , Debit_Unit , Refund_Amount , Refund_Unit , and Spending_Limit_Report .

Table 3-2 (Cont.) ECE Charging Client Metrics

Metric Name	Type	Description
<code>ece_total_requests_by_result_code</code>	counter	Tracks the total requests processed by Diameter Gateway by using the result code. This metric shows the total number of Gy and Sy requests processed by a Diameter Gateway instance for each result code.
<code>ece_current_request_rate_milliseconds_by_result_code</code>	gauge	Tracks the current processing rate of Diameter Gateway (in milliseconds) by using the Diameter protocols (Sy/Gy) and the result code. This metric provides the current processing rate for Gy and Sy operations handled by a Diameter Gateway instance for each result code.
<code>ece_session_metrics</code>	gauge	Tracks the overall number of Gy sessions that are open or close for an ECE component, such as Diameter Gateway and ECE server nodes. This metric shows the total number of open and closed sessions for each Diameter Gateway instance and rating group. Note: By default, ECE Metrics Collector aggregates ECE server metrics by rating group at the system-level. You can refine this metric to include the information per ECS node by running ECE Metrics Collector with the <code>-mp</code> parameter.

Example

The following are the examples for the metrics in the `BRS_CLIENT_METRICS` group:

- **`ece_brs_total_task_count`**. This example shows that `diameterGateway1` has processed 340845 requests without error and rejected or timed-out 0 requests:

```
ece_brs_total_task_count{nodeName="diameterGateway1", role="diameterGateway",
machine="abc.us.example.com", type="accepted"} 340845
ece_brs_total_task_count{nodeName="diameterGateway1", role="diameterGateway",
machine="abc.us.example.com", type="processed"} 340845
ece_brs_total_task_count{nodeName="diameterGateway1", role="diameterGateway",
machine="abc.us.example.com", type="rejected"} 0
ece_brs_total_task_count{nodeName="diameterGateway1", role="diameterGateway",
machine="abc.us.example.com", type="timeout"} 0
```

- **`ece_brs_current_task_count`**. This example shows that `diameterGateway1` has accepted and processed 54 requests in the current scrape interval:

```
ece_brs_current_task_count{nodeName="diameterGateway1", role="diameterGateway",
machine="abc.us.example.com" type="pending"} 0
ece_brs_current_task_count{nodeName="diameterGateway1", role="diameterGateway",
machine="abc.us.example.com" type="accepted"} 54
ece_brs_current_task_count{nodeName="diameterGateway1", role="diameterGateway",
machine="abc.us.example.com" type="processed"} 54
ece_brs_current_task_count{nodeName="diameterGateway1", role="diameterGateway",
machine="abc.us.example.com" type="timeout"} 0
ece_brs_current_task_count{nodeName="diameterGateway1", role="diameterGateway",
machine="abc.us.example.com" type="rejected"} 0
```

- **ece_brs_current_throughput_per_second.** This example shows that diameterGateway1 has processed Initiate, Update, and Terminate (IUT) for voice calls at a rate Of 30 calls per second:

```
ece_brs_current_throughput_per_second{nodeName="diameterGateway1",
role="diameterGateway", machine="abc.us.example.com" opType="initiate"} 26
ece_brs_current_throughput_per_second{nodeName="diameterGateway1",
role="diameterGateway", machine="abc.us.example.com" opType="update"} 27
ece_brs_current_throughput_per_second{nodeName="diameterGateway1",
role="diameterGateway", machine="abc.us.example.com" opType="terminate"} 29
```

- **ece_brs_current_latency_milliseconds.** This example shows the minimum, maximum, and average latency in milliseconds for Initiate and Update requests:

```
ece_brs_current_latency_milliseconds{nodeName="diameterGateway1",
role="diameterGateway", machine="abc.us.example.com" type="average",
opType="initiate"} 5.9
ece_brs_current_latency_milliseconds{nodeName="diameterGateway1",
role="diameterGateway", machine="abc.us.example.com" type="min",
opType="initiate"} 4.6
ece_brs_current_latency_milliseconds{nodeName="diameterGateway1",
role="diameterGateway", machine="abc.us.example.com" type="max",
opType="initiate"} 7.0
ece_brs_current_latency_milliseconds{nodeName="diameterGateway1",
role="diameterGateway", machine="abc.us.example.com" type="average",
opType="update"} 0.0
ece_brs_current_latency_milliseconds{nodeName="diameterGateway1",
role="diameterGateway", machine="abc.us.example.com" type="min",
opType="update"} 0.0
ece_brs_current_latency_milliseconds{nodeName="diameterGateway1",
role="diameterGateway", machine="abc.us.example.com" type="max",
opType="update"} 0.0
ece_brs_current_latency_milliseconds{nodeName="diameterGateway1",
role="diameterGateway", machine="abc.us.example.com" type="average",
opType="terminate"} 0.0
ece_brs_current_latency_milliseconds{nodeName="diameterGateway1",
role="diameterGateway", machine="abc.us.example.com" type="min",
opType="terminate"} 0.0
ece_brs_current_latency_milliseconds{nodeName="diameterGateway1",
role="diameterGateway", machine="abc.us.example.com" type="max",
opType="terminate"} 0.0
```

- **ece_total_requests_by_result_code.** This example shows the total number of Gy and Sy requests processed by diameterGateway1 for the result code 2001:

```
ece_total_requests_by_result_code{nodeName="diameterGateway1",
role="diameterGateway", machine="abc.us.example.com", protocol="Gy",
resultCode="2001"} 237114
ece_total_requests_by_result_code{nodeName="diameterGateway1",
role="diameterGateway", machine="abc.us.example.com", protocol="Sy",
resultCode="2001"} 45883
```

- **ece_current_request_rate_milliseconds_by_result_code.** This example shows the current processing rate in milliseconds for Gy and Sy requests processed by diameterGateway1 for the result code 2001:

```
ece_current_request_rate_milliseconds_by_result_code{instance="abc.us.example
.com:5050",job="ece_brs",machine="abc.us.example.com",nodeName="diameterGatew
ay1",protocol="Gy",resultCode="2001",role="diameterGateway"} 140.6
ece_current_request_rate_milliseconds_by_result_code{instance="abc.us.example
.com:5050",job="ece_brs",machine="abc.us.example.com",nodeName="diameterGatew
ay1",protocol="Sy",resultCode="2001",role="diameterGateway"} 10.4
```

- **ece_session_metrics.** This example shows the total number of open and closed sessions for diameterGateway1 and the rating group 10:

```
ece_session_metrics{nodeName="diameterGateway1",rating_group="*",role="diameterGateway",type="Close"} 8140.0
ece_session_metrics{nodeName="diameterGateway1",rating_group="*",role="diameterGateway",type="Open"} 11750.0
ece_session_metrics{rating_group="10",type="Close"} 7639.0
ece_session_metrics{rating_group="10",type="Open"} 10645.0
```

CACHE_METRICS

The CACHE_METRICS group contains metrics for ECE caches. You can use this metric to track the overall growth rate of certain caches along with other metrics.

Table 3-3 describes the metrics that ECE Metrics Collector retrieves for ECE caches.

Table 3-3 ECE Cache Metrics

Metric Name	Type	Description
ece_cache_avg_size	Gauge	Contains the average size of objects (in bytes) in the ECE caches.
ece_cache_entries	Gauge	Contains the total number of entries present in an ECE cache on an ECE node at the time the query is run.
ece_cache_gets	Counter	Contains the total number of gets performed on an ECE cache.
ece_cache_puts	Counter	Contains the total number of updates made to an ECE cache. It is the mirror of the ece_cache_gets metric.
ece_cache_store_millis	Counter	Contains the cumulative time spent (in milliseconds) on the cache-store operations for an ECE cache. This metric is applicable only for the following caches: RatedEvent and ServiceContext.
ece_cache_store_writes	Counter	Contains the number of updates attempted on a cache store, such as RatedEvent and ServiceContext.
ece_cache_total_size	Gauge	Contains the total size of an ECE cache (by default in megabytes). You can also define the unit of the metric by using the -u parameter.

Examples

The following are the examples for the metrics in the CACHE_METRICS group:

- **ece_cache_avg_size.** This example shows the average size in bytes for the ActiveSession, Customer, and RatedEvent caches:

```
ece_cache_avg_size{nodeName="ecs1", service="BRMFederatedCache", machine="abc.us.example.com", cache="ActiveSession"} 1968
ece_cache_avg_size{nodeName="ecs1", service="BRMFederatedCache", machine="abc.us.example.com", cache="Customer"} 3344
ece_cache_avg_size{nodeName="ecs1", service="BRMFederatedCache", machine="abc.us.example.com", cache="RatedEvent"} 2557
```

- **ece_cache_entries.** This example shows the total number of entries in the ActiveSession, Customer, and RatedEvent caches:

```
ece_cache_entries{nodeName="ecs1", service="BRMFederatedCache",
machine="abc.us.example.com", cache="ActiveSession"} 3068
ece_cache_entries{nodeName="ecs1", service="BRMFederatedCache",
machine="abc.us.example.com", cache="Customer"} 1117
ece_cache_entries{nodeName="ecs1", service="BRMFederatedCache",
machine="abc.us.example.com", cache="RatedEvent"} 836
```

- **ece_cache_gets.** This example shows the total number of gets for the ActiveSession and Customer caches:

```
ece_cache_gets{nodeName="ecs1", service="BRMFederatedCache",
machine="abc.us.example.com", cache="ActiveSession", tier="back"} 13906794
ece_cache_gets{nodeName="ecs1", service="BRMFederatedCache",
machine="abc.us.example.com", cache="Customer", tier="back"} 8720803
```

- **ece_cache_store_millis.** This example shows that 244.931 milliseconds and 5.019 milliseconds are spent on the RatedEvent and ServiceContext cache updates respectively:

```
ece_cache_store_millis{nodeName="ecs1", service="BRMFederatedCache",
machine="abc.us.example.com", cache="RatedEvent", tier="back"} 244.931
ece_cache_store_millis{nodeName="ecs1", service="BRMFederatedCache",
machine="abc.us.example.com", cache="ServiceContext", tier="back"} 5.019
```

- **ece_cache_store_writes.** This example shows the number of updates attempted for the RatedEvent and ServiceContext caches on 2 ECE charging server nodes:

```
ece_cache_store_writes{nodeName="ecs1", service="BRMFederatedCache",
machine="abc.us.example.com", cache="RatedEvent", tier="back"} 3182
ece_cache_store_writes{nodeName="ecs1", service="BRMFederatedCache",
machine="abc.us.example.com", cache="ServiceContext", tier="back"} 1
ece_cache_store_writes{nodeName="ecs2", service="BRMFederatedCache",
machine="abc.us.example.com", cache="RatedEvent", tier="back"} 2281
ece_cache_store_writes{nodeName="ecs2", service="BRMFederatedCache",
machine="abc.us.example.com", cache="ServiceContext", tier="back"} 0
```

- **ece_cache_total_size.** This example shows the size of the ActiveSession, Balance, and Customer cache in megabytes:

```
ece_cache_total_size{nodeName="ecs1", service="BRMFederatedCache",
machine="abc.us.example.com", cache="ActiveSession", unit="mb"} 1.73
ece_cache_total_size{nodeName="ecs1", service="BRMFederatedCache",
machine="abc.us.example.com", cache="Balance", unit="mb"} 0.52
ece_cache_total_size{nodeName="ecs1", service="BRMFederatedCache",
machine="abc.us.example.com", cache="Customer", unit="mb"} 1.17
```

JVM_METRICS

The JVM_METRICS group contains standard metrics about the central processing unit (CPU) and memory utilization of JVMs, which are members of the ECE grid.

[Table 3-4](#) describes the metrics that ECE Metrics Collector retrieves for JVMs.

Table 3-4 JVM Metrics

Metric Name	Type	Description
<code>ece_jvm_memory</code>	Gauge	Contains the Java heap usage information (by default in megabytes) for each ECE charging server (ECS) node. This metric contains a lot of memory-related attributes of JVMs, which you can use to get a detailed view of the memory usage for different ECE components. Note: The off-heap memory data is tracked only if the <code>-M</code> parameter is not used while running the ECE Metrics Collector.
<code>ece_jvm_process_cpu_percent</code>	Gauge	Contains the CPU usage information (in percentage) for each ECE component on the server. This data is collected from the corresponding MBean attributes by JVMs.
<code>ece_jvm_process_file_descriptors</code>	Gauge	Contains the total number of file-descriptors currently available for an ECE component and the descriptors that are in use for that ECE component.
<code>ece_jvm_server_cpu_percent</code>	Gauge	Contains the CPU load information (in percentage) for each system in the cluster. These statistics are based on the average data collected from all the ECE grid members running on a server.
<code>ece_jvm_server_load_average</code>	Gauge	Contains the system load average (the number of items waiting in the CPU run-queue) information for each machine in the cluster. These statistics are based on the average data collected from all the ECE grid members running on a server.
<code>ece_jvm_server_swap</code>	Gauge	Contains system swap usage information (by default in megabytes) for each system in the cluster. These statistics are based on the average data collected from all the ECE grid members running on a server.

Example

The following are the examples for the metrics in the `JVM_METRICS` group:

- **`ece_jvm_memory`**. This example shows the values in megabytes for a single ECS node:

```
ece_jvm_memory{nodeName="ecs1", role="server", machine="abc.us.example.com",
type="heap", class="committed"} 1536.00
ece_jvm_memory{nodeName="ecs1", role="server", machine="abc.us.example.com",
type="heap", class="init"} 1536.00
ece_jvm_memory{nodeName="ecs1", role="server", machine="abc.us.example.com",
type="heap", class="percent"} 42.56
ece_jvm_memory{nodeName="ecs1", role="server", machine="abc.us.example.com",
type="heap", class="used"} 653.79
ece_jvm_memory{nodeName="ecs1", role="server", machine="abc.us.example.com",
type="mapped", class="count"} 0.0
ece_jvm_memory{nodeName="ecs1", role="server", machine="abc.us.example.com",
```

```

type="mapped", class="memory_used"} 0.00
ece_jvm_memory{nodeName="ecs1", role="server", machine="abc.us.example.com",
type="mapped", class="total_capacity"} 0.00
ece_jvm_memory{nodeName="ecs1", role="server", machine="abc.us.example.com",
type="off_heap", class="committed"} 173.06
ece_jvm_memory{nodeName="ecs1", role="server", machine="abc.us.example.com",
type="off_heap", class="init"} 2.44

```

- **ece_jvm_process_cpu.** This example shows that the 3 ECS nodes are using 6.41%, 6.31%, and 6.04% of CPU respectively:

```

ece_jvm_process_cpu_percent{nodeName="ecs1", role="server",
machine="abc.us.example.com"} 6.409681757059616
ece_jvm_process_cpu_percent{nodeName="ecs2", role="server",
machine="abc.us.example.com"} 6.308724832214765
ece_jvm_process_cpu_percent{nodeName="ecs3", role="server",
machine="abc.us.example.com"} 6.037567084078712

```

- **ece_jvm_process_file_descriptors.** This example shows that BRM Gateway is using 144 descriptors and there are maximum of 65536 descriptors available for BRM Gateway:

```

ece_jvm_process_file_descriptors{nodeName="brmGateway", role="brmGateway",
machine="abc.us.example.com", type="max"} 65536
ece_jvm_process_file_descriptors{nodeName="brmGateway", role="brmGateway",
machine="abc.us.example.com", type="open"} 144

```

- **ece_jvm_server_cpu_percent.** The following example shows that the abc server is using approximately 31% of CPU:

```

ece_jvm_server_cpu_percent{machine="abc.us.example.com"} 30.95

```

- **ece_jvm_server_load_average.** This example shows the abc server's load average is 3.08 at the time of query (if it is an 8 CPU sever, this indicates that the server is not heavily loaded):

```

ece_jvm_server_load_average{machine="abc.us.example.com"} 3.08

```

- **ece_jvm_server_swap.** This example shows that abc server has 4096 megabytes of swap in a total of 4 gigabytes and has about 2.8 gigabytes free space:

```

ece_cache_total_size{nodeName="ecs1", service="BRMFederatedCache",
machine="abc.us.example.com", cache="ActiveSession", unit="mb"} 1.73
ece_cache_total_size{nodeName="ecs1", service="BRMFederatedCache",
machine="abc.us.example.com", cache="Balance", unit="mb"} 0.52
ece_cache_total_size{nodeName="ecs1", service="BRMFederatedCache",
machine="abc.us.example.com", cache="Customer", unit="mb"} 1.17

```

SESSION_AND_EVENT_METRICS

The `SESSION_AND_EVENT_METRICS` group contains metrics on ECE server sessions and rated events processed by these sessions. See ["BRS_CLIENT_METRICS"](#) for the Diameter session metrics.

[Table 3-5](#) describes the metrics that ECE Metrics Collector retrieves for ECE server sessions and rated events.

Table 3-5 ECE Server Sessions and Rated Events Metrics

Metric Name	Type	Description
<code>ece_rated_events_current_by_node</code>	Gauge	Contains the total number of RatedEvent objects extracted from the Oracle NoSQL database since the last query. Note: In single-schema systems, only one Rated Event Formatter instance is tracked for collecting metrics.
<code>ece_rated_events_total_by_node</code>	Counter	Contains the total number of rated events processed by each ECS node. This includes both the rated events that are stored in the RatedEvent cache and the Oracle NoSQL database. This data is tracked only when you run ECE Metrics Collector with the -mp parameter.
<code>ece_rated_events_total</code>	Counter	Tracks the processing of rated events by ECE server nodes.
<code>ece_session_metrics</code>	Counter	Contains the total number of sessions opened or closed on each ECE server node by rating group. This metric can be refined when you run ECE Metrics Collector with the -mp parameter.

Example

The following are the examples for the metrics in the `SESSION_AND_EVENT_METRICS` group:

- `ece_rated_events_current_by_node`.** This example shows that 4477 events were extracted in the last scrape interval:

```
ece_rated_events_current_by_node{nodeName="ratedEventFormatter1",
machine="abc.us.example.com", role="ratedEventFormatter", type="extracted"} 4477.0
```
- `ece_rated_events_total_by_node`.** This example shows the rated events extracted and processed by ECS node `ecs1`. There are 158421 events stored in the RatedEvent cache, of which 158290 have been stored in the Oracle NoSQL database:

```
ece_rated_events_total_by_node{nodeName="ecs3", machine="abc.us.example.com",
role="server", type="inserted"} 177644.0
ece_rated_events_total_by_node{nodeName="ecs3", machine="abc.us.example.com",
role="server", type="pushed"} 177502.0
ece_rated_events_total_by_node{nodeName="ratedEventFormatter1",
machine="abc.us.example.com", role="ratedEventFormatter", type="extracted"}
503713.0
```
- `ece_rated_events_total`.** This example shows the number of rated events extracted from Rated Event Formatter and the number of rated events stored in the RatedEvent cache and Oracle NoSQL database respectively:

```
ece_rated_events_total{type="extracted"} 503713.0
ece_rated_events_total{type="inserted"} 511258.0
ece_rated_events_total{type="pushed"} 510842.0
```
- `ece_session_metrics`.** This example shows the number of sessions opened and closed for rating group 10 for the ECE server node `ecs3` (if **-mp** parameter is used):


```

ece_session_metrics{nodeName="ecs3", machine="abc.us.example.com",
role="server", type="Close", rating_group="10"} 174647.0
ece_session_metrics{nodeName="ecs3", machine="abc.us.example.com",
role="server", type="Open", rating_group="10"} 184504.0
ece_session_metrics{type="Close", rating_group="10"} 506267.0
ece_session_metrics{type="Open", rating_group="10"} 534832.0

```

SERVICE_METRICS

The `SERVICE_METRICS` group contains metrics for the Oracle Coherence cache services. You can use this data to monitor the server latency and load (per node) and the backlogs which may accumulate on the ECE nodes. You can configure the frequency to reset the Coherence service statistics by running the ECE Metrics Collector with the `-r <duration>` parameter.

[Table 3-6](#) describes the metrics that ECE Metrics Collector retrieves for Oracle Coherence cache services.

Table 3-6 Oracle Coherence Cache Services Metrics

Metric Name	Type	Description
<code>ece_service_avg_thread_count</code>	Gauge	Contains the average active thread count as determined by Coherence.
<code>ece_service_endangered_partitions</code>	Gauge	Contains the total number of endangered partitions on an ECE service. The metric value is 0 if the partition is not endangered and a number greater than zero if any node has failed.
<code>ece_service_ha_status</code>	Gauge	Contains a numeric representation of the Coherence high-availability status. The metric value is 0 for ENDANGERED node, 1 for NODE-SAFE, 2 for RACK-SAFE, 3 for MACHINE-SAFE, and 4 for SITE-SAFE. Typically, nodes can be endangered temporarily during rebalancing operations when the nodes are added or removed.
<code>ece_service_request_avg_duration</code>	Gauge	Contains the average server-side request latency in milliseconds.
<code>ece_service_request_count</code>	Counter	Contains the total number of requests run by the service since the last reset.
<code>ece_service_request_max_duration</code>	Gauge	Contains the maximum server-side request latency (in milliseconds) since the last reset.
<code>ece_service_request_pending_count</code>	Gauge	Contains the total number of requests currently pending for a service. Large number of pending tasks may indicate a performance or capacity problem.
<code>ece_service_request_pending_duration</code>	Gauge	Contains the duration of the request (in milliseconds) pending in a service.
<code>ece_service_task_avg_duration</code>	Gauge	Contains the average server-side task latency in milliseconds since the last reset.

Table 3-6 (Cont.) Oracle Coherence Cache Services Metrics

Metric Name	Type	Description
<code>ece_service_task_backlog</code>	Gauge	Contains the current server-side task backlog for each service. A large backlog is indicative of a performance or capacity problem.
<code>ece_service_task_count</code>	Counter	Contains the total number of tasks processed by a service since the last reset.
<code>ece_service_task_max_backlog</code>	Gauge	Contains the maximum task backlog for each service since the last reset.
<code>ece_service_unbalanced_partitions</code>	Gauge	Contains the total number of unbalanced partitions for a particular service. Typically, unbalanced partitions can occur temporarily during rebalancing operations when the nodes are added or removed.

Example

The following are the examples for the metrics in the `SERVICE_METRICS` group:

- **`ece_service_avg_thread_count`**. This example shows the number of `BRMFederatedCache` and `InvocationService` service threads in use. By default, ECE provides 4 threads for the `BRMFederatedCache` service and 33 threads for the `InvocationService`:

```
ece_service_avg_thread_count{nodeName="ecs1", machine="abc.us.example.com",
service="BRMFederatedCache"} 0.5684553
ece_service_avg_thread_count{nodeName="ecs1", machine="abc.us.example.com",
service="InvocationService"} 1.3715283
```

- **`ece_service_endangered_partitions`**. This example shows that there are no endangered partitions for the ECS server node `ecs1`:

```
ece_service_endangered_partitions{nodeName="ecs1", machine="abc.us.example.com",
service="BRMFederatedCache"} 0
ece_service_endangered_partitions{nodeName="ecs1", machine="abc.us.example.com",
service="InvocationService"} -1
```

- **`ece_service_ha_status`**. This example shows that the `BRMFederatedCache` services on the ECS server nodes `ecs1` and `ecs2` are 1 (`NODE-SAFE`):

```
ece_service_ha_status{nodeName="ecs1", machine="abc.us.example.com",
service="BRMFederatedCache"} 1
ece_service_ha_status{nodeName="ecs1", machine="abc.us.example.com",
service="InvocationService"} -1
ece_service_ha_status{nodeName="ecs2", machine="abc.us.example.com",
service="BRMFederatedCache"} 1
ece_service_ha_status{nodeName="ecs2", machine="abc.us.example.com",
service="InvocationService"} -1
```

- **`ece_service_request_avg_duration`**. This example shows the ECE server node `ecs1` with average request for the `BRMFederatedCache` service as 5 milliseconds and 0 for the `InvocationService` service (this service processes only tasks):

```
ece_service_request_avg_duration{nodeName="ecs1", machine="abc.us.example.com",
service="BRMFederatedCache"} 5.173
```

```
ece_service_request_avg_duration{nodeName="ecs1",
machine="abc.us.example.com", service="InvocationService"} 0.000
```

- **ece_service_request_count.** This example shows the ECS server node ecs1 has processed approximately 30000 requests since the last statistics reset:

```
ece_service_request_count{nodeName="ecs1", machine="abc.us.example.com",
service="BRMFederatedCache"} 30044
ece_service_request_count{nodeName="ecs1", machine="abc.us.example.com",
service="InvocationService"} 0
```

- **ece_service_request_max_duration.** This example shows the maximum request duration on the ECE server node ecs1 is approximately 66 milliseconds:

```
ece_service_request_max_duration{nodeName="ecs1",
machine="abc.us.example.com", service="BRMFederatedCache"} 67
ece_service_request_max_duration{nodeName="ecs1",
machine="abc.us.example.com", service="InvocationService"} 0
```

- **ece_service_request_pending_count.** This example shows the ECE server node ecs1 with 12 pending requests:

```
ece_service_request_pending_count{nodeName="ecs1",
machine="abc.us.example.com", service="BRMFederatedCache"} 12
ece_service_request_pending_count{nodeName="ecs1",
machine="abc.us.example.com", service="InvocationService"} 0
```

- **ece_service_request_pending_duration.** This example shows the ECE server node ecs1 with requests pending for 13 milliseconds:

```
ece_service_request_pending_duration{nodeName="ecs1",
machine="abc.us.example.com", service="BRMFederatedCache"} 13
ece_service_request_pending_duration{nodeName="ecs1",
machine="abc.us.example.com", service="InvocationService"} 0
```

- **ece_service_task_avg_duration.** This example shows the average task durations on the ECE server node ecs1 for the InvocationService and BRMFederatedCache services in milliseconds:

```
ece_service_task_avg_duration{nodeName="ecs1",
machine="abc.us.example.com", service="BRMFederatedCache"} 3.276
ece_service_task_avg_duration{nodeName="ecs1",
machine="abc.us.example.com", service="InvocationService"} 9.142
```

- **ece_service_task_backlog.** This example shows the ECE server node ecs1 with a backlog of 4 requests on the BRMFederatedCache service and 1 task on the InvocationService on ECE server node ecs2:

```
ece_service_task_backlog{nodeName="ecs1",
machine="slc15ejg.us.example.com", service="BRMFederatedCache"} 4
ece_service_task_backlog{nodeName="ecs1",
machine="slc15ejg.us.example.com", service="InvocationService"} 0
ece_service_task_backlog{nodeName="ecs2",
machine="slc15ejg.us.example.com", service="BRMFederatedCache"} 0
ece_service_task_backlog{nodeName="ecs2",
machine="slc15ejg.us.example.com", service="InvocationService"} 1
```

- **ece_service_task_count.** This example shows that all services have handled approximately 15000 tasks:

```
ece_service_task_count{nodeName="ecs1", machine="abc.us.example.com",
service="BRMFederatedCache"} 14995
ece_service_task_count{nodeName="ecs1", machine="abc.us.example.com",
service="InvocationService"} 15010
ece_service_task_count{nodeName="ecs2", machine="abc.us.example.com",
```

```
service="BRMFederatedCache"} 14272
ece_service_task_count{nodeName="ecs2", machine="abc.us.example.com",
service="InvocationService"} 14264
```

- **ece_service_task_max_backlog.** This example shows the ECE server node ecs1 and ecs2 with a backlog of 26 and 13 tasks for the BRMFederatedCache and 0 and 1 for the InvocationService service:

```
ece_service_task_max_backlog{nodeName="ecs1", machine="abc.us.example.com",
service="BRMFederatedCache"} 26
ece_service_task_max_backlog{nodeName="ecs1", machine="abc.us.example.com",
service="InvocationService"} 0
ece_service_task_max_backlog{nodeName="ecs2", machine="abc.us.example.com",
service="BRMFederatedCache"} 13
ece_service_task_max_backlog{nodeName="ecs2", machine="abc.us.example.com",
service="InvocationService"} 1
```

- **ece_service_unbalanced_partitions.** This example shows that there are no unbalanced partitions on ECE server node ecs1:

```
ece_service_unbalanced_partitions{nodeName="ecs1", machine="abc.us.example.com",
service="BRMFederatedCache"} 0
ece_service_unbalanced_partitions{nodeName="ecs1", machine="abc.us.example.com",
service="InvocationService"} -1
```

FEDERATION_METRICS

The FEDERATION_METRICS group contains metrics for ECE federated caches when ECE persistence is disabled. The metrics in this group provide information regarding the volume of data transferred, the number of objects transferred, and so on. You can use this metric to monitor the data transferred from the primary production system to the remote or backup systems. This data is typically used for disaster recovery where the Oracle NoSQL database is used for storing rated events.

You can run the ECE Metrics Collector with the **-f ClusterName** and **-F ServiceName** parameters to retrieve the federated cache metrics.

[Table 3-7](#) describes the metrics that ECE Metrics Collector retrieves for ECE federated caches when ECE persistence is disabled.

Table 3-7 ECE Federated Caches Metrics

Metric Name	Type	Description
ece_federated_service_bandwidth	Gauge	Tracks the current or maximum bandwidth used to transfer data from ECE charging nodes to a secondary ECE cluster.
ece_federated_service_rate	Gauge	Tracks the approximate rate of data transfer in bytes and number of messages sent. This metric uses the Coherence Mbean attributes for tracking data.
ece_federated_service_replicate_millis	Gauge	Contains the cache replication latency (in milliseconds) for initial data replication. The type can be: total (total time taken for replication), estimate_ttc (estimated time to complete).
ece_federated_service_replicate_percent	Gauge	Contains the percentage of cache replication completed for a service.

Table 3-7 (Cont.) ECE Federated Caches Metrics

Metric Name	Type	Description
ece_federated_service_status	Gauge	Contains the status of the service which is on federation. The state of a service can be: 1 (Initial), 2 (Idle), 3 Ready, 4 (Sending), 5 (Connecting), 6 (Connect_Wait), 7 (Stopped), 8 (Paused), 9 (Error), 10 (Yielding), 11 (Backlog_Excessive), 12 (Backlog_Normal), and 13 (Disconnected). The status of a service can be: 1 (OK), 2 (Warning), and 3 (Error).
ece_federated_service_time_millis	Gauge	Contains the cache replication latency (in milliseconds) for data replicated to the remote cache. The type can be: apply , backlog_delay , and round_trip . These are the 90th percentile latency times.
ece_federated_service_total	Gauge	Contains the total number of bytes, cache entries, and messages that are replicated to the remote cache. The entity type can be: records (total number of journal records), bytes (total number of bytes sent), entries (total number of cache entries sent), message (total number of messages sent), response (total number of message responses received). The status can be: sent (entity shipped to remote cluster), unacked (messages sent without acknowledgment), and error (messages failed).

Example

The following are the examples for the metrics in the FEDERATION_METRICS group:

- **ece_federated_service_bandwidth.** This example shows the bandwidth at the current moment for the BRMFederatedCache and ReplicatedFederatedCache services on the ECE charging server node ecs1 (in MBps). It also shows that there is no maximum bandwidth configured (as the value is -1.0) for these services:

```
ece_federated_service_bandwidth{nodeName="ecs1",
machine="abc.us.example.com", service="BRMFederatedCache", type="current"}
9.159515380859375
ece_federated_service_bandwidth{nodeName="ecs1",
machine="abc.us.example.com", service="BRMFederatedCache", type="max"} -1.0
ece_federated_service_bandwidth{nodeName="ecs1",
machine="abc.us.example.com", service="ReplicatedFederatedCache",
type="current"} 0.0
ece_federated_service_bandwidth{nodeName="ecs1",
machine="abc.us.example.com", service="ReplicatedFederatedCache",
type="max"} -1.0
```

- **ece_federated_service_rate.** This example shows the approximate number of bytes sent per second and number of messages sent for the BRMFederatedCache service on the ECE charging server node ecs1:

```
ece_federated_service_rate{nodeName="ecs1", machine="abc.us.example.com",
service="BRMFederatedCache", entity="bytes", type="sent"} 1183714
ece_federated_service_rate{nodeName="ecs1", machine="abc.us.example.com",
service="BRMFederatedCache", entity="message", type="sent"} 182
```

- **ece_federated_service_replicate_millis.** This example shows the replication completed time in milliseconds for the BRMFederatedCache service on the ECE charging server node ecs1:

```
ece_federated_service_replicate_millis{nodeName="ecs1",
machine="abc.us.example.com", service="BRMFederatedCache", type="estimate_ttc"} 0
ece_federated_service_replicate_millis{nodeName="ecs1",
machine="abc.us.example.com", service="BRMFederatedCache", type="total"} 378
ece_federated_service_replicate_millis{nodeName="ecs1",
machine="abc.us.example.com", service="ReplicatedFederatedCache",
type="estimate_ttc"} 0
ece_federated_service_replicate_millis{nodeName="ecs1",
machine="abc.us.example.com", service="ReplicatedFederatedCache", type="total"} 938
```

- **ece_federated_service_replicate_percent.** This example shows that the replication is 100% complete for BRMFederatedCache and ReplicatedFederatedCache services on the ECE charging server node ecs1:

```
ece_federated_service_replicate_percent{nodeName="ecs1",
machine="abc.us.example.com", service="BRMFederatedCache"} 100
ece_federated_service_replicate_percent{nodeName="ecs1",
machine="abc.us.example.com", service="ReplicatedFederatedCache"} 100
```

- **ece_federated_service_status.** This example shows the state and statuses of the BRMFederatedCache and ReplicatedFederatedCache services on the ECE charging server node ecs1. The service status is directly reflected from Coherence.

```
ece_federated_service_status{nodeName="ecs1", machine="abc.us.example.com",
service="BRMFederatedCache", type="state"} 2
ece_federated_service_status{nodeName="ecs1", machine="abc.us.example.com",
service="BRMFederatedCache", type="status"} 0
ece_federated_service_status{nodeName="ecs1", machine="abc.us.example.com",
service="ReplicatedFederatedCache", type="state"} 2
ece_federated_service_status{nodeName="ecs1", machine="abc.us.example.com",
service="ReplicatedFederatedCache", type="status"} 0
```

- **ece_federated_service_time_millis.** This example shows the time taken to apply the changes to the BRMFederatedCache service on the ECE charging server node ecs1, with a round-trip time of approximately 5 minutes:

```
ece_federated_service_time_millis{nodeName="ecs1", machine="abc.us.example.com",
service="BRMFederatedCache", type="apply"} 20
ece_federated_service_time_millis{nodeName="ecs1", machine="abc.us.example.com",
service="BRMFederatedCache", type="backlog_delay"} 10
ece_federated_service_time_millis{nodeName="ecs1", machine="abc.us.example.com",
service="BRMFederatedCache", type="round_trip"} 5
```

- **ece_federated_service_total.** This example shows the total number of BRMFederatedCache service entries, messages, data bytes, records, and responses replicated from the ECE charging server node ecs1. The changes to the cache entries are grouped into journal records and then into messages:

```
ece_federated_service_total{nodeName="ecs1", machine="abc.us.example.com",
service="BRMFederatedCache", entity="bytes", type="sent"} 28868414
ece_federated_service_total{nodeName="ecs1", machine="abc.us.example.com",
service="BRMFederatedCache", entity="entries", type="sent"} 11292
ece_federated_service_total{nodeName="ecs1", machine="abc.us.example.com",
service="BRMFederatedCache", entity="message", type="sent"} 5590
```

```
ece_federated_service_total{nodeName="ecs1", machine="abc.us.example.com",
service="BRMFederatedCache", entity="message", type="unacked"} 0
ece_federated_service_total{nodeName="ecs1", machine="abc.us.example.com",
service="BRMFederatedCache", entity="records", type="sent"} 5646
ece_federated_service_total{nodeName="ecs1", machine="abc.us.example.com",
service="BRMFederatedCache", entity="response", type="error"} 0
```

Setting Log Levels by Using Scripts

You can set the log level of the ECE module or modules by ECE functional domain.

To set log levels by using the **ece_log** script:

1. On the driver machine, change to the *ECE_home/tools* directory.
2. Start ECC, if not started:

```
./ecc
```

3. Run the following command:

```
ece_log -l Levels -c Classes -S Level
```

where:

- *Levels* specifies the log-levels to be selected; for example, "debug|info".
- *Classes* specifies the log class names.
- *Level* specifies the log-level to be set.

For example:

```
ece_log -l debug -c "rating|client" -S error
```

This command sets all the DEBUG-level log entries to ERROR for the rating and client classes.

Using cohql_file_processor for Query

The **cohql_file_processor** tool is a wrapper around the query tool (**query.sh**), which runs queries on ECE Coherence caches. You can use the following scripts to query ECE data instead of running the **query.sh** script frequently:

- **start_cohql_file_processor**. Use this script to run the query in the non-interactive mode using input files.
- **interactive_cohql**. Use this script to run the query in the interactive mode using command-line.

You can stop the query by using the **stop_cohql_file_processor** script.

For information on the parameters, default values, and examples for this tool, run the help (**-h**) command.

Duplicate Check Enhancement in ECE

ECE performs duplicate check on the incoming TERMINATE, DIRECT_DEBIT, and REFUND requests to verify the session ID in the requests. By default, the session ID is stored in the aggregate usage and rated event objects. In this case, sometimes the aggregate usage and rated event objects have to be kept in the ECE cache for a longer duration when duplicate checks are performed on these requests.

With this enhancement, when data persistence is enabled in ECE, the session IDs are stored in the new Session ID cache, **TerminatedSessionHistory**. This cache stores only the session IDs of the requests that are processed. For performing duplicate checks on TERMINATE, DIRECT_DEBIT, and REFUND requests, ECE checks the session IDs in this new cache. This ensures that the aggregate usage and rated event objects are evicted from the cache immediately after the successful cache update.

Charging Operation Type Can be Configured for Expired Active Session Cleanup

In the previous releases, ECE removed the expired active session and the corresponding expired balance reservations only when TERMINATE or CANCEL operation type in usage requests were processed. It was not possible to configure other charging operation types for cleaning up the expired objects.

With this enhancement, you can configure for which charging operation types the expired active sessions and balance reservations must be cleaned up. For example, you can configure ECE to remove the expired active sessions and balance reservations when the INITIATE or UPDATE operation type in a usage request is processed. You can specify the charging operation types to be used by setting the **asoCleanupOperationList** Mbean attribute. See "[Configuring Expired Objects Cleanup](#)".

Configuring Expired Objects Cleanup

1. Access the ECE MBeans:
 - a. Log on to the driver machine.
 - b. Start the ECE charging servers (if they are not started).
 - c. Start a JMX editor, such as JConsole, that enables you to edit MBean attributes.
 - d. Connect to the ECE charging server node set to **start CohMgt = true** in the *ECE_home/occeserver/config/eceTopology.conf* file.

The **eceTopology.conf** file also contains the host name and port number for the node.
 - e. In the editor's MBean hierarchy, expand the **ECE Configuration** node.
2. Expand **charging.server**.
3. Expand **Attributes**.
4. Set the **asoCleanupOperationList** attribute to the charging operation types for which the expired active session and balance reservation objects must be cleaned up.

The default values are TERMINATE, CANCEL. If this attribute is set to null, the default values are used for the cleanup.

For the list of charging operation types supported by ECE, see the discussion about the Charging API in *ECE Implementing Charging*.

ECE Now Generates POID for Events

In the previous releases, ECE was using the Portal object IDs (POIDs) received from BRM for tracking events rated by ECE.

With this enhancement, POIDs can be generated in ECE for tracking the rated events. ECE uses Rated Event Formatter to generate the required POIDs and persists the last allocated POID ID in the Oracle NoSQL database. This ensures that the POIDs are generated without any duplication even if the ECE system is restarted.

The POID generated in ECE contains the following information:

```
event_type date cluster_id BRM_schema_id unique_id
```

See [Table 3-8](#) for the description of each entry in the POID.

Table 3-8 POID Entries in ECE

Entry	Description
<i>event_type</i>	A unique 4-bit number assigned to each event type. For example, 0 is assigned to subscription events, 1 is assigned to postpaid events (USAGE_POSTPAID), and 2 to 7 is assigned to prepaid events (USAGE_PREPAID) depending on the prepaidPartitionSet value specified in BRM. The default value for <i>event_type</i> is 0.
<i>date</i>	The 16-bit date on which the POID is generated. The date is determined based on ECE virtualTime if it is enabled. For more information on virtualTime , see the discussion about changing time and date to test ECE in <i>ECE Implementing Charging</i> .
<i>cluster_id</i>	A unique 4-bit number assigned to the Coherence cluster to identify ECE in the cluster. The <i>cluster_id</i> is limited to 0 to 15 and the maximum number of ECE clusters allowed in a deployment is 16. The default value for <i>cluster_id</i> is 0. If ECE is configured for disaster recovery, you must specify the cluster ID for each cluster used in the Active-hot standby or Active-cold standby systems.
<i>BRM_schema_id</i>	A unique 6-bit number assigned to the BRM schema. The <i>BRM_schema_id</i> is limited to 0 to 31.
<i>unique_id</i>	A unique 34-bit number assigned to each POID.

For tracking the events rated by ECE, Rated Event Formatter uses the POIDs generated in ECE. You can configure multiple instances of Rated Event Formatter to ensure high availability and uninterrupted POID allocation. In case if the primary Rated Event Formatter instance fails, the secondary Rated Event Formatter instance ensures that the POIDs are allocated without any interruption. In a disaster recovery deployment, if the Rated Event Formatter instance in the *primary* site fails, the Rated Event Formatter instance in the *backup* site continues the POID allocation for the events. To connect the instances in different sites or systems, you must specify the name of the primary Rated Event Formatter instance in the primary and secondary Rated Event Formatter instances.

For tracking the bill items created in ECE, ECE continues to use the POIDs received from BRM. However, ECE now persists the POID pool received from BRM in the Oracle NoSQL database. This ensures that the reserved POID pool is retained in ECE even after the ECE restart. It allows ECE to continue the POID allocation for the bill items using the existing POID pool, which in turn reduces the dependency on BRM.

To enable POID generation in ECE for events, you must perform the following:

1. Enable prepaid-event partitions in BRM. For instructions, see "[Enabling Prepaid-Event Partitions](#)".

2. Ensure that the cluster ID is configured for ECE clusters. The cluster ID must be specified if you have ECE configured for disaster recovery. See "[Configuring Cluster ID](#)".
3. Ensure that the name of the primary Rated Event Formatter instance is specified in each Rated Event Formatter instance. The primary Rated Event Formatter instance must be specified if you have ECE configured for disaster recovery. See "[Connecting Rated Event Formatter Instances](#)".
4. Enable prepaid-event partitions in ECE. See "[Enabling Prepaid-Event Partitions](#)".

Configuring Cluster ID

To configure the cluster ID for ECE clusters:

1. Access the ECE MBeans:
 - a. Log on to the driver machine.
 - b. Start the ECE charging servers (if they are not started).
 - c. Start a JMX editor, such as JConsole, that enables you to edit MBean attributes.
 - d. Connect to the ECE charging server node set to **start CohMgt = true** in the *ECE_home/occeserver/config/eceTopology.conf* file.

The **eceTopology.conf** file also contains the host name and port number for the node.
 - e. In the editor's MBean hierarchy, expand the **ECE Configuration** node.
2. Expand **charging.clusters.Cluster_Name**, where *Cluster_Name* is the name of the ECE cluster that you are configuring.
3. Expand **Attributes**.
4. Set the **id** attribute to a unique number that indicates the ID of the cluster in the POID generated in ECE.

Rated Event Formatter uses the cluster ID in the POID to identify the ECE clusters. The cluster ID must be unique for each cluster.

Connecting Rated Event Formatter Instances

To connect the Rated Event Formatter instances in different sites or systems, you must perform this for each Rated Event Formatter instance

To connect Rated Event Formatter instances:

1. Access the ECE MBeans:
 - a. Log on to the driver machine.
 - b. Start the ECE charging servers (if they are not started).
 - c. Start a JMX editor, such as JConsole, that enables you to edit MBean attributes.
 - d. Connect to the ECE charging server node set to **start CohMgt = true** in the *ECE_home/occeserver/config/eceTopology.conf* file.

The **eceTopology.conf** file also contains the host name and port number for the node.
 - e. In the editor's MBean hierarchy, expand the **ECE Configuration** node.

2. Expand **charging.ratedEventFormatters.Instance_Name**, where *Instance_Name* is the name of the instance you want to configure; for example, `ratedEventFormatter2`.
3. Expand **Attributes**.
4. Set the **primaryInstanceName** attribute to the name of the primary Rated Event Formatter instance.

For example, if the name of the primary Rated Event Formatter instance is `ratedEventFormatter1`, specify `ratedEventFormatter1` as **primaryInstanceName** in the primary and all secondary instances.

5. Change directory to the `ECE_home/occeserver/bin` directory.
6. Start ECC:

```
./ecc
```

7. Stop and restart any Rated Event Formatter instances that you configured.

Each instance reads its configuration information by name.

For information about stopping and starting Rated Event Formatter instances, see the discussion about starting and stopping ECE in *BRM Elastic Charging Engine System Administrator's Guide*.

Enabling Prepaid-Event Partitions

To enable prepaid-event partitions:

1. Access the ECE MBeans:
 - a. Log on to the driver machine.
 - b. Start the ECE charging servers (if they are not started).
 - c. Start a JMX editor, such as JConsole, that enables you to edit MBean attributes.
 - d. Connect to the ECE charging server node set to **start CohMgt = true** in the `ECE_home/occeserver/config/eceTopology.conf` file.

The `eceTopology.conf` file also contains the host name and port number for the node.
 - e. In the editor's MBean hierarchy, expand the **ECE Configuration** node.
2. Expand **charging.brmCdrPlugins.Instance_Name**, where *Instance_Name* is the name of the `BrmCdrPluginDirect` Plug-in instance you are configuring.
3. Expand **Attributes**.
4. Set the **prepaidPartitionSet** attribute to the value that you specified in the **prepaid_partition_set** entry in the `BRM_Home/sys/dm_oracle/pin.conf` file.

To enable prepaid-event partitions, you need to set this attribute to a number between 2 and 7. If this attribute is set to 0, ECE continues to use the POIDs received from BRM for events instead of generating them.

New Tool for Querying ECE Cache Data

In previous releases, the ECE cache data query took a long time to complete if the data was not properly indexed.

With this enhancement, a new query tool, **query_cache.sh**, has been introduced to query data in the ECE caches that are associated with customer keys. The new query tool provides access to the ECE cache content by way of CohQL. This tool is included with the ECE Server software in *ECE_home/occesdk/bin/query*, where *ECE_home* is the directory in which ECE Server software is installed. You can use this tool for debugging, development, generating reports, and other-related queries. This tool supports only the non-interactive mode.

The following is the syntax for the non-interactive use of the new query tool:

```
sh query_cache.sh run cache_name Associated_key ID
```

where:

- *Cache_name* specifies the name of the cache you are querying.
- *Associated_key* specifies the customer ID (composite key) that is associated with the cache.
- *ID* specifies the unique identifier, such as balance ID, subscriber ID, and so on.

For example, to query a customer's balance, you can run the query on the Balance cache using the associated customer ID (479665838) and balance ID (479664046):

```
sh query_cache.sh run Balance 479665838 479664046
```

This query returns the balance information for the specified customer ID.

For information on the query tool options, use the help command:

```
sh query_cache.sh -h
```

ECE Now Supports Wildcard in Item Type Selectors

ECE now supports wildcard (*) in item type selectors for services and events. You can use the wildcard to substitute one or more characters in the service or event type to indicate that any value is acceptable; for example, */service/telco/gsm/**.

If wildcard is used in the service or event type, ECE uses the **applicableToAllChildServices** and **applicableToAllChildEvents** values to identify if the service or event type and item type selector is applicable for all the child services or events. If the value is **true**, the item type selector is considered for all the child services or events. If the value is **false**, the item type selector is not considered for the child services or events.

For more information on using wildcard in item type selectors, see *PDC Creating Product Offerings*.

Support for Persisting BRS Configuration Data

In the previous releases, batch request service (BRS) parameters were not persisted in the ECE cache and were lost during ECE restart. You had to manually set these parameters again after ECE restart.

With this enhancement, you can persist the BRS parameters in the ECE cache by adding them using the **brsConfigurations** MBean attribute. You can now use this attribute instead of the **BatchRequestService** and **ChargingClient.BatchRequestService** MBean attributes for configuring system overload protection and client-side ECE request queues.

 **Note:**

If you have multiple instances of a component and you want to use the same BRS configuration for all the instances, you can add the BRS configuration for the role instead of each instance. For example, if you have `diameterGateway1`, `diameterGateway2`, and `diameterGateway3`, add the `BrsConfiguration` instance as `diameterGateway` to use the same BRS configuration for all the three instances.

If the BRS configuration is not added or removed for any of the instance, the BRS configuration for the role is applied to that instance by default. If the BRS configuration for the role is not defined, the default (system-level) BRS configuration is applied to that instance.

To add the BRS configuration:

1. Access the ECE MBeans:
 - a. Log on to the driver machine.
 - b. Start the ECE charging servers (if they are not started).
 - c. Start a JMX editor, such as JConsole, that enables you to edit MBean attributes.
 - d. Connect to the ECE charging server node set to **start CohMgt = true** in the `ECE_home/occeserver/config/eceTopology.conf` file.

The `eceTopology.conf` file also contains the host name and port number for the node.
 - e. In the editor's MBean hierarchy, expand the **ECE Configuration** node.
2. Expand **charging.brsConfigurations**.
3. Expand **Operations**.
4. Click **addBrsConfiguration**.
5. Set the **name** attribute to the name of the `BrsConfiguration` instance; for example, `diameterGateway1`.
6. Click **addBrsConfiguration**.
The new BRS configuration is added. For example, `charging.brsConfigurations.diameterGateway1`.
7. Expand **charging.brsConfigurations.instanceName**, where *instanceName* is the name of the instance for which you are configuring BRS.
8. Expand **Attributes**.
9. Specify the values for the attributes.

For descriptions of each attribute, see the discussion about configuring system overload protection and configuring client-side ECE request queues in *BRM System Administrator's Guide*. For the default values, see the **charging.brsConfigurations.default** Mbean attribute.

You can also delete the BRS configuration. To delete the BRS configuration:

1. Access the ECE MBeans:

- a. Log on to the driver machine.
 - b. Start the ECE charging servers (if they are not started).
 - c. Start a JMX editor, such as JConsole, that enables you to edit MBean attributes.
 - d. Connect to the ECE charging server node set to **start CohMgt = true** in the *ECE_home/occeserver/config/eceTopology.conf* file.
The **eceTopology.conf** file also contains the host name and port number for the node.
 - e. In the editor's MBean hierarchy, expand the **ECE Configuration** node.
2. Expand **charging.brsConfigurations**.
 3. Expand **Operations**.
 4. Click **removeBrsConfiguration**.
 5. Set the **name** attribute to the name of the BrsConfiguration instance that you want to remove.
 6. Click **removeBrsConfiguration**.
The BRS Configuration is deleted.

Rated Event Partition in the ECE Persistent Database

In the previous releases, Rated Event Formatter was using SQL-DELETE query for deleting the rated events stored in the ECE persistence database. When SQL-DELETE query was used for deleting rated events, there was a considerable increase in the table space. This might impact the overall performance.

With this enhancement, Rated Event Formatter uses Oracle Drop Partition instead of SQL-DELETE query for deleting rated events. By default, the partition time for creating rated event partitions is set to 5 minutes. You can change the default partition time depending on your sizing and performance requirements by setting the **ratedEventTablePartitionByMinute** MBean attribute. During installation you can configure the initial storage and subpartition values for rated event table. For more information, see *Elastic Charging Engine Installation Guide*.

To change the partition time:

1. Access the ECE MBeans:
 - a. Log on to the driver machine.
 - b. Start the ECE charging servers (if they are not started).
 - c. Start a JMX editor, such as JConsole, that enables you to edit MBean attributes.
 - d. Connect to the ECE charging server node set to **start CohMgt = true** in the *ECE_home/occeserver/config/eceTopology.conf* file.
The **eceTopology.conf** file also contains the host name and port number for the node.
 - e. In the editor's MBean hierarchy, expand the **ECE Configuration** node.
2. Expand **charging.connectionConfigurations.Connection_Name**; where *Connection_Name* is the name of the persistence database connection.
3. Expand **Attributes**.
4. Click **ratedEventTablePartitionByMinute** to set the partition time.

5. Save your changes.

New Features in ECE 12.0/12.0 Patch Set 1

For the new features in ECE 12.0/12.0 Patch Set 1 (12.0.0.1.0), see *ECE Release Notes*.

4

New Features in PDC

The Oracle Communications Pricing Design Center (PDC) 12.0 Patch Set releases include several new features.

Topics in this document:

- [New Features in PDC 12.0 Patch Set 8](#)
- [New Features in PDC 12.0 Patch Set 7](#)
- [New Features in PDC 12.0 Patch Set 6](#)
- [New Features in PDC 12.0 Patch Set 5](#)
- [New Features in PDC 12.0 Patch Set 4](#)
- [New Features in PDC 12.0 Patch Set 3](#)
- [New Features in PDC 12.0 Patch Set 2](#)
- [New Features in PDC 12.0 Patch Set 1](#)

New Features in PDC 12.0 Patch Set 8

PDC 12.0 Patch Set 8 includes the following enhancements:

- [PDC Supports Flexible Proration Options](#)
- [PDC Enhancement to Consume Allowances before Charging](#)
- [PDC UI Support for Tax Selectors and Tax Exemption Selectors](#)
- [PDC UI Support for Automated Product and Discount Sharing Groups](#)
- [PDC UI Support for 30-Day Proration](#)
- [PDC UI Support for Add-On Charge Offers in Bundles](#)
- [PDC Cloud Native Support for SSO](#)
- [Support for ProductOffering and ProductOfferingPrice GET APIs](#)
- [PDC Support for Discount Offer Stacking](#)

PDC Supports Flexible Proration Options

In previous releases, you could configure time stamp rounding only at a systemwide level using the **timestamp_rounding** entry in the CM **pin.conf** file. You could specify that all time stamps were either:

- **Rounded to midnight.** This meant that offer validity periods started at midnight, even if the offer was purchased later. In this case, purchases of recurring bundles would be valid starting at midnight, causing any delayed usage between midnight and the purchase time to be incorrectly consumed from the new grant.

- **Set to the purchase time.** This meant that charges had to be prorated for the first day of the billing cycle. For example, if a product was purchased at 15:00 for a 30-day billing cycle, the customer was charged for 29 days and 9 hours.

To make proration more flexible and to avoid any balance validity or charging issues, PDC now allows you to create a charge offer with different time stamp rounding values for the following:

- **The validity period:** You can specify that the validity period starts at the purchase time or midnight of the day the product is purchased. Alternatively, you can set to use the systemwide setting in the CM **pin.conf** file.
- **The charging scale:** You can specify whether to charge for a full day or a partial day for the first day of the billing cycle. Alternatively, you can set to use the validity period setting.

Configure the proration settings using the **ImportExportPricing** utility or the PDC UI.

To configure proration settings using an XML file and the **ImportExportPricing** utility, set these new XML elements under the **<ChargeOffering>** element:

- **<validityRounding>**: Specifies whether to start the charge offer's validity period at the purchase time or midnight of the purchase day.
 - **OFF**: Starts at the time of purchase. This overrides the CM **pin.conf** setting at the charge offer level.
 - **ON**: Starts at midnight (00:00:00) of the day that the charge offer is purchased. This overrides the CM **pin.conf** setting at the charge offer level.
 - **NOT_SET**: Uses the systemwide setting in the CM **pin.conf** file. This is the default.
- **<scaleRounding>**: Specifies how to calculate the scale:
 - **OFF**: Calculate it based on the **<validityRounding>** setting.
 - **ON**: Calculate it based on full days.

To configure proration settings using the PDC UI, use the **Validity Rounding** and **Scale Rounding** fields in the **Offer Settings** section of the **Create Charge Offer** page.

For more information, see "Configuring Full Day Proration" in *PDC Creating Product Offerings*.

PDC Enhancement to Consume Allowances before Charging

PDC includes enhancements for creating product offerings that grant free allowances, such as 100 free minutes or 5 free movie rentals. PDC now allows you to create offerings that consume all free allowances before applying any usage charges. For example, you could create a monthly product offering that grants 5 free movie rentals and then charges \$4 for any additional movie rentals.

You configure PDC to grant allowances using this method when you configure charges in your charge offers. In a charge, add a pricing tier definition that contains one or more pricing tier ranges. Each pricing tier range must contain an upper bound value, which can now be either a number or a noncurrency balance element ID.

The pricing tier ranges will be processed in order of the upper bound's numeric value and the order in which the pricing tier range is inserted in the database.

For more information, see "Configuring Pricing to Consume Granted Allowances Before Charging" in *PDC Creating Product Offerings*.

PDC UI Support for Tax Selectors and Tax Exemption Selectors

The PDC UI now supports the tax selectors and tax exemption selectors added in Patch Set 6. Also, PDC (including the GUI) now supports profile attributes as tax code selectors.

You can use the PDC UI to configure tax selectors and tax exemption selectors to apply taxes based on account, service, event, and profile attributes. This allows you to choose whether to use the direct tax code or to choose the tax code using the selectors in the rate plan while creating the product.

For more information, see "Creating a Tax Selector" and "Creating a Tax Exemption Selector" in *PDC Online Help*.

PDC UI Support for Automated Product and Discount Sharing Groups

You can now use the PDC UI to enable automated creation of the following:

- Product sharing groups, which allow your customers to share a product with a group of accounts. For example, a corporate account could purchase a Wireless product with special pricing, and then automatically share the package with all of its employees. For more information, see "Creating Charge Offers with Automated Sharing Enabled" in *BRM Managing Customers*.
- Discount sharing groups, which enable your customers to share a discount offer with a group of accounts. For example, a corporate account could purchase an Internet discount offer that includes a 15% discount on the monthly fee and three free movie rentals, and then automatically share the discount offer with all of its employees. For more information, see "Creating Discount Offers with Automated Sharing Enabled" in *BRM Managing Customers*.

Previously, this functionality could be implemented only using the **ImportExportPricing** utility.

PDC UI Support for 30-Day Proration

When creating bundles in the PDC UI, you can now specify how cycle charges and discounts are prorated at the charge offer or discount offer level. Previously, this functionality could be implemented only at the system level and only using the **ImportExportPricing** utility.

You can specify to calculate prorated cycle charges and discounts using:

- **A 30-day month:** Prorated cycle charges and discounts are calculated based on a 30-day month, regardless of the number of days in the billing cycle. For example, if a bundle was owned for 6 days in a cycle, the prorated charge would be the cycle charge multiplied by 0.20 ($6 \div 30$).
- **The actual number of days in a billing cycle:** Prorated cycle charges and discounts are calculated based on the actual number of days in a particular cycle, such as 28 days in February, 31 days in March, and 30 days in April. For example, if a bundle was owned for 6 days in March, the prorated charge would be the cycle charge multiplied by 0.19 ($6 \div 31$).

- **The system-wide proration setting:** Prorated cycle charges and discounts are calculated according to the systemwide setting in the CM `pin.conf` file's `enable_30_day_proration` entry.

To configure 30-day proration using the PDC UI, use the **Proration** list in the Offer Customization subsection of the **Create Bundle** page. For more information, see "Offer Customization Subsection" in *PDC Online Help*.

PDC UI Support for Add-On Charge Offers in Bundles

When creating bundles in the PDC UI, you can now include add-on charge offers. Previously, add-on charge offers could be implemented using only an XML file and the **ImportExportPricing** utility. See "[Support for Add-On Charge Offers in Bundles](#)" for more information.

By default, all charge offers in bundles are base charge offers, which means they are automatically included when the customer purchases the bundle. Add-on charge offers are optional and can be purchased with the bundle or later on. To configure add-on offers in bundles using the PDC UI, use the **This is an add-on charge offer** check box in the Offer Customization subsection of the **Create Bundle** page.

For more information, see "Offer Customization Subsection" in *PDC Online Help*.

PDC Cloud Native Support for SSO

PDC cloud native now supports the single sign-on (SSO) login method using SAML 2.0. SSO allows you to log in to applications using a single user name and password combination.

You set up SSO using SAML 2.0 using these keys in your `override-values.yaml` file for `oc-cn-op-job-helm-chart`:

```
ocpdc:
  configEnv:
    # Mandatory: Set to "true" to configure and use SAML 2.0, SSO service
    isSSOEnabled:
    # Name of your SAML Asserter
    samlAsserterName: "pdcSAML2IdentityAsserter"
    # Base URL that is used to construct endpoint URLs, Load Balancer host and
    # port at which the server is visible externally.
    # Must be appended with "/saml2". Example: https://LB_HOST:LB_PORT/saml2
    # (Mandatory if isSSOEnabled is set to true)
    ssoPublishedSiteURL:
    # URL to which unsolicited authentication responses are sent if they do
    # not contain an
    # accompanying target URL (Mandatory if isSSOEnabled is set to true)
    ssoDefaultURL:
    # URL where user will be redirected after logging out from the application
    # (Mandatory if isSSOEnabled is set to true)
    ssoLogoutURL:
```

For more information, see "Adding PDC Keys for oc-cn-op-job-helm-chart" in *BRM Cloud Native Deployment Guide*.

Support for ProductOffering and ProductOfferingPrice GET APIs

PDC REST Services Manager now supports the TMF 620 Open API GET endpoints for Product Offering and Product Offering Price.

For more information, see "ProductOffering/{Name}" and "ProductOfferingPrice/{Name}" in *PDC REST Services Manager Integration Guide*.

PDC Support for Discount Offer Stacking

Previously, when a discount was purchased a second time, it was treated as separate from the original purchase. It is now possible to extend the validity of a discount subscription by purchasing the discount multiple times using the **EXTEND** and **LONGEST_DATE** modes. You can also define a grace period (with units and offset) for resubscribing to a discount offer.

You can configure these settings using the **ImportExportPricing** utility. For more information, see "Purchasing the Same Offer in a Bundle Multiple Times" in *PDC Creating Product Offerings*.

New Features in PDC 12.0 Patch Set 7

PDC 12.0 Patch Set 7 includes the following enhancements:

- [GUI Support for Credit Limit Enforcement Enhancements](#)
- [First Usage Activation Enhancements in PDC](#)
- [Design-Time GUI Improvements](#)
- [Improved Handling of Balance Elements](#)
- [Proration Can Be Configured at the Offer Level](#)
- [Support for Add-On Charge Offers in Bundles](#)

GUI Support for Credit Limit Enforcement Enhancements

GUI support has been added for the credit limit enhancements added in Patch Set 6.

You can now configure a minimum amount to charge if the customer does not have the entire amount. You can choose to prorate the service based on the amount charged. If the subscription fails, you can cancel the subscription after the retries are exhausted. Alternatively, you can choose not to cancel the subscription, and retries will be suspended until the account is recharged.

For more information, see "Properties for All Quantity Ranges" in *PDC Online Help*.

First Usage Activation Enhancements in PDC

PDC now supports configuring a bundle or a package to start on first usage without having to configure each charge offer and discount offer in the bundle or package to start on first usage. When you use this option, the first usage of any offering will activate all of the offerings in the bundle or package.

You can implement the first usage activation feature by using the PDC UI or an XML file with the **ImportExportPricing** utility. See the following for more information:

- "About the Validity Periods of Offers in Bundles" in *PDC Creating Product Offerings*
- "Activating Offers in Bundles" and "Activating Offers in Packages" in *PDC Creating Product Offerings*
- "Specifying Bundle General Information and Settings" and "Creating a Package" in *PDC Online Help*

Design-Time GUI Improvements

GUI support has been added for the following features. In Patch Set 6, these features could be configured only through the **ImportExportPricing** utility.

- You can now create tax codes in the PDC GUI, and they will be synchronized with both BRM and ECE.
- You can now use the PDC GUI to specify whether to consume noncurrency resources for discounts in order of earliest expiry, or in order of any discount rules configured.
- You can now use the PDC GUI to set thresholds for offering loans at the package level.

For more information, see *PDC Online Help*.

Improved Handling of Balance Elements

Prior to Patch Set 7, PDC forced charge offers and discount offers to impact all balance elements referenced in the policy. Now, unnecessary validations have been removed from PDC, so that it no longer forces all balance elements to be affected.

Proration Can Be Configured at the Offer Level

Previously, you configured proration at the system-wide level. Now, you can also configure in bundles whether an offer is prorated based on 30 days or on the actual number of days in the month.

See "Setting Proration for Offers in a Bundle" in *PDC Creating Product Offerings* for more information.

Support for Add-On Charge Offers in Bundles

You can include add-on charge offers in your bundles. All charge offers in bundles are base charge offers by default, which means they are automatically included when the customer purchases the bundle. Add-on charge offers can be purchased with the bundle or later on.

When you create an add-on charge offer, you specify how to determine its validity start date. The add-on charge offer's validity start date is the end date of the charge offer that you specify. For example, assume charge offer A has a validity period from June 1 through June 15. If you specify to align add-on charge offer B's validity period with charge offer A, charge offer B's validity start date would be June 15.

You can specify that an add-on charge offer's validity dates align with:

- The base charge offer that you specify

- The active base charge offer that expires first
- The active base charge offer that expires last
- The active charge offer that expires first
- The active charge offer that expires last

See "About Add-On Charge Offers in Bundles" in *PDC Creating Product Offerings* for more information.

New Features in PDC 12.0 Patch Set 6

PDC 12.0 Patch Set 6 includes the following enhancements:

- [New Features Supported by PDC REST Services Manager](#)
- [New XML Options for Exceeding Credit Limits](#)
- [Consuming Noncurrency Resources for Discounts in Order of Expiration](#)
- [Tax Codes Are Now Created in PDC](#)
- [PDC XML Tags](#)
- [PDC Now Supports Single Sign-On](#)

New Features Supported by PDC REST Services Manager

PDC REST Services Manager now supports creating pricing configurations based the following:

- TMF620 payloads using the following extended object fields:
 - **isTaxInclusive**: For charge offers, lets you map tax inclusive tax codes.
 - **appliesTo**: For discount offers, lets you specify that a discount offer in a bundle applies to all current and future offers or all current and future offers of a particular type in the bundle instead of specifying specific offers.
- Custom attributes (also known as product specification characteristics or product specification attributes).
- New service types for digital TV, devices, and accessories. (This assumes you have set up the corresponding services in BRM and PDC).
You can sell devices and accessories by themselves or as part of service bundles. PDC REST Services Manager creates dummy parent bundles for devices and accessories being sold independently that appear directly under a package.
- Additional Helidon security logging details with the ALL log level.
- Ability to change the default log manager.

For information about PDC REST Services Manager, see "PDC REST Services Manager Overview" in *PDC REST Services Manager Integration Guide*.

New XML Options for Exceeding Credit Limits

When you define your product offerings in XML and upload them using the **ImportExportPricing** utility, in addition to the options for exceeding credit limits introduced in Patch Set 5, you can now use the following options for the **<enforceCreditLimit>** element in the price tier:

- **INSUFFICIENT_BALANCE**: Lets you set an minimum amount to charge when a customer doesn't have the full amount available. You can choose to prorate the service according to the minimum amount.
- **AUTO_RENEW_CANCEL**: Lets you cancel the subscription completely if it fails.

For these two new options and the existing RENTAL_FAILURE option, you can also define the maximum number of retries and the retry interval at the offer level. Previously you could only define the number of retries at the system level.

See "Enforcing or Exceeding Credit Limits" in *PDC Creating Product Offerings* for more information about these options.

 **Note:**

BRM also supports these options with the **loadpricelist** utility. See "Allowing Customers to Exceed Their Credit Limit" in *Configuring Pipeline Rating and Discounting*.

Consuming Noncurrency Resources for Discounts in Order of Expiration

When you define your product offerings in XML and upload them using the **ImportExportPricing** utility, you can specify whether to consume noncurrency resources for the discounts in order of earliest expiry, or in order of any discount rules configured.

See "Consuming Noncurrency Resources for Discounts in Order of Expiration" in *PDC Creating Product Offerings* for more information.

 **Note:**

This is only supported for rating in Elastic Charging Engine (ECE). It is not supported for Pipeline Manager.

Tax Codes Are Now Created in PDC

Before BRM 12.0 Patch Set 6, you created tax codes in BRM and synchronized them to Pricing Design Center (PDC). In Patch Set 6 and later releases, you create tax codes in PDC and publish them to BRM and Elastic Charging Engine (ECE) using the **ImportExportPricing** utility.

See "About Creating Tax Codes (Patch Set 6 and Later)" in *BRM Calculating Taxes* for more information.

If you are applying Patch Set 6 on top of a previous BRM 12.0 Patch Set, before you can use PDC to create new tax codes or modify tax codes created in previous patch sets, you must run the **SyncPDC** utility to synchronize the tax codes from BRM to PDC.

When you run SyncPDC after upgrading to Patch Set 6 or a later release, the following new actions will be taken:

- Maintain data consistency between your existing TaxCode table and the newly introduced BCTaxCode tables in the PDC database schema.
- Update data in the XREF_RRE_TAXCODE table in the transformation cross-reference (XREF) database schema.

See "Synchronizing Pricing Setup Components" in *PDC Creating Product Offerings* for information about **SyncPDC**.

PDC Support for More Granular Tax Code Application

You can now configure tax selectors and tax exemption selectors to apply taxes based on account, service, event, and profile attributes using the **ImportExportPricing** utility. This allows you to choose whether to use the direct tax code or to choose the tax code using the selectors. You can also apply tax exemptions to charge offers using selectors.

See "Selector Configuration Examples" in *PDC Creating Product Offerings* for more information.

PDC XML Tags

PDC XML tag information is useful when exporting or importing pricing or config data using the **ImportExportPricing** utility. PDC documentation now lists all the available tags and describes each field. For more information, see "PDC XML Tags" in *PDC Creating Product Offerings*.

PDC Now Supports Single Sign-On

PDC now supports the single sign-on (SSO) login method using SAML 2.0. SSO allows you to log in to applications using a single user name and password combination

For more information, see "Setting Up Single Sign-On for Pricing Design Center" in *PDC Installation Guide*.

New Features in PDC 12.0 Patch Set 5

PDC 12.0 Patch Set 5 includes the following enhancements:

- [PDC Supports Event Notifications](#)
- [Charges in PDC Support Dynamic Taxation](#)
- [Moving Tax Codes to PDC](#)
- [Distributing Noncurrency Balances in Incremental Buckets](#)
- [Now Specify Whether to Continue Charging for SuspendedActive Accounts](#)
- [Support for Purchasing the Same Offer Multiple Times](#)
- [Prorating Billing DOM Changes, Package Transitions, and Bundle Transitions](#)
- [Setting Charge Offer Cycle Alignment for Reactivated Bundles](#)
- [PDC UI Supports Recurring Charges on Specific DOM](#)
- [PDC UI Supports Versioning with Date Ranges](#)
- [PDC UI Supports Options for Exceeding Credit Limits](#)

- [PDC UI Supports Product Specification Attributes](#)
- [PDC REST Services Manager Supports OAuth with Oracle Access Management](#)
- [PDC REST Services Manager Supports Synchronizing New Elements](#)

PDC Supports Event Notifications

You can now configure PDC to generate notification events when the following occurs:

- A balance element's validity date is about to expire. For example, when the validity date for a balance of free minutes is about to expire. For more information, see "Configuring Balance Elements" in *PDC Creating Product Offerings*.
- A subscription is about to expire or is due for renewal. For more information, see "Configuring Charge Offer Usage and Ownership" in *PDC Creating Product Offerings*.

To use this new functionality, you must run a database upgrade script after you install PDC 12.0 Patch Set 5 or later:

1. Go to the `PDC_home/PDC/interimpatch/upgrade` directory.
2. Run this command, which opens SQL*Plus:

```
% sqlplus login@ORACLE_SID
Enter password: password
```

where:

- `login` is the user name for the PDC database schema.
 - `ORACLE_SID` is the database alias of the PDC database schema.
 - `password` is the password for the specified user name.
3. Run one of the following commands:

```
SQL> @AddExpiryNotification.sql
```

4. Exit SQL*Plus.

Charges in PDC Support Dynamic Taxation

In PDC, you can now configure charges to apply dynamic tax calculation, which defers tax calculation until the end of a billing cycle but calculates taxes using the tax rate at the time an event occurred. For example, assume a recurring charge's tax rate changes from 2% to 3% on May 15. If a billing cycle ends on May 30, BRM would use the 2% rate to calculate the charge's April 30 through May 14 taxes and the 3% rate to calculate the charge's May 15 through May 29 taxes. This provides the benefits of billing-time taxation while allowing you to change tax rates in the middle of a billing cycle.

For more information, see "Configuring Taxation in Charges" in *PDC Creating Product Offerings*.

Moving Tax Codes to PDC

You cannot synchronize tax codes from BRM to PDC using the **SyncPDC** utility for the following cases. Instead, you must manually re-create the BRM tax codes in a PDC-compliant XML file and then load them into the PDC database.

- For Patch Set 5 (full installation or upgrade), manually re-create the BRM tax codes in a PDC-compliant XML file and then load them into the PDC database.
- For a Patch Set 6 upgrade (from Patch Set 5 or previous Patch Sets), a one-time manual step to synchronize taxcodes in PDC is not present. Hence, **SyncPDC** and taxcode export may fail. You must manually re-create the BRM tax codes in a PDC-compliant XML file and then load them into the PDC database.

 **Note:**

In a full installation of PDC 12.0 Patch Set 6 and later releases, you use the **SyncPDC** utility to synchronize tax codes from BRM to PDC.

To re-create your BRM 12.0 Patch Set 5 tax codes in PDC 12.0 Patch Set 5:

1. Manually replicate the tax codes you created in BRM in a PDC-compliant XML file. For examples, see the following:
 - [Vertex Tax Code Example](#)
 - [Simple VAT Tax Code Example](#)
2. Load the XML file into the PDC database by using the **ImportExportPricing** utility:

```
ImportExportPricing -import -ow -config XMLFile
```

See "ImportExportPricing" in *PDC Creating Product Offerings* for information about the utility's syntax and parameters.

Vertex Tax Code Example

The following provides an example for replicating Vertex tax codes that you defined in BRM to a PDC-compliant XML file:

1. On the machine where BRM is installed, open the *BRM_home/sys/data/config/config_taxcodes_map.xml* file.
2. Look at the tax code configuration in the file. Below shows the configuration for a sample **qt_usage** tax code:

```
<ConfigObjectconfigName="/config/taxcodes_map"configMode="recreate">
  <DESCR>Taxcodes Map Configuration</DESCR>
  <NAME>taxcodes_map</NAME>
  <TAXES elem="1">
    <TAX_CODE>qt_usage</TAX_CODE>
    <TAXPKG_TAX_CODE>Q</TAXPKG_TAX_CODE>
    <CODE1>01</CODE1>
    <CODE2>01</CODE2>
    <SALES_INDICATOR>S</SALES_INDICATOR>
  </TAXES>
</ConfigObject>
```

3. Create an XML file and add information about the **qt_usage** tax code to the equivalent PDC elements:
 - The BRM **<TAX_CODE>** element is equivalent to the PDC **<code>** element.
 - The BRM **<TAXPKG_TAX_CODE>** element is equivalent to the PDC **<taxPackageType>** element. The possible values:

- BRM value of **C** equals a PDC value of **VERTEX_COMMTAX_21**
- BRM value of **Q** equals a PDC value of **VERTEX_QUANTUM**
- The BRM **<CODE1>** element is equivalent to the PDC **<transType_CategoryCode_Rate>** element.
- The BRM **<CODE2>** element is equivalent to the PDC **<transSubType_ServiceCode>** element.
- The BRM **<SALES_INDICATOR>** element is equivalent to the PDC **<salesIndicator>** element. The possible values:
 - BRM value of **S** equals a PDC value of **SALE**
 - BRM value of **R** equals a PDC value of **RESALE**

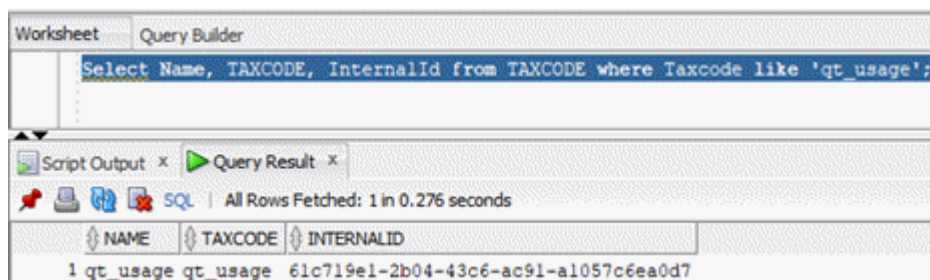
For example, this shows the contents from step 2 in a PDC-compliant XML file:

```
<taxCodes xmlns:cim="http://xmlns.oracle.com/communications/platform/model/Config">
  <name>qt_usage</name>
  <priceListName>Default</priceListName>
  <obsolete>>false</obsolete>
  <code>qt_usage</code>
  <taxPackageType>VERTEX_QUANTUM</taxPackageType>
  <taxCodeValidityPeriods>
    <taxCodeMaps>
      <transType_CategoryCode_Rate>1</transType_CategoryCode_Rate>
      <transSubType_ServiceCode>1</transSubType_ServiceCode>
      <salesIndicator>SALE</salesIndicator>
    </taxCodeMaps>
  </taxCodeValidityPeriods>
</taxCodes>
```

4. Using a database viewer, open the TAXCODE table in the PDC database schema. Write down the internal ID associated with the tax code.

For example, [Figure 4-1](#) shows the internal ID for the sample **qt_usage** tax code.

Figure 4-1 qt_usage Internal ID



5. In your PDC-compliant XML file, add the tax code's internal ID to a new **<internalId>** element. For example:

```
<taxCodes xmlns:cim="http://xmlns.oracle.com/communications/platform/model/Config">
  <name>qt_usage</name>
  <internalId>61c719e1-2b04-43c6-ac91-a1057c6ea0d7</internalId>
  <priceListName>Default</priceListName>
  <obsolete>>false</obsolete>
  <code>qt_usage</code>
  <taxPackageType>VERTEX_QUANTUM</taxPackageType>
```

```

<taxCodeValidityPeriods>
  <taxCodeMaps>
    <transType_CategoryCode_Rate>1</transType_CategoryCode_Rate>
    <transSubType_ServiceCode>1</transSubType_ServiceCode>
    <salesIndicator>SALE</salesIndicator>
  </taxCodeMaps>
</taxCodeValidityPeriods>
</taxCodes>

```

6. Save and close the file.

The XML file is ready to be loaded into the PDC database.

Simple VAT Tax Code Example

The following provides an example for replicating a simple VAT tax code that you defined in BRM to a PDC-compliant XML file:

1. On the machine where BRM is installed, open the *BRM_home/sys/data/config/config_taxcodes_map.xml* file.
2. Look at the tax code configuration in the file. Below shows the configuration for a sample VAT tax code:

```

<ConfigObjectconfigName="/config/taxcodes_map"configMode="recreate">
  <DESCR>Taxcodes Map Configuration</DESCR>
  <NAME>taxcodes_map</NAME>
  <TAXES elem="1">
    <TAX_CODE>VAT</TAX_CODE>
    <TAXPKG_TAX_CODE>U</TAXPKG_TAX_CODE>
    <PERCENT>4.5</PERCENT>
    <VALID_FROM_STR>02/01/22</VALID_FROM_STR>
    <VALID_TO_STR>01/31/23</VALID_TO_STR>
    <TAX_JURISDICTION>Fed</TAX_JURISDICTION>
    <TAX_JURISDICTION_LIST>GB</TAX_JURISDICTION_LIST>
    <DESCR>VAT-GB</DESCR>
    <RULE_TYPE>Std</RULE_TYPE>
  </TAXES>
</ConfigObject>

```

3. Create an XML file and add information about the VAT tax code to the equivalent PDC elements:
 - The BRM **<TAX_CODE>** element is equivalent to the PDC **<code>** element.
 - The BRM **<TAXPKG_TAX_CODE>** element is equivalent to the PDC **<taxPackageType>** element. For simple and custom tax codes, set the value of **<taxPackageType>** to **CUSTOM**.
 - The BRM **<PERCENT>** element is equivalent to the PDC **<percent>** element.
 - The BRM **<VALID_FROM_STR>** element is equivalent to the PDC **<validFrom>** element. Enter the date in YYYYMMDD format.
 - The BRM **<VALID_TO_STR>** element is equivalent to the PDC **<validTo>** element. Enter the date in YYYYMMDD format.
 - The BRM **<TAX_JURISDICTION>** element is equivalent to the PDC **<taxJurisdictionLevel>** element.
 - BRM value of **Fed** equals a PDC value of **FEDERAL**
 - BRM value of **Sta** equals a PDC value of **STATE**
 - BRM value of **Cou** equals a PDC value of **COUNTY**

- BRM value of **Cit** equals a PDC value of **CITY**
- BRM value of **Loc** equals a PDC value of **LOCATION**
- The BRM **<TAX_JURISDICTION_LIST>** element is equivalent to the PDC **<taxJurisdictions>** element.
- The BRM **<DESCR>** element is equivalent to the PDC **<description>** element.
- The BRM **<RULE_TYPE>** element is equivalent to the PDC **<taxRuleType>** element. The possible values:
 - BRM value of **Std** equals a PDC value of **STANDARD**
 - BRM value of **Tax** equals a PDC value of **TAX**
 - BRM value of **Inc** equals a PDC value of **INCLUSIVE**
 - BRM value of **NCS** equals a PDC value of **NCS**
 - BRM value of **NCT** equals a PDC value of **NCT**

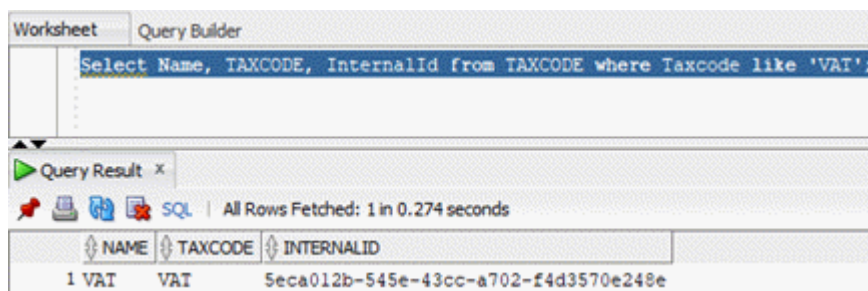
For example, this shows the contents from step 2 in a PDC-compliant XML file:

```
<taxCodes>
  <name>VAT</name>
  <priceListName>Default</priceListName>
  <obsolete>>false</obsolete>
  <code>VAT</code>
  <taxPackageType>CUSTOM</taxPackageType>
  <taxCodeValidityPeriods>
    <validFrom>20010201</validFrom>
    <validTo>20300131</validTo>
    <taxCodeMaps>
      <description>VAT-GB</description>
      <percent>4.5</percent>
      <taxJurisdictionLevel>FEDERAL</taxJurisdictionLevel>
      <taxJurisdictions>GB</taxJurisdictions>
      <taxRuleType>STANDARD</taxRuleType>
    </taxCodeMaps>
  </taxCodeValidityPeriods>
</taxCodes>
```

4. Using a database viewer, open the TAXCODE table in the PDC database schema. Write down the internal ID associated with the tax code.

For example, [Figure 4-2](#) shows the internal ID for the sample VAT tax code.

Figure 4-2 VAT Internal ID



5. Add the tax code's internal ID to a new **<internalId>** element. For example:

```

<taxCodes>
  <name>VAT</name>
  <internalId>5eca012b-545e-43cc-a702-f4d3570e248e</internalId>
  <priceListName>Default</priceListName>
  <obsolete>>false</obsolete>
  <code>VAT</code>
  <taxPackageType>CUSTOM</taxPackageType>
  <taxCodeValidityPeriods>
    <validFrom>20010201</validFrom>
    <validTo>20300131</validTo>
    <taxCodeMaps>
      <description>VAT-EU</description>
      <percent>5</percent>
      <taxJurisdictionLevel>FEDERAL</taxJurisdictionLevel>
      <taxJurisdictions>GB</taxJurisdictions>
      <taxJurisdictions>FR</taxJurisdictions>
      <taxRuleType>STANDARD</taxRuleType>
    </taxCodeMaps>
    <taxCodeMaps>
      <description>VAT-GB</description>
      <percent>4.5</percent>
      <taxJurisdictionLevel>FEDERAL</taxJurisdictionLevel>
      <taxJurisdictions>GB</taxJurisdictions>
      <taxRuleType>STANDARD</taxRuleType>
    </taxCodeMaps>
    <taxCodeMaps>
      <description>VAT-FR</description>
      <percent>4.0</percent>
      <taxJurisdictionLevel>FEDERAL</taxJurisdictionLevel>
      <taxJurisdictions>FR</taxJurisdictions>
      <taxRuleType>STANDARD</taxRuleType>
    </taxCodeMaps>
  </taxCodeValidityPeriods>
</taxCodes>

```

6. Save and close the file.

The XML file is ready to be loaded into the PDC database.

Distributing Noncurrency Balances in Incremental Buckets

 **Note:**

This functionality is supported only for recurring and one-time charges that have validity periods with a specified end date.

You can now grant noncurrency balances to your customers in smaller portions on an incremental basis. For example, if you grant customers 500 MB of data with a five-hour validity period, you could dole out the data to customers in hourly buckets of 100 MB each. To configure PDC do this, you create multiple validity periods in your noncurrency balances.

See "Granting Noncurrency Balances in Increments" in *PDC Creating Product Offerings*.

Now Specify Whether to Continue Charging for SuspendedActive Accounts

When you add charges to a charge offer, you can now specify whether BRM continues charging or stops charging accounts that have a **SuspendedActive** custom life cycle state.

See "Specifying Whether to Continue Charging Inactive, Canceled, or SuspendedActive Accounts" in *PDC Creating Product Offerings*.

Support for Purchasing the Same Offer Multiple Times

When creating bundles in PDC, you can specify what happens if customers purchase the same offer more than once. The additional offer can be:

- Purchased as a new subscription
- Purchased as a replacement of the existing subscription
- Purchased as an extension of the existing subscription if it is purchased within a specified grace period.

The grace period can be any number of seconds, minutes, hours, or days. The default grace period is 0 days, which specifies that the grace period never ends.

For more information, see "About Purchasing the Same Offer Multiple Times" in *PDC Creating Product Offerings*.

To be able to add a grace period for charge offer extensions, you must run the **addBundledProductOfferingItemGracePeriod.sql** script after you install PDC 12.0 Patch Set 5 or later:

1. Go to the *PDC_home/PDC/interimpatch/upgrade* directory.
2. Run this command, which opens SQL*Plus:

```
% sqlplus login@ORACLE_SID
Enter password: password
```

where:

- *login* is the user name for the PDC database schema.
- *ORACLE_SID* is the database alias of the PDC database schema.
- *password* is the password for the specified user name.

3. Run this command:

```
SQL> @addBundledProductOfferingItemGracePeriod.sql
```

4. Exit SQL*Plus.

Prorating Billing DOM Changes, Package Transitions, and Bundle Transitions

PDC now allows you to specify how to handle recurring charges when your customers do the following in the middle of their billing cycle:

- Change their billing day of month (DOM). You can specify whether to apply the full cycle charge, prorate the cycle charge, or apply no charge. For more information, see "About Prorating Recurring Charges and Rollovers" in *PDC Creating Product Offerings*.
- Transition from one bundle to another. You can specify whether to apply the full cycle charge from the original bundle, apply the full cycle charge from the new bundle, or prorate the cycle charges from both bundles. For more information, see "Prorating Charges During Bundle Transitions" in *PDC Creating Product Offerings*.
- Transition from one package to another. You can specify whether to apply the full cycle charge from the original package, apply the full cycle charge from the new package, or prorate the cycle charges from both packages. For more information, see "Prorating Charges During Package Transitions" in *PDC Creating Product Offerings*.

To use this new functionality, you must run a database upgrade script after you install PDC 12.0 Patch Set 5 or later:

1. Go to the `PDC_home/PDC/interimpatch/upgrade` directory.
2. Run this command, which opens SQL*Plus:

```
% sqlplus login@ORACLE_SID
Enter password: password
```

where:

- `login` is the user name for the PDC or XREF database schema.
 - `ORACLE_SID` is the database alias of the PDC or XREF database schema.
 - `password` is the password for the specified user name.
3. Run one of the following commands:
 - For PDC schema users:

```
SQL> @AddProrationOptionForCycleAndTransition.sql
```
 - For XREF schema users:

```
SQL> @AddProrateCycleXrefRRECharges.sql
```
 4. Exit SQL*Plus.

Setting Charge Offer Cycle Alignment for Reactivated Bundles

When you add a charge offer to a bundle, you also specify, if a customer suspends and later reactivates their subscription, whether the reactivated charge offer's cycle alignment is based on the original billing or purchase date, or the reactivation date. You use the **Renewal Mode** option on the charge offer to configure this. You can also configure it in XML using the `ImportExportPricing` utility.

For more information, see "About Creating Bundles" and "Setting Charge Offer Cycle Alignment for Reactivated Bundles" in *PDC Creating Product Offerings*.

PDC UI Supports Recurring Charges on Specific DOM

When you create charge offers in the PDC UI, you can now specify to apply recurring charges on a specific day of the month (DOM) rather than on the customer's billing DOM or on the purchase date. Previously, this functionality could be implemented using only an XML file and the `ImportExportPricing` utility.

For more information, see "Applying Recurring Charges on a Specific DOM" in *PDC Creating Product Offerings* and "Specifying Charge Offer Settings" in *PDC Online Help*.

PDC UI Supports Versioning with Date Ranges

When creating a charge offer in PDC, you can now configure how pricing configurations for date ranges are applied to support versioning.

You can configure whether customers who subscribe during one date range move to the new pricing configuration after the first range ends, or keep the same pricing configuration, effectively maintaining multiple versions of the same charge offer. You can also determine whether the pricing configuration is chosen based on the service instantiation date or the purchase date.

Previously, this functionality could be implemented using only an XML file and the **ImportExportPricing** utility.

For more information, see "Configuring How Pricing Configurations for Date Ranges are Applied" in *PDC Creating Product Offerings* and "Specifying Charge Offer Settings" in *PDC Online Help*.

PDC UI Supports Options for Exceeding Credit Limits

In the PDC UI, when configuring properties for quantity ranges in charges, you can specify what happens to subscriptions when customers exceed their credit limits.

You can prevent customers from exceeding their credit limit and either prorate the resources according to the available balance, or fail the subscription and notify an external system for further processing.

You can allow customers to exceed their credit limit and:

- Use all available balance and record the remaining amount as an outstanding amount.
- Use all available balance and grant a loan for the remaining amount.
- Leave the available balance and record the entire amount as an outstanding amount.
- Skip billing for this cycle.

For more information, see "Allowing Customers to Exceed Their Credit Limit" in *PDC Creating Product Offerings*.

Note:

If you attempt to import charge offers with the **ImportExportPricing** utility using XML files that were exported before installing Patch Set 5, the import will fail with a validation error. You must update the values of the **<enforceCreditLimit>** element from **true** to **NORMAL** and from **false** to **DEFAULT** before importing.

PDC UI Supports Product Specification Attributes

The PDC UI now supports product specification attributes, which are created in a configuration template file. You can now use the PDC UI to provide values for the attributes for charge offers, discount offers, charginshare offers, bundles, and packages. Previously, this functionality could be implemented using only an XML file and the **ImportExportPricing** utility.

For more information, see "Configuring Product Specification Attributes for Pricing Components" in *PDC Creating Product Offerings* and "Specifying Product Specification Attribute Values" in *PDC Online Help*.



Note:

Previously, product specification attributes were called extended attributes. The name has been updated to align with the industry-standard TM Forum term.

PDC REST Services Manager Supports OAuth with Oracle Access Management

You can now implement OAuth to secure communication with PDC REST Services Manager by using Oracle Access Management. Previously you were required to use Oracle Identity Cloud Service.

See "PDC REST Services Manager Security" in *BRM Security Guide* for more information about implementing OAuth using either provider.

PDC REST Services Manager Supports Synchronizing New Elements

PDC REST Services Manager can now synchronize the following elements in TMF620 payloads from your enterprise product catalog to PDC:

- Charge selectors with value maps
- Packages and bundles containing multiple charge offers and discounts
- Subscription terms for packages and bundles

See "PDC REST Services Manager Overview" in *PDC REST Services Manager Integration Guide* for more information about elements you can synchronize using PDC REST Services Manager.

New Features in PDC 12.0 Patch Set 4

PDC 12.0 Patch Set 4 includes the following enhancements:

- [Creating Subscription Terms](#)
- [Creating Deliverables](#)
- [XML Product Offering Enhancements for PDC](#)
- [Integrating PDC with External Enterprise Product Catalogs](#)

Creating Subscription Terms

You can now create subscription terms in PDC and add them to your packages and bundles.

Subscription terms define the commitment period and options for canceling and renewing the goods and services you offer. For example, a subscription term could have a commitment period such as 1 month, 1 year, or 2 years. When you create a subscription term, you specify whether customers can cancel their subscriptions early and whether they incur any fees for doing so.

After you create subscription terms, you associate them with packages and bundles. When customers purchase a package and accept its terms, it becomes a contract.

See the following topics in *PDC Creating Product Offerings*:

- Creating Subscription Terms
- Creating Packages
- Creating Bundles

Creating Deliverables

You can now create deliverables in XML and load them to PDC using the **ImportExportPricing** utility.

A deliverable is a good or service that provides value to your customers, such as an eBook, a night in a hotel room, or a month of telephony service. It defines the type of good or service you are selling, its standalone selling price, its revenue earning schedule, and its revenue G/L ID.

You can add deliverables to your charge offers in PDC.

See the following topics in *PDC Creating Product Offerings*:

- Configuring Deliverables
- Configuring Charge Offers

XML Product Offering Enhancements for PDC

PDC now supports the following enhancements when you define your product offerings in XML and upload them using the **ImportExportPricing** utility:

- **Creating extended attributes:** You can create extended attributes for charge offers, discount offers, chargeshare offers, bundles, and packages. Extended attributes save extra information that may be useful to external applications. You create extended attributes in a configuration template file, then provide values for them while creating your pricing components. The utility validates the attributes on the pricing components against the template file. Although BRM does not act on the extended attributes, the information is stored in the BRM database, and can be queried by external applications.

See "Configuring Extended Attributes for Pricing Components" in *PDC Creating Product Offerings*.

- **Purchasing the same charge offer multiple times:** When creating bundles, you can specify what happens if customers purchase the same charge offer more than once. See "Purchasing the Same Charge Offer Multiple Times" in *PDC Creating Product Offerings*.
- **Offering hourly balance impacts:** When creating charges, you can configure hourly validity periods for the balance impacts that make up charges. This is useful for creating short-lived rewards, like free games or minutes. See "Offering Hourly Balance Impacts" in *PDC Creating Product Offerings*.
- **Applying recurring charges on a specific day of the month:** When creating charge offers with hourly validity, you can apply recurring charges on a specific day of the month instead of the customer's billing date or the purchase date. See "Applying Recurring Hourly Charges on a Specific Day of the Month" in *PDC Creating Product Offerings*.
- **Using date ranges for versioning:** When creating charge offers, you can add charges with new date ranges to create new versions of the same charge offer and determine whether existing subscriptions move to the new charge or continue with the old charge. Charge offers created before PDC 12.0 Patch Set 4 automatically use the existing functionality, where existing subscriptions move to the new charge. You can update the value of the new **dateRangeImpactType** field for the charge offers to change to the new versioning options. See "Using Date Ranges for Versioning" in *PDC Creating Product Offerings*.
- **Service is optional for charges:** When creating charges for the ECE rating, online rating, and subscription target engine profiles (ECE_RATING, ECE_SUBSCRIPTION, RRE_RATING, RRE_SUBSCRIPTION), specifying a service for a charge is optional.

**Note:**

BRM also supports these enhancements with the **loadpricelist** utility. See "XML Examples of Pricing Components" in *Configuring Pipeline Rating and Discounting*.

Integrating PDC with External Enterprise Product Catalogs

You can now integrate PDC with external product catalogs, such as Oracle Digital Experience for Communications Launch Experience, by using the new PDC REST Services Manager.

This integration allows you to create product offerings with charges and discounts in an external product catalog while using PDC and BRM to do the rating and billing.

For information about PDC REST Services Manager, see "PDC REST Services Manager Overview" in *PDC REST Services Manager Integration Guide*.

For information about installing PDC REST Services Manager, see "Installing Only PDC REST Services Manager" in *PDC Installation Guide*.

New Features in PDC 12.0 Patch Set 3

PDC 12.0 Patch Set 3 includes the following enhancements:

- [PDC Now Supports 5G Rating in ECE](#)
- [PDC Now Localized](#)

PDC Now Supports 5G Rating in ECE

PDC has been enhanced to support 5G rating in ECE. The PDC event definition file includes these new attributes, which indicate that the network attribute is from a 5G network:

- **networkAttributeItem5g**
- **persistedName5g**

For more information, see "Enabling Charging for Custom Events" in *PDC Creating Product Offerings*.

PDC Now Localized

PDC is now available in localized versions. The following languages are supported: French, Italian, Spanish, Japanese, Korean, Chinese Simplified, Chinese Traditional, Russian, and Portuguese Brazilian.

Localized versions of the software are available in the full installer JAR file: **pdcserver-12.0.0.3.0_generic_full.jar** file.

New Features in PDC 12.0 Patch Set 2

PDC 12.0 Patch Set 2 includes the following enhancements:

- [Support for Configuring Item Type Selectors in the PDC User Interface](#)
- [PDC Web Services for Creating and Exporting Components](#)
- [PDC Synchronizes Event Data Using Event Types](#)
- [PDC Now Supports Wildcard in Item Type Selectors](#)
- [Support for Enabling SSO Using SAML](#)

Support for Configuring Item Type Selectors in the PDC User Interface

In previous releases, you could configure item type selector rules only by importing them into the PDC database by using the **ImportExportPricing** utility.

You can now configure item type selector rules by using the PDC user interface. For more information, see the discussion about configuring the item type selectors in the PDC Online Help.

PDC Web Services for Creating and Exporting Components

In previous releases, you could use PDC Web services only to create and modify pricing components. With this enhancement, you can also use PDC Web services to perform the following:

- Export operation for Pricing components.
- Create, modify, and export operations for Setup components.
- Create, and export operations for Metadata, and Custom data.

Creating or Modifying Components using PDC Web Services

The PDC Web service uses the **PricingGateway.xsd** file for creating or modifying components in PDC. The XSD files describe the structure of the XML document. The XML file you create must comply with the structure defined in the XSD. The **PricingGateway.xsd** file is available at:

```
http://hostName:sslPortNumber/pdc/PricingGatewayPort?xsd=1
```

where:

- *hostName* is the host name of the machine on which PDC is deployed.
- *sslPortNumber* is the SSL port number of the domain on which PDC is deployed.

You can use the PDC Web service to do the following:

- Create the setup components defined in an XML file in PDC. See "[createBusinessConfig](#)" for more information.
- Create the setup components defined in an XML file in PDC and publish the setup components to a billing system, such as Oracle Communications Billing and Revenue Management (BRM). See "[createBusinessConfigAndSubmit](#)" for more information.
- Modify the setup components in PDC as defined in an XML file. See "[modifyBusinessConfig](#)" for more information.
- Modify the setup components in PDC as defined in an XML file and publish the setup components to a billing system. See "[modifyBusinessConfigAndSubmit](#)" for more information.
- Create the metadata defined in an XML file in PDC. See "[createMetadata](#)" for more information.
- Create the custom data defined in an XML file in PDC. See "[createCustomFields](#)" for more information.

createBusinessConfig

This Web service operation validates the input XML by comparing the XML fields and values against the values in the **PricingGateway.xsd** file and the rules for each type of Business config component. If the validation is successful, it retrieves the data from the XML file and creates Business config components in PDC.

The **createBusinessConfig** operation does not publish the Business config components to the BRM database.

Syntax

```
public oracle.communications.brm.pdc.server.service.types.PDCResponseType
createBusinessConfig(oracle.communications.brm.pdc.server.service.types.PricingInputXML
Type param) throws
oracle.communications.brm.pdc.server.service.PricingExceptionResponse;
```

createBusinessConfigAndSubmit

This Web service operation validates the input XML by comparing the XML fields and values against the values in the **PricingGateway.xsd** file and the rules for each type of Business config component. If the validation is successful, it retrieves the data from the XML file,

creates Business config components in PDC, and publishes the Business config components to the BRM database.

Syntax

```
public oracle.communications.brm.pdc.server.service.types.PDCResponseType  
createBusinessConfigAndSubmit(oracle.communications.brm.pdc.server.service.types.  
PricingInputXMLType param) throws  
oracle.communications.brm.pdc.server.service.PricingExceptionResponse;
```

modifyBusinessConfig

This Web service operation validates the input XML by comparing the XML fields and values against the values in the **PricingGateway.xsd** file and the rules for each type of Business config component. If the validation is successful, it retrieves the data from the XML file and updates the existing Business config components in PDC.

The **modifyBusinessConfig** operation does not publish the Business config components to the BRM database.

Syntax

```
public oracle.communications.brm.pdc.server.service.types.PDCResponseType  
modifyBusinessConfig(oracle.communications.brm.pdc.server.service.types.PricingIn  
putXMLType param) throws  
oracle.communications.brm.pdc.server.service.PricingExceptionResponse;
```

modifyBusinessConfigAndSubmit

This Web service operation validates the input XML by comparing the XML fields and values against the values in the **PricingGateway.xsd** file and the rules for each type of Business config component. If the validation is successful, it retrieves the data from the XML file and updates the existing Business config components in PDC, and publishes the Business config components to the BRM database.

Syntax

```
public oracle.communications.brm.pdc.server.service.types.PDCResponseType  
modifyBusinessConfigAndSubmit(oracle.communications.brm.pdc.server.service.types.  
PricingInputXMLType param) throws  
oracle.communications.brm.pdc.server.service.PricingExceptionResponse;
```

createMetadata

This Web service operation validates the input XML by comparing the XML fields and values against the values in the **PricingGateway.xsd** file and the rules for each type of Metadata component. If the validation is successful, it retrieves the data from the XML file and creates Metadata components in PDC.

The **createMetadata** operation does not publish the Metadata components to the BRM database.

Syntax:

```
public oracle.communications.brm.pdc.server.service.types.PDCResponseType  
createMetadata(oracle.communications.brm.pdc.server.service.types.PricingInputXML  
Type param) throws  
oracle.communications.brm.pdc.server.service.PricingExceptionResponse;
```

createCustomFields

This Web service operation validates the input XML by comparing the XML fields and values against the values in the **PricingGateway.xsd** file and the rules for each type of Custom Fields. If the validation is successful, it retrieves the data from the XML file and creates Custom Fields in PDC.

The **createCustomFields** operation does not publish the Custom Fields to the BRM database.

Syntax

```
public oracle.communications.brm.pdc.server.service.types.PDCResponseType  
createCustomFields(oracle.communications.brm.pdc.server.service.types.PricingInputXMLType  
param) throws oracle.communications.brm.pdc.server.service.PricingExceptionResponse;
```

getPricingObjects

This Web service operation validates the input XML by comparing the XML fields and values against the values in the **RetrievePricingGateway.xsd** file. If the validation is successful, it retrieves the Pricing Objects from PDC and returns the data in **RetrieveResponseType** object.

Syntax

```
public oracle.communications.brm.pdc.server.service.types.RetrieveResponseType  
getPricingObjects(oracle.communications.brm.pdc.server.service.types.RetrieveInputXMLType  
param) throws Exception;
```

getBusinessConfigObjects

This Web service operation validates the input XML by comparing the XML fields and values against the values in the **RetrievePricingGateway.xsd** file. If the validation is successful, it retrieves the Business Config from PDC and returns the data in **RetrieveResponseType** object.

Syntax

```
public oracle.communications.brm.pdc.server.service.types.RetrieveResponseType  
getBusinessConfigObjects(oracle.communications.brm.pdc.server.service.types.RetrieveInputXMLType  
param) throws Exception;
```

getMetadataObjects

This Web service operation validates the input XML by comparing the XML fields and values against the values in the **RetrievePricingGateway.xsd** file. If the validation is successful, it retrieves the Metadata Object from PDC and returns the data in **RetrieveResponseType** object.

Syntax

```
public oracle.communications.brm.pdc.server.service.types.RetrieveResponseType  
getMetadataObjects(oracle.communications.brm.pdc.server.service.types.RetrieveInputXMLType  
param) throws Exception;
```


getBRMObjects

This Web service operation validates the input XML by comparing the XML fields and values against the values in the **RetrievePricingGateway.xsd** file. If the validation is successful, it retrieves the BRM Object from PDC and returns the data in **RetrieveResponseType** object.

Syntax

```
public oracle.communications.brm.pdc.server.service.types.RetrieveResponseType  
getBRMObjects(oracle.communications.brm.pdc.server.service.types.RetrieveInputXML  
Type param) throws Exception;
```

getCustomFields

This Web service operation validates the input XML by comparing the XML fields and values against the values in the **RetrievePricingGateway.xsd** file. If the validation is successful, it retrieves the Custom Fields from PDC and returns the data in **RetrieveResponseType** object.

Syntax

```
public oracle.communications.brm.pdc.server.service.types.RetrieveResponseType  
getCustomFields(oracle.communications.brm.pdc.server.service.types.RetrieveInputX  
MLType param) throws Exception;
```

Exporting Components using PDC Web Services

You can use the PDC Web services to do the following:

- Export the pricing components. See "[getPricingObjects](#)" for more information.
- Export the setup components. See "[getBusinessConfigObjects](#)" for more information.
- Export the metadata components. See "[getMetadataObjects](#)" for more information.
- Export the BRM components. See "[getBRMObjects](#)" for more information.
- Export custom fields. See "[getCustomFields](#)" for more information.

PDC Web services use the **RetrieveInputXMLType** object as input for exporting the PDC components. You can create the **RetrieveInputXMLType** object.

- Using the Setter Method. See "[Creating RetrieveInputXMLType Object Using Setter Method](#)".
- Using the XML. See "[Creating RetrieveInputXMLType Object Using XML](#)".

Creating RetrieveInputXMLType Object Using Setter Method

You can create the **RetrieveInputXMLType** object using setter methods (for example, `pojos`) with the required parameters to export components from PDC system.

For example, to export pricing components you can create the object as follows:

```
getPricingObjects() {  
    RetrieveResponseType response = new RetrieveResponseType();  
    RetrieveInputXMLType request = new RetrieveInputXMLType();  
    ObjectType objectType = new ObjectType();  
}
```

```

WS PricingObject pricingObject = new WSPricingObject();
pricingObject.getPricingObjectType().add(WSPricingObjectType.ALTERATION_EXCLUSION);
objectType.setPricingObject(pricingObject);
request.setObjectType(objectType);
request.setAllReferences(false);
request.setIncludeFailedObjects(false);
request.setObsolete(false);
request.setReferences(false);
UserContextType userContext = new UserContextType();
userContext.setUserid("pdcuser");
request.setUserContext(userContext);
response = pricingGatewayPortType.getPricingObjects(request);
return response;
}

```

Creating RetrieveInputXMLType Object Using XML

The PDC Web services use the **RetrievePricingGateway.xsd** file for exporting components from PDC. The XSD file describes the structure of the XML file. The XML file you create must comply with the structure defined in the XSD. The **RetrievePricingGateway.xsd** file is available at:

<http://hostName:sslPortNumber/pdc/PricingGatewayPort?xsd=2>

where:

- *hostName* is the host name of the machine on which PDC is deployed.
- *sslPortNumber* is the SSL port number of the domain on which PDC is deployed.

You need to manually set the required parameters in the XML file for indicating the objects that you want to export from the PDC system. The **RetrieveInputXMLType** object is then created by parsing this XML.

Parameters for Exporting PDC Components

[Table 4-1](#) provides the description of parameters used for exporting PDC components.

Table 4-1 Parameters for Exporting PDC Components

Parameter	Description
objectType	Specifies the object type to export. The valid values are pricingObject , businessConfigObject , brmObject , and metadataObject . You can specify the pricing object type for exporting. For example, ALTERATION_RATE_PLAN, USC_MAP and so on. If pricing object type is not specified, all the pricing components are considered for exporting. This parameter is also applicable for setup components, metadata, and cross-reference data.
references	Specifies whether to export data including references.
allReferences	Specifies whether to export data including all references. This is applicable only when exporting pricing objects. When allReferences is set to true for pricing objects, the response is written separately into pricingObjectResponse , configObjectResponse , and metadataObjectResponse fields. If the export allReferences is not set for pricing objects, the pricing object responses are written into response field.

Table 4-1 (Cont.) Parameters for Exporting PDC Components

Parameter	Description
obsolete	Specifies whether to export obsolete data.
includeFailedObjects	Specifies whether to export data including failed objects.
modifiedAfterDate	Specifies whether to export data based on modified date. ModifiedAfterDate supports the following formats: <ul style="list-style-type: none"> <i>yyyy-mm-dd</i> Example: 2010-01-05 <i>yyyy-mm-dd'T'hh:mm:ss</i> Example: 2010-01-05T23:59:59 <i>yyyy-mm-dd'T'hh:mm:ssz</i> Example: 2010-01-05T19:05:09GMT+05:30
modifiedByUser	Specifies whether to export data based on the user who modified the data.
objectName	Specifies whether to export data based on object name.
UserContext	Specifies whether to export data for a specific user.
productSpecName	Specifies whether to export data based on product specification. This is applicable only when exporting pricing components.

About Web Service Response

The PDC Web service operation first validates the input XML by comparing the XML fields and its values against the values in the **RetrievePricingGateway.xsd** file. If the validation is successful, it retrieves the objects from PDC and returns the data in **RetrieveResponseType** object.

Structure of the **RetrieveResponseType** object is as follows:

```
<xsd:complexType name="RetrieveResponseType">
  <xsd:annotation>
    <xsd:documentation>Get Object Webservice Response</xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="status" type="xsd:string"/>
    <xsd:element name="errors" type="xsd:string" maxOccurs="unbounded"
minOccurs="0"/>
    <xsd:element name="response" type="xsd:base64Binary"/>
    <xsd:element name="pricingObjectResponse" type="xsd:base64Binary"/>
    <xsd:element name="configObjectResponse" type="xsd:base64Binary"/>
    <xsd:element name="metadataObjectResponse" type="xsd:base64Binary"/>
  </xsd:sequence>
</xsd:complexType>
```

The **RetrieveResponseType** object contains the status of operation and the exported object fields in the XML binary format based on the input provided in **RetrieveInputXml** file. The status field value is either SUCCEEDED or FAILED. The **RetrieveResponseType** object also capture errors occurred during the operation.

Processing the Response from Export/GET API

When an export or GET operation is successful, you receive a response with the requested object data. You get the response as follows:

```
response = pricingGatewayPortType.getPricingObjects(request);
```

You can process the response object by using the following code in your application:

```
processResponse(RetrieveResponseType response) {
byte[] zippedBinaryStr = response.getResponse();
for (int i = 0; i < response.getErrors().size(); i++) {
    System.out.println("Cause : \n" + response.getErrors().get(i)+ "\n");
}
String status = response.getStatus(); //status values can be either SUCCEEDED or FAILED
}
```

Optionally, you can convert the response into the XML format by using the following code in your application:

```
private static void writeToFile(byte[] zippedBinaryStr, String filename) throws
Exception {
    try {
        FileOutputStream fpStream = new FileOutputStream(filename);
        BufferedOutputStream bufferedStream = new BufferedOutputStream(fpStream);
        extractZippedContentAndWriteXMLStream(zippedBinaryStr, bufferedStream);
        bufferedStream.flush();
        bufferedStream.close();
        fpStream.flush();
        fpStream.close();
    } catch (IOException e) {
        System.out.println("Exception caught when writing data to file");
        throw e;
    }
}

private static void extractZippedContentAndWriteXMLStream(byte[] binaryXMLString,
BufferedOutputStream bufferedFileStream) throws Exception {
    try (ByteArrayInputStream byteStrm = new ByteArrayInputStream(binaryXMLString);
        ZipInputStream in = new ZipInputStream((byteStrm))) {
        ZipEntry entry = in.getNextEntry();
        //Transfer bytes from the ZIP file to the output file
        if (entry != null && in.available() > 0) {
            byte[] buf = new byte[BUFFER_SIZE];
            int len = in.read(buf);
            while (len > 0) {
                bufferedFileStream.write(buf, 0, len);
                len = in.read(buf);
            }
            bufferedFileStream.flush();
        }
    } catch (Exception e) {
        throw e;
    }
}
```

where *filename* is the path to the XML file for storing the output.

PDC Synchronizes Event Data Using Event Types

With this enhancement, the BRM event data is synchronized with PDC based on the event type defined in BRM. For information on the list of event types in BRM, refer to "[BRM Supports POID Generation in ECE](#)".

Note:

When you set an event type in BRM, note the following:

- If the event type is set to **NONE** in BRM and the corresponding event is not mapped in the **pin_event_map** file, the event does not synchronize with PDC.
- If the event type is set to **NONE** in BRM and the corresponding event is mapped in the **pin_event_map** file, the event synchronizes with PDC. However, you should modify the event type in BRM from **NONE** to the relevant event type and then run the **SyncPDC** utility again if you want to use this event type for creating a pricing component.

After you create an event and synchronize it with PDC, it is not recommended to modify the event type of the event. However, if you have set the incorrect event type, you can modify the incorrect event type by updating it in BRM and then synchronizing in PDC. You can modify the event type only for the custom events. For example, if the event type for the parent event **/event/session/telco/gprs** is set as **USAGE_PREPAID** and the event type for the child event **/event/session/telco/gprs/master** is set as **NONE**, you must change the event type for the child event as **USAGE_PREPAID** in BRM, synchronize it with PDC, and then publish to ECE. See "[Synchronizing and Publishing Event Type to ECE](#)" for synchronizing the modified data to PDC and publishing it to ECE.

Synchronizing and Publishing Event Type to ECE

To synchronize the event type with PDC and publish it to ECE:

1. Ensure that the event type is changed in BRM. For more information, see the discussion about changing the event type in "[BRM Supports POID Generation in ECE](#)".
2. Run the **SyncPDC** utility.
3. Go to **PDC_home/apps/bin**, where **PDC_home** is the directory in which the PDC software is installed.
4. Export the event that you want to modify by using the following command:

```
ImpotExportPricing -export filename.xml -metadata
```

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

5. Verify the exported data in the XML file and set the **<TRUE<** value to **TRUE** in the following syntax:

```
<  
<1.0<
```

```
<TRUE<  
<
```

6. Import the modified XML file into PDC database by running the following command:

```
ImpotExportPricing -import -metadata filename -ow
```

Modified event type is published to ECE.

PDC Now Supports Wildcard in Item Type Selectors

PDC now supports wildcard (*) in item type selectors for services and events. You can use the wildcard to substitute one or more characters in the service or event name to indicate that any value is acceptable; for example, /service/telco/gsm*. The following elements are added in PDC to support wildcard: **applicableToAllChildServices** and **applicableToAllChildEvents**. By setting the **true** or **false** value to these elements, you can indicate whether the item type selector is applicable to child services or events.

If wildcard is used in the services and events, Oracle Communications Billing and Revenue Management Elastic Charging Engine (ECE), real-time rating engine, and batch rating engine use the **applicableToAllChildServices** and **applicableToAllChildEvents** values to identify if the services or events are applicable for all child services or events. If the value is set to **true**, the item type selector is considered for all the child services and events. If the value is set to **false**, the child services or events are not considered.

For more information, see the discussion about configuring item type selectors in *PDC Creating Product Offerings*.

Support for Enabling SSO Using SAML

You can now use SAML to enable single-sign on (SSO) in PDC. SSO allows you to log in to applications using a single user name and password combination.

For more information, see the discussion about configuring SAML for SSO in *PDC Installation Guide*.

New Features in PDC 12.0 Patch Set 1

PDC 12.0 Patch Set 1 includes the following enhancements:

- [Support for Rolling Back the PDC Patch Set](#)
- [Support for Secure Communication With the PDC Database](#)

Support for Rolling Back the PDC Patch Set

PDC now allows you to roll back a PDC patch set. For example, if you experience issues after installing PDC 12.0 Patch Set 1, you can roll back PDC to 12.0.

For more information, see the discussion about rolling back a patch set in *PDC Installation Guide*.

Support for Secure Communication With the PDC Database

PDC now supports Secure Sockets Layer (SSL)-enabled database to ensure secure communication between the PDC application and database.

For more information, see the discussions about configuring SSL for the Oracle database and configuring PDC to use SSL-enabled database in *PDC Installation Guide*.

5

New Features in Billing Care

The Oracle Communications Billing Care 12.0 Patch Set releases include several new features.

Topics in this document:

- [New Features in Billing Care 12.0 Patch Set 8](#)
- [New Features in Billing Care 12.0 Patch Set 7](#)
- [New Features in Billing Care 12.0 Patch Set 6](#)
- [New Features in Billing Care 12.0 Patch Set 5](#)
- [New Features in Billing Care 12.0 Patch Set 4](#)
- [New Features in Billing Care 12.0 Patch Set 3](#)
- [New Features in Billing Care 12.0 Patch Set 2](#)
- [New Features in Billing Care 12.0 Patch Set 1](#)

New Features in Billing Care 12.0 Patch Set 8

Billing Care 12.0 Patch Set 8 includes the following enhancements:

- It is now possible to set a flag for whether a notification for a business event needs to be opted in, needs to be opted out, or is mandatory (cannot be opted out). Also, at the subscriber preference level, there are separate lists for business events that can be opted in or opted out, based on this new flag. This flag is available in both the GUI and the REST API. See "Viewing and Adding Subscriber Preferences" in *Billing Care Online Help*.
- Billing Care and the Billing Care REST API deployment in a cloud native environment now supports a zero-downtime upgrade without the use of an alternate namespace. This simplified approach to upgrading in place, and the reduced dependence on other components of the solution like load balancer (to reroute traffic), will enable faster upgrades.

See "Configuring Billing Care, Billing Care REST API, and Business Operations Center for Cloud Native" and "Upgrading Your Billing Care and Billing Care REST API Cloud Native Services" in *BRM Cloud Native Deployment Guide* for more information.

- Billing Care now supports overriding credit limits when creating accounts, subscribing to new services, purchasing bundles, transitioning packages, and viewing the balance overlay for currency and non-currency resource flows. Two new fields have been introduced.

A new **Configure Bundle** section is added when you are creating an account, purchasing a service, purchasing a bundle, or transitioning a package. This section contains the **Credit limit check** field, which can have the following values:

- **Override credit limit:** The credit limit will be overridden
- **Prorate based on credit limit:** Prorated until the credit limit is reached

- **Enforce credit limit:** The transaction will fail if the credit limit check fails

A new **Credit limit check** field is added to the Balance view overlay, which can have the following values:

- Enforce credit limit
- Override credit limit

These new fields are available in both the GUI and the REST API. See "Creating an Account" in *Billing Care Online Help*.

- Billing Care now supports selecting multiple events and specifying a single amount (for example, \$5.00) by which to adjust each of the selected events. See "Adjusting Multiple Events for an Account" in *Billing Care Online Help*.
- The Billing Care Asset view now supports a tabular product-focused view with the option to apply filters and sort based on any column. See "Working with Assets in a Product-Focused Table" in *Billing Care Online Help*.
- Billing Care is now certified to integrate with Oracle Identity Cloud Service using SAML. See "Configuring SAML for SSO Using a Service Provider" in *Billing Care Installation Guide*.

New Features in Billing Care 12.0 Patch Set 7

Billing Care 12.0 Patch Set 7 includes the following enhancements:

- Billing Care now allows you to create promise-to-pay agreements in which customers can pay off the amount due through multiple installments. The installment schedule and amount can be set up automatically by Collections Manager or manually according to the customer. You can now also configure whether credit limits are automatically increased when an account enters a promise-to-pay agreement and then decreased when the amount due is paid off. See "Working with Promise-to-Pay Agreements" in *Billing Care Online Help*.
- Billing Care now supports the dynamic credit floor feature added in Patch Set 7. The dynamic credit floor is calculated using the granted amounts from all sub-balances that are valid for a resource's current cycle. Dynamic credit floor configuration is supported in both the GUI and the REST API.

This feature allows BRM to set the credit floor automatically when your customers are granted limited-time resources.

See "Working with Credit Limits, Credit Floors, and Thresholds" in *Billing Care Online Help*.
- Billing Care now supports rolling up a child account's credit limits to the parent billing account. This configuration is supported both in the GUI and the REST API.

See "Working with Credit Limits, Credit Floors, and Thresholds" in *Billing Care Online Help*.
- You can use the Billing Care GUI to configure notifications to be delivered after the due dates of certain business events, making use of the functionality added to BRM in Patch Set 7.

This feature allows you to send reminders to customers at fixed intervals after a triggering event has occurred.

See "Creating a Notification Specification for Post-Event Delivery" in *Billing Care Online Help*.

New Features in Billing Care 12.0 Patch Set 6

Billing Care 12.0 Patch Set 6 includes the following enhancements:

- [Billing Care GUI Enhancements in Patch Set 6](#)
- [Billing Care REST API Enhancements in Patch Set 6](#)
- [Billing Care SDK Enhancements in Patch Set 6](#)

Billing Care GUI Enhancements in Patch Set 6

The Billing Care GUI includes the following enhancements:

- The notification screens have been updated. See "Notifications" in *Billing Care Online Help*.
- You can create, collect, and refund customer deposits for devices, services, packages, or accounts. See "Deposits" in *Billing Care Online Help*.
- The payment and adjustment screens have been updated to display more detailed tax information. See "Payments" and "Working with Adjustments" in *Billing Care Online Help*.
- The bill details now display any dynamic taxation that has been configured. For information about dynamic taxation in BRM, see "About Creating Tax Codes (Patch Set 6 and Later)" in *BRM Calculating Taxes*.

Billing Care REST API Enhancements in Patch Set 6

The Billing Care REST API has been enhanced to support managing loans, notifications, deposits, and installments. See *REST API Reference for Billing Care* for more information.

Billing Care SDK Enhancements in Patch Set 6

You can now use the Billing Care SDK to make these customizations:

- You can now customize whether failed transactions in the Billing Care REST API are recorded in the BRM database. To do so, use the new **request.record.failure** flag in the **Configurations.xml** file. See "Recording Billing Care REST API Request Failures" in *Billing Care SDK Guide*.
- You can now customize the fields displayed in the Billing Care **Make a Payment** screen according to the selected payment method. See "Customizing the Make a Payment Screen" in *Billing Care SDK Guide*.
- You can now customize Billing Care to display a Success toast message when a payment or adjustment is submitted successfully. Success toast messages appear at the top of the Billing Care page and then disappear after a few seconds. See "Displaying Success Toast Messages in Billing Care" in *Billing Care SDK Guide*.

New Features in Billing Care 12.0 Patch Set 5

Billing Care 12.0 Patch Set 5 includes the following enhancements:

- You can now use Billing Care to create notification specifications, which provide the rules for sending messages to your end customers through an external notification application.

They define which customers can receive each type of message as well as how and when to deliver the message. See "Notifications" in *Billing Care Online Help*.

- You can now use Billing Care to enable your customers to pay their bills in installments. See "Installments" in *Billing Care Online Help*.
- When you purchase a product in Billing Care, you can now change the length of the grace period. Customers can reactivate and extend their expired product if they repurchase it within the grace period. See "Purchasing New Products and Services" in *Billing Care Online Help*.
- Your customers can now specify their preferences for receiving messages from an integrated system with both BRM and an external notification application. For example, customers can specify their messages' preferred language, delivery time, and delivery method. See "Viewing and Adding Subscriber Preferences" in *Billing Care Online Help*.

New Features in Billing Care 12.0 Patch Set 4

Billing Care 12.0 Patch Set 4 includes the following enhancements:

- [Monitoring Billing Care and Billing Care REST API](#)
- [Billing Care Application Enhancements in Patch Set 4](#)
- [Billing Care SDK Enhancements in Patch Set 4](#)

Monitoring Billing Care and Billing Care REST API

You can now use external applications, such as Prometheus and Grafana, to monitor Billing Care and the Billing Care REST API. For more information, see "Monitoring Billing Care and Billing Care REST API" in *BRM System Administrator's Guide*.

Billing Care Application Enhancements in Patch Set 4

The Billing Care application includes the following enhancements in Patch Set 4:

- Billing Care now allows you to manage contracts, which are subscriptions that include terms that your customers agree to when they purchase it. You can use Billing Care to:
 - Modify a contract's auto-renewal options
 - Purchase optional bundles that are associated with a contract
 - Cancel optional bundles in a customer's contract, applying any early termination fees for doing so
 - Cancel a customer's contract before the end of the commitment period, if allowed by the contract's terms, incurring any early termination fees for doing so

 **Note:**

Contracts are supported only when deliverable-based revenue recognition is enabled in your system.

- Your customers' subscription cards now show more details about the contracts and no-contract subscriptions that they have purchased.

 **Note:**

Contracts and no-contract subscriptions are supported only when deliverable-based revenue recognition is enabled in your system.

- You can now search for entries in the Billing Care Newsfeed by account number or payment ID.
- You can now add account-level notes. Account-level notes consolidate all comments that CSRs have entered for an account, allowing you to easily track and view all interactions that have occurred with your customers. Each note includes the name of the CSR that posted it, the date that it was entered, the note's comments, and any follow-up comments in the note's thread.
- You can now transfer negative balances (credits) from one bill unit to another bill unit that owes money. For example, if a customer's mobile phone bill has a balance of -\$10, that balance could be transferred to their \$50 Internet bill. In this case, the balance would become \$0 for the mobile phone bill and \$40 for the Internet bill.
- Your customers' Asset cards now show more details.

For more information on these enhancements, see *Billing Care Online Help*.

Billing Care SDK Enhancements in Patch Set 4

You can now use the Billing Care SDK and OPSS policies to make these customizations:

- You can extend the Billing Care REST framework to process custom headers sent in HTTP requests and responses and then perform additional functionality such as order tracking. You do this by adding filters to the Billing Care SDK that do the following:
 - Intercept HTTP requests before they are sent to the resource, and then make decisions based on the header value. For example, a filter could check whether the request is a duplicate and, if so, reject the request.
 - Intercept HTTP responses before they are sent to the client, and then make decisions based on the header value. For example, a filter could construct tracking objects and persist them in the BRM database.

For more information, see "Extending REST Services to Filter Custom Headers" in *Billing Care SDK Guide*.

- You can extend the Billing Care REST API to return complex data, such as all data stored in a BRM storable class, in a JAXB-annotated class. For example, it could return all fields in an **/account** object or the **/profile** object linked with an account. When you extend the BRM REST API to enrich response data, it returns object information in the **extension** field of response objects in JSON or XML format.

For more information, see "Extending REST API Response Objects" in *Billing Care SDK Guide*.

- You can extend the Billing Care REST API to support any OAuth token management tool. By default, the Billing Care REST API uses Oracle Access Manager to generate and validate OAuth requests from clients.

For more information, see "Using Custom OAuth Providers with Billing Care REST API" in *Billing Care SDK Guide*.

New Features in Billing Care 12.0 Patch Set 3

Billing Care 12.0 Patch Set 3 includes the following enhancements:

- [Hierarchy Enhancements](#)
- [Billing Care Installation Enhancements](#)
- [Billing Care SDK Enhancements](#)
- [Enhanced Billing Care Security](#)

For more information on these enhancements, see *Billing Care Online Help*.

Hierarchy Enhancements

You can now view organization hierarchies and billing hierarchies in Billing Care from the parent account through to the child accounts.

For more information, see "Organization Hierarchies" and "Billing Hierarchies" in *Billing Care Online Help*.

Billing Care Installation Enhancements

The Billing Care installer now allows you to:

- Deploy Billing Care on a WebLogic Cluster.
- Deploy Billing Care on a managed server that is currently offline.
- Deploy Billing Care on a managed server that is also part of a cluster.

The Billing Care GUI installer now displays the state (Running or Shut Down) of target servers on which to deploy Billing Care. At the end of the installation process, it also provides a sample link for accessing Billing Care.

For more information, see *Billing Care Installation Guide*.

Billing Care SDK Enhancements

You can now use the Billing Care SDK and OPSS policies to make these customizations:

- Open custom views in Billing Care in full screen mode. See "Opening Custom Views From Landing Page" in *Billing Care SDK Guide*.
- Set the time in Billing Care to a time zone that automatically handles Daylight Saving Time (DST) changes. To do so, use the **brmsserver.timezone** registry entry in the **Configurations.xml** file. See "Default Configuration File Entries" in *Billing Care SDK Guide*.
- Display summary or detailed account balance information in the Billing Care **Home** tab. See "About Customizing the Billing Care Home Tab" in *Billing Care SDK Guide*.
- Remove the **Balances** section from an account's home page in Billing Care. See "About Removing the Balances Area" in *Billing Care SDK Guide*.

- Create Billing Care search templates using your choice of attributes. See the sample XML search templates in the *SDK_home/BillingCare_SDK/samples/SDKTemplatesCustomization* directory.

Enhanced Billing Care Security

To secure your Billing Care Web cookies, you must now set the **SECURE_COOKIE** variable to **true** in your WebLogic Server **plan.xml** file. See "Securing Web Cookies" in *BRM Security Guide*.

New Features in Billing Care 12.0 Patch Set 2

Billing Care 12.0 Patch Set 2 includes the following enhancements:

- [SDK Enhancements for Customizing Billing Care](#)
- [View Bills Generated Before Moving the Account to a Hierarchy](#)
- [Support for Account Number and Bill Number in Billing Care Embeddable URLs](#)
- [Embeddable URLs for Purchase Package and Purchase Bundle Screens](#)

For more information on these enhancements, see the *Billing Care Online Help*.

SDK Enhancements for Customizing Billing Care

You can now use the Billing Care SDK and OPSS policies to perform the following in Billing Care:

- Customize reason codes displayed for event adjustments.
- Limit the adjustment percentage entered by customer service representatives (CSRs) for event adjustments.
- Disable the event adjustment options based on user roles.
- Restrict debit and credit adjustment options for event adjustments.
- Make the Notes field mandatory for additional product purchase and event adjustments.
- Display only event adjustments in the Bills section for performing adjustments.
- Restrict the purchase of additional bundles based on user roles.
- Restrict the validity or end date set by the CSRs while purchasing additional products or services.
- Set the maximum adjustment limit based on the currency resources used for event adjustments.
- Filter the bundles available for purchase.
- Filter start and end dates for additional purchase.
- Disable the **Charges not related to services** link and the link to child accounts in the **Bills** tab, **My Charges** area, and Payment Details dialog box.

For more information, see *Billing Care SDK Guide*.

View Bills Generated Before Moving the Account to a Hierarchy

In previous releases, when an account was moved to a hierarchy as a child account, the bills generated for the account earlier were not displayed in the bills list.

With this enhancement, you can view the bills that are generated before and after moving the account to a hierarchy in the account's **Bills > Switch Bills** menu. For more information, see the discussion about switch bills in the *Billing Care Online Help*.

Support for Account Number and Bill Number in Billing Care Embeddable URLs

In the previous releases, only account ID and bill ID was supported in the Billing care embeddable URLs.

With this enhancement, you can also use account number and bill number instead of the corresponding POIDs in Billing Care URLs that are embedded in external applications.

For more information, see the discussion about embedding billing care screens in *Billing Care SDK Guide*.

Embeddable URLs for Purchase Package and Purchase Bundle Screens

Billing Care supports embedding screens into CRM applications and online account management interfaces to directly access the Billing Care functionality.

With this enhancement, you can embed Purchase Package and Purchase Bundle screens into external applications.

For more information, see the discussion about embedding billing care screens in *Billing Care SDK Guide*.

New Features in Billing Care 12.0 Patch Set 1

Billing Care 12.0 Patch Set 1 includes the following enhancements:

- [Additional Options for Searching Bills](#)
- [Billing Care Is Now Certified with Mozilla Firefox 58.0](#)
- [Billing Care is Now Certified with Oracle Identity and Access Management 12c Release 2](#)
- [SDK for Customizing the Purchase Flow and Assets Display](#)
- [Support for Backdated Account Creation and Product Purchase](#)
- [Support for Creating Wholesale Accounts and Billing Hierarchies](#)
- [Support for Customizing Product Information Post Account Creation](#)
- [Support for Deferred Actions](#)
- [Support for Multischema](#)

- [Support for Undefined Payment Method](#)
- [Support for Enabling SSO using SAML](#)

For more information on these enhancements, see the *Billing Care Online Help*.

Additional Options for Searching Bills

Billing Care now allows you to search for:

- Specific bills by using bill number
- The last generated bill
- Bills generated between specific dates by using date range

Billing Care Is Now Certified with Mozilla Firefox 58.0

Currently, Billing Care 12.0 is certified with Mozilla Firefox 54.0.1. With this patch, Billing Care 12.0 is certified with Mozilla Firefox 58.0.

For more information, see *Billing Care Installation Guide*.

Billing Care is Now Certified with Oracle Identity and Access Management 12c Release 2

Currently, Billing Care 12.0 is certified with Oracle Identity and Access Management (OIAM) 11g Release 2 (11.1.2.3.0). With this patch, Billing Care 12.0 is certified with OIAM 12c R2 (12.2.1.3.0).

For more information, see *Billing Care Installation Guide*.

SDK for Customizing the Purchase Flow and Assets Display

You can now customize the Purchase Catalogue screen, add custom fields, and customize the Assets area to display the custom fields by using the Billing Care SDK. For example, this SDK allows you hide notes, fields, and the purchase start date or end date in the Purchase Catalogue screen and view the newly added custom fields in the Assets area.

For more information on customizing Billing Care to customize Purchase Catalogue and Assets display, see *Billing Care SDK Guide*.

Support for Backdated Account Creation and Product Purchase

Billing Care now allows you to backdate the account creation and product purchase. You can now enter or select an account creation date or a product purchase date that is earlier than the current date.

Support for Creating Wholesale Accounts and Billing Hierarchies

Billing Care now allows you to create wholesale accounts and billing hierarchies. You can create wholesale parent and child accounts and set up a wholesale billing hierarchy by assigning the bill unit of the wholesale parent account to the wholesale business profile and the bill unit of each wholesale child account to the wholesale parent account for payment.

Support for Customizing Product Information Post Account Creation

Billing Care now allows you to customize products after account creation by using the **Rate Customizations** option. For example, you can modify a package or bundle and the start date and end date for charges (such as recurring, one time, and usage charges).

Support for Deferred Actions

Billing Care now allows you to perform actions on the account, service, and collections-related deferred actions that are already created. By using the **Deferred Actions** option, you can edit, run, and delete the deferred actions.

Support for Multischema

Billing Care now supports multischema. You can now view the accounts in the different schemas in Billing Care.

Support for Undefined Payment Method

Billing Care now allows you to create account and access services without defining any specific payment method. You can now set the payment method for a bill unit to **No Payment Method** when there is no payment method currently defined for an account or service. This enables you offer free trial for customers. You can define a specific payment method after the free period has ended.

Support for Enabling SSO using SAML

You can now use SAML to enable single-sign on (SSO) in Billing Care. SSO allows you to log in to applications using a single user name and password combination.

For more information, see the discussion about configuring SAML for SSO in *Billing Care Installation Guide*.

6

New Features in Business Operations Center

The Oracle Communications Business Operations Center 12.0 Patch Set releases include several new features.

Topics in this document:

- [New Features in Business Operations Center 12.0 Patch Set 8](#)
- [New Features in Business Operations Center 12.0 Patch Set 4](#)
- [New Features in Business Operations Center 12.0 Patch Set 3](#)
- [New Features in Business Operations Center 12.0 Patch Set 2](#)
- [New Features in Business Operations Center 12.0 Patch Set 1](#)

New Features in Business Operations Center 12.0 Patch Set 8

Business Operations Center 12.0 Patch Set 8 includes the following enhancements:

- Previously, Business Operations Center only displayed a **View Failure Report** link for ready-to-use jobs, like billing and invoicing.
Now, Business Operations Center has been enhanced to display a **View Failure Report** link for all custom jobs that are based on the multithreaded application (MTA) framework.
For custom jobs that are *not* based on the MTA framework, it is the customer's responsibility to populate the audit table. If there is data present in the audit table, the system will display the **View Failure Report** link. For more information, see "Viewing Details of Failed Jobs" in *Business Operations Center Online Help*.
- The Business Operations Center deployment in a cloud native environment now supports a zero-downtime upgrade without the use of an alternate namespace. This simplified approach to upgrading in place, and the reduced dependence on other components of the solution like load balancer (to reroute traffic), will enable faster upgrades.
See "Configuring the Billing Care, Billing Care REST API, and Business Operations Center Services" and "Upgrading Your Business Operations Center Cloud Native Services" in *BRM Cloud Native Deployment Guide* for more information.
- Business Operations Center is now certified to integrate with Oracle Identity Cloud Service using SAML. See "Configuring SAML for SSO" in *Business Operations Center Installation Guide*.

New Features in Business Operations Center 12.0 Patch Set 4

Business Operations Center 12.0 Patch Set 4 includes the following enhancements:

- The General Ledger job now includes options for generating preview or export reports that comply with deliverable-based revenue recognition. Previously, you could generate reports for only event-based revenue recognition. See "Generating General Ledger Reports" in *Business Operations Center Online Help*.

- Business Operations Center now runs business operations jobs by calling the PCM_OP_JOB_EXECUTE and PCM_OP_JOB_PROCESS_TEMPLATE opcodes. In previous releases, it ran business operations jobs by running the **pin_job_executor** utility. See "How Business Operations Center Runs Jobs" in *BRM System Administrator's Guide*.
- Business Operations Center now supports multischema systems. You can choose to run a business operations job on the primary schema, on one of the secondary schemas, or on all of the schemas. By default, business operations jobs are run on all schemas. See "Running Business Operations Jobs" in *Business Operations Center Online Help*.
- You can now customize Business Operations Center to run your custom applications against a BRM multischema system. See "Setting Up Business Operations Center to Run Custom Applications on Multischema Systems" in *BRM System Administrator's Guide*.
- In the **Business Dashboard** page of Business Operations Center, the **Subscriptions** graphs have been removed and replaced with new **Product Popularity** graphs. The **Product Popularity** graphs consist of the following two bar charts:
 - **Most Popular Products:** This shows the number of subscriptions for each of your top ten most popular products.
 - **Least Popular Products:** This shows the number of subscriptions for each of your bottom ten least popular products.See "Viewing Business Trends" in *Business Operations Center Online Help*.
- You can now monitor Business Operations Center by using external applications, such as Prometheus and Grafana. For more information, see "Monitoring Business Operations Center" in *BRM System Administrator's Guide*.

New Features in Business Operations Center 12.0 Patch Set 3

Business Operations Center 12.0 Patch Set 3 includes the following enhancements:

- [Business Operations Center Installation Enhancements](#)
- [Business Operations Center Supports Custom Jobs](#)
- [Business Operations Center Upgrades Now Supported](#)
- [Support for Enabling SSO using SAML](#)

Business Operations Center Installation Enhancements

The Business Operations Center installer now allows you to:

- Deploy Business Operations Center on a WebLogic Cluster.
- Deploy Business Operations Center on a managed server that is currently offline.
- Deploy Business Operations Center on a managed server that is also part of a cluster.

The Business Operations Center GUI installer now displays the state (Running or Shut Down) of target servers on which to deploy Business Operations Center.

For more information, see *Business Operations Center Installation Overview*.

Business Operations Center Supports Custom Jobs

You can now set up custom jobs in Business Operations Center that run your custom MTA and non-MTA applications.

See "Setting Up Custom Jobs" and "Running Custom Jobs" in *Business Operations Center Online Help*.

Business Operations Center Upgrades Now Supported

You can now upgrade from these versions of Business Operations Center to Business Operations Center 12.0 Patch Set 3:

- Business Operations Center 7.5
- Business Operations Center 7.5 Patch Set 1 and later patch sets
- Business Operations Center 12.0
- Business Operations Center 12.0 Patch Set 1 and Patch Set 2

For information about how to upgrade, see "Upgrading Business Operations Center" in *BRM Upgrade Guide*.

Support for Enabling SSO using SAML

You can now use SAML to enable single-sign on (SSO) in Business Operations Center. SSO allows you to log in to applications using a single user name and password combination.

For more information, see "Configuring SAML for SSO" in *Business Operations Center Installation Guide*.

New Features in Business Operations Center 12.0 Patch Set 2

Business Operations Center 12.0 Patch Set 2 includes the following enhancements:

- [Workflow Job to Automate Billing Process](#)

Workflow Job to Automate Billing Process

In previous releases, you had to create separate business operation center jobs for billing and invoicing and manage the dependency between the jobs using scheduled time.

In this case, if the jobs were not run in the correct order, the bills could reflect incorrect charges.

With this enhancement, you can run a workflow job, which can include billing, payment collections, invoices, and refunds. Workflow jobs enable you to automate an end-to-end billing process.

 **Note:**

Only BOC Super Admin can run workflow jobs.

For more information on the Admin role, see the discussion about roles in the Business Operations Center Help.

You can include one or more of these jobs in a workflow job, in this order:

- Billing
- Invoicing
- Payment collections, including the option to settle any previously authorized one-time payments.
- Refunds

 **Note:**

You can't change the order. A workflow job stops if any individual job fails.

And, you can specify these attributes to determine which accounts are included:

- The status of the accounts to bill: active, inactive, or closed.
- The payment method.
 - Payment card (credit card or debit card)
 - Invoice
- The account's billing day of month. You can run billing for accounts whose billing day of month is:
 - On or before the job is run. The job creates bills for accounts whose billing date is any day before midnight of the day that you run the job. If you run billing every day, the accounts billed are from the previous day. If you run billing every two weeks, all accounts with a billing date in the previous two weeks are billed.
 - A specified number of days before the job is run.
 - On a specific day of the month. You can enter multiple days.

If invoice jobs are included, specify the level of information to include in the invoices: summary or detailed.

You can view the workflow job status and metrics from the **Job History** page by clicking the link displayed under **Processed** in each row. However, you cannot rerun the failed workflow jobs from this page.

New Features in Business Operations Center 12.0 Patch Set 1

Business Operations Center 12.0 Patch Set 1 includes the following enhancements:

- [Business Operations Center is Now Certified with Jersey Bundle 2.27 \(Patch Set 1\)](#)
- [Business Operations Center is Now Certified with Jackson 2.9.6 \(Patch Set 1\)](#)
- [Business Operations Center is Now Certified with Log4j2 2.11.1 \(Patch Set 1\)](#)
- [Business Operations Center Is Now Certified with Mozilla Firefox 58.0 \(Patch Set 1\)](#)
- [Business Operations Center is Now Certified with Oracle Identity and Access Management 12c Release 2 \(Patch Set 1\)](#)
- [Business Operations Center is Now Certified with Quartz Scheduler 2.3.0 \(Patch Set 1\)](#)

Business Operations Center is Now Certified with Jersey Bundle 2.27 (Patch Set 1)

Currently, Business Operations Center 12.0 is certified with Jersey Bundle 2.25.1. With this patch, Business Operations Center 12.0 is certified with Jersey Bundle 2.27.

For more information, see *Business Operations Center Installation Guide*.

Business Operations Center is Now Certified with Jackson 2.9.6 (Patch Set 1)

Currently, Business Operations Center 12.0 is certified with Jackson version 2.8.9. With this patch, Business Operations Center 12.0 is certified with Jackson version 2.9.6.

For more information, see *Business Operations Center Installation Guide*.

Business Operations Center is Now Certified with Log4j2 2.11.1 (Patch Set 1)

Currently, Business Operations Center 12.0 is certified with Log4j2 2.8.2. With this patch, Business Operations Center 12.0 is certified with Log4j2 2.11.1.

For more information, see *Business Operations Center Installation Guide*.

Business Operations Center Is Now Certified with Mozilla Firefox 58.0 (Patch Set 1)

Currently, Business Operations Center 12.0 is certified with Mozilla Firefox 54.0.1. With this patch, Business Operations Center 12.0 is certified with Mozilla Firefox 58.0.

For more information, see *Business Operations Center Installation Guide*.

Business Operations Center is Now Certified with Oracle Identity and Access Management 12c Release 2 (Patch Set 1)

Currently, Business Operations Center 12.0 is certified with Oracle Identity and Access Management (OIAM) 11g Release 2 (11.1.2.3.0). With this patch, Business Operations Center 12.0 is certified with OIAM 12c R2 (12.2.1.3.0).

For more information, see *Business Operations Center Installation Guide*.

Business Operations Center is Now Certified with Quartz Scheduler 2.3.0 (Patch Set 1)

Currently, Business Operations Center 12.0 is certified with Quartz Scheduler 2.2.3. With this patch, Business Operations Center 12.0 is certified with Quartz Scheduler 2.3.0.

For more information, see *Business Operations Center Installation Guide*.

7

Opcode Changes

The Oracle Communications Billing and Revenue Management (BRM) 12.0 Patch Sets include new and modified opcodes.

Topics in this document:

- [New Standard Opcodes](#)
- [New Policy Opcodes](#)
- [Changed Standard Opcodes](#)
- [Changed Policy Opcodes](#)

For more information about the opcodes, see "Opcode Descriptions" in *BRM Opcode Guide*.

New Standard Opcodes

[Table 7-1](#) lists the FM standard opcodes that have been introduced in the BRM 12.0 patch set releases.

Table 7-1 New FM Standard Opcodes

New Standard Opcode	Description
PCM_OP_CUST_CONVERT_WHOLE_SALE_HIERARCHY	(Patch Set 8) Converts existing individual account hierarchies to use wholesale billing.
PCM_OP_LOAN_PULLBACK_LOAN	(Patch Set 8) Pulls back the loan amount from the specified account or service.
PCM_OP_LOAN_RECOVER_LOAN	(Patch Set 8) Applies and recovers a late payment fee if the payment is late.
PCM_OP_NOTIFICATION_CALC_DELIVERY_TIME	(Patch Set 8) Calculates and returns an event's delivery time based on the preferred time, scheduled time, silent period, and silent day settings.
PCM_OP_NOTIFICATION_GET_LAST_NOTIFY_TSTAMP	(Patch Set 8) Retrieves the PIN_FLD_LAST_NOTIFICATION_T and PIN_FLD_LAST_NOTIFICATION_OFFSET values for a given set of timestamps.
PCM_OP_NOTIFICATION_PROCESS_NOTIFICATION	(Patch Set 8) A wrapper opcode for handling in-advance and post-expiration notifications, which calls the appropriate notification opcode.
PCM_OP_NOTIFICATION_VERIFY_PUBLISH_REQUIRED	(Patch Set 8) Verifies whether a notification should be published for this business event.
PCM_OP_BILL_GROUP_REMOVE_PAYING_MEMBERS	(Patch Set 7) Removes all paying accounts under a specified account hierarchy.
PCM_OP_DEPOSIT_HIERARCHY_DEPOSIT_TRANSFER	(Patch Set 7) Transfers deposits from child bills to parent bills during a hierarchy transfer.
PCM_OP_ACT_REQUEST_CREATE	(Patch Set 6) Creates a /request object. Records details about a failed operation.
PCM_OP_ACT_REQUEST_RETRIEVE	(Patch Set 6) Retrieves /request objects.

Table 7-1 (Cont.) New FM Standard Opcodes

New Standard Opcode	Description
PCM_OP_BILL_GET_LIMIT	(Patch Set 6) Retrieves temporary and permanent credit limits for both currency and noncurrency resources.
PCM_OP_INSTALLMENT_GET_BILL_ITEMS	(Patch Set 6) Retrieves bills and items with pending due amounts.
PCM_OP_INSTALLMENT_GET_INSTALLMENT_DETAILS	(Patch Set 6) Retrieves all charge details associated with a specified installment schedule and account.
PCM_OP_INSTALLMENT_UPDATE_INSTALLMENT	(Patch Set 6) Modifies the amount due for remaining installments which are in an open state.
PCM_OP_BAL_TRANSFER_BALANCE	(Patch Set 5) Transfers balances and noncurrency resources from one subscriber to another.
PCM_OP_INSTALLMENT_APPLY_CHARGE	(Patch Set 5) Applies the installment charge or amount for the customer on the effective date.
PCM_OP_INSTALLMENT_CANCEL_INSTALLMENT	(Patch Set 5) Sums any remaining amounts, charges the customer, and cancels the remaining installments.
PCM_OP_INSTALLMENT_CREATE_INSTALLMENT	(Patch Set 5) Creates installments for a specific customer.
PCM_OP_INSTALLMENT_CREATE_SCHEDULE_SPEC	(Patch Set 5) Creates installment schedule specifications.
PCM_OP_INSTALLMENT_GET_INSTALLMENTS	(Patch Set 5) Retrieves information about all installments associated with an account and service.
PCM_OP_INSTALLMENT_GET_SCHEDULE_SPEC	(Patch Set 5) Retrieves information about the specified installment schedule specification.
PCM_OP_INSTALLMENT_MODIFY_SCHEDULE_SPEC	(Patch Set 5) Modifies an existing installment schedule specification.
PCM_OP_INSTALLMENT_PROPOSAL	(Patch Set 5) Prepares the installment proposal.
PCM_OP_INSTALLMENT_UPDATE_INSTALLMENT_STATUS	(Patch Set 5) Modifies an installment's status.
PCM_OP_INSTALLMENT_UPDATE_INSTALLMENT	(Patch Set 5) Updates open installments.
PCM_OP_LOAN_APPLY_LOAN	(Patch Set 5) Grants a loan to a customer.
PCM_OP_LOAN_ELIGIBILITY	(Patch Set 5) Checks whether a customer is eligible for a loan or not.
PCM_OP_LOAN_GET_LOAN	(Patch Set 5) Retrieves loan information.
PCM_OP_LOAN_RESET_CYCLE	(Patch Set 5) Resets the loan cycle so customers can take out more loans.
PCM_OP_NOTIFICATION_CREATE_SPECIFICATION	(Patch Set 5) Creates a notification specification.
PCM_OP_NOTIFICATION_GET_SPECIFICATION	(Patch Set 5) Retrieves notification specifications.
PCM_OP_NOTIFICATION_MODIFY_SPECIFICATION	(Patch Set 5) Modifies an existing notification specifications.
PCM_OP_PRICE_CREATE_SELECTOR	(Patch Set 5) Creates a tax exemption selector.
PCM_OP_PRICE_GET_SELECTOR	(Patch Set 5) Retrieves tax exemption selectors.
PCM_OP_RATE_EVALUATE_SELECTOR	(Patch Set 5) Retrieves tax exemption data based on specified attributes.

Table 7-1 (Cont.) New FM Standard Opcodes

New Standard Opcode	Description
PCM_OP_SUBSCRIPTION_CRUD_OFFER_OVERRIDE	(Patch Set 5) Creates, reads, updates, or deletes an /offering_override_values object based on input flag.
PCM_OP_SUBSCRIPTION_HANDLE_PROMO_EVENT	(Patch Set 5) Retrieves promotion events and applies promotions.
PCM_OP_CONTRACT_CREATE_CONTRACT	(Patch Set 4) Creates a contract for a customer. It creates a /subscriber_contract object and generates a contract creation event. This opcode is called by Billing Care.
PCM_OP_CONTRACT_MODIFY_CONTRACT	(Patch Set 4) Modifies a contract owned by a customer. This opcode validates the contract input, updates the existing contract, and generates a contract modification event. This opcode is called by Billing Care.
PCM_OP_CONTRACT_CANCEL_CONTRACT	(Patch Set 4) Cancels a customer's contract. This opcode cancels an existing contract, cancels all pricing object associated objects, generates a contract cancellation event, and applies an early termination fee, if required by the terms. This opcode is called by the pin_contracts utility.
PCM_OP_CONTRACT_RENEW_CONTRACT	(Patch Set 4) Renews customer contracts that expire today. This opcode is called by the pin_contracts utility.

New Policy Opcodes

Table 7-2 lists the FM policy opcodes that have been introduced in the BRM 12.0 patch set releases.

Table 7-2 New FM Policy Opcodes

New Policy Opcode	Description
PCM_OP_SUBSCRIPTION_POL_PURCHASE_PRODUCT	(Patch Set 8) Enables you to customize purchased products.
PCM_OP_PYMT_POL_EXEC_COLLECTIONS_ACTION	(Patch Set 7) Checks whether a payment is for closing a promise-to-pay agreement.
PCM_OP_SUBSCRIPTION_POL_ORDERED_BALGRP_PRIORITY	(Patch Set 7) Enables you to reorder an ordered balance group for automatic sharing groups based on business needs. By default, this opcode does nothing.
PCM_OP_ACT_POL_CUST_CREATE	(Patch Set 6) Enables you to customize the information stored in a /request object.
PCM_OP_BAL_POL_VAL_BAL_TRANSFER	(Patch Set 6) Customizes business validation logic to allow or disallow balance transfers.
PCM_OP_BILL_POL_POST_SET_LIMIT_AND_CR	(Patch Set 6) Customizes the notifications for temporary credit limits.
PCM_OP_BILL_POL_PRE_SET_LIMIT_AND_CR	(Patch Set 6) Customizes the validity dates for a temporary credit limit.
PCM_OP_LOAN_POL_PRE_APPLY_LOAN	(Patch Set 6) Enables you to customize the loan grant process.
PCM_OP_LOAN_POL_PRE_RECOVER_LOAN	(Patch Set 6) Enables you to customize the loan recovery process.
PCM_OP_LOAN_POL_RESET_CYCLE	(Patch Set 6) Enables you to reset fields in the /profile/loan object. Was previously a standard opcode, PCM_OP_LOAN_RESET_CYCLE.

Table 7-2 (Cont.) New FM Policy Opcodes

New Policy Opcode	Description
PCM_OP_SUBSCRIPTION_POL_CANCEL_ON_FAILURE	(Patch Set 6) Enables you to cancel ownership of a deal or product for a specified account or service.
PCM_OP_BAL_POL_SET_SUB_BALANCES	(Patch Set 5) Customizes balances before they are debited from the source account, service, or balance group, or after they are credited to the destination account, service, or balance group.
PCM_OP_INSTALLMENT_POL_GET_SCHEDULE_SPEC	(Patch Set 5) Customizes how installment schedule specifications are retrieved.
PCM_OP_INSTALLMENT_POL_PREP_INSTALLMENT	(Patch Set 5) Prepares installment fields.
PCM_OP_INSTALLMENT_POL_PREP_SCHEDULE_SPEC	(Patch Set 5) Prepares installment schedule specification creation fields.
PCM_OP_INSTALLMENT_POL_VALIDATE_INSTALLMENT	(Patch Set 5) Validates installment fields.
PCM_OP_INSTALLMENT_POL_VALIDATE_SCHEDULE_SPEC	(Patch Set 5) Validates the installment schedule specification information.
PCM_OP_LOAN_POL_ELIGIBILITY	(Patch Set 5) Customizes the loan eligibility check. By default, this opcode does nothing.
PCM_OP_NOTIFICATION_POL_GET_SPECIFICATION	(Patch Set 5) Validates and implements any customizations while retrieving notification specifications.
PCM_OP_NOTIFICATION_POL_PREP_SPECIFICATION	(Patch Set 5) Prepares fields for the notification specification and implements any customizations while creating notification specifications.
PCM_OP_NOTIFICATION_POL_VALIDATE_SPECIFICATION	(Patch Set 5) Validates and implements any customizations while creating notification specifications.
PCM_OP_PYMT_POL_TOPUP_SET_VALIDITY	(Patch Set 5) Customizes the validity dates for noncurrency resources during the top-up process. By default, this opcode does nothing.
PCM_OP_PYMT_POL_POST_TOPUP	(Patch Set 5) Customizes how to calculate the due date for the next recurring top-up. By default, this opcode does nothing.
PCM_OP_SUBSCRIPTION_POL_CREATE_OFFER_OVERRIDE	(Patch Set 5) Specifies when and how to override charges and discounts during specified time periods. It creates, reads, updates, or deletes a specified /offering_override_value object based on the value of an input flag.
PCM_OP_SUBSCRIPTION_POL_POST_PROMO_EVENT	(Patch Set 5) Customizes promotion events after they are rated.
PCM_OP_SUBSCRIPTION_POL_PREP_PROMO_EVENT	(Patch Set 5) Customizes promotion events before they are rated.
PCM_OP_CONTRACT_POL_POST_CREATE_CONTRACT	(Patch Set 4) Customizes how contracts are created. By default, the opcode does nothing.
PCM_OP_CONTRACT_POL_PREP_CONTRACT	(Patch Set 4) Customize how contracts are created. By default, the opcode does nothing.
PCM_OP_CONTRACT_POL_VALIDATE_CONTRACT	(Patch Set 4) Validates a customer's contract for valid start and end dates, pricing objects, account objects, and contract objects (during modification). The opcode is triggered during contract creation, modification, and renewal.
PCM_OP_PUBLISH_POL_PREP_EVENT	(Patch Set 4) Modifies business event payloads before the CM sends them to the Data Manager. By default, the opcode does nothing.
PCM_OP_SUBSCRIPTION_POL_VALIDATE_OFFERING	(Patch Set 4) Validates extended attributes defined on pricing components. By default, the opcode does nothing.

Table 7-2 (Cont.) New FM Policy Opcodes

New Policy Opcode	Description
PCM_OP_SUBSCRIPTION_POL_P R E_CYCLE_DISCOUNT	(Patch Set 1) Determines the balance elements (resources) of the nonpaying child bill units for rolling them up to the paying parent bill unit during wholesale billing. This opcode is called before applying the cycle discounts.

Changed Standard Opcodes

Table 7-3 lists the FM standard opcodes that have been modified in the BRM 12.0 patch set releases.

Table 7-3 Changed Standard Opcodes

Changed Standard Opcode	Description
PCM_OP_ACT_REQUEST_CREATE	(Patch Set 8) Modified to include the following new input and output flist field: <ul style="list-style-type: none"> PIN_FLD_NOTIFICATION_SPEC array passes additional details with the request record in the notification event. This array and any field that it contains will not be persisted with the request.
PCM_OP_AR_BILL_DISPUTE	(Patch Set 8) Modified to include the following new output flist field under the PIN_FLD_RESULTS array: <ul style="list-style-type: none"> PIN_FLD_EVENT_OBJ field specifies the POID of the bill dispute event object that was created for a bill dispute.
PCM_OP_AR_GET_ACTION_ITEMS	(Patch Set 8) Modified to include the following new output flist field under the PIN_FLD_RESULTS array: <ul style="list-style-type: none"> PIN_FLD_ITEM_POID_LIST field holds the list of items in a bill.
PCM_OP_AR_GET_DISPUTES	(Patch Set 8) Modified to include the following new output flist field under the PIN_FLD_RESULTS array: <ul style="list-style-type: none"> PIN_FLD_ITEM_POID_LIST field holds the list of items in a bill.
PCM_OP_CUST_CREATE_BAL_GRP	(Patch Set 8) Modified to include the following new input flist field: <ul style="list-style-type: none"> PIN_FLD_PROGRAM_NAME field specifies the name of the program.
PCM_OP_DEPOSIT_GET_SPECIFIC ATION	(Patch Set 8) The PIN_FLD_STATUS_STR field was renamed to PIN_FLD_DRAFT_FLAG. The PIN_FLD_DRAFT_FLAG field can be set to draft (1) or non-draft (0).
PCM_OP_LOAN_GET_LOAN	(Patch Set 8) Modified the order of output flist fields. The following fields now appear directly under the PIN_FLD_RESULTS output flist array: <ul style="list-style-type: none"> PIN_FLD_CREDIT_LIMIT PIN_FLD_AVAILABLE_LOAN_LIMIT In addition, the PIN_FLD_LOAN_AMOUNT field was renamed to PIN_FLD_AMOUNT.

Table 7-3 (Cont.) Changed Standard Opcodes

Changed Standard Opcode	Description
PCM_OP_NOTIFICATION_CREATE_SPECIFICATION PCM_OP_NOTIFICATION_GET_SPECIFICATION PCM_OP_NOTIFICATION_MODIFY_SPECIFICATION	(Patch Set 8) Modified to include the following new flist fields: <ul style="list-style-type: none"> PIN_FLD_AGGREGATE_MODE field specifies whether to send separate notifications for each event (0) or to combine notifications at the search level (1). PIN_FLD_NOTIFY_OPT_TYPE field specifies to opt-out customers from receiving notification messages by default (1), to opt-in customers for receiving notification messages by default (2), or to enforce opt-in preferences for customers (3).
PCM_OP_PRICE_GET_PRICE_LIST PCM_OP_PRICE_SET_PRICE_LIST	(Patch Set 8) Modified to include the following new output flist fields: <ul style="list-style-type: none"> PIN_FLD_OFFER_VALIDITY_ROUNDING field specifies the validity rounding at the product level: not set (0), on (1), or off (2). PIN_FLD_SCALE_ROUNDING field specifies whether product scale rounding is disabled (1) or enabled (0).
PCM_OP_DEPOSIT_GET_SPECIFICATION	(Patch Set 7) Modified to include the following new input and output flist field: <ul style="list-style-type: none"> PIN_FLD_CONTEXT_INFO substruct passes information from external clients. Modified to replace the PIN_FLD_STATUS_STR input flist field with the following input flist field, making the input flist more like the output flist: <ul style="list-style-type: none"> PIN_FLD_DRAFT_FLAG field indicates the status of the deposit specification where 1 indicates draft and 0 indicates non-draft.
PCM_OP_BILL_SET_LIMIT_AND_CR	(Patch Set 7) Modified to support dynamic credit floors. It includes the following new input flist fields: <ul style="list-style-type: none"> PIN_FLD_DYNAMIC_CREDIT_FLOOR field specifies the dynamic credit floor setting: use the credit floor or the sum of original balances for the credit thresholds check. PIN_FLD_OVERRIDE_CREDIT_LIMIT field specifies whether to consider the credit limit: the credit limit will be decided based on rating modes (0) or override the credit limit (0x01). PIN_FLD_ROLL_UP_CREDIT_LIMIT field specifies whether to roll up the credit limit in a Payment Responsible Real-Time Credit Enforcement (PR_RTCE) balance monitoring group.
PCM_OP_COLLECTIONS_GET_BILL_INFOS	(Patch Set 7) Modified to include the following under the PIN_FLD_SCENARIO_INFO substruct in the input flist: <ul style="list-style-type: none"> PIN_FLD_PROMISE_TO_PAY array contains details about a promise-to-pay agreement.
PCM_OP_COLLECTIONS_INVOKE_PROMISE_TO_PAY	(Patch Set 7) Modified to include the following additions to the flist: <ul style="list-style-type: none"> PIN_FLD_SPEC_NAME input flist field specifies the name of the promise-to-pay specification. PIN_FLD_MILESTONES input and output flist array contains unequal installment payment and unequal interval details.
PCM_OP_CUSTCARE_MOVE_ACCT	(Patch Set 7) Modified to include the following new input flist fields under the PIN_FLD_GROUP_MOVE_MEMBER array: <ul style="list-style-type: none"> PIN_FLD_ROLL_UP_CREDIT_LIMIT field specifies whether to roll up the credit limit in a Payment Responsible Real-Time Credit Enforcement (PR_RTCE) balance monitoring group. PIN_FLD_TYPE_STR field specifies the group sharing monitor type. Payment Responsible Real-Time Credit Enforcement group will be created automatically if the type is PR_RTCE and parent account doesn't have group sharing monitor of type PR_RTCE.

Table 7-3 (Cont.) Changed Standard Opcodes

Changed Standard Opcode	Description
PCM_OP_CUST_COMMIT_CUSTOM ER	<p>(Patch Set 7) Modified to include the following new input and output flist fields:</p> <ul style="list-style-type: none"> • PIN_FLD_OVERRIDE_CREDIT_LIMIT field specifies whether to consider the credit limit: the credit limit will be decided based on rating modes (0) or override the credit limit (0x01). • PIN_FLD_DYNAMIC_CREDIT_FLOOR specifies the dynamic credit floor setting, whether to use credit_floor or sum of original balances for credit thresholds check • PIN_FLD_ROLL_UP_CREDIT_LIMIT field specifies whether to roll up the credit limit in a Payment Responsible Real-Time Credit Enforcement (PR_RTCE) balance monitoring group. • PIN_FLD_VALIDITY_ALIGN_MODE field specifies the validity align mode: align with the base product (7), align with any base product earliest (8), align with any base product latest (9), align with any product earliest (10), or align with any product latest (11). • PIN_FLD_BASE_PRODUCT_OBJ specifies the base product POID to which an add-on product's validity must align. • PIN_FLD_TYPE_STR specifies the group sharing monitor type. The Payment Responsible Real-Time Credit Enforcement group will be created automatically if the monitor type is PR_RTCE. • PIN_FLD_FLAGS specifies the type of proration to apply at the product level: default system configuration (0x00), 30-day proration (0x01), or the number of days in a month (0x02).
PCM_OP_CUST_CREATE_ACCT PCM_OP_CUST_CREATE_BAL_GRP PCM_OP_CUST_CREATE_CUSTOM ER	<p>(Patch Set 7) Modified to include the following new input flist fields under the PIN_FLD_LIMIT array:</p> <ul style="list-style-type: none"> • PIN_FLD_OVERRIDE_CREDIT_LIMIT field specifies whether to consider the credit limit: the credit limit will be decided based on rating modes (0) or override the credit limit (0x01). • PIN_FLD_DYNAMIC_CREDIT_FLOOR specifies the dynamic credit floor setting, whether to use credit_floor or sum of original balances for credit thresholds check

Table 7-3 (Cont.) Changed Standard Opcodes

Changed Standard Opcode	Description
PCM_OP_CUST_MODIFY_CUSTOM ER	(Patch Set 7) Modified to include the following new input and output flist fields: <ul style="list-style-type: none"> • PIN_FLD_OVERRIDE_CREDIT_LIMIT field specifies whether to consider the credit limit: the credit limit will be decided based on rating modes (0) or override the credit limit (0x01). • PIN_FLD_DYNAMIC_CREDIT_FLOOR specifies the dynamic credit floor setting, whether to use credit_floor or sum of original balances for credit thresholds check • PIN_FLD_ROLL_UP_CREDIT_LIMIT field specifies whether to roll up the credit limit in a Payment Responsible Real-Time Credit Enforcement (PR_RTCE) balance monitoring group. • PIN_FLD_VALIDITY_ALIGN_MODE field specifies the validity align mode: align with the base product (7), align with any base product earliest (8), align with any base product latest (9), align with any product earliest (10), or align with any product latest (11). • PIN_FLD_BASE_PRODUCT_OBJ specifies the base product POID to which an add-on product's validity must align. • PIN_FLD_TYPE_STR specifies the group sharing monitor type. The Payment Responsible Real-Time Credit Enforcement group will be created automatically if the monitor type is PR_RTCE. • PIN_FLD_FLAGS specifies the type of proration to apply at the product level: default system configuration (0x00), 30-day proration (0x01), or the number of days in a month (0x02).
PCM_OP_CUST_SET_BAL_GRP	(Patch Set 7) Modified to include the following input flist fields under the PIN_FLD_LIMIT array: <ul style="list-style-type: none"> • PIN_FLD_TYPE_STR specifies the group sharing monitor type. The Payment Responsible Real-Time Credit Enforcement group will be created automatically if the monitor type is PR_RTCE. • PIN_FLD_OVERRIDE_CREDIT_LIMIT field specifies whether to consider the credit limit: the credit limit will be decided based on rating modes (0) or override the credit limit (0x01). • PIN_FLD_DYNAMIC_CREDIT_FLOOR specifies the dynamic credit floor setting, whether to use credit_floor or sum of original balances for credit thresholds check
PCM_OP_CUST_UPDATE_CUSTOM ER	(Patch Set 7) Modified to include the following new input and output flist fields: <ul style="list-style-type: none"> • PIN_FLD_OVERRIDE_CREDIT_LIMIT field specifies whether to consider the credit limit: the credit limit will be decided based on rating modes (0) or override the credit limit (0x01). • PIN_FLD_DYNAMIC_CREDIT_FLOOR specifies the dynamic credit floor setting, whether to use credit_floor or sum of original balances for credit thresholds check • PIN_FLD_ROLL_UP_CREDIT_LIMIT field specifies whether to roll up the credit limit in a Payment Responsible Real-Time Credit Enforcement (PR_RTCE) balance monitoring group.

Table 7-3 (Cont.) Changed Standard Opcodes

Changed Standard Opcode	Description
PCM_OP_DEPOSIT_ADD_INTEREST PCM_OP_DEPOSIT_COLLECT_PAYMENT PCM_OP_DEPOSIT_CREATE_SPECIFICATION_PROFILE PCM_OP_DEPOSIT_CREATE_SPECIFICATION PCM_OP_DEPOSIT_DELETE_DRAFT PCM_OP_DEPOSIT_GET_REFUND_REQUEST PCM_OP_DEPOSIT_GET_SPECIFICATION_PROFILE PCM_OP_DEPOSIT_GET_TRANSACTIONS PCM_OP_DEPOSIT_HIERARCHY_DEPOSIT_TRANSFER PCM_OP_DEPOSIT_MODIFY_SPECIFICATION_PROFILE PCM_OP_DEPOSIT_MODIFY_SPECIFICATION PCM_OP_DEPOSIT_REFUND_REQUEST PCM_OP_DEPOSIT_RELEASE_DEPOSIT PCM_OP_DEPOSIT_REVERSE_DEPOSIT PCM_OP_DEPOSIT_TRANSFER_DEPOSIT PCM_OP_DEPOSIT_TRIGGER_DEPOSIT PCM_OP_DEPOSIT_UPDATE_DEPOSIT_RECEIVED PCM_OP_DEPOSIT_UPDATE_DEPOSIT PCM_OP_DEPOSIT_UPDATE_REFUND_REQUEST	(Patch Set 7) Modified to include the following new input and output flist field: <ul style="list-style-type: none"> • PIN_FLD_CONTEXT_INFO substruct passes information from external clients.
PCM_OP_DEPOSIT_GET_DEPOSITS	(Patch Set 7) Modified to include the following new input flist field: <ul style="list-style-type: none"> • PIN_FLD_END_DETAILS field is an encoded field containing the mode, unit, and offset. Mode is always relative (4) occupying 8 bits starting from the least significant bit. Unit can be months (0), days (1), weeks (2), or years (3), occupying 4 bits after mode. Offset is the value for the specified unit, occupying the 20 bits after unit. Both the input and output flists have been modified to include the following new field: <ul style="list-style-type: none"> • PIN_FLD_CONTEXT_INFO substruct passes information from external clients.

Table 7-3 (Cont.) Changed Standard Opcodes

Changed Standard Opcode	Description
PCM_OP_DEPOSIT_PURCHASE_DEPOSIT	<p>(Patch Set 7) Modified to include the following new input flist field:</p> <ul style="list-style-type: none"> PIN_FLD_PARENT field specifies the POID of the parent / purchased deposit object. <p>Both the input and output flists have been modified to include the following new field:</p> <ul style="list-style-type: none"> PIN_FLD_CONTEXT_INFO substruct passes information from external clients.
PCM_OP_INSTALLMENT_CREATE_SCHEDULE_SPEC	<p>(Patch Set 7) The input flist contains the following changes:</p> <ul style="list-style-type: none"> The PIN_FLD_MAX_INSTALLMENT_UNEQUAL_INTERVALS field was renamed to PIN_FLD_MAX_UNEQUAL_INTERVALS. The PIN_FLD_MAX_NUMBER_UNEQUAL_INSTALLMENT field was renamed to PIN_FLD_MAX_UNEQUAL_INSTALLMENTS.
PCM_OP_MONITOR_SETUP_MEMBERS	<p>(Patch Set 7) Modified to support automatic balance monitoring groups. The input and output flists have not changed.</p>
PCM_OP_NOTIFICATION_MODIFY_SPECIFICATION PCM_OP_NOTIFICATION_CREATE_SPECIFICATION	<p>(Patch Set 7) Under the PIN_FLD_NOTIFICATION_SPEC array, the PIN_FLD_NOTIFICATION_TYPE input flist field supports the following new value: AFTER_DUE_DATE (5).</p>
PCM_OP_PRICE_GET_PRICE_LIST	<p>(Patch Set 7) Modified to support dynamic credit floors. It contains the following new output flist fields:</p> <ul style="list-style-type: none"> PIN_FLD_OVERRIDE_CREDIT_LIMIT field specifies whether to consider the credit limit: the credit limit will be decided based on rating modes (0) or override the credit limit (0x01). PIN_FLD_DYNAMIC_CREDIT_FLOOR field specifies the dynamic credit floor setting: use the credit floor or the sum of original balances for the credit thresholds check. PIN_FLD_BASE_PRODUCT_NAME field specifies the name of the base product. PIN_FLD_BASE_PRODUCT_CODE fields specifies the code of the base product. PIN_FLD_VALIDITY_ALIGN_MODE field specifies the validity align mode: align with the base product (7), align with any base product earliest (8), align with any base product latest (9), align with any product earliest (10), or align with any product latest (11).
PCM_OP_PRICE_SET_PRICE_LIST	<p>(Patch Set 7) Modified to support dynamic credit floors. It contains the following new input flist fields:</p> <ul style="list-style-type: none"> PIN_FLD_OVERRIDE_CREDIT_LIMIT field specifies whether to consider the credit limit: the credit limit will be decided based on rating modes (0) or override the credit limit (0x01). PIN_FLD_DYNAMIC_CREDIT_FLOOR field specifies the dynamic credit floor setting: use the credit floor or the sum of original balances for the credit thresholds check. PIN_FLD_BASE_PRODUCT_NAME field specifies the name of the base product. PIN_FLD_BASE_PRODUCT_CODE fields specifies the code of the base product. PIN_FLD_VALIDITY_ALIGN_MODE field specifies the validity align mode: align with the base product (7), align with any base product earliest (8), align with any base product latest (9), align with any product earliest (10), or align with any product latest (11).

Table 7-3 (Cont.) Changed Standard Opcodes

Changed Standard Opcode	Description
PCM_OP_SUBSCRIPTION_PURCHASE_DEAL	(Patch Set 7) Modified to include the following new input flist fields: <ul style="list-style-type: none"> PIN_FLD_FLAGS specifies the type of proration to apply at the product level: default system configuration (0x00), 30-day proration (0x01), or the number of days in a month (0x02). PIN_FLD_BASE_PRODUCT_OBJ specifies the base product POID to which an add-on product's validity must align. PIN_FLD_VALIDITY_ALIGN_MODE specifies the validity align mode: align with the base product (7), align with any base product earliest (8), align with any base product latest (9), align with any product earliest (10), or align with any product latest (11).
PCM_OP_SUBSCRIPTION_PURCHASE_PRODUCT	(Patch Set 7) Modified to support automatic product sharing groups. It contains the following new input flist field under the PIN_FLD_PRODUCTS array: <ul style="list-style-type: none"> PIN_FLD_BASE_PRODUCT_OBJ field specifies the base product POID to which an add-on product's validity must align.
PCM_OP_SUBSCRIPTION_SHARING_GROUP_CREATE	(Patch Set 7) Modified to support product sharing groups. It contains the following new input flist fields: <ul style="list-style-type: none"> PIN_FLD_ROLL_UP_CREDIT_LIMIT field specifies whether to roll up the credit limit in a Payment Responsible Real-Time Credit Enforcement (PR_RTCE) balance monitoring group. PIN_FLD_PRODUCTS array contains the list of products in the product sharing group
PCM_OP_SUBSCRIPTION_SHARING_GROUP_DELETE PCM_OP_SUBSCRIPTION_SHARING_GROUP_MODIFY	(Patch Set 7) Modified to delete or change the products associated with a product sharing group. They contain the following new input flist field: <ul style="list-style-type: none"> PIN_FLD_PRODUCTS array contains the list of product instances to delete or change.
PCM_OP_SUBSCRIPTION_SHARING_GROUP_SET_PARENT	(Patch Set 7) Modified to support the change of parent for a product sharing group. It contains the following new input flist field: <ul style="list-style-type: none"> PIN_FLD_PRODUCTS array contains the products of the new parent.
PCM_OP_BILL_SET_LIMIT_AND_CREATE	(Patch Set 6) Modified to create and update temporary credit limits. It contains the following new input and output flist fields: <ul style="list-style-type: none"> PIN_FLD_TEMP_LIMIT array provides details about a temporary credit limit.
PCM_OP_CUST_COMMIT_CUSTOMER	(Patch Set 6) Modified to support PINless direct debit payments and temporary credit limits. The opcode flist contains these new fields: <ul style="list-style-type: none"> PIN_FLD_TEMP_LIMIT array provides details about a temporary credit limit. Also, the following field was removed from the opcode's flist: PIN_FLD_TYPE_STR.
PCM_OP_CUST_CREATE_ACCOUNT	(Patch Set 6) Modified to support PINless direct debit payments. The following field was removed from the opcode's flist: PIN_FLD_TYPE_STR.
PCM_OP_CUST_CREATE_CUSTOMER	(Patch Set 6) Modified to support PINless direct debit payments. The following field was removed from the opcode's flist: PIN_FLD_TYPE_STR.
PCM_OP_CUST_MODIFY_CUSTOMER	(Patch Set 6) Modified to support PINless direct debit payments. The following field was removed from the opcode's flist: PIN_FLD_TYPE_STR.

Table 7-3 (Cont.) Changed Standard Opcodes

Changed Standard Opcode	Description
PCM_OP_CUST_PREP_CUSTOMER	(Patch Set 6) Modified to support PINless direct debit payments. The following field was removed from the opcode's flist: PIN_FLD_TYPE_STR.
PCM_OP_CUST_SET_BAL_GRP	(Patch Set 6) Modified to support temporary credit limits. The opcode flist contains these new fields: <ul style="list-style-type: none"> PIN_FLD_TEMP_LIMIT array provides details about a temporary credit limit.
PCM_OP_CUST_UPDATE_CUSTOMER	(Patch Set 6) Modified to support PINless direct debit payments. The following field was removed from the opcode's flist: PIN_FLD_TYPE_STR.
PCM_OP_CUST_VALIDATE_CUSTOMER	(Patch Set 6) Modified to support PINless direct debit payments. The following field was removed from the opcode's flist: PIN_FLD_TYPE_STR.
PCM_OP_LOAN_APPLY_LOAN	(Patch Set 6) The input flist contains these new fields: <ul style="list-style-type: none"> PIN_FLD_CHANNEL indicates the channel requesting the loan PIN_FLD_TYPE indicates the type of loan The output flist contains these new fields: <ul style="list-style-type: none"> PIN_FLD_AVAILABLE_LOAN_BALANCE: The sum of the previous remaining loan balance and the current loan balance PIN_FLD_CURRENT_BAL: The main balance PIN_FLD_LOAN_FEE: The loan fee PIN_FLD_TAX: The tax amount PIN_FLD_CREDIT_LIMIT: The total loan credit limit PIN_FLD_AVAILABLE_LOAN_LIMIT: The available loan credit limit after applying the loan PIN_FLD_BALANCES.PIN_FLD_GRANTED_BAL: The noncurrency resource granted as part of loan.
PCM_OP_LOAN_ELIGIBILITY	(Patch Set 6) The opcode checks the values of additional fields in the <i>/profile/loan</i> object and has new reason codes.
PCM_OP_LOAN_GET_LOAN	(Patch Set 6) The input flist includes the PIN_FLD_FLAGS field, to indicate whether to return only active loans (0) or all loans (1). The output flist contains additional fields. See <i>BRM Opcode Flist Reference</i> for full details.
PCM_OP_CUST_CREATE_PROFILE PCM_OP_CUST_MODIFY_PROFILE	(Patch Set 6) The input flist contains additional fields for loan profiles: <ul style="list-style-type: none"> PIN_FLD_OPT_LOAN: Whether the customer opts in or out of loans PIN_FLD_MAX_ACTIVE_LOANS: The maximum number of active loans a customer can have PIN_FLD_LOAN_THRESHOLDS_FIXED: The fixed threshold for offering a loan to a customer. Overrides the threshold configured on the package. PIN_FLD_LOAN_THRESHOLDS: The percent threshold for offering a loan to a customer. Overrides the threshold configured on the package. PIN_FLD_LIMIT: An array for specifying a loan credit limit that overrides the limit specified in the <i>/config/loan</i> object. See <i>BRM Opcode Flist Reference</i> for full details.
PCM_OP_BAL_GET_BALANCES	(Patch Set 6) Modified to include the PIN_FLD_SUBTYPE field in the PIN_FLD_SUB_BALANCES array. This field indicates whether the sub-balance is for a loan or not.

Table 7-3 (Cont.) Changed Standard Opcodes

Changed Standard Opcode	Description
PCM_OP_AR_EVENT_ADJUSTMENT	(Patch Set 5) Modified to prevent event adjustments for loan events.
PCM_OP_AR_EVENT_DISPUTE	(Patch Set 5) Modified to prevent event disputes for loan events.
PCM_OP_AR_GET_ACCT_BILLS	(Patch Set 5) Modified to support bill states. The output flist contains the new PIN_FLD_STATE field.
PCM_OP_AR_GET_BILLS	(Patch Set 5) Modified to support bill states. The output flist contains the new PIN_FLD_STATE field.
PCM_OP_AR_ITEM_ADJUSTMENT	(Patch Set 5) Modified to prevent item adjustments for loan items.
PCM_OP_AR_ITEM_DISPUTE	(Patch Set 5) Modified to prevent item disputes for loan items.
PCM_OP_BILL_ITEM_TRANSFER	(Patch Set 5) Modified to support bill states. The opcode updates the new PIN_FLD_STATE field.
PCM_OP_BILL_SET_LIMIT_AND_CR	(Patch Set 5) Modified to support: <ul style="list-style-type: none"> Setting the PIN_FLD_DELTA_CREDIT_LIMIT field in the balance group. This credit limit includes the original credit limit, loans, and any fees and taxes related to loans. Setting or modifying the loan thresholds in the PIN_FLD_RESRC_FIXED_THRESHOLD and PIN_FLD_RESRC_PERC_THRESHOLD fields of /config/credit_profile objects.
PCM_OP_BILL_TAX_EVENT	(Patch Set 5) Modified to support real-time location-based taxation. The opcode flist includes this new field under the PIN_FLD_EVENT substruct: <ul style="list-style-type: none"> PIN_FLD_LOCALE specifies the real-time location. The format of the string is "City;State;Zipcode;Country". PIN_FLD_ZONEMAP_TARGET specifies the search target. PIN_FLD_ZONEMAP_NAME specifies the name of the zone map to search.
PCM_OP_CUST_COMMIT_CUSTOMER	(Patch Set 5) Modified to support /profile/loan objects in the input flist. It contains these new input flist fields: <ul style="list-style-type: none"> PIN_FLD_OVERRIDE_PRODUCTS PIN_FLD_OVERRIDE_DISCOUNTS
PCM_OP_CUST_CREATE_PROFILE	(Patch Set 5) Modified to support creating /profile/loan objects.
PCM_OP_CUST_CREATE_TOPUP	(Patch Set 5) Modified to support recurring top-ups.
PCM_OP_CUST_DELETE_PROFILE	(Patch Set 5) Modified to support deleting /profile/loan objects.
PCM_OP_CUST_FIND_PROFILE	(Patch Set 5) Modified to support finding /profile/loan objects.
PCM_OP_CUST_MODIFY_PROFILE	(Patch Set 5) Modified to support modifying /profile/loan objects.
PCM_OP_CUST_MODIFY_TOPUP	(Patch Set 5) Modified to support recurring top-ups.
PCM_OP_CUST_SET_TOPUP	(Patch Set 5) Modified to support recurring top-ups.

Table 7-3 (Cont.) Changed Standard Opcodes

Changed Standard Opcode	Description
PCM_OP_PRICE_GET_PRICE_LIST	<p>(Patch Set 5) Modified to support splitting balances into multiple validity periods. The opcode flist contains these new fields:</p> <ul style="list-style-type: none">• PIN_FLD_CALENDAR_DOM specifies the specific day of the month when the offer was purchased.• PIN_FLD_FIXED_PRICE_TAG specifies the price tag for the fixed amount.• PIN_FLD_PRORATION_FLAG specifies whether proration or no proration is supported during plan or deal transitions.• PIN_FLD_RENEWAL_MODE specifies whether a reactivated product's cycle is aligned with the original purchase date or the reactivation date.• PIN_FLD_SCALED_PRICE_TAG specifies the price tag for the scaled amount.• PIN_FLD_SPLIT_DETAILS is a 32-bit field that specifies the logic for creating multiple validity periods:<ul style="list-style-type: none">– The lower 8 bits specify the expiration mode for the incremental validity period: bucket (5) or total (6)– The second 4 bits specify the units for the incremental validity period: seconds (1), minutes (2), hours (3), days (4), months (5)– The remaining 20 MSB bits specify the amount for the validity period

Table 7-3 (Cont.) Changed Standard Opcodes

Changed Standard Opcode	Description
PCM_OP_PRICE_SET_PRICE_LIST	<p>(Patch Set 5) Modified to support splitting balances into multiple validity periods, changing cycle alignment for package reactivation, grace periods for when the same product is purchased multiple times, and modifying the rate type. The opcode flist contains these new and modified fields:</p> <ul style="list-style-type: none"> • PIN_FLD_GRACE_PERIOD_OFFSET specifies the amount of time in the grace period. The default is 0. • PIN_FLD_GRACE_PERIOD_UNIT specifies the unit of the grace period: 0 (seconds), 1 (minutes), 2 (hours), or 3 (days). The default is 3. • PIN_FLD_RENEWAL_MODE specifies whether a reactivated product's billing cycle is aligned with the original purchase date or the reactivation date. • PIN_FLD_FIXED_PRICE_TAG specifies the price tag for the fixed amount. • PIN_FLD_SCALED_PRICE_TAG specifies the price tag for the scaled amount. • PIN_FLD_SPLIT_DETAILS is a 32-bit field that specifies the logic for creating multiple validity periods: <ul style="list-style-type: none"> – The lower 8 bits specify the expiration mode for the incremental validity period: bucket (5) or total (6) – The second 4 bits specify the units for the incremental validity period: seconds (1), minutes (2), hours (3), days (4), months (5) – The remaining 20 MSB bits specify the amount for the validity period • PIN_FLD_TYPE in the rate array in the product flist is modified to support new values for exceeding credit limits: <ul style="list-style-type: none"> – 742: Use available balance and grant a loan for the rest. – 743: Use available balance and record the rest as an outstanding amount. – 744: Don't use available balance and record the entire amount as an outstanding amount. – 745: Don't exceed the credit limit and fail the subscription. – 746: Skip billing for this cycle.
PCM_OP_PYMT_TOPUP	<p>(Patch Set 5) Modified to support:</p> <ul style="list-style-type: none"> • Top-ups of noncurrency balances, such as minutes or Gigabytes. • Triggering cycle fees when a top-up is applied. • Applying top-ups to loans before other balances. • Real-time location-based taxation. <p>The opcode flist includes this new field:</p> <ul style="list-style-type: none"> • PIN_FLD_LOCALE specifies the real-time location. The format of the string is "City;State;Zipcode;Country". • PIN_FLD_ZONEMAP_TARGET specifies the search target. • PIN_FLD_ZONEMAP_NAME specifies the name of the zone map to search.
PCM_OP_RATE_EVENT	<p>(Patch Set 5) Modified to support splitting balances into multiple validity periods.</p>

Table 7-3 (Cont.) Changed Standard Opcodes

Changed Standard Opcode	Description
PCM_OP_RATE_TAX_EVENT	<p>(Patch Set 5) Modified to support real-time location-based taxation. The opcode flist includes these new fields:</p> <ul style="list-style-type: none"> PIN_FLD_LOCALE specifies the real-time location. The format of the string is "City;State;Zipcode;Country". PIN_FLD_ZONEMAP_TARGET specifies the search target. PIN_FLD_ZONEMAP_NAME specifies the name of the zone map to search.
PCM_OP_SUBSCRIPTION_PURCHASE_DEAL	<p>(Patch Set 5) Modified to support adding grace periods when the same deal is purchased multiple times. The opcode flist includes these new fields under the PIN_FLD_PRODUCTS array:</p> <ul style="list-style-type: none"> PIN_FLD_GRACE_PERIOD_OFFSET specifies the amount of time in the grace period. The default is 0. PIN_FLD_GRACE_PERIOD_UNIT specifies the unit of the grace period: 0 (seconds), 1 (minutes), 2 (hours), or 3 (days). The default is 3. PIN_FLD_OVERRIDE_PRODUCTS specifies the product whose pricing should be overridden. <p>It is also modified to support real-time location-based taxation. The opcode flist includes this new field:</p> <ul style="list-style-type: none"> PIN_FLD_LOCALE specifies the real-time location. The format of the string is "City;State;Zipcode;Country". PIN_FLD_ZONEMAP_TARGET specifies the search target. PIN_FLD_ZONEMAP_NAME specifies the name of the zone map to search.
PCM_OP_SUBSCRIPTION_PURCHASE_DISCOUNT	<p>(Patch Set 5) Modified to support overriding purchased discounts for a limited period of time. The opcode flist includes these new fields under the PIN_FLD_DISCOUNTS array:</p> <ul style="list-style-type: none"> PIN_FLD_OVERRIDE_DISCOUNTS specifies the discount whose pricing should be overridden.
PCM_OP_SUBSCRIPTION_PURCHASE_PRODUCT	<p>(Patch Set 5) Modified to support adding grace periods when the same product is purchased multiple times. The opcode flist includes these new fields under the PIN_FLD_PRODUCTS array:</p> <ul style="list-style-type: none"> PIN_FLD_GRACE_PERIOD_OFFSET specifies the amount of time in the grace period. The default is 0. PIN_FLD_GRACE_PERIOD_UNIT specifies the unit of the grace period: 0 (seconds), 1 (minutes), 2 (hours), or 3 (days). The default is 3. <p>It is also modified to support real-time location-based taxation. The opcode flist includes this new field:</p> <ul style="list-style-type: none"> PIN_FLD_LOCALE specifies the real-time location. The format of the string is "City;State;Zipcode;Country". PIN_FLD_ZONEMAP_TARGET specifies the search target. PIN_FLD_ZONEMAP_NAME specifies the name of the zone map to search.

Table 7-3 (Cont.) Changed Standard Opcodes

Changed Standard Opcode	Description
PCM_OP_SUBSCRIPTION_SET_DISCOUNTINFO	(Patch Set 5) Modified to support dynamic charging. The opcode flist includes these new fields under the PIN_FLD_DISCOUNTS array: <ul style="list-style-type: none"> PIN_FLD_OFFERING_OVERRIDE_VALUES_OBJ specifies the POID of the /offering_override_values object associated with the product. PIN_FLD_OVERRIDE_DISCOUNTS array specifies whether to create, read, update, and delete the /offering_override_values object based on different flag values.
PCM_OP_SUBSCRIPTION_SET_PRODUCTINFO	(Patch Set 5) Modified to support adding grace periods when the same product is purchased multiple times, and to support dynamic charging. The opcode flist includes these new fields under the PIN_FLD_PRODUCTS array: <ul style="list-style-type: none"> PIN_FLD_GRACE_PERIOD_OFFSET specifies the amount of time in the grace period. The default is 0. PIN_FLD_GRACE_PERIOD_UNIT specifies the unit of the grace period: 0 (seconds), 1 (minutes), 2 (hours), or 3 (days). The default is 3. PIN_FLD_OFFERING_OVERRIDE_VALUES_OBJ specifies the POID of the /offering_override_values object associated with the product. PIN_FLD_OVERRIDE_PRODUCTS array specifies whether to create, read, update, and delete the /offering_override_values object based on different flag values.
PCM_OP_SUBSCRIPTION_SET_PRODUCT_STATUS	(Patch Set 5) Modified to support cycle alignment for package reactivation. The input flist contains the new PIN_FLD_RENEWAL_MODE field.
PCM_OP_SET_DD	(Patch Set 5) Modified to support creating storable classes with residency type 9. For classes with this residency type, data dictionary definitions are updated without creating new tables.
PCM_OP_CREATE_OBJ PCM_OP_BULK_CREATE_OBJ PCM_OP_WRITE_FLDS PCM_OP_BULK_WRITE_FLDS PCM_OP_INC_FIELDS PCM_OP_DELETE_OBJ PCM_OP_BULK_DELETE_OBJ PCM_OP_DELETE_FLDS	(Patch Set 5) Modified to prevent insert, update, and delete operations for database views created for storable classes with residency type 9 (GLOBAL_DB_VIEW).
PCM_OP_CUST_COMMIT_CUSTOMER	(Patch Set 4) Modified to support customer contracts.
PCM_OP_CUST_CREATE_CUSTOMER	(Patch Set 4) Modified to support customer contracts.
PCM_OP_CUST_SET_STATUS	(Patch Set 4) Modified to support customer contracts.
PCM_OP_GET_PRICE_LIST	(Patch Set 4) Modified to support customer contracts.
PCM_OP_JOB_EXECUTE	(Patch Set 4) Modified to support Business Operations Center on multischema systems. The output flist now contains an array of POIDs for the /job/boc objects of each schema.

Table 7-3 (Cont.) Changed Standard Opcodes

Changed Standard Opcode	Description
PCM_OP_JOB_PROCESS_TEMPLAT E	(Patch Set 4) Modified to support Business Operations Center on multischema systems. It now includes the PIN_FLD_SCHEMAS array in its input list, which specifies the schema on which to run the business operations job.
PCM_OP_PRICE_COMMIT_DEAL	(Patch Set 4) Modified to support extended attributes.
PCM_OP_PRICE_COMMIT_DISCOU NT	(Patch Set 4) Modified to support extended attributes.
PCM_OP_PRICE_COMMIT_PLAN	(Patch Set 4) Modified to support extended attributes.
PCM_OP_PRICE_COMMIT_PRODU CT	(Patch Set 4) Modified to support extended attributes.
PCM_OP_PRICE_COMMIT_SPONSO RSHIP	(Patch Set 4) Modified to support extended attributes.
PCM_OP_PRICE_GET_PRICE_LIST	(Patch Set 4) Modified to support versioning with date ranges and checking the PIN_FLD_MODE field.
PCM_OP_PRICE_SET_PRICE_LIST	(Patch Set 4) Modified to support customer contracts, versioning with date ranges, checking the PIN_FLD_MODE field, and extended attributes.
PCM_OP_RATE_EVENT	(Patch Set 4) Modified to support versioning with date ranges.
PCM_OP_SUBSCRIPTION_CALC_Q UOTE	(Patch Set 4) Modified to support customer contracts.
PCM_OP_SUBSCRIPTION_PURCHA SE_DEAL	(Patch Set 4) Modified to support customer contracts, extended attributes, and checking the PIN_FLD_MODE field.
PCM_OP_SUBSCRIPTION_PURCHA SE_PRODUCT	(Patch Set 4) Modified to support extended attributes and checking the PIN_FLD_MODE field.
PCM_OP_SUBSCRIPTION_PURCHA SE_DISCOUNT	(Patch Set 4) Modified to support extended attributes and checking the PIN_FLD_MODE field.
PCM_OP_SUBSCRIPTION_SHARIN G_GROUP_CREATE	(Patch Set 4) Modified to support extended attributes.
PCM_OP_SUBSCRIPTION_SHARIN G_GROUP_MODIFY	(Patch Set 4) Modified to support extended attributes.
PCM_OP_JOB_PROCESS_TEMPLAT E	(Patch Set 3) Modified to support custom job templates.
PCM_OP_CUST_DELETE_ACCT	(Patch Set 3) Includes a new PIN_FLD_FLAGS list field. When this input field is set to 1, ECE can delete accounts that have active open sessions.
PCM_OP_AR_GET_BILLS	(Patch Set 2) Modified to support enhanced wholesale billing.
PCM_OP_AR_GET_ACCT_BILLS	(Patch Set 2) Modified to support enhanced wholesale billing.
PCM_OP_AR_GET_ACCT_BAL_SUM MARY	(Patch Set 2) Modified to support enhanced wholesale billing.
PCM_OP_AR_GET_ACTION_ITEMS	(Patch Set 2) Modified to support enhanced wholesale billing.
PCM_OP_AR_GET_ACCT_ACTION_I TEMS	(Patch Set 2) Modified to support enhanced wholesale billing.
PCM_OP_AR_GET_BAL_SUMMARY	(Patch Set 2) Modified to support enhanced wholesale billing.
PCM_OP_AR_GET_BILL_ITEMS	(Patch Set 2) Modified to support enhanced wholesale billing.
PCM_OP_AR_GET_DISPUTES	(Patch Set 2) Modified to support enhanced wholesale billing.
PCM_OP_AR_GET_DISPUTE_DETAI LS	(Patch Set 2) Modified to support enhanced wholesale billing.

Table 7-3 (Cont.) Changed Standard Opcodes

Changed Standard Opcode	Description
PCM_OP_CUST_COMMIT_CUSTOM ER	(Patch Set 2) Modified to include the PIN_FLD_TRANSACTIONS array for storing card credentials.
PCM_OP_CUST_CREATE_PAYINFO	(Patch Set 2) Modified to include the PIN_FLD_TRANSACTIONS array for storing card credentials.
PCM_OP_CUST_SET_PAYINFO	(Patch Set 2) Modified to include the PIN_FLD_TRANSACTIONS array for storing card credentials.
PCM_OP_PYMT_CHARGE	(Patch Set 2) Modified to use the stored card credentials for payment transactions.
PCM_OP_PYMT_CHARGE_CC	(Patch Set 2) Modified to use the stored card credentials for payment transactions.
PCM_OP_PYMT_COLLECT	(Patch Set 2) Modified to use the stored card credentials for payment transactions.
PCM_OP_AR_ACCOUNT_ADJUSTM ENT	(Patch Set 1) Modified to skip account-level adjustment for the nonpaying child bill units in the wholesale bill unit hierarchy.
PCM_OP_AR_ACCOUNT_WRITEOF F	(Patch Set 1) Modified to skip write-off adjustments for the nonpaying child bill units in the wholesale bill unit hierarchy.
PCM_OP_AR_BILL_ADJUSTMENT	(Patch Set 1) Modified to perform the following: <ul style="list-style-type: none"> • Limit the search to only the paying parent bill unit in the wholesale bill unit hierarchy. • Perform the validations to ensure that the bill amount is not over-adjusted during the adjustment.
PCM_OP_AR_BILL_DISPUTE	(Patch Set 1) Modified to perform the following: <ul style="list-style-type: none"> • Limit the search to only the paying parent bill unit in the wholesale bill unit hierarchy. • Perform the validations to ensure that the dispute items of the nonpaying child bill units are not provided in the input list.
PCM_OP_AR_BILL_SETTLEMENT	(Patch Set 1) Modified to perform the following: <ul style="list-style-type: none"> • Limit the search to only the paying parent bill unit in the wholesale bill unit hierarchy. • Perform the validations to ensure that the settlement items of the nonpaying child bill units are not provided in the input list.
PCM_OP_AR_BILL_WRITEOFF	(Patch Set 1) Modified to perform the following: <ul style="list-style-type: none"> • Limit the search to only the paying parent bill unit in the wholesale bill unit hierarchy. • Skip write-off adjustments for the nonpaying child bill units in the wholesale bill unit hierarchy.
PCM_OP_AR_BILLINFO_WRITEOFF	(Patch Set 1) Modified to limit the search to only the paying parent bill unit in the wholesale bill unit hierarchy and skip write-off adjustments for the nonpaying child bill units in the wholesale bill unit hierarchy.
PCM_OP_AR_EVENT_ADJUSTMEN T	(Patch Set 1) Modified to adjust a billed event's balance impact of a nonpaying child bill unit in the wholesale bill unit hierarchy. For wholesale billing, this opcode creates temporary A/R items (/tmp_ar_item_to_roll_up object) with the details of the adjustment and the due amount of the nonpaying child bill unit after adjustment.

Table 7-3 (Cont.) Changed Standard Opcodes

Changed Standard Opcode	Description
PCM_OP_AR_EVENT_DISPUTE	(Patch Set 1) Modified to open disputes for the billed events of the nonpaying child bill units in the wholesale bill unit hierarchy. For wholesale billing, this opcode creates temporary A/R items (/ tmp_ar_item_to_roll_up object) with the details of the dispute and the amount to be transferred to the paying parent bill unit.
PCM_OP_AR_EVENT_SETTLEMENT	(Patch Set 1) Modified to settle the billed dispute events of the nonpaying child bill units in the wholesale bill unit hierarchy. For wholesale billing, this opcode creates temporary A/R items (/ tmp_ar_item_to_roll_up object) with the details of the settlement and the latest disputed, transferred, and total amounts of the nonpaying child bill unit.
PCM_OP_AR_ITEM_ADJUSTMENT	(Patch Set 1) Modified to skip item-level adjustments for the billed items of the nonpaying child bill units in the wholesale bill unit hierarchy.
PCM_OP_AR_ITEM_DISPUTE	(Patch Set 1) Modified to skip item-level dispute for the billed items of the nonpaying child bill units in the wholesale bill unit hierarchy.
PCM_OP_AR_ITEM_SETTLEMENT	(Patch Set 1) Modified to settle the billed dispute items of the nonpaying child bill units in the wholesale bill unit hierarchy. For wholesale billing, this opcode creates temporary A/R items (/ tmp_ar_item_to_roll_up object) with the details of the settlement and the latest disputed, transferred, and total amounts of the nonpaying child bill unit.
PCM_OP_AR_ITEM_WRITEOFF	(Patch Set 1) Modified to skip write-off adjustments for the bill items of the nonpaying child bill units in the wholesale bill unit hierarchy.
PCM_OP_BILL_MAKE_BILL	(Patch Set 1) Modified to perform the following: <ul style="list-style-type: none"> roll charges (bill items) and journals of the nonpaying child bill units up to the paying parent bill unit for wholesale billing. Update the bill objects of the nonpaying child bill units for wholesale billing.
PCM_OP_COLLECTIONS_INVOKE_PROMISE_TO_PAY	(Patch Set 1) Modified to return the payment milestones, milestone amount, and milestone due date when called in the calc-only mode.
PCM_OP_CUST_COMMIT_CUSTOM ER	(Patch Set 1) Modified to support wholesale account creation.
PCM_OP_CUST_CREATE_CUSTOM ER	(Patch Set 1) Modified to support wholesale account creation.
PCM_OP_CUST_DELETE_ACCT	(Patch Set 1) Modified to delete all the objects (such as / event and / bill) and audit entries when the corresponding account is deleted.
PCM_OP_CUST_SET_BILLINFO	(Patch Set 1) Modified to perform the following: <ul style="list-style-type: none"> Set the items of the bill unit moved in or out of a wholesale bill unit hierarchy to point to the appropriate AR_ITEM_OBJ and ITEM_CLASS. Set the PIN_FLD_PARENT_FLAGS field to 1 only if the bill unit is assigned to the wholesale business profile or if the system-wide wholesale billing is enabled. Perform the validations to ensure that there is only one paying parent bill unit in the wholesale bill unit hierarchy.
PCM_OP_INV_MAKE_INVOICE	(Patch Set 1) Modified to create invoice only for the items of the paying parent bill unit in the wholesale bill unit hierarchy.
PCM_OP_INV_POL_PREP_INVOICE	(Patch Set 1) Modified to retrieve only the items of the paying parent bill unit in the wholesale bill unit hierarchy.

Table 7-3 (Cont.) Changed Standard Opcodes

Changed Standard Opcode	Description
PCM_OP_PYMT_COLLECT	(Patch Set 1) Modified to consider only the items of the paying parent bill unit in the wholesale bill unit hierarchy.
PCM_OP_PYMT_MBI_DISTRIBUTE	(Patch Set 1) Modified to consider only the items of the paying parent bill unit in the wholesale bill unit hierarchy.
PCM_OP_SUBSCRIPTION_RERATE_REBILL	(Patch Set 1) Modified to create temporary A/R items (<i>/tmp_ar_item_to_roll_up</i> object) after rerating nonpaying child bill units in the wholesale bill unit hierarchy.

Changed Policy Opcodes

Table 7-4 lists the FM policy opcodes that have been modified in the BRM 12.0 patch set releases.

Table 7-4 Changed Policy Opcodes

New Policy Opcode	Description
PCM_OP_ACT_POL_REQUEST_CREATE	(Patch Set 8) Modified to include the following new input and output flist field: <ul style="list-style-type: none"> PIN_FLD_NOTIFICATION_SPEC array passes additional details with the request record in the notification event. This array and any field that it contains will not be persisted with the request.
PCM_OP_COLLECTIONS_POL_CALC_DUE_DATE	(Patch Set 7) Modified to include these new input flist fields: <ul style="list-style-type: none"> PIN_FLD_ACTION_NAME specifies the name of the collections action, for example, courtesy phone call. PIN_FLD_ACTION_TYPE specifies the type of collections action, for example, manual_action.
PCM_OP_COLLECTIONS_POL_INVOLVE_PROMISE_TO_PAY	(Patch Set 7) Modified to include the new PIN_FLD_RESULTS array in the input flist. The array contains the results of validation based on your configuration in the promise-to-pay specification passed in the opcode's input flist.
PCM_OP_CUST_POL_GET_DEALS	(Patch Set 7) Modified to include these new output flist fields in the PIN_FLD_PRODUCTS array: <ul style="list-style-type: none"> PIN_FLD_BASE_PRODUCT_OBJ specifies the POID of the base product. PIN_FLD_VALIDITY_ALIGN_MODE specifies the validity align mode for add-on products: PIN_VALIDITY_ALIGN_BASE (7), PIN_VALIDITY_ALIGN_ANY_BASE_EARLIEST (8), PIN_VALIDITY_ALIGN_ANY_BASE_LATEST (9), PIN_VALIDITY_ALIGN_EARLIEST (10), and PIN_VALIDITY_ALIGN_LATEST (11). PIN_FLD_BASE_PRODUCT_NAME specifies the base product name. PIN_FLD_BASE_PRODUCT_CODE specifies the base product code.

Table 7-4 (Cont.) Changed Policy Opcodes

New Policy Opcode	Description
PCM_OP_CUST_POL_GET_PLANS	(Patch Set 7) The output flist contains the following changes: <ul style="list-style-type: none"> • New PIN_FLD_ATTRIBUTE_OBJ field in the PIN_FLD_PLAN array contains the offer attribute POID. • Existing PIN_FLD_CREDIT_THRESHOLDS_FIXED field in the PIN_FLD_LIMIT array changed from decimal to string format. • Existing PIN_FLD_CREDIT_THRESHOLDS field in the PIN_FLD_LIMIT array changed from decimal to string format.
PCM_OP_NOTIFICATION_POL_VALID_SPECIFICATION PCM_OP_NOTIFICATION_POL_PREP_SPECIFICATION	(Patch Set 7) Under the PIN_FLD_NOTIFICATION_SPEC array, the PIN_FLD_NOTIFICATION_TYPE input flist field supports the new value AFTER_DUE_DATE (5).
PCM_OP_PYMT_POL_UNDER_PAYMENT	(Patch Set 7) Modified to include the following new output flist array: <ul style="list-style-type: none"> • PIN_FLD_TAXES specifies the itemized tax details. This array is included when the ItemizedTaxCalculation business parameter is set to enabled.
PCM_OP_PUBLISH_POL_PREP_EVENT	(Patch Set 6) Modified to support the Kafka DM. It contains this new flist substruct: <ul style="list-style-type: none"> • PIN_FLD_NOTIFICATION_STATUS_INFO specifies the delivery identifiers for each delivery method.
PCM_OP_CUST_POL_TAX_INIT	(Patch Set 5) Modified to support reading tax codes from the /config/taxcodes_map file.
PCM_OP_CUST_POL_POST_COMMIT	(Patch Set 4) Modified to support contracts.
PCM_OP_CUST_POL_PREP_PASSWORD	(Patch Set 4) Modified to generate a random eight-character password for prepaid services by default.
PCM_OP_PYMT_POL_PRE_COLLECT	(Patch Set 2) Modified to support PIN_FLD_TRANSACTIONS array for stored-credential transactions.
PCM_OP_CUST_POL_VALID_PAYINFO	(Patch Set 2) Modified to support PIN_FLD_TRANSACTIONS array for stored-credential transactions.
PCM_OP_INV_POL_PREP_INVOICE	(Patch Set 1) Modified to prepare the invoice information for wholesale accounts.

8

Storable Class Changes

The Oracle Communications Billing and Revenue Management (BRM) 12.0 Patch Sets include new and modified storable classes.

Topics in this document:

- [New Storable Classes](#)
- [Changed Storable Classes](#)

New Storable Classes

[Table 8-1](#) lists the storable classes that have been introduced in the BRM 12.0 patch set releases.

Table 8-1 New Storable Classes

New Storable Class	Description
/config/collections/ promise_to_pay_spec	(Patch Set 7) Store details about a promise-to-pay specification for bills that are in collections. It lists the minimum installment amount, the number of days between installment payments, and so on.
/config/stored_procedure	(Patch Set 7) Stores information about all stored procedures that can be run by client applications connected to the CM. The object contains information about stored procedures included with BRM, but you can add custom stored procedures.
/event/group/sharing/products	(Patch Set 7) Stores information about product sharing groups.
/event/group/sharing/products/ create	(Patch Set 7) This event is generated when a /group/sharing/products object is created for an account or service.
/event/group/sharing/products/ delete	(Patch Set 7) This event is generated when a /group/sharing/products object is deleted for an account or service.
/event/group/sharing/products/ modify	(Patch Set 7) This event is generated when a /group/sharing/products object is modified for an account or service.
/group/sharing/products	(Patch Set 7) Stores a list of products for the owner of a product sharing group.
/config/calendar/silent_days	(Patch Set 6) Stores all calendar days that are considered silent days. Messages cannot be sent to customers on these days.
/config/deposit	(Patch Set 6) Stores configuration data for deposits.
/config/deposit_spec_profile	(Patch Set 6) Stores the common business rules for the deposits in your organization.
/deposit_refund_request	(Patch Set 6) Stores information about an account's deposit refund status.
/deposit_specification	(Patch Set 6) Stores the configuration for a deposit specification.
/event/billing/deposit	(Patch Set 6) Stores details about a deposit event.

Table 8-1 (Cont.) New Storable Classes

New Storable Class	Description
/event/billing/deposit/reversal	(Patch Set 6) Stores details about a deposit reversal event.
/event/billing/deposit/interest	(Patch Set 6) Event created when interest is applied to a deposit.
/event/billing/deposit/interest/credit	(Patch Set 6) Event created when interest is applied to a credit card deposit.
/event/billing/deposit/interest/debit	(Patch Set 6) Event created when interest is applied to a debit deposit.
/event/billing/loan_grant	(Patch Set 6) Stores details about a loan grant event.
/event/billing/loan_late_fee	(Patch Set 6) Event created when a late payment fee is applied for a late loan repayment.
/event/billing/loan_pullback	(Patch Set 6) Event created when the amount due for a loan is pulled from the customer's main balance.
/event/billing/loan_recovery	(Patch Set 6) Event created when a loan grant is recovered from a customer.
/event/billing/loan_recovery_reversal	(Patch Set 6) Event created when a customer reverts a loan payment.
/event/billing/product/fee/purchase/deposit	(Patch Set 6) Event created when a deposit fee associated with a product is applied.
/event/billing/product/fee/purchase/deposit/refund	(Patch Set 6) Event created when a deposit refund fee associated with a product is applied.
/event/deposit	(Patch Set 6) Stores a deposit event.
/event/deposit/collected	(Patch Set 6) Event created when a deposit is collected.
/event/deposit/create	(Patch Set 6) Event created when a deposit is created.
/event/deposit/receive	(Patch Set 6) Event created when a deposit is received.
/event/deposit/refund	(Patch Set 6) Event created when a deposit is refunded.
/event/deposit/refund/approved	(Patch Set 6) Event created when the request for a deposit refund is approved.
/event/deposit/refund/rejected	(Patch Set 6) Event created when the request for a deposit refund is rejected.
/event/deposit/refund/request	(Patch Set 6) Event created when a deposit refund is requested.
/event/deposit/release	(Patch Set 6) Event created when a deposit is released.
/event/deposit/reversal	(Patch Set 6) Event created when a deposit is reversed.
/event/deposit/transfer	(Patch Set 6) Event created when a deposit is transferred.
/event/deposit/update	(Patch Set 6) Event created when a deposit is updated.
/item/deposit	(Patch Set 6) Stores details about a deposit.
/item/deposit/charge	(Patch Set 6) Stores details about a deposit charge.
/item/deposit/interest	(Patch Set 6) Stores details about interest on a deposit.
/item/deposit/interest/credit	(Patch Set 6) Stores details about deposit interest applied to a credit card.
/item/deposit/interest/debit	(Patch Set 6) Stores details about deposit interest applied to debits.
/item/deposit/refund	(Patch Set 6) Stores details about a refunded deposit.

Table 8-1 (Cont.) New Storable Classes

New Storable Class	Description
<code>/item/deposit/refund/fee</code>	(Patch Set 6) Stores details about a refund deposit fee.
<code>/item/deposit/reversal</code>	(Patch Set 6) Stores details about a deposit reversal.
<code>/item/loan_late_fee</code>	(Patch Set 6) Stores details about late fees applied to a loan.
<code>/purchased/deposit</code>	(Patch Set 6) Stores details about a purchased deposit.
<code>/request</code>	(Patch Set 6) A base class for recording a request or transaction for review.
<code>/request/failed</code>	(Patch Set 6) Stores the error summary for a failed request.
<code>/request/failed/opcode</code>	(Patch Set 6) Stores information about the opcode called as part of the failed request.
<code>/request/failed/rest</code>	(Patch Set 6) Stores information about the failed request from the REST API.
<code>/config/delivery_methods</code>	(Patch Set 5) Stores the supported delivery methods, such as email or IVR, for notifications.
<code>/config/event_promo_map</code>	(Patch Set 5) Stores the mapping between events and promotional events for applying event-based promotions.
<code>/config/event_promo_tag_map</code>	(Patch Set 5) Stores the mapping between promotional tags and promotional events.
<code>/config/loan</code>	(Patch Set 5) A configuration object that stores loan parameters.
<code>/config/notification_spec</code>	(Patch Set 5) Stores notification specifications.
<code>/config/taxcodes_map</code>	(Patch Set 5) Stores your tax codes.
<code>/config/taxexemption_code_map</code>	(Patch Set 5) Stores details about each tax exemption code.
<code>/event/audit/transfer_balance</code>	(Patch Set 5) Stores an audit event for tracking individual balance transfers between balance groups.
<code>/event/billing/fee/balance_transfer</code>	(Patch Set 5) Stores the balance transfer fee for the target or source account.
<code>/event/billing/loan_credit</code>	(Patch Set 5) Stores details about a loan event.
<code>/event/billing/loan_debit</code>	(Patch Set 5) Stores details about a loan event.
<code>/event/billing/loan_fee</code>	(Patch Set 5) Stores details about a loan fee event.
<code>/event/billing/loan_grant</code>	(Patch Set 5) Stores details about a loan that also grants non-currency resources. You configure charge offers in PDC to grant the resources for this event.
<code>/event/billing/topup</code>	(Patch Set 5) Stores details about a top-up applied to an account.
<code>/installment_schedule</code>	(Patch Set 5) Stores details about an installment payment plan.
<code>/item/installment</code>	(Patch Set 5) Stores details about an installment.
<code>/item/loan_credit</code>	(Patch Set 5) Stores details about a loaned amount.
<code>/item/loan_debit</code>	(Patch Set 5) Stores details about the asset representing a loan.
<code>/item/loan_fee</code>	(Patch Set 5) Stores details about a loan fee.
<code>/offering_override_values</code>	(Patch Set 5) Stores details for overriding a rate for a period of time.
<code>/profile/loan</code>	(Patch Set 5) A profile object that stores loan parameters specific to a customer or service.

Table 8-1 (Cont.) New Storable Classes

New Storable Class	Description
/profile/specialdates	(Patch Set 5) Stores extended account attributes such as special dates. These attributes are used to apply promotions based on the occurrence of special dates.
/schedule/notification	(Patch Set 5) Stores deferred actions for notifications.
/tax_exemption_selector	(Patch Set 5) Stores information about tax exemption selectors, which can be used to apply tax exemptions based of different attributes.
/config/note_type	(Patch Set 4) Stores information for account-level notes in Billing Care.
/deliverable	(Patch Set 4) Stores details about deliverables.
/deliverable/product	(Patch Set 4) Stores details for a physical good deliverable type.
/deliverable/service	(Patch Set 4) Stores details about these deliverable types: one-time activities, metered services, and continuous services.
/entitlement	(Patch Set 4) Stores entitlement information about deliverables.
/entitlement/allowance	(Patch Set 4) Stores entitlement information for continuous service deliverables.
/entitlement/allowance/onetime	(Patch Set 4) Stores entitlement information for one-time activity deliverables.
/entitlement/allowance/recurring	(Patch Set 4) Stores entitlement information for metered service deliverables.
/entitlement/feature	(Patch Set 4) Stores entitlement information for metered service deliverables.
/event/activity/job_request	(Patch Set 4) Stores audit data when the PCM_OP_JOB_EXECUTE opcode calls an application in a non-job context.
/offer_attribute_group	(Patch Set 4) Stores extended attributes defined for pricing components.
/performance_snapshot/realtime	(Patch Set 4) Store monitoring information for BRM cloud native.
/performance_sql_data	(Patch Set 4) Store monitoring information for BRM cloud native.
/revenue_basis_item	(Patch Set 4) Stores entitlement information for metered service deliverables.
/revenue_basis_item/rate_based	(Patch Set 4) Stores revenue basis data, such as the standalone selling price and earning schedule, for continuous service and physical good deliverables.
/revenue_basis_item/value_based	(Patch Set 4) Stores revenue basis data, such as the standalone selling price and earning schedule, for metered service and one-time activity deliverables.
/event/job_template	(Patch Set 3) Stores a notification event for job_template processing.
/event/job_template/create	(Patch Set 3) Stores a create event for the job_template.
/event/job_template/modify	(Patch Set 3) Stores a modify event for job_template.
/event/job_template/delete	(Patch Set 3) Stores a delete event for job_template.
/job_definition	(Patch Set 3) Stores the job's running location and usage summary.

Table 8-1 (Cont.) New Storable Classes

New Storable Class	Description
/job_template/custom	(Patch Set 3) Stores the command for running the custom Business Operations job.
/history_subscriber_contract	(Patch Set 2) Stores a copy of the subscriber contract data. This class is created when a contract is renewed or modified.
/subscriber_contract	(Patch Set 2) Stores the subscriber contract data.
/cmt_balances	(Patch Set 2) Stores the balance data temporarily.
/tmp_ar_item_to_roll_up	(Patch Set 1) Stores information for the A/R items of the nonpaying child bill units to roll them up to the paying parent bill unit.
/event/activity/roll_up	(Patch Set 1) Records activity related to rolling of items up to the paying parent bill unit.
/recurring_bundle_history	(Patch Set 1) Stores information about recurring bundles.

Changed Storable Classes

Table 8-2 lists the storable classes that have been modified in the BRM 12.0 patch set releases.

Table 8-2 Changed Storable Classes

Changed Storable Class	Description
/config/business_events	<p>(Patch Set 8) Contains this new field under the PIN_FLD_EVENTS array:</p> <ul style="list-style-type: none"> PIN_FLD_NOTIFY_SEARCH_LEVEL specifies the level to consolidate multiple events into a notification: create a separate notification for each event (1), consolidate an account's events into one notification (2), consolidate a balance group's events into one notification (3), consolidate a service's events into one notification (4), and consolidate a bill unit's events into one notification (5). <p>Note: This field is applicable only for notifications sent in advance or after expiration.</p>
/config/distribution	<p>(Patch Set 8) Contains this new field under the PIN_FLD_DISTRIBUTION array:</p> <ul style="list-style-type: none"> PIN_FLD_SCHEMA_NAME specifies the name of the schema.
/config/notification_spec	<p>(Patch Set 8) Contains these new fields under the PIN_FLD_NOTIFICATION_SPEC array:</p> <ul style="list-style-type: none"> PIN_FLD_NOTIFY_OPT_TYPE specifies to opt-out customers from receiving notification messages by default (1), to opt-in customers for receiving notification messages by default (2), or to enforce opt-in preferences for customers (3). PIN_FLD_AGGREGATE_MODE specifies whether to send separate notifications for each event (0) or to consolidate details from multiple events into one notification (1).

Table 8-2 (Cont.) Changed Storable Classes

Changed Storable Class	Description
/deal /event/billing/discount/action /purchased_discount	(Patch Set 8) Contains this new field under the PIN_FLD_DISCOUNTS array: <ul style="list-style-type: none"> PIN_FLD_GRACE_PERIOD specifies the grace period for this offering.
/process_audit/billing	(Patch Set 8) Contains this new field under the PIN_FLD_FAILED_ACCOUNTS array: <ul style="list-style-type: none"> PIN_FLD_SCOPE_OBJ contains a link to the failed object's POID.
/product	(Patch Set 8) Contains these new fields: <ul style="list-style-type: none"> PIN_FLD_OFFER_VALIDITY_ROUNDING specifies the offer validity rounding at the product level. The default is 0. PIN_FLD_SCALE_ROUNDING specifies the scale rounding at the product level. The default is 0.
/service	(Patch Set 8) Contains these new fields: <ul style="list-style-type: none"> PIN_FLD_SVC_EXP_LAST_NOTI_T specifies the last time this object was processed for a SERVICE_STATE_EXPIRATION_T notification. PIN_FLD_SVC_EXP_LAST_NOTI_OFFSET specifies the offset the last time this object was processed for a SERVICE_STATE_EXPIRATION_T notification.
/balance_group	(Patch Set 7) Contains these new fields under the PIN_FLD_BALANCES array: <ul style="list-style-type: none"> PIN_FLD_ROLLED_UP_CREDIT_LIMIT specifies the credit limit to roll up in a billing hierarchy.
/collections_scenario	(Patch Set 7) Contains these new fields under the PIN_FLD_PROMISE_TO_PAY array: <ul style="list-style-type: none"> PIN_FLD_STATUS specifies the status of a collections action: pending (0), canceled (1), completed (2), kept (3), or broken (4). PIN_FLD_CONFIG_SPEC_OBJ contains a link to a /config/collections/promise_to_pay_spec object.
/config/business_events	(Patch Set 7) The PIN_FLD_EVENT_TYPE field includes this new value: <ul style="list-style-type: none"> AFTER-DUE-DATE specifies that notifications for this business event can be sent after an event occurs.
/config/credit_profile	(Patch Set 7) Contains these new fields in the PIN_FLD_PROFILES array: <ul style="list-style-type: none"> PIN_FLD_DYNAMIC_CREDIT_FLOOR specifies whether to use a dynamic credit floor: use the existing credit floor for credit limit breach checks (0) or dynamically update the credit floor for a given period of time (1). PIN_FLD_OVERRIDE_CREDIT_LIMIT specifies whether to override the credit limit: the credit limit is set based on the rating mode (0) or the credit limit is overridden (0x01).

Table 8-2 (Cont.) Changed Storable Classes

Changed Storable Class	Description
/config/notification_spec	(Patch Set 7) The following fields include new values: <ul style="list-style-type: none"> PIN_FLD_EVENT_NAME includes these additional values: PostBalanceExpiry, PostBillDue, PostProductExpiry, PostSubscriptionRenewalDue, PostCollectionsActionDue, PostInstallmentDue. PIN_FLD_EVENT_TYPE includes the new AFTER-DUE-DATE value.
/deal	(Patch Set 7) Contains these new fields under the PIN_FLD_PRODUCTS array: <ul style="list-style-type: none"> PIN_FLD_FLAGS specifies how to prorate at the product level: default system configuration (0), 30-day proration (1), or number of days in the month (2). PIN_FLD_BASE_PRODUCT_OBJ contains a link to the base product object.
/discount	(Patch Set 7) Contains this new field: <ul style="list-style-type: none"> PIN_FLD_APPLY_MODE specifies whether to trigger automatic sharing group creation (1) or not (0).
/event	(Patch Set 7) Contains these new fields under the PIN_FLD_SUB_BAL_IMPACTS array: <ul style="list-style-type: none"> PIN_FLD_CURRENT_BAL specifies the total available balance for this resource. PIN_FLD_AVAILABLE_LOAN_BALANCE specifies the loan balance available for this resource.
/event/activity/collections/promise_to_pay	(Patch Set 7) Contains this new array: <ul style="list-style-type: none"> PIN_FLD_MILESTONES contains the promise-to-pay information for unequal installments and unequal intervals.
/event/billing/limit	(Patch Set 7) Contains these new fields in the PIN_FLD_LIMIT array: <ul style="list-style-type: none"> PIN_FLD_DYNAMIC_CREDIT_FLOOR specifies whether to use a dynamic credit floor: use the existing credit floor for credit limit breach checks (0) or dynamically update the credit floor for a given period of time (1). PIN_FLD_OVERRIDE_CREDIT_LIMIT specifies whether to override the credit limit: the credit limit is set based on the rating mode (0) or the credit limit is overridden (0x01).
/event/billing/product/action	(Patch Set 7) Contains this new field in the PIN_FLD_ACTION_INFO array: <ul style="list-style-type: none"> PIN_FLD_BASE_PRODUCT_OBJ contains a link to the base product object.
/plan	(Patch Set 7) Contains these new fields in the PIN_FLD_LIMIT array: <ul style="list-style-type: none"> PIN_FLD_OVERRIDE_CREDIT_LIMIT specifies whether to override the credit limit: the credit limit is set based on the rating mode (0) or the credit limit is overridden (0x01). PIN_FLD_DYNAMIC_CREDIT_FLOOR specifies whether to use a dynamic credit floor: use the existing credit floor for credit limit breach checks (0) or dynamically update the credit floor for a given period of time (1).

Table 8-2 (Cont.) Changed Storable Classes

Changed Storable Class	Description
/product	(Patch Set 7) Contains this new field: <ul style="list-style-type: none"> PIN_FLD_APPLY_MODE specifies whether to trigger automatic sharing group creation (1) or not (0).
/purchased_deposit	(Patch Set 7) Contains these new fields: <ul style="list-style-type: none"> PIN_FLD_PARENT contains a link to the parent purchased deposit object. This is set when the deposit amount is updated. PIN_FLD_END_DETAILS is an encoded field holding mode, unit, and offset. The mode is always relative (4) occupying 8 bits starting from LSB. The unit can be months (0), days (1), weeks (2), and years (3), occupying 4 bits after mode. The offset is the value for unit, occupying 20 bits after unit.
/purchased_product	(Patch Set 7) Contains this new field: <ul style="list-style-type: none"> PIN_FLD_BASE_PRODUCT_OBJ contains a link to the base product object.
/balance_group	(Patch Set 6) Contains these new fields under the PIN_FLD_SUB_BALANCES array: <ul style="list-style-type: none"> PIN_FLD_SUBTYPE specifies the type of sub-balance, which can be default (0), loan (1), or temporary credit limit (2).
/collections_action	(Patch Set 6) Modified to sending messages to customers through external notification applications. It contains these new fields: <ul style="list-style-type: none"> PIN_FLD_LAST_NOTIFICATION_T specifies the last time this object was processed for expiration notifications. PIN_FLD_LAST_NOTIFICATION_OFFSET specifies the offset the last time this object was processed.
/config/business_events	(Patch Set 6) Modified to support notifications in Billing Care. The object's PIN_FLD_EVENT_TYPE field is now set to one of the following: <ul style="list-style-type: none"> IN_ADVANCE: Specifies that notifications for this business event can be sent several days before an event occurs. REGULAR: Specifies that notifications for this business event can be sent immediately or at a specific time of day.
/config/credit	(Patch Set 6) Contains these new fields: <ul style="list-style-type: none"> PIN_FLD_LOAN_THRESHOLDS: Percent threshold for loan notifications. PIN_FLD_LOAN_THRESHOLDS_FIXED: Fixed threshold for loan notifications.
/config/delivery_methods	(Patch Set 6) Modified to sending messages to customers through external notification applications. It contains this new field: <ul style="list-style-type: none"> PIN_FLD_DELIVERY_IDENTIFIER_TAG specifies the delivery identifier tag.
/event/billing/cycle/tax	(Patch Set 6) The PIN_FLD_TAXES array contains the new PIN_FLD_ITEM_OBJ field, which holds the corresponding charge item POID.

Table 8-2 (Cont.) Changed Storable Classes

Changed Storable Class	Description
/event/billing/limit	(Patch Set 6) Modified to support temporary credit limits. It contains these new fields: <ul style="list-style-type: none"> • PIN_FLD_SUB_BAL_IMPACTS array specifies the temporary credit limit.
/event/billing/loan_fee	(Patch Set 6) Modified to include the PIN_FLD_AMOUNT field to indicate the loan amount.
/event/billing/payment/cc	(Patch Set 6) Contains these new fields: <ul style="list-style-type: none"> • PIN_FLD_TRACE_NUMBER specifies the trace number returned by Paymentech for PINless debit authorization. • PIN_FLD_AUTH_MOP specifies the Method of Payment (MOP) used to process the PINless debit authorization.
/event/billing/product/action	(Patch Set 6) The PIN_FLD_NEXT_RETRY_T field is added to record when a subscription for which the subscriber has insufficient balance will next be retried.
/installment_schedule	(Patch Set 6) Modified to sending messages to customers through external notification applications. It contains these new fields: <ul style="list-style-type: none"> • PIN_FLD_LAST_NOTIFICATION_T specifies the last time this object was processed for expiration notifications. • PIN_FLD_LAST_NOTIFICATION_OFFSET specifies the offset the last time this object was processed for an end date notification. • In the PIN_FLD_INSTALLMENTS array, the PIN_FLD_LAST_NOTIFICATION_OFFSET field specifies the offset the last time this object was processed for an installment due notification.
/item	(Patch Set 6) Modified to include the PIN_FLD_TAX_DATA field, which holds tax-related data about an item such as the tax code, amount taxed, tax, and amount of tax adjusted.
/payinfo/cc	(Patch Set 6) Contains these new fields: <ul style="list-style-type: none"> • PIN_FLD_TYPE_STR specifies the card type indicator. The first character indicates whether the card supports PINless debit transactions: Y specifies supports, N specifies does not support, and X specifies unknown.
/plan	(Patch Set 6) Contains these new fields: <ul style="list-style-type: none"> • PIN_FLD_LOAN_THRESHOLDS: Percent threshold for loan notifications. • PIN_FLD_LOAN_THRESHOLDS_FIXED: Fixed threshold for loan notifications.

Table 8-2 (Cont.) Changed Storable Classes

Changed Storable Class	Description
/profile/loan	(Patch Set 6) Contains these new fields: <ul style="list-style-type: none"> • PIN_FLD_OPT_LOAN: This value indicates whether customer opted for loan or not. 0 indicates opt_out and 1 indicates opt_in. By default, the value is set to 0. • PIN_FLD_MAX_ACTIVE_LOANS: This value indicates maximum number of active loans • PIN_FLD_REPAYMENT_DAYS: This field stores the days for loan repayment • PIN_FLD_PULLBACK: This field provides an option to enable or disable loan pullback if the loan is not repaid within the • PIN_FLD_REPAYMENT_DAYS. 0 indicates disable pullback and 1 indicates enable pullback. By default, the value is set to 0. • PIN_FLD_LIMIT: This array stores the CREDIT_LIMIT which will override the FIXED_MAXIMUM or SCALED_MAXIMUM in /config/loan. • PIN_FLD_VALID_FROM: Start date for PIN_FLD_CREDIT_LIMIT • PIN_FLD_CREDIT_LIMIT: Maximum loan amount which can be granted
/product	(Patch Set 6) These new fields are used when PIN_FLD_TYPE for the rate is set to 745, 747, or 748 : <ul style="list-style-type: none"> • PIN_FLD_MAX_RETRY_COUNT: The maximum number of times to retry the subscription before failing. • PIN_FLD_RETRY_UNIT: The unit for the retries. Valid values are: <ul style="list-style-type: none"> – 1: Seconds – 2: Minutes – 3: Hours – 4: Days – 5: Months – 6: Years – 10: Weeks • PIN_FLD_RETRY_OFFSET: The frequency of retries.
/purchased_product	(Patch Set 6) The PIN_FLD_NEXT_RETRY_T field is added to record when a subscription for which the subscriber has insufficient balance will next be retried.

Table 8-2 (Cont.) Changed Storable Classes

Changed Storable Class	Description
/rate	<p>(Patch Set 6) PIN_FLD_TYPE is modified to support new values for exceeding credit limits:</p> <ul style="list-style-type: none"> • 747: Charge a minimum amount if a customer has less than the full amount. You can optionally prorate the service according to the amount charged. • 748: Subscription fails and is canceled after the configured number of retries. Retries are configured in a business parameter or in the /product object. <p>These new fields are used when PIN_FLD_TYPE for the rate is set to 747:</p> <ul style="list-style-type: none"> • PIN_FLD_MINIMUM_SCALED_AMOUNT: The minimum amount to charge for the service if the customer does not have sufficient balance. • PIN_FLD_PRORATE_QUANTITY: Whether to prorate the service if the minimum amount is charged (1) or not (0). The default is 1.
/balance_group	<p>(Patch Set 5) Contains these new fields:</p> <ul style="list-style-type: none"> • PIN_FLD_LAST_NOTIFICATION_T specifies the last time this object was processed for expiration notifications. • PIN_FLD_LAST_NOTIFICATION_OFFSET specifies the offset the last time this object was processed. • PIN_FLD_DELTA_CREDIT_LIMIT specifies the credit limit for the balance group that includes loans and any related fees and taxes. • PIN_FLD_OUTSTANDING_AMOUNT specifies outstanding amounts remaining after an existing balance is used up when exceeding a credit limit.
/batch/rel	<p>(Patch Set 5) Contains these new fields to handle new processing options for the Rated Event Manager:</p> <p>PIN_FLD_FILES is an array that contains elements that each correspond to a file of CDR data for this batch Rated Event Loader session. Each element contains the following fields:</p> <ul style="list-style-type: none"> • PIN_FLD_STATUS contains the processing status of the file. • PIN_FLD_FILENAME contains the tracking file name used for this CDR data file. • PIN_FLD_NUM_TOTAL_RECORDS contains the number of distinct events in the data file. • PIN_FLD_BUFFER contains the content of a CDR file in ZIP format. <p>PIN_FLD_STATUSES is an array that contains elements that each correspond to the status of a retry attempt for a Rated Event Loader session. Each element contains the following fields:</p> <ul style="list-style-type: none"> • PIN_FLD_CREATED_T contains the creation time for the retry record. • PIN_FLD_PROCESS_START_T contains the time that the retry attempt was started. • PIN_FLD_PROCESS_END_T contains the time that the retry attempt was completed. • PIN_FLD_STATUS contains the processing state of the retry attempt.

Table 8-2 (Cont.) Changed Storable Classes

Changed Storable Class	Description
/bill	(Patch Set 5) Contains these new fields: <ul style="list-style-type: none"> • PIN_FLD_LAST_NOTIFICATION_T specifies the last time this object was processed for expiration notifications. • PIN_FLD_LAST_NOTIFICATION_OFFSET specifies the offset the last time this object was processed. • PIN_FLD_STATE specifies the state of the bill: UNDEFINED, INPROGRESS, NEW, PARTIALLYPAID, SETTLED, ONHOLD, SENT, or VALIDATED.
/config/beid	(Patch Set 5) Contains the new PIN_FLD_ENABLE_EXPIRY_NOTIFICATION field, which specifies whether to generate notifications if a customer's balance is about to expire.
/config/credit_profile	(Patch Set 5) Contains new fields to configure the thresholds for triggering automatic loans: <ul style="list-style-type: none"> • PIN_FLD_RESRC_FIXED_THRESHOLD • PIN_FLD_RESRC_PERC_THRESHOLD
/config/subscriber_preferences_map	(Patch Set 5) The PIN_FLD_TYPE field supports three new values: 6 (TIME), 7 (MULTI SELECT ENUM), and 262 (TIME RANGE).
/deal	(Patch Set 5) Contains these new fields in the PIN_FLD_PRODUCTS array: <ul style="list-style-type: none"> • PIN_FLD_RENEWAL_MODE specifies whether a reactivated product's cycle is aligned with the original purchase date or the reactivation date. • PIN_FLD_GRACE_PERIOD_OFFSET specifies the amount of the grace period. • PIN_FLD_GRACE_PERIOD_UNIT specifies the unit of the grace period, which can be seconds, minutes, hours, or days.
/event	(Patch Set 5) The PIN_FLD_OUTSTANDING_AMOUNT field is added in the PIN_FLD_BAL_IMPACTS array. This field specifies outstanding amounts remaining after an existing balance is used up when exceeding a credit limit.
/event/billing/discount/action	(Patch Set 5) Contains these new fields: <ul style="list-style-type: none"> • PIN_FLD_OFFERING_OVERRIDE_VALUES_OBJ contains the POID of the /offering_override_values object associated with this discount.
/event/billing/product/action	(Patch Set 5) Contains these new fields: <ul style="list-style-type: none"> • PIN_FLD_RENEWAL_MODE specifies whether a reactivated product's cycle is aligned with the original purchase date or the reactivation date. • PIN_FLD_GRACE_PERIOD_OFFSET specifies the amount of the grace period. • PIN_FLD_GRACE_PERIOD_UNIT specifies the unit of the grace period, which can be seconds, minutes, hours, or days. • PIN_FLD_OFFERING_OVERRIDE_VALUES_OBJ contains the POID of the /offering_override_values object associated with the product.

Table 8-2 (Cont.) Changed Storable Classes

Changed Storable Class	Description
/history_bills	(Patch Set 5) Contains these new fields: <ul style="list-style-type: none"> • PIN_FLD_LAST_NOTIFICATION_T specifies the last time this object was processed for expiration notifications. • PIN_FLD_LAST_NOTIFICATION_OFFSET specifies the offset the last time this object was processed.
/product	(Patch Set 5) Contains the new PIN_FLD_ENABLE_NOTIFICATION field, which specifies whether to generate notification events for subscriptions that contain this charge offer when it is about to expire or renew. Possible values are: <ul style="list-style-type: none"> • Both product expiration and subscription renewal notifications are enabled (3) • Product expiration notifications are disabled and subscription renewal notifications are enabled (2) • product expiration notifications are enabled and subscription renewal notifications are disabled (1) • Both product expiration and subscription renewal notifications are disabled (0) Also, the existing PIN_FLD_FLAGS field in the PIN_FLD_USAGE_MAP array includes a new possible value of RATE_LIFECYCLE_SUSPEND (0x04), which specifies to continue charging the product when the life cycle state is suspended.
/profile/subscriber_preferences	(Patch Set 5) Contains the new PIN_FLD_SUBSCRIBER_PREFERENCES array, which specifies an account's subscriber preferences.
/purchased_discount	(Patch Set 5) Contains these new fields: <ul style="list-style-type: none"> • PIN_FLD_OFFERING_OVERRIDE_VALUES_OBJ contains the POID of the /offering_override_values object associated with this discount.
/purchased_product	(Patch Set 5) Contains these new fields: <ul style="list-style-type: none"> • PIN_FLD_LAST_NOTIFICATION_T specifies the last time this object was processed for expiration notifications. • PIN_FLD_LAST_NOTIFICATION_OFFSET specifies the offset the last time this object was processed. • PIN_FLD_RENEWAL_MODE specifies whether a reactivated product's cycle is aligned with the original purchase date or the reactivation date. • PIN_FLD_GRACE_PERIOD_OFFSET specifies the amount of the grace period. • PIN_FLD_GRACE_PERIOD_UNIT specifies the unit of the grace period, which can be seconds, minutes, hours, or days. • PIN_FLD_RETRY_COUNT specifies how many times the subscription has been retried when the credit limit has been exceeded. • PIN_FLD_OFFERING_OVERRIDE_VALUES_OBJ contains the POID of the /offering_override_values object associated with this product.

Table 8-2 (Cont.) Changed Storable Classes

Changed Storable Class	Description
/rate	<p>(Patch Set 5) Contains these new fields:</p> <ul style="list-style-type: none"> • PIN_FLD_PRORATION_FLAG specifies how to apply cycle fees when customers change their billing DOM in the middle of a billing cycle: apply the full cycle fee (701), do not apply the cycle fee (703), or prorate the cycle fee (0). • PIN_FLD_SCALED_PRICE_TAG specifies the price tag for the scaled amount. • PIN_FLD_FIXED_PRICE_TAG specifies the price tag for the fixed amount. • PIN_FLD_SPLIT_DETAILS is a 32-bit field that specifies the logic for creating multiple validity periods: <ul style="list-style-type: none"> – The lower 8 bits specify the expiration mode for the incremental validity period: bucket (5) or total (6). – The second 4 bits specify the units for the incremental validity period: seconds (1), minutes (2), hours (3), days (4), months (5). – The remaining 20 MSB bits specify the amount for the validity period. • PIN_FLD_TYPE is modified to support new values for exceeding credit limits: <ul style="list-style-type: none"> – 742: Use available balance and grant a loan for the rest. – 743: Use available balance and record the rest as an outstanding amount. – 744: Don't use available balance and record the entire amount as an outstanding amount. – 745: Don't exceed the credit limit and fail the subscription. – 746: Skip billing for this cycle.
/topup	<p>(Patch Set 5) Modified to support top-ups of noncurrency resources and recurring top-ups, and proration during state changes. It contains these new fields:</p> <ul style="list-style-type: none"> • PIN_FLD_RESOURCE_ID specifies the resource to top up during recurring top ups. • PIN_FLD_TOPUP_INTERVAL specifies the interval between recurring top ups. • PIN_FLD_NEXT_TOPUP_T specifies the date on which to perform the next top up. • PIN_FLD_NUM_TOPUPS specifies the maximum number of top ups that can be done for recurring top ups. A value of 0 indicates that the recurring top up continues forever. • PIN_FLD_NUM_TOPUPS_DONE specifies the number of recurring top ups that have been completed. When this reaches PIN_FLD_NUM_TOPUPS, the recurring top up will be stopped. • PIN_FLD_STATUS specifies the object's status: active or closed. • PIN_FLD_BAL_GRP_OBJ specifies the balance group in which sub-balances need to be topped up. • PIN_FLD_TYPE specifies the top-up type: automatic (0), one-time (1), or recurring (2).

Table 8-2 (Cont.) Changed Storable Classes

Changed Storable Class	Description
/transition	(Patch Set 5) Modified to support proration of cycle fees during state changes. It contains a new PIN_FLD_PRORATION_FLAG field, which specifies how to apply cycle fees when customers transition from one package or bundle to another: prorate both the current and new cycle fees (1), apply the current bundle's fee to this cycle, and then start applying the new bundle's fee next cycle (2), or start applying the new bundle's fee to this cycle, and fully refund the current bundle's fee (4).
/config/glid	(Patch Set 4) Contains a new PIN_FLD_TYPE field that specifies the GLID type: Standard (0), A/R (1), Revenue (2), Contract (3).
/deal	(Patch Set 4) Contains new fields: <ul style="list-style-type: none"> • PIN_FLD_ATTRIBUTE_OBJ specifies the POID of the /offer_attribute_group object containing extended attributes for the /deal object • PIN_FLD_MODE, under the PIN_FLD_PRODUCTS and PIN_FLD_DISCOUNTS arrays, specifies what happens if customers purchase the same product in the deal more than once.
/discount	(Patch Set 4) Contains a new PIN_FLD_ATTRIBUTE_OBJ field that specifies the POID of the /offer_attribute_group object containing extended attributes for the /discount object.
/event/billing/charge/cc	(Patch Set 4) Modified to support Card Type Indicators (CTIs). It includes a new PIN_FLD_TYPE_STR field under PIN_FLD_CC_INFO, which is used to store the CTI returned by Paymentech.
/event/billing/product/action	(Patch Set 4) Contains these new fields: <ul style="list-style-type: none"> • PIN_FLD_CALENDAR_DOM specifies a specific calendar day to apply cycle, usage, and purchase fees. • PIN_FLD_MODE specifies what happens if customers purchase the same product more than once.
/event/billing/validate/cc	(Patch Set 4) Modified to support CTIs. It includes a new PIN_FLD_TYPE_STR field under PIN_FLD_CC_INFO, which is used to store the CTI returned by Paymentech.
/job_template	(Patch Set 4) Modified to support Business Operations Center on a multischema system. It contains a new PIN_FLD_SCHEMAS array for storing the schema number on which to run a job.
/plan	(Patch Set 4) Contains a new PIN_FLD_ATTRIBUTE_OBJ field that specifies the POID of the /offer_attribute_group object containing extended attributes for the /plan object.
/product	(Patch Set 4) Contains new fields: <ul style="list-style-type: none"> • PIN_FLD_ATTRIBUTE_OBJ specifies the POID of the /offer_attribute_group object containing extended attributes for the /product object • PIN_FLD_CALENDAR_DOM specifies a specific calendar day to apply cycle, usage, and purchase fees. • PIN_FLD_DATE_RANGE_TYPE specifies whether to select a rate for the product based on the event date, purchase date, or service instantiation date

Table 8-2 (Cont.) Changed Storable Classes

Changed Storable Class	Description
/purchased_product	(Patch Set 4) Contains new fields: <ul style="list-style-type: none"> • PIN_FLD_CALENDAR_DOM specifies a specific calendar day to apply cycle, usage, and purchase fees • PIN_FLD_MODE specifies what happens if customers purchase the same product more than once
/sponsorship	(Patch Set 4) Contains a new PIN_FLD_ATTRIBUTE_OBJ field that specifies the POID of the /offer_attribute_group object containing extended attributes for the /sponsorship object.
/event/billing/charge/cc	(Patch Set 2) Modified to support stored-credential transactions.
/event/billing/validate/cc	(Patch Set 2) Modified to support stored-credential transactions.
/event/customer/nameinfo	(Patch Set 2) Modified to store the POID of the /nameinfo object that changed and is populated for service-level contact information.
/job/boc	(Patch Set 2) Modified to run workflow jobs in Business Operations Center.
/job_template/billing	(Patch Set 2) Modified to run workflow jobs in Business Operations Center.
/job_template/collect	(Patch Set 2) Modified to support workflow jobs in Business Operations Center.
/job_template/invoice	(Patch Set 2) Modified to support workflow jobs in Business Operations Center.
/job_template/ledger_report	(Patch Set 2) Modified to support workflow jobs in Business Operations Center.
/payinfo/cc	(Patch Set 2) Modified to store card credentials for future transactions.
/rate	(Patch Set 2) Modified to support enhancements in BRM 12.0 Patch Set 2.
/service	(Patch Set 2) Modified to support enhancements in BRM 12.0 Patch Set 2.
/tmp_journals_to_process	(Patch Set 1) Modified to store information for the journals of the nonpaying child bill units to roll them up to the paying parent bill unit.

9

Event Notification Changes

The Oracle Communications Billing and Revenue Management (BRM) 12.0 Patch Sets include new and modified event notifications.

Topics in this document:

- [New Event Notifications](#)
- [Changed Event Notifications](#)

New Event Notifications

[Table 9-1](#) lists the event notifications that have been introduced in the BRM 12.0 patch set releases.

Table 9-1 New Event Notifications

New Event Notification	Description
<code>/event/notification/service/state_change/post_expiry</code>	(Patch Set 8) Generated when a service life cycle state has recently expired.
<code>/event/notification/service/state_change/pre_expiry</code>	(Patch Set 8) Generated when a service life cycle state is about to expire.
<code>/event/notification/balance/post_expiry</code>	(Patch Set 7) Generated a few days after a sub-balance has expired.
<code>/event/notification/billing/post_due</code>	(Patch Set 7) Generated a few days after a bill is due.
<code>/event/notification/collections_action/post_due</code>	(Patch Set 7) Generated a few days after a collections action is due.
<code>/event/notification/installment_schedule/post_due</code>	(Patch Set 7) Generated a few days after a pending installment payment is due.
<code>/event/notification/product/post_expiry</code>	(Patch Set 7) Generated a few days after a product has expired.
<code>/event/notification/subscription/post_renewal_due</code>	(Patch Set 7) Generated a few days after a subscription is due for renewal.
<code>/event/notification/billinfo/paytype/modify</code>	(Patch Set 6) Details about a payment type that was modified.
<code>/event/notification/billing/temp_limit</code>	(Patch Set 6) Details about a temporary credit limit that was added.
<code>/event/notification/billing/temp_limit/expiry</code>	(Patch Set 6) Details about a temporary credit limit that has expired.
<code>/event/notification/collections_action</code>	(Patch Set 6) Details about a collections action.
<code>/event/notification/collections_action/due</code>	(Patch Set 6) Details about a collections action that is due.
<code>/event/notification/installment_schedule</code>	(Patch Set 6) Details about an installment payment that is due.

Table 9-1 (Cont.) New Event Notifications

New Event Notification	Description
<code>/event/notification/installment_schedule/due</code>	(Patch Set 6) Details about an installment payment that is due or ending.
<code>/event/notification/minimum_amount</code>	(Patch Set 6) Details about a rental that is overridden by a minimum scaled amount during a credit limit breach.
<code>/event/notification/offering_override/create</code>	(Patch Set 6) Details about a product that was purchased with an overridden offer.
<code>/event/notification/offering_override/delete</code>	(Patch Set 6) Details about a product with an overridden offer that was deleted.
<code>/event/notification/offering_override/modify</code>	(Patch Set 6) Details about a product with an overridden offer that was modified.
<code>/event/notification/payment</code>	(Patch Set 6) Details about a payment that was made.
<code>/event/notification/payment/failed</code>	(Patch Set 6) Details about a payment that failed.
<code>/event/notification/request/create</code>	(Patch Set 6) Details about a <code>/request/failed</code> , <code>/request/failed/rest</code> , or <code>/request/failed/opcode</code> object that was created.
<code>/event/notification/threshold_loan</code>	(Patch Set 6) Details about a threshold that was breached.
<code>/event/notification/loan_reject</code>	(Patch Set 5) Details for rejecting a loan, including details of why it was rejected.
<code>/event/notification/loan_success</code>	(Patch Set 5) Details for successfully granting a loan, including the amount loaned.
<code>/event/notification/rental_failure</code>	(Patch Set 5) Details about a failed subscription.

Changed Event Notifications

Table 9-2 lists the event notifications that have been modified in the BRM 12.0 patch set releases.

Table 9-2 Changed Event Notifications

Changed Event Notifications	Description
<code>/event/notifications/balance/expiry</code>	(Patch Set 8) Modified to include these new fields: <ul style="list-style-type: none"> PIN_FLD_BAL_GRP_OBJ contains a link to the <code>/balance_group</code> object to which the notification is being sent. PIN_FLD_SERVICE_OBJ contains a link to the <code>/service</code> object to which the notification is being sent.
<code>/event/notification/balance/post_expiry</code>	(Patch Set 8) Modified to include these new fields: <ul style="list-style-type: none"> PIN_FLD_BAL_GRP_OBJ contains a link to the <code>/balance_group</code> object to which the notification is being sent. PIN_FLD_SERVICE_OBJ contains a link to the <code>/service</code> object to which the notification is being sent.
<code>/event/notification/billing/due</code>	(Patch Set 8) Modified to include these new fields: <ul style="list-style-type: none"> PIN_FLD_BILL_OBJ contains a link to the <code>/bill</code> object to which the notification is being sent.

Table 9-2 (Cont.) Changed Event Notifications

Changed Event Notifications	Description
/event/notification/billing/post_due	(Patch Set 8) Modified to include these new fields: <ul style="list-style-type: none"> PIN_FLD_BILL_OBJ contains a link to the /bill object to which the notification is being sent.
/event/notification/collections_action/due	(Patch Set 8) Modified to include these new fields: <ul style="list-style-type: none"> PIN_FLD_BILLINFO_OBJ contains a link to the /billinfo object to which the notification is being sent. PIN_FLD_ACTION_OBJ contains a link to the /collection_action object to which the notification is being sent.
/event/notification/collections_action/post_due	(Patch Set 8) Modified to include these new fields: <ul style="list-style-type: none"> PIN_FLD_BILLINFO_OBJ contains a link to the /billinfo object to which the notification is being sent. PIN_FLD_ACTION_OBJ contains a link to the /collection_action object to which the notification is being sent.
/event/notification/installment_schedule/due	(Patch Set 8) Modified to include these new fields: <ul style="list-style-type: none"> PIN_FLD_INSTALLMENT_OBJ contains a link to the /installment_schedule object to which the notification is being sent.
/event/notification/installment_schedule/post_due	(Patch Set 8) Modified to include these new fields: <ul style="list-style-type: none"> PIN_FLD_INSTALLMENT_OBJ contains a link to the /installment_schedule object to which the notification is being sent.
/event/notification/product.expiry	(Patch Set 8) Modified to include these new fields: <ul style="list-style-type: none"> PIN_FLD_SERVICE_OBJ contains a link to the /service object to which the notification is being sent.
/event/notification/product/post_expiry	(Patch Set 8) Modified to include these new fields: <ul style="list-style-type: none"> PIN_FLD_SERVICE_OBJ contains a link to the /service object to which the notification is being sent.
/event/notification/request/create	(Patch Set 8) Modified to include these new fields: <ul style="list-style-type: none"> PIN_FLD_OBJECT specifies the POID of the /request object recorded. PIN_FLD_EXTENDED_INFO substruct contains details about the recorded request. PIN_FLD_NOTIFICATION_SPEC array passes additional details with this request record in the notification event. This array and any field that it contains will not be persisted with the request object in database
/event/notification/subscription/post_renewal_due	(Patch Set 8) Modified to include these new fields: <ul style="list-style-type: none"> PIN_FLD_SERVICE_OBJ contains a link to the /service object, to which the subscription renewal post due notification is being sent.
/event/notification/subscription/renewal_due	(Patch Set 8) Modified to include these new fields: <ul style="list-style-type: none"> PIN_FLD_SERVICE_OBJ contains a link to the /service object, to which the subscription renewal due notification is being sent.

Table 9-2 (Cont.) Changed Event Notifications

Changed Event Notifications	Description
<code>/event/notification/subscription/renewal</code>	(Patch Set 6) Modified to include rollover details. It includes these new fields: <ul style="list-style-type: none">• CycleStateStr: Set to LastCycle when the subscription renewal has happened for the last time. Set to MiddleCycle when more subscription renewals are required.• DealName: Includes additional information about the deal of the renewed product.• RatePlanName: Includes additional information about the plan of the renewed product.• RolloverInfo: Includes the list of resources rolled over, with details such as amounts and validity dates.
<code>/event/notification/rental_failure</code>	(Patch Set 6) Modified to include these new fields: <ul style="list-style-type: none">• PIN_FLD_MODE: Indicates the option selected for exceeding the credit limit in PIN_FLD_TYPE field of the /rate object.• PIN_FLD_RETRY_AVAILABLE: Indicates how many more times the subscription can be retried. 0 indicates that the maximum has been reached.

10

Business Parameter Changes

The Oracle Communications Billing and Revenue Management (BRM) 12.0 Patch Sets include new business parameters.

Topics in this document:

- [New Accounts Receivable Business Parameter Entries](#)
- [New Billing Business Parameter Entries](#)
- [New Customer Business Parameter Entries](#)
- [New Invoicing Business Parameter Entries](#)
- [New Multibalance Business Parameter Entries](#)
- [New Notification Business Parameter Entries](#)
- [New Rating Business Parameter Entries](#)
- [New Subscription Business Parameter Entries](#)
- [New System Business Parameter Entries](#)

New Accounts Receivable Business Parameter Entries

[Table 10-1](#) lists the accounts receivable business parameter entries introduced in the BRM 12.0 patch set releases.

Table 10-1 New Entries in bus_params_AR.xml

Business Parameter	Description
Cvv2Required	(Patch Set 8) For Paymentech credit-card processor users, specifies whether to require credit-card verification (CVV) data for Visa card transactions as a method of fraud prevention. The default is disabled . Note: In Patch Set 7 and earlier releases, this functionality is controlled by using the cvv2_required CM pin.conf entry.
DDcollect	(Patch Set 8) Specifies whether to collect a customer's current balance during registration. The default is enabled . Note: In Patch Set 7 and earlier releases, this functionality is controlled by using the dd_collect CM pin.conf entry.

Table 10-1 (Cont.) New Entries in bus_params_AR.xml

Business Parameter	Description
DDRRevalidationInterval	(Patch Set 8) Specifies the time limit, in seconds, BRM waits before revalidating a customer's direct debit account during registration. BRM will not attempt to validate an account if a previous validation failed and the specified time has not elapsed. The default is 3600 (1 hour). Note: In Patch Set 7 and earlier releases, this functionality is controlled by using the dd_revalidation_interval CM pin.conf entry.
CCRRevalidationInterval	(Patch Set 8) Specifies the time limit, in seconds, BRM waits before revalidating a customer's credit card during registration. BRM will not attempt to validate an account if a previous validation failed and the specified time has not elapsed. The default is 3600 . Note: In Patch Set 7 and earlier releases, this functionality is controlled by using the cc_revalidation_interval CM pin.conf entry.
CCcollect	(Patch Set 8) Specifies whether to collect a customer's current balance during registration. The default is enabled . Note: In Patch Set 7 and earlier releases, this functionality is controlled by using the cc_collect CM pin.conf entry.
CidRequired	(Patch Set 7) Specifies whether credit-card verification (CID) data is required for American Express credit card transactions. The parameter values are enabled or disabled . The default is disabled .
PINlessDebitProcessing	(Patch Set 6) Specifies whether BRM can collect payments through PINless direct debit. The valid values are enabled and disabled . The default is disabled . See "Enabling PINless Debit Payments in BRM" in <i>BRM Configuring and Collecting Payments</i> .

New Billing Business Parameter Entries

Table 10-2 lists the billing business parameter entries introduced in the BRM 12.0 patch set releases.

Table 10-2 New Entries in bus_params_billing.xml

Business Parameter	Description
ApplyFolds	(Patch Set 8) Specifies whether to apply folds and enable fold calculation. The default is enabled . Note: In Patch Set 7 and earlier releases, this functionality is controlled by using the apply_folds CM pin.conf entry.

Table 10-2 (Cont.) New Entries in bus_params_billing.xml

Business Parameter	Description
ApplyRollover	(Patch Set 8) Specifies whether to apply rollovers. The default is enabled . Note: In Patch Set 7 and earlier releases, this functionality is controlled by using the apply_rollover CM pin.conf entry.
CancelTolerance	(Patch Set 8) Specifies the cancellation tolerance for account products, in minutes. This is the time after a product is purchased when it can be canceled with a full refund. For example, this tolerance is needed when a CSR assigns the wrong product to a customer and needs to cancel it. The default is 15 . Note: In Patch Set 7 and earlier releases, this functionality is controlled by using the cancel_tolerance CM pin.conf entry.
ItemizedTaxCalculation	(Patch Set 6) Specifies whether to retrieve the total tax and tax due at each item level. The possible values are enabled and disabled . The default is disabled .

New Customer Business Parameter Entries

Table 10-3 lists the billing business parameter entries introduced in the BRM 12.0 patch set releases.

Table 10-3 New Entries in bus_params_customer.xml

Business Parameter	Description
ActgDom	(Patch Set 8) Specifies the default day of the month for billing accounts. The default is 0 . Note: In Patch Set 7 and earlier releases, this functionality is controlled by using the actg_dom CM pin.conf entry.
BillWhen	(Patch Set 8) Specifies the default number of accounting cycles before a customer is billed. The default is 1 . Note: In Patch Set 7 and earlier releases, this functionality is controlled by using the bill_when CM pin.conf entry.
CCchecksum	(Patch Set 8) Specifies whether to run a checksum validation on the customer's credit card during registration. The default is enabled . Note: In Patch Set 7 and earlier releases, this functionality is controlled by using the cc_checksum CM pin.conf entry.

New Invoicing Business Parameter Entries

Table 10-4 lists the invoicing business parameter entries introduced in the BRM 12.0 patch set releases.

Table 10-4 New Entries in bus_params_invoicing.xml

Business Parameter	Description
DisableFilterItemByPruneTime	(Patch Set 8) Specifies whether to filter invoice items by pruning time. The valid values are: <ul style="list-style-type: none"> enabled: Items are not filtered by pruning time. disabled: Items are filtered by pruning time. This is the default.

New Multibalance Business Parameter Entries

Table 10-5 lists the multibalance business parameter entries introduced in the BRM 12.0 patch set releases.

Table 10-5 New Entries in bus_params_multi_bal.xml

Business Parameter	Description
ConsumeSubType	(Patch Set 6) Specifies the order of balance consumption processing. The valid values are: <ul style="list-style-type: none"> main: Main balance consumption processing. loan: Loan balance consumption process. This is the default.

New Notification Business Parameter Entries

Patch Set 8 introduces a new **notification** instance of the `/config/business_params` object. You edit the notification-related business parameters by using the `bus_params_notification.xml` file.

Table 10-6 lists the notification business parameter entries introduced in the BRM 12.0 patch set releases.

Table 10-6 New Entries in bus_params_notification.xml

Business Parameter	Description
KafkaDBNumber	(Patch Set 8) The KafkaDBNumber for which the DynamicKey is populated. The default is 0.0.9.6 / 0 .
KafkaDynamicKeyEnabled	(Patch Set 8) Specifies whether the Kafka Dynamic Key is enabled or not. If enabled and KafkaDBNumber matches the POID, the PIN_FLD_NOTIFICATION_KEY_INFO field is added to the PCM_OP_PUBLISH_POL_PREP_EVENT input flist. The default is disabled .

New Rating Business Parameter Entries

Table 10-7 lists the rating business parameter entries introduced in the BRM 12.0 patch set releases.

Table 10-7 New Entries in bus_params_rating.xml

Business Parameter	Description
ExtraRateFlags	(Patch Set 8) Allows you to turn on or off rating features, such as credit floor checking. For information, see "Setting Optional Rating Flags" in <i>BRM Configuring Pipeline Rating and Discounting</i> . The default is 0 . Note: In Patch Set 7 and earlier releases, this functionality is controlled by using the extra_rate_flags CM pin.conf entry.
ResetMemberCreditLimit	(Patch Set 7) Specifies whether to set a member account's credit limit to zero when rolling up balances to a parent account. The valid values are enabled and disabled . The default is enabled .
SharingGroupsEnabled	(Patch Set 7) Specifies whether the sharing groups feature is enabled. The valid values are enabled and disabled . The default is enabled .
ApplyDiscountOnZeroCharge	(Patch Set 5) Specifies whether to apply discounts when the product scale is 0. The valid values are: <ul style="list-style-type: none"> disabled: Rating skips the pipeline call so that discounts are not processed if the product scale is 0 enabled: Discounts are applied. This is the default.

New Subscription Business Parameter Entries

Table 10-8 lists the subscription business parameter entries introduced in the BRM 12.0 patch set releases.

Table 10-8 New Entries in bus_params_subscription.xml

Business Parameter	Description
CdcLineCancelDayInclude	(Patch Set 8) Specifies whether the day on which a line was canceled should be counted when decrementing contract days. The default is 1 . Note: In Patch Set 7 and earlier releases, this functionality is controlled by using the cdc_line_cancel_day_include CM pin.conf entry.
CdcLineCreateDayInclude	(Patch Set 8) Specifies whether the day on which a line was created should be counted when incrementing contract days. The default is 1 . Note: In Patch Set 7 and earlier releases, this functionality is controlled by using the cdc_line_create_day_include CM pin.conf entry.

Table 10-8 (Cont.) New Entries in bus_params_subscription.xml

Business Parameter	Description
EventFetchSize	<p>(Patch Set 8) Size used for step search while searching events in rerating/rebilling. This value can be changed as per the required number of events to be searched. For more information, see "Setting the Rerating Event Cache Size (Fetch Size)" in <i>BRM Rerating Events</i>.</p> <p>The default is 10000.</p> <p>Note: In Patch Set 7 or earlier releases, use the event_fetch_size CM pin.conf entry instead.</p>
KeepCancelledProductsOrDiscounts	<p>(Patch Set 8) Specifies whether to automatically delete / purchased_product objects when a charge offer is canceled.</p> <p>The default is enabled.</p> <p>Note: In Patch Set 7 and earlier releases, this functionality is controlled by using the keep_cancelled_products_or_discounts CM pin.conf entry.</p>
NumBillingCycles	<p>(Patch Set 8) Specifies the maximum number of billing cycles allowed between the current time and the backdated event date for rerating events.</p> <p>The default is 1.</p> <p>Note: In Patch Set 7 and earlier releases, this functionality is controlled by using the num_billing_cycles CM pin.conf entry.</p>
PropagateDiscount	<p>(Patch Set 8) Specifies whether to immediately propagate shared discounts when a new discount is added or removed from the group or a member subscribes or unsubscribes to the group. For more information, see "Configuring the Start and End Times for Discount Sharing" in <i>BRM Managing Customers</i>.</p> <p>The default is disabled.</p> <p>Note: In Patch Set 7 and earlier releases, use the propagate_discount CM pin.conf entry instead.</p>
RateChange	<p>(Patch Set 8) Specifies whether to enable the enhanced rate change management feature. When enabled, rate changes for cycle events that occur in the middle of a cycle will be prorated using both the old and new rate. When disabled, only the old rate will be applied.</p> <p>The default is disabled.</p> <p>Note: In Patch Set 7 and earlier releases, use the rate_change CM pin.conf entry instead.</p>
TimestampRoundingDuringApplyLoan	<p>(Patch Set 8) Specifies whether to round the timestamps of loan sub-balance buckets when applying loans.</p> <p>The default is disabled.</p>
ReduceLoanBalDuringRecovery	<p>(Patch Set 7) Specifies whether the loan balance should be reduced during recovery. The valid values are enabled (1) and disabled (0).</p> <p>The default value is 0.</p>

Table 10-8 (Cont.) New Entries in bus_params_subscription.xml

Business Parameter	Description
BackdateTriggerAutoRerate	<p>(Patch Set 7) Specifies whether to automatically create rerate objects for backdated actions, so that the pin_rerate utility can determine which events need rerating. The valid values are:</p> <ul style="list-style-type: none"> • enabled: Specifies to automatically create rerate objects for backdated actions. The pin_rerate utility can look for these objects and then use them to determine which events need rerating and how far back to go. • disabled: Specifies to not automatically create rerate objects for backdated actions. This is the default.
AutomatedGroupSharingSetup	<p>(Patch Set 7) Specifies whether to automatically create a discount sharing group or product sharing group for the top-level parent or billing account. The valid values are:</p> <ul style="list-style-type: none"> • 0: Disabled. This is the default. • 1: Automatic discount sharing groups are enabled. • 2: Automatic product sharing groups are enabled. • 3: Both automatic discount sharing groups and automatic product sharing groups are enabled.
ServiceTransferEnabled	<p>(Patch Set 7) Specifies whether the service transfer feature is enabled. The valid values are enabled and disabled. The default is enabled.</p>
ApplyChargeOnInactiveOrCancelProduct	<p>(Patch Set 6) Specifies whether to apply charges to inactive or canceled products. The valid values are enabled and disabled. The default is disabled. See "Specifying Whether to Charge Inactive, Canceled, or Suspended Active Accounts" in <i>PDC Creating Product Offerings</i>.</p>
ApplyRolloverBeforeCycleFees	<p>(Patch Set 6) Specifies whether PCM_OP_SUBSCRIPTION_CYCLE_FORWARD adds rollover details to /event/notification/subscription/renewal notification events. The valid values are enabled and disabled. The default is disabled. See "Adding Rollover Details to Subscription Renewals" in <i>BRM Managing Customers</i>.</p>
LoanRepaymentPercent	<p>(Patch Set 6) In cases where customers make an underpayment, specifies the percentage of the payment that is applied to the loan repayment. Valid values are 0 through 100, where 0 specifies that no amount is applied to the loan repayment and 100 specifies that the entire amount is applied to the loan repayment. The default is 100.</p>
CreateRerateJobDuringCancel	<p>(Patch Set 5) Specifies whether to automatically create rerate jobs when a product with noncurrency resources is canceled. The valid values are:</p> <ul style="list-style-type: none"> • 0: Disabled. This is the default. • 1: Enabled. <p>See "Enabling Rerating for Canceled Noncurrency Resources" in <i>BRM Rerating Events</i>.</p>
DefaultZoneMapName	<p>(Patch Set 5) Specifies the name of the default zone map name to use for location-based operations. The default is DefaultZonemap.</p>

Table 10-8 (Cont.) New Entries in bus_params_subscription.xml

Business Parameter	Description
MaxRetryCount	<p>(Patch Set 5) When a customer's subscription fails due to insufficient credit, the subscription can be retried your specified number of times before sending a failure notification to an external system.</p> <p>The default is 0, which sends the failure notification immediately without retrying.</p> <p>See "Configuring the Maximum Number of Subscription Retries" in <i>PDC Creating Product Offerings</i>.</p>

New System Business Parameter Entries

Table 10-9 lists the system business parameter entries introduced in the BRM 12.0 patch set releases.

Table 10-9 New Entries in bus_params_system.xml

Business Parameter	Description
CrossSchemaSharingGroup	<p>(Patch Set 8) Specifies whether the parent and member accounts in a sharing group can reside in different database schemas. See "Enabling Group Members to Reside in Multiple Schemas" in <i>BRM Managing Customers</i> for more information.</p> <p>The default is disabled.</p>
PrepaidEngine	<p>(Patch Set 7) Specifies the rating engine used for prepaid events. The valid values are:</p> <ul style="list-style-type: none"> 0: The BRM rating engine is used. This is the default. 1: The ECE rating engine is used.
AcceptableDelayTime	<p>(Patch Set 6) Specifies how long after the delivery time that messages can be delivered to your customers through an external notification application. The default is 2 hours.</p> <p>See "Setting System-Wide Values for Notifications" in <i>BRM Managing Customers</i>.</p>
EnableExternalIds	<p>(Patch Set 6) Specifies whether to generate unique IDs for <i>/event</i>, <i>/balance_group</i>, <i>/billinfo</i>, and <i>/payinfo</i> objects. A value of 1 enables the generation of unique IDs, and 0 disables it. The default is 0.</p>
SilentDaysCalendarName	<p>(Patch Set 6) Specifies the name of the calendar for determining the list of silent days. Notifications cannot be delivered to your customers on silent days. The default is NotificationSilentDays.</p> <p>See "Setting System-Wide Values for Notifications" in <i>BRM Managing Customers</i>.</p>
NotificationSilentPeriod	<p>(Patch Set 5) Specifies the system-wide silent period during which messages cannot be sent to customers. The default is 21:00 through 07:00.</p> <p>See "Setting System-Wide Values for Notifications" in <i>BRM Managing Customers</i>.</p>

Table 10-9 (Cont.) New Entries in bus_params_system.xml

Business Parameter	Description
NotificationSubscriberPreferences	(Patch Set 5) Specifies the custom fields to add to notifications during the enrichment process. The default is empty. See "Adding Custom Fields during Enrichment" in <i>BRM Managing Customers</i> .

11

Schema and Index Changes

The Oracle Communications Billing and Revenue Management (BRM) 12.0 Patch Sets include new and changed schemas and indexes.

- [Patch Set 8 Schema and Index Changes](#)
- [Patch Set 7 Schema and Index Changes](#)
- [Patch Set 6 Schema and Index Changes](#)
- [Patch Set 5 Schema and Index Changes](#)
- [Patch Set 4 Schema and Index Changes](#)
- [Patch Set 3 Schema and Index Changes](#)
- [Patch Set 2 Schema and Index Changes](#)
- [Patch Set 1 Schema and Index Changes](#)

Patch Set 8 Schema and Index Changes

BRM 12.0 Patch Set 8 includes the following schema and index changes.



Note:

No views were added in BRM 12.0 Patch Set 8.

The following tables have been added:

- THREAD_INFO_FAILED_ACCT_T

The following indexes have been added:

- I_EVT_CONTRACT_CANCEL__ID

[Table 11-1](#) lists the columns that have been added.

Table 11-1 Columns Added

Table	Column Added
AU_CONFIG_BUSINESS_EVENTS_T	NOTIFY_SEARCH_LEVEL NOTIFY_OPCODE
AU_PROC_AUD_BILL_ERR_ACCT_T	SCOPE_OBJ_DB SCOPE_OBJ_ID0 SCOPE_OBJ_TYPE SCOPE_OBJ_REV

Table 11-1 (Cont.) Columns Added

Table	Column Added
AU_PRODUCT_T	OFFER_VALIDITY_ROUNDING SCALE_ROUNDING
AU_SERVICE_T	SVC_EXP_LAST_NOTI_T SVC_EXP_LAST_NOTI_OFFSET
CONFIG_BUSINESS_EVENTS_T	NOTIFY_SEARCH_LEVEL NOTIFY_OPCODE
CONFIG_DISTRIBUTION_T	SCHEMA_NAME
CONFIG_NOTIFICATION_SPEC_T	NOTIFY_OPT_TYPE AGGREGATE_MODE
PROC_AUD_BILL_ERR_ACCT_T	SCOPE_OBJ_DB SCOPE_OBJ_ID0 SCOPE_OBJ_TYPE SCOPE_OBJ_REV
PRODUCT_T	OFFER_VALIDITY_ROUNDING SCALE_ROUNDING
SERVICE_T	SVC_EXP_LAST_NOTI_T SVC_EXP_LAST_NOTI_OFFSET

Patch Set 7 Schema and Index Changes

BRM 12.0 Patch Set 7 includes the following schema and index changes.



Note:

No views were added in BRM 12.0 Patch Set 7.

The following tables have been added:

- CONFIG_STORED_PROCEDURE_T
- CONFIG_PROMISE_TO_PAY_SPEC_T
- GROUP_SHARING_PRODUCTS_T
- AU_GROUP_SHARING_PRODUCTS_T
- EVENT_GROUP_SHARE_PRODUCTS_T

The following indexes have been added:

- I_PAYINFO_UNIQUE_ID
- I_GROUP_SHARING_PRODUCTS_ID
- I_EVENT_GRP_SHARE_PROD_ID

[Table 11-2](#) lists the columns that have been added.

Table 11-2 Columns Added

Table	Column Added
AU_BAL_GRP_BALS_T	ROLLED_UP_CREDIT_LIMIT
AU_BAL_INFO_LIMIT_T	DYNAMIC_CREDIT_FLOOR OVERRIDE_CREDIT_LIMIT
AU_CFG_CREDIT_PROFILE_T	DYNAMIC_CREDIT_FLOOR OVERRIDE_CREDIT_LIMIT
AU_DEAL_PRODUCTS_T	BASE_PRODUCT_OBJ_ID0 BASE_PRODUCT_OBJ_TYPE BASE_PRODUCT_OBJ_REV BASE_PRODUCT_OBJ_DB FLAGS
AU_DISCOUNT_T	APPLY_MODE
AU_PLAN_LIMIT_T	DYNAMIC_CREDIT_FLOOR OVERRIDE_CREDIT_LIMIT
AU_PRODUCT_T	APPLY_MODE
AU_PURCHASED_DEPOSIT_T	END_DETAILS PARENT_DB PARENT_ID0 PARENT_TYPE PARENT_REV
AU_PURCHASED_PRODUCT_T	BASE_PRODUCT_OBJ_ID0 BASE_PRODUCT_OBJ_TYPE BASE_PRODUCT_OBJ_REV BASE_PRODUCT_OBJ_DB
BAL_GRP_BALS_T	ROLLED_UP_CREDIT_LIMIT
BAL_INFO_LIMIT_T	DYNAMIC_CREDIT_FLOOR OVERRIDE_CREDIT_LIMIT
CFG_CREDIT_PROFILE_T	DYNAMIC_CREDIT_FLOOR OVERRIDE_CREDIT_LIMIT
COLL_SCEN_PROMISETOPAY_T	CONFIG_OBJ_DB CONFIG_OBJ_ID0 CONFIG_OBJ_TYPE CONFIG_OBJ_REV STATUS
CONFIG_PROMISE_TO_PAY_SPEC_T	MIN_PERC_THRESHOLD
DEAL_PRODUCTS_T	BASE_PRODUCT_OBJ_ID0 BASE_PRODUCT_OBJ_TYPE BASE_PRODUCT_OBJ_REV BASE_PRODUCT_OBJ_DB FLAGS
DISCOUNT_T	APPLY_MODE

Table 11-2 (Cont.) Columns Added

Table	Column Added
EVENT_ACT_COLL_P2P_UNEQUAL_T	MILESTONE_AMOUNT MILESTONE_PERCENTAGE MILESTONE_INTERVAL
EVENT_BILLING_LIMIT_T	DYNAMIC_CREDIT_FLOOR OVERRIDE_CREDIT_LIMIT
EVENT_BILLING_PRODUCT_ACTION_T	BASE_PRODUCT_OBJ_ID0 BASE_PRODUCT_OBJ_TYPE BASE_PRODUCT_OBJ_REV BASE_PRODUCT_OBJ_DB
PLAN_LIMIT_T	DYNAMIC_CREDIT_FLOOR OVERRIDE_CREDIT_LIMIT
PRODUCT_T	APPLY_MODE
PURCHASED_DEPOSIT_T	END_DETAILS PARENT_DB PARENT_ID0 PARENT_TYPE PARENT_REV
PURCHASED_PRODUCT_T	BASE_PRODUCT_OBJ_ID0 BASE_PRODUCT_OBJ_TYPE BASE_PRODUCT_OBJ_REV BASE_PRODUCT_OBJ_DB

Patch Set 6 Schema and Index Changes

BRM 12.0 Patch Set 6 includes the following schema and index changes.



Note:

No views were added in BRM 12.0 Patch Set 6.

The following tables have been added:

- CONFIG_DELIVERY_METHODS_T
- CFG_NOTIF_SPEC_ADV_NOTIF_T
- CFG_NOTIF_SPEC_CRITERIA_T
- CFG_NOTIF_SPEC_DELIV_METHODS_T
- CFG_NOTIF_SPEC_DELIVERY_SPEC_T
- CONFIG_NOTIFICATION_SPEC_T
- CONFIG_DEPOSIT_SPEC_PROFILE_T

- DEPOSIT_SPECIFICATION_T
- DEPOSIT_REFUND_REQUEST_T
- EVENT_DEPOSIT_DETAILS_T
- EVENT_DEPOSIT_RECEIVE_T
- EVENT_DEPOSIT_REFUND_T
- EVENT_DEPOSIT_RELEASE_T
- EVENT_DEPOSIT_TRANSFER_T
- EVENT_DEPOSIT_UPDATE_DETAILS_T
- PURCHASED_DEPOSIT_EVENT_T
- PURCHASED_DEPOSIT_T
- REQUEST_T
- REQUEST_FAILED_T
- REQUEST_FAILED_OPCODE_T
- REQUEST_FAILED_ERROR_INFO_T
- REQUEST_FAILED_REST_T
- REQUEST_FAILED_HEADER_INFO_T
- EVENT_BILLING_LOAN_LATE_FEE_T
- EVENT_BILLING_LOAN_PULLBACK_T
- EVENT_BILLING_LOAN_RECOVERY_T
- EVENT_BILLING_LOAN_REC_REV_T
- EVENT_BILLING_LIMIT_T
- CONFIG_INSTL_SCH_CRITERIA_T
- CONFIG_INSTL_SCH_SPEC_T
- CONFIG_INSTL_NEXT_STATE_T
- CONFIG_INSTL_STATE_T
- EVT_BILLING_INSTL_T
- EVENT_INSTL_CANCEL_T
- EVENT_INSTL_CREATE_T
- INSTALLMENT_SOURCE_T
- INSTALLMENT_SCH_INSTALLMENT_T
- INSTALLMENT_SCHEDULE_T

The following indexes have been added:

- I_REQUEST__ID
- I_REQUEST_ACC_ID
- I_REQUEST_STATUS__ID
- I_REQUEST_PROGRAM_NAME__ID

[Table 11-3](#) lists the columns that have been added.

Table 11-3 Columns Added

Table	Column Added
PROFILE_LOAN_T	PULLBACK REPAYMENT_DAYS MAX_ACTIVE_LOANS OPT_LOAN
PROFILE_LOAN_LIMIT_T	CREDIT_LIMIT VALID_TO VALID_FROM
EVENT_BILLING_LOAN_FEE_T	AMOUNT
PURCHASED_PRODUCT_T	NEXT_RETRY_T
RATE_BAL_IMPACTS_T	MINIMUM_SCALED_AMOUNT
RATE_T	PRORATE_QUANTITY TYPE
PRODUCT_T	RETRY_UNIT RETRY_OFFSET MAX_RETRY_COUNT
EVENT_BILLING_PRODUCT_ACTION_T	NEXT_RETRY_T
ITEM_T	TAX_DATA
EVENT_BILLING_TAXES_T	ITEM_OBJ_DB ITEM_OBJ_ID0 ITEM_OBJ_TYPE ITEM_OBJ_REV
CONFIG_DELIVERY_METHODS_T	DELIVERY_IDENTIFIER_TAG
BAL_GRP_SUB_BALS_T	SUBTYPE
EVENT_BILLING_LOAN_DEBIT_T	CHANNEL TYPE
CFG_CREDIT_PROFILE_T	LOAN_THRESHOLDS_FIXED LOAN_THRESHOLDS
PLAN_LIMIT_T	LOAN_THRESHOLDS_FIXED LOAN_THRESHOLDS
BAL_INFO_LIMIT_T	LOAN_THRESHOLDS_FIXED LOAN_THRESHOLDS
PROFILE_LOAN_T	NO_OF_ACTIVE_LOANS UNIT FREQUENCY
COLLECTIONS_ACTION_T	LAST_NOTIFICATION_OFFSET LAST_NOTIFICATION_T
INSTALLMENT_SCH_INSTALLMENT_T	LAST_NOTIFICATION_OFFSET
INSTALLMENT_SCHEDULE_T	LAST_NOTIFICATION_OFFSET

Table 11-3 (Cont.) Columns Added

Table	Column Added
EVENT_BILLING_FEE_TRANS_INFO_T	ZONEMAP_NAME ZONEMAP_TARGET LOCALE

Patch Set 5 Schema and Index Changes

BRM 12.0 Patch Set 5 includes the following schema and index changes.

The following tables have been added:

- BATCH_REL_RETRY_T
- BATCH_REL_ZIP_FILE_BUF_T
- BATCH_REL_ZIP_FILE_T
- BILL_STRUCTURE_T
- CFG_NOTIF_SPEC_ADV_NOTIF_T
- CFG_NOTIF_SPEC_CRITERIA_T
- CFG_NOTIF_SPEC_DELIV_METHODS_T
- CFG_NOTIF_SPEC_DELIVERY_SPEC_T
- CONFIG_DEPOSIT_SPEC_PROFILE_T
- CONFIG_EVENT_PROMO_MAP_T
- CONFIG_EVENT_PROMO_TAG_MAP_T
- CONFIG_INSTL_SCH_CRITERIA_T
- CONFIG_INSTL_SCH_SPEC_T
- CONFIG_INSTL_STATE_T
- CONFIG_LOAN_LOC_T
- CONFIG_NOTI_DELIVERY_METHODS_T
- CONFIG_NOTIFICATION_SPEC_T
- CONFIG_TAXCODES_MAP_T
- CONFIG_TAXEXEMPTION_CODE_MAP_T
- DEPOSIT_SPECIFICATION_T
- EVENT_AUDIT_TRANS_OUT_T
- TAX_SELECTOR_T
- EVENT_BILLING_FEE_TRANS_INFO_T
- EVENT_BILLING_LOAN_GRANT_T
- EVENT_INSTL_CANCEL_T
- EVENT_INSTL_CREATE_T

- EVT_BILLING_INSTL_T
- INSTALLMENT_SCHEDULE_T
- INSTALLMENT_SOURCE_T
- INSTALLMENT_SCH_INSTALLMENT_T
- NO_DB_IMPACTS
- OFFERING_OVERRIDE_T
- PROFILE_LOAN_T
- PROFILE_SPECIALDATES_T
- TAX_EXEMPTION_SELECTOR_T

The following indexes have been added:

- I_BATCH_REL_RETRY_ID
- I_BATCH_REL_RETRY_ID
- I_BATCH_REL_RETRY_READY
- I_BATCH_REL_ZIP_OBJ_ID0
- I_BATCH_REL_ZIP_OBJ_ID0
- I_BATCH_REL_ZIP_BUF_OBJ_ID0
- I_BATCH_REL_ZIP_BUF_OBJ_ID0
- I_BATCH_REL_ZIP_INCOMPLETE
- I_BILL_BILLINFO_ID
- I_CONFIG_INSTL_SCH_CRIT_VALUE
- I_CONFIG_INSTL_SCH_SPEC_NAME
- I_TOPUP_AUTH_ID

The following views have been added:

- BALANCE_BUCKET_ACCOUNT_V
- BALANCE_BUCKET_V
- BALANCE_BUCKET_SERVICE_ALIAS_V
- BALANCE_BUCKET_SERVICE_V

[Table 11-4](#) lists the columns that have been added.

Table 11-4 Columns Added

Table	Column Added
BAL_GRP_T	UNIQUE_ID
BAL_GRP_BALS_T	DELTA_CREDIT_LIMIT OUTSTANDING_AMOUNT
BAL_GRP_SUB_BALS_T	GRANTED_BAL LAST_NOTIFICATION_OFFSET UNIQUE_ID

Table 11-4 (Cont.) Columns Added

Table	Column Added
BATCH_REL_ZIP_FILE_T	FILENAME NUM_RECORDS STATUS
BILL_T	LAST_NOTIFICATION_OFFSET STATE
BILLINFO_T	UNIQUE_ID
CFG_CREDIT_PROFILE_T	GRANT_FIXED_THRESHOLD GRANT_PERC_THRESHOLD
CONFIG_BEID_BALANCES_T	ENABLE_EXPIRY_NOTIFICATION
CONFIG_BUSINESS_PARAMS_T	PARAM_VALUE
DEAL_PRODUCTS_T	GRACE_PERIOD RENEWAL_MODE
EVENT_BAL_IMPACTS_T	OUTSTANDING_AMOUNT TAX_EXEMPTION_CODE
EVENT_BILLING_CHARGE_CC_T	CARD_TYPE
EVENT_BILLING_DISC_ACTION_T	OFFERING_OVERRIDE_VALUES_OBJ_DB OFFERING_OVERRIDE_VALUES_OBJ_ID0 OFFERING_OVERRIDE_VALUES_OBJ_REV OFFERING_OVERRIDE_VALUES_OBJ_TYPE
EVENT_BILLING_LIMIT_T	GRANT_FIXED_THRESHOLD GRANT_PERC_THRESHOLD
EVENT_BILLING_PAYMENT_CC_T	TRACE_NUM AUTH_MOP
EVENT_BILLING_PRODUCT_ACTION_T	GRACE_PERIOD LAST_NOTIFICATION_T LAST_NOTIFICATION_OFFSET OFFERING_OVERRIDE_VALUES_OBJ_DB OFFERING_OVERRIDE_VALUES_OBJ_ID0 OFFERING_OVERRIDE_VALUES_OBJ_TYPE OFFERING_OVERRIDE_VALUES_OBJ_REV RENEWAL_MODE RETRY_COUNT
EVENT_BILLING_TAXES_T	TAX_EXEMPTION_CODE

Table 11-4 (Cont.) Columns Added

Table	Column Added
EVENT_BILLING_TOPUP_T	AUTH_ID BAL_GRP_OBJ_DB BAL_GRP_OBJ_ID0 BAL_GRP_OBJ_TYPE BAL_GRP_OBJ_REV SUB_BAL_ID TOPUP_TYPE
HISTORY_BILLS_T	LAST_NOTIFICATION_OFFSET
JOURNAL_T	TAX_EXEMPTION_CODE
PAYINFO_T	UNIQUE_ID
PAYINFO_CC_T	CARD_TYPE TYPE_STR
PRODUCT_T	ENABLE_NOTIFICATION
PURCHASED_DISCOUNT_T	OFFERING_OVERRIDE_VALUES_OBJ_DB OFFERING_OVERRIDE_VALUES_OBJ_ID0 OFFERING_OVERRIDE_VALUES_OBJ_REV OFFERING_OVERRIDE_VALUES_OBJ_T TYPE
PURCHASED_PRODUCT_T	GRACE_PERIOD LAST_NOTIFICATION_OFFSET OFFERING_OVERRIDE_VALUES_OJB_DB OFFERING_OVERRIDE_VALUES_OBJ_ID0 OFFERING_OVERRIDE_VALUES_OBJ_REV OFFERING_OVERRIDE_VALUES_OBJ_T TYPE RENEWAL_MODE RETRY_COUNT
RATE_T	PRORATION_FLAG
RATE_BAL_IMPACTS_T	FIXED_PRICE_TAG SCALED_PRICE_TAG SPLIT_DETAILS
RATE_PLAN_T	SELECTOR_OBJ_ID0 SELECTOR_OBJ_DB SELECTOR_OBJ_REV SELECTOR_OBJ_TYPE
TMP_JOURNALS_TO_PROCESS_T	TAX_EXEMPTION_CODE
TOPUP_T	TOPUP_TYPE
TRANSITION_T	PRORATION_FLAG

Patch Set 4 Schema and Index Changes

BRM 12.0 Patch Set 4 includes the following schema and index changes:

- The following tables are added:
 - EVENT_ACTIVITY_JOB_REQ_T
 - PERFORMANCE_SNAPSHOT_T
 - PERFORMANCE_RESULTS_T
 - PERFORMANCE_SQL_DATA_T
 - OFFER_ATTRIBUTE_GROUP_T
 - OFFER_ATTRIBUTE_T
 - JOB_TEMPLATE_SCHEMAS_T
 - AU_JOB_TEMPLATE_SCHEMAS_T
 - EVENT_BILLING_TOPUP_T
 - EVENT_AUDIT_TRANS_INFO_T
 - BRM_PS_T
- The following column is added in the EVENT_BILLING_CHARGE_CC_T table:
 - TYPE_STR
- The following column is added in the EVENT_BILLING_VALID_CC_T table:
 - TYPE_STR
- The following column is added in the EVENT_BILLING_DISC_ACTION_T table:
 - PURCHASE_MODE
- The following columns are added in the EVENT_BILLING_PRODUCT_ACTION_T table:
 - PURCHASE_MODE
 - CALENDAR_DOM
- The following columns are added in the PRODUCT_T table:
 - CALENDAR_DOM
 - DATE_RANGE_TYPE
 - ATTRIBUTE_OBJ_REV
 - ATTRIBUTE_OBJ_TYPE
 - ATTRIBUTE_OBJ_ID0
 - ATTRIBUTE_OBJ_DB
- The following columns are added in the DISCOUNT_T table:
 - ATTRIBUTE_OBJ_REV
 - ATTRIBUTE_OBJ_TYPE
 - ATTRIBUTE_OBJ_ID0
 - ATTRIBUTE_OBJ_DB

- The following columns are added in the PLAN_T table:
 - ATTRIBUTE_OBJ_REV
 - ATTRIBUTE_OBJ_TYPE
 - ATTRIBUTE_OBJ_ID0
 - ATTRIBUTE_OBJ_DB
- The following columns are added in the DEAL_T table:
 - ATTRIBUTE_OBJ_REV
 - ATTRIBUTE_OBJ_TYPE
 - ATTRIBUTE_OBJ_ID0
 - ATTRIBUTE_OBJ_DB
- The following columns are added in the PURCHASED_PRODUCT_T table:
 - CALENDAR_DOM
 - PURCHASE_MODE
- The following column is added in the DEAL_PRODUCTS_T table:
 - PURCHASE_MODE
- The following column is added in the DEAL_DISCOUNTS_T table:
 - PURCHASE_MODE
- The following column is added in the PURCHASED_DISCOUNT_T table:
 - PURCHASE_MODE
- The following columns are added in the AU_PLAN_T table:
 - ATTRIBUTE_OBJ_REV
 - ATTRIBUTE_OBJ_TYPE
 - ATTRIBUTE_OBJ_ID0
 - ATTRIBUTE_OBJ_DB
- The following columns are added in the AU_DISCOUNT_T table:
 - ATTRIBUTE_OBJ_REV
 - ATTRIBUTE_OBJ_TYPE
 - ATTRIBUTE_OBJ_ID0
 - ATTRIBUTE_OBJ_DB
- The following columns are added in the AU_PRODUCT_T table:
 - ATTRIBUTE_OBJ_REV
 - ATTRIBUTE_OBJ_TYPE
 - ATTRIBUTE_OBJ_ID0
 - ATTRIBUTE_OBJ_DB
 - DATE_RANGE_TYPE
 - CALENDAR_DOM
- The following columns are added in the AU_DEAL_T table:

- ATTRIBUTE_OBJ_REV
- ATTRIBUTE_OBJ_TYPE
- ATTRIBUTE_OBJ_ID0
- ATTRIBUTE_OBJ_DB
- The following column is added in the AU_PURCHASED_DISCOUNT_T table:
 - PURCHASE_MODE
- The following column is added in the AU_DEAL_DISCOUNTS_T table:
 - PURCHASE_MODE
- The following column is added in the AU_DEAL_PRODUCTS_T table:
 - PURCHASE_MODE
- The following columns are added in the AU_PURCHASED_PRODUCT_T table:
 - PURCHASE_MODE
 - CALENDAR_DOM
- The following columns are added in the AU_SPONSORSHIP_T table:
 - ATTRIBUTE_OBJ_REV
 - ATTRIBUTE_OBJ_TYPE
 - ATTRIBUTE_OBJ_ID0
 - ATTRIBUTE_OBJ_DB
- The following columns are added in the TOPUP_T table:
 - BAL_GRP_OBJ_REV
 - BAL_GRP_OBJ_TYPE
 - BAL_GRP_OBJ_ID0
 - BAL_GRP_OBJ_DB
 - STATUS
 - TOPUPS_DONE
 - NUM_TOPUPS
 - NEXT_TOPUP_T
 - TOPUP_INTERVAL
 - RESOURCE_ID
- The following columns are added in the SPONSORSHIP_T table:
 - ATTRIBUTE_OBJ_REV
 - ATTRIBUTE_OBJ_TYPE
 - ATTRIBUTE_OBJ_ID0
 - ATTRIBUTE_OBJ_DB
- The following column is added in the THREAD_INFO_T table:
 - THREAD_ID_DB
 - THREAD_ID_ID0

- THREAD_ID_TYPE
- THREAD_ID_REV
- The following indexes are added:
 - i_PERF_SNAPSHOT__ID
 - i_PERF_SNAPSHOT__HOST_NAME
 - i_PERF_RESULTS_ADD
 - i_PERF_RESULTS__ID
 - i_PERF_SQL_DATA__ID
 - i_PERF_SQL_DATA_BUF__ID

Patch Set 3 Schema and Index Changes

BRM 12.0 Patch Set 3 includes the following schema and index changes:

- The following tables are added:
 - JOB_TEMPLATE_CUSTOM_INFO_T
 - JOB_DEFINITION_T
 - EVENT_JOB_TEMPLATE_INFO_T
 - BOC_VERSION_T

Patch Set 2 Schema and Index Changes

BRM 12.0 Patch Set 2 includes the following schema and index changes:

- The following tables are added:
 - CMT_BALANCES_T
 - EVT_BILLING_CHARGE_CC_TRANS_T
 - EVT_BILL_VLDT_CC_TRANS_T
 - JOB_TEMPLATE_TBILL_BILLINFO_T
 - JOB_TEMPLATE_TBILL_INFO_T
 - JOB_TEMPLATE_TBILL_PAY_TYPES_T
 - JOB_TEMPLATE_TBILL_STATUSES_T
 - JOB_TEMPLATE_TBILLP_BILLINFO_T
 - JOB_TEMPLATE_TBILLP_INFO_T
 - JOB_TEMPLATE_TBILLP_STATUSES_T
 - JOB_TEMPLATE_WF_CATEGORIES_T
 - NAMEINFO_T
 - PAYINFO_CC_TRANS_T
- The following columns are added in the EVENT_CUSTOMER_NAMEINFO_T table:
 - POID_DB

- POID_ID0
- POID_TYPE
- POID_REV
- The following columns are added in the JOB_BOC_T table:
 - WORKFLOW_OBJ_ID
 - WORKFLOW_OBJ_TYPE
 - WORKFLOW_OBJ_REV
- The following column is added in the JOB_TEMPLATE_T table:
 - WORKFLOW_FLAG
- The following columns are added in the HISTORY_SUBSCRIBER_CONTRACT_T table:
 - GROUP_CONTRACT_OBJ_ID0
 - GROUP_CONTRACT_OBJ_TYPE
 - GROUP_CONTRACT_OBJ_REV
 - PLAN_CONTRACT_OBJ_DB
 - PLAN_CONTRACT_OBJ_ID0
 - PLAN_CONTRACT_OBJ_TYPE
 - PLAN_CONTRACT_OBJ_REV
- The following column is added in the RATE_BAL_IMPACTS_T table:
 - TAXABLE_AMOUNT
- The following columns are added in the SERVICE_T table:
 - NAMEINFO_OBJ_ID0
 - NAMEINFO_OBJ_TYPE
 - NAMEINFO_OBJ_REV
- The following columns are added in the SUBSCRIBER_CONTRACT_T table:
 - GROUP_CONTRACT_OBJ_DB
 - GROUP_CONTRACT_OBJ_ID0
 - GROUP_CONTRACT_OBJ_TYPE
 - GROUP_CONTRACT_OBJ_REV
 - PLAN_CONTRACT_OBJ_DB
 - PLAN_CONTRACT_OBJ_ID0
 - PLAN_CONTRACT_OBJ_TYPE
 - PLAN_CONTRACT_OBJ_REV
- The following indexes are added:
 - I_PROC_AUD_POCESS_END__ID
 - I_NAMEINFO__ID
 - I_NAMEINFO_ACCOUNT_OBJ__ID
 - I_NAMEINFO_PHONES__ID

- I_PROFILE_SUBS_PREF__ID
- I_ALLOWANCE__ID
- I_ASS_BUS_PROFILE_TEMPLATES__ID
- I_CFG_PROV_TAX_CODE_EFFECT__ID
- I_CFG_PROV_TAXES_REC__ID
- I_CFG_PROV_TAX_INFO__ID
- I_CFG_PROV_TAX_JUR_REC__ID
- I_DELIVERABLE_PRODUCT__ID
- I_EVT_ACT_SSP_MS__ID
- I_EVT_CONTRACT__ID
- I_EVT_CONTRACT_MODIFY__ID
- I_EVT_CONTRACT_RENEW__ID
- I_EVT_CONTRACT_START__ID
- I_EVT_PRODUCT_FEE_PENALTY__ID
- I_FEATURE__ID
- I_HIS_SUB_CNTR_ACCT_PKG__ID
- I_NOTE_DATA__ID
- I_RATE_RBI_INFO__ID
- I_RECURRING_ALLOWANCE__ID
- I_REV_DISTRIB_SC__ID
- I_REVENUE_DETAILS__ID
- I_REVENUE_PRODUCT_ACCT__ID
- I_REVENUE_PRODUCT_SC__ID
- I_REVENUES_PROD_T__ID
- I_REV_EVENT_BAL_IMPT__ID
- I_REV_EVENT_DEL__ID
- I_REV_EVENT__ID
- I_REV_PRODUCT_DEL__ID
- I_SERVICE_ENTITLEMENTS__ID
- I_SUB_CNTR_ACCT_PKG__ID
- I_ASSOCIATED_BUS_PROFILE__ID
- I_DELIVERABLE__ID
- I_ENTITLEMENT__ID
- I_HIS_SUBSCRIBER_CONTRACT__ID
- I_REV_DISTRIB__ID
- I_REVENUE_BASIS_ITEM__ID
- I_REVENUE_PRODUCT__ID

- I_SUBSCRIBER_CONTRACT__ID
- I_SUBSCRIPTION_TERMS__ID

Patch Set 1 Schema and Index Changes

BRM 12.0 Patch Set 1 includes the following schema and index changes:

- The following tables are added:
 - EVENT_ACT_ROLLUP_ITEMS_T
 - TMP_AR_ITEM_TO_ROLL_UP_T
- The following columns are added in the ITEM_T table:
 - ITEM_CLASS
 - AR_ITEM_OBJ
- The following column is added in the TMP_JOURNALS_TO_PROCESS_T table:
 - AR_BILLINFO_OBJ
- The following indexes are added:
 - I_TMP_AR_ITM_ROLLUP__ID
 - I_TMP_AR_ITM_ROLLUP__STATUS
 - I_ITEM_AR_ITEM_OBJ__ID
 - I_TMP_JOURNALS_TO_PROCESS__AR

12

Utility Changes

The Oracle Communications Billing and Revenue Management (BRM) 12.0 Patch Sets include new, modified, and removed utilities.

Topics in this document:

- [New Utilities](#)
- [Changed Utilities](#)
- [Removed Utilities](#)

New Utilities

[Table 12-1](#) lists the utilities that have been introduced in the BRM 12.0 patch set releases.

Table 12-1 New Utilities

New Utility	Description
pin_apply_promotion	(Patch Set 5) Introduced to apply promotions based on special dates. For more information, see "Applying Promotions On Special Dates" in <i>BRM Configuring Pipeline Rating and Discounting</i> .
pin_clean_offer_override	(Patch Set 5) Introduced to remove expired override date ranges. For more information, see "pin_clean_offer_override" in <i>BRM Configuring Pipeline Rating and Discounting</i> .
pin_gen_notifications	(Patch Set 5) Introduced to generate notification events when a customer's balance is about to expire, product is about to expire, subscription is about to renew, or bill is almost due. For more information, see "Sending Messages to Customers through External Applications" in <i>BRM Managing Customers</i> .
rel_manager	(Patch Set 5) Introduced to provide functionality for the Rated Event Loader, primarily for use with the new Rated Event Manager. For more information, see "Rated Event Loader Manager Utility" in <i>Loading Rated Events</i> .
pin_contracts	(Patch Set 4) Introduced to renew customer contracts or cancel contracts that have expired. This utility calls the PCM_OP_CONTRACT_RENEW_CONTRACT opcode to renew contracts, and the PCM_OP_CONTRACT_CANCEL_CONTRACT opcode to cancel expired contracts. For more information, see "Processing Contract Auto-Renewals" and "Canceling Expired Contracts" in <i>BRM Managing Customers</i> .
pin_del_closed_accts	(Patch Set 1) Introduced to delete closed accounts in BRM. This utility calls the PCM_OP_CUST_DELETE_ACCT opcode to delete the closed accounts that are older than the specified retention period. For more information, see " Enhanced Data Protection ".

Table 12-1 (Cont.) New Utilities

New Utility	Description
pin_roll_up_ar_items	(Patch Set 1) Introduced to process the temporary A/R items (<i>/tmp_ar_item_to_roll_up</i> object) of the nonpaying child bill units and roll the balance impact up to the corresponding A/R items in the paying parent bill unit. You can run multiple threads of pin_roll_up_ar_items to process A/R items for different paying parent bill unit. For more information, see " Improved Performance for Large Accounts ".
pin_update_journals	(Patch Set 1) Enhanced to process the temporary journals (<i>/tmp_journals_to_process</i> object) of the nonpaying child bill units and roll them up to the paying parent bill unit. You must run this utility before billing the paying parent bill unit. For more information, see " Improved Performance for Large Accounts ".

Changed Utilities

Table 12-2 lists the utilities that have been modified in the BRM 12.0 patch set releases.

Table 12-2 Changed Utilities

Changed Utility	Description
pin_cc_migrate	(Patch Set 6) Modified to support PINless debit transactions.
pin_sub_balance_cleanup	(Patch Set 6) Modified to support temporary credit limits. It includes the new -t parameter for purging sub-balances with temporary credit limits. See "pin_sub_balance_cleanup" in <i>BRM System Administrator's Guide</i> .
pin_collect	(Patch Set 6) Modified to support PINless debit transactions.
pin_deposit	(Patch Set 6) Modified to support PINless debit transactions.
pin_gen_notifications	(Patch Set 6) Modified to support notifications for installments and collections. It includes two parameters: <ul style="list-style-type: none"> • -collections_action <i>action</i> that generates an /event/notification/collections_action/due notification event for each specified collections action that needs to be performed for a customer. Replace <i>action</i> with the name of the /collections_action subclass, such as close_billinfo, late_fee, or promise_to_pay. • -installments [end_date]due_date that generates an /event/notification/installment_schedule/due notification event for each installment that is due or ends on the current date. See "pin_gen_notifications" in <i>BRM Managing Customers</i> .
pin_balance_transfer	(Patch Set 5) Modified to support recurring standard top-ups. It includes a new -standard parameter, which specifies to perform recurring standard top-ups. See "pin_balance_transfer" in <i>BRM Configuring and Collecting Payments</i> .

Table 12-2 (Cont.) Changed Utilities

Changed Utility	Description
loadpricelist	<p>(Patch Set 6) Modified to support:</p> <ul style="list-style-type: none"> • Additional values for PIN_FLD_TYPE in /rate objects • Setting the following fields in /rate objects: <ul style="list-style-type: none"> – PIN_FLD_PRORATE_QUANTITY set according to the prorate_quantity flag of the rate element. – PIN_FLD_MINIMUM_SCALED_AMOUNT set according to the minimum_scaled_amount element for the balance impact. • Setting the following fields in /product objects: <ul style="list-style-type: none"> – PIN_FLD_MAX_RETRY_COUNT – PIN_FLD_RETRY_UNIT – PIN_FLD_RETRY_OFFSET <p>(Patch Set 5) Modified to support:</p> <ul style="list-style-type: none"> • Setting PIN_FLD_TYPE in /rate objects according to the values provided in the type flag of the rate element. • Setting PIN_FLD_RENEWAL_MODE in /deal.products objects according to the values provided in the renewal_mode element.
ImportExportPricing	<p>(Patch Set 6) Modified to support:</p> <ul style="list-style-type: none"> • Additional values for the enforceCreditLimit element • Setting the following fields in /rate objects: <ul style="list-style-type: none"> – PIN_FLD_PRORATE_QUANTITY set according to the creditLimitProrateQuantity element in the price tier. – PIN_FLD_MINIMUM_SCALED_AMOUNT set according to the creditLimitMinimumAmount element for the charge. • Setting the following fields in /product objects: <ul style="list-style-type: none"> – PIN_FLD_MAX_RETRY_COUNT – PIN_FLD_RETRY_UNIT – PIN_FLD_RETRY_OFFSET <p>(Patch Set 5) Modified to support:</p> <ul style="list-style-type: none"> • Setting PIN_FLD_TYPE in /rate objects according to the values provided in the enforceCreditLimit element. • Setting PIN_FLD_RENEWAL_MODE in /deal.products objects according to the values provided in the renewalMode element. <p>(Patch Set 4) Modified to support subscription terms and deliverables. See "ImportExportPricing" in <i>PDC Creating Product Offerings</i>.</p>
pin_clean	<p>(Patch Set 4) Modified to support multischema systems. It includes a new -schema schema_number parameter that specifies the schema on which to run the utility on multischema BRM systems. By default, the utility runs on all schemas. See "pin_clean" in <i>BRM Configuring and Collecting Payments</i>.</p>
pin_recover	<p>(Patch Set 4) Modified to support multischema systems. It includes a new -schema schema_number parameter that specifies the schema on which to run the utility on multischema BRM systems. By default, the utility runs on all schemas. See "pin_recover" in <i>BRM Configuring and Collecting Payments</i>.</p>
purge_audit_tables.pl	<p>(Patch Set 4) Modified the utility to read the database user name and password from the Oracle Wallet rather than from a command-line parameter. To connect the utility to the database, you now use the -l /@connection parameter rather than -l login/password@connection. See "purge_audit_tables.pl" in <i>BRM System Administrator's Guide</i>.</p>

Table 12-2 (Cont.) Changed Utilities

Changed Utility	Description
pin_ifw_sync_oracle	(Patch Set 4) Modified the utility to read the database user name and password from the Oracle Wallet rather than from a command-line parameter. To connect the utility to the database, you now use the <code>-I /@connection</code> parameter rather than <code>-l login/password@connection</code> . See "pin_ifw_sync_oracle" in <i>BRM Installation Guide</i> .
db2irules.pl	(Patch Set 4) Modified the utility to read the database user name and password from the Oracle Wallet rather than at the command line. See "db2irules.pl" in <i>BRM Configuring Pipeline Rating and Discounting</i> .
irules2db.pl	(Patch Set 4) Modified the utility to read the database user name and password from the Oracle Wallet rather than at the command line. See "irules2db.pl" in <i>BRM Configuring Pipeline Rating and Discounting</i> .
stateconfigtool	(Patch Set 4) Modified the utility to read the database user name and password from the Oracle Wallet rather than at the command line. See "stateconfigtool" in <i>BRM Configuring Pipeline Rating and Discounting</i> .
pin_job_executor	(Patch Set 3) Modified to support custom MTA applications. See "pin_job_executor" in <i>BRM System Administrator's Guide</i> .
pin_crypt_app	(Patch Set 3) Modified to support the AES encryption method. The AES encryption method is supported only for backwards compatibility with BRM 7.5. See "pin_crypt_app" in <i>BRM Developer's Guide</i> .
pin_del_closed_accts	(Patch Set 3) Modified to include a new -force parameter, which allows BRM to close accounts that still have active open sessions in ECE. See "pin_del_closed_accts" in <i>BRM System Administrator's Guide</i> .
cmt_mta_cycle_fees	(Patch Set 2) Renamed as cmt_mta_deploy and modified to support incremental migration of legacy BRM data. See " Support for Migration of Legacy Data into BRM and ECE in Real Time " for more information.
partition_utils	(Patch Set 2) Modified to support creation of partitions for the prepaid events. See " BRM Supports POID Generation in ECE " for more information.
pin_cmt	(Patch Set 2) Modified to migrate legacy service and balance data incrementally into the BRM system. You can now run the pin_cmt utility with the following new parameters to migrate data: <ul style="list-style-type: none"> • deploy_db • deploy_ece See " Support for Migration of Legacy Data into BRM and ECE in Real Time " for more information.
pin_collect	(Patch Set 2) Modified to support stored credentials for payment transactions. See " Support for Stored-Credential Transactions for Payments " for more information.
pin_deposit	(Patch Set 2) Modified to support stored credentials for payment transactions. See " Support for Stored-Credential Transactions for Payments " for more information.

Removed Utilities

Table 12-3 lists the utilities that have been removed in the BRM 12.0 patch set releases.

Table 12-3 Removed Utilities

New Utility	Description
pin_installment_notifications	(Patch Set 6) You now use the pin_gen_notifications utility to trigger notifications for installments. See "Running the pin_gen_notifications Utility" in <i>BRM Managing Customers</i> .
pin_update_items_journals	(Patch Set 1) This utility has been replaced by the pin_update_journal utility. See " Improved Performance for Large Accounts " for more information.

13

Known Problems

The Oracle Communications Billing and Revenue Management (BRM) Patch Sets include several known problems and workarounds.

Topics in this document:

- [Known Problems in BRM](#)
- [Known Problems in Business Operations Center](#)
- [Known Problems in Billing Care](#)
- [Known Problems in PDC](#)
- [Known Problems in PCC](#)
- [Known Problems in BRM Cloud Native Deployment](#)
- [Known Problems in BRM REST Services Manager](#)

Known Problems in BRM

This section describes known problems and workarounds for BRM 12.0 patch sets.

A/R Actions between Billing and Invoicing Not Captured in Invoice

The invoicing process does not pick up the details of any A/R action that occurs between the time when a bill is generated and when the invoice for that bill is produced.

Workaround: Run billing and invoicing as an atomic operation without other A/R actions in between. For more information, see *BRM Configuring and Running Billing*.

Bill Now Generates Two Audit Objects for Nonpaying Child Items

When you run Bill Now on an account, selecting items corresponding to a nonpaying child bill unit, two audit objects are created: one audit object with an amount of 0 as revenue for the account that owns the paying parent bill unit and another audit object for the account that owns the nonpaying child bill unit.

Currently, there is no workaround.

Cannot Install BRM Reports and BRM Invoices in the Same Home Directory as Oracle Analytics Server

When you try to install the BRM Reports and BRM Invoices components in the same home directory as Oracle Analytics Server, the installation fails.

Workaround: Install BRM Reports and BRM Invoices in a different Oracle home directory than Oracle Analytics Server, then copy the **BRM_Reports** and **BRM_Invoices** directories as follows:

1. Install the BRM Reports and BRM Invoices components in one location as described in "Installing Individual BRM Components." For example, use the following installation location:

```
$PIN_HOME/opt/portal
```
2. Install Oracle Fusion Middleware and Oracle Analytics Server in a separate location from BRM Reports and BRM Invoices. For example, use the following installation location:

```
$PIN_HOME/Oracle/Middleware/Oracle_Home
```
3. Create the necessary database schemas and configure Oracle Analytics Publisher as described in "Creating the Database Schemas" and "Configuring Oracle Analytics Server Domain with the Configuration Assistant" in *Oracle Analytics Installing and Configuring Oracle Analytics Server*.
4. Copy the **BRM_Reports** and **BRM_Invoices** directories from the location where you installed them. For example:

```
$PIN_HOME/opt/portal/user_projects/domains/bi/bidata/components/bipublisher/repository/Reports
```
5. Paste the **BRM_Reports** and **BRM_Invoices** directories to the new location in the Oracle Fusion Middleware and Oracle Analytics Server directory structure. For example:

```
$PIN_HOME/Oracle/Middleware/Oracle_Home/user_projects/domains/bi/bidata/components/bipublisher/repository/Reports
```

BRM Server and Client OUI Installers Not Localized in Russian

Localized versions of BRM server and client applications, such as Pricing Design Center and Pipeline Configuration Center, are available in Russian. However, their Oracle Universal Installers (OUI) are not localized in Russian.

Currently, there is no workaround.

For information about localized client applications, see "Using Localized Client Applications" in *BRM Developer's Guide*.

Default Invoices Do Not Display Custom Events and Items

The default BI Publisher invoice templates shipped with BRM do not display data from custom **levent** and **litem** objects.

Workaround: If you create custom **levent** or **litem** objects, you must also customize the BI Publisher invoice templates to display your custom data. For more information, see the discussion about customizing invoices in *BRM Designing and Generating Invoices*.

Error When Configuring or Modifying Security Policies for Web Services Manager for JAX-WS

If you have configured security policies for Web Services Manager for JAX-WS or if you have modified the security policies, the request sent by the client to the server fails and the following error message appears:

headers:[{http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd}Security] are not understood

Workaround: If your client caches service WSDLs, refresh the WSDLs before sending the requests to the server.

Event Browser Displays Data Internal to BRM in the Event Description

For usage events that were rated by Oracle Communications Elastic Charging Engine (ECE), Event Browser displays data internal to BRM in the event description.

Currently, there is no workaround.

Invoices Do Not Display Charge Sharing Details Separately

The balances associated with charge sharing are not split into original charges and sponsored/shared charges in the invoices. The total balances are correct and include any effect of charge sharing.

Currently, there is no workaround.

Invoices Might Display Wrong Discount Amount

Invoices might display the wrong discount amount for accounts with nonpaying bill units in a discount sharing group.

Currently, there is no workaround.

pin_inv_accts Fails for Migrated Accounts if Business Intelligence Publisher Invoicing Integration Is Enabled

If Oracle Business Intelligence Publisher (BI Publisher) invoicing integration is enabled, the **pin_inv_accts** utility fails for migrated accounts and an error is logged in the **cm.pinlog** file.

Workaround: Do the following before migrating the accounts:

1. Open the *BRM_home/sys/amt/data/create_generate_amt_metadata.sql* file in a text editor.
2. Search for the following lines:

```
ELSIF (column_r.column_name LIKE '%POID_LIST') THEN
  --
  -- operation: rewrite the POID_DB reference in %_POID_LIST fields
  -- (i.e. account_t.next_item_poid_list, account_t.item_poid_list)
  -- while preserving the null value
  --
  -- assumption: POID_DB reference is starts at character position 1 and the format
  -- is 0.0.0.x
  --
  -- call custom Java stored procedure convert_poid_list()
  --
  select_column_list := select_column_list||
  ', convert_poid_list('||column_r.column_name||', #)';
  dbms_output.put_line('replace list reference with dest_db_id');
```

3. Add the following lines after the ELSIF condition you searched for in step 2:

```

ELSIF (column_r.column_name LIKE '%OBJ_LIST') THEN
  select_column_list := select_column_list||
  ', convert_poid_list('||column_r.column_name||', #)';
dbms_output.put_line('replace list reference with dest_db_id');

```

4. Save and close the file.
5. Run the **pin_amt_install.pl** script.
6. Stop and start the Data Manager (DM) and Connection Manager (CM) services.
7. Start migrating the accounts.

Refund Is Incorrectly Calculated When Remaining Charge Discount Canceled

On canceling a remaining charge discount on a cycle forward arrears event, the refund amount is calculated incorrectly.

Currently, there is no workaround.

RE Loader Daemon Fails to Process Files Under Load

The RE Loader daemon fails and throws the following error message when processing a large number of Pipeline Manager output files:

```
The infranet.rel.password key is missing from the properties file.
```

This occurs because, under load, RE Loader fails to read the database password from the Oracle wallet.

Workaround: In the *BRM_home/apps/pin_rel/Infranet.properties* file, set this entry to the password for connecting to the BRM database. The password can be in plain text or encrypted in OZT format.

```
infranet.rel.password = Password
```

See *BRM Loading Rated Events*.

Sample Prepaid Service Life Cycle Does Not Support Provisioning of Telco Services

The sample prepaid service life cycle does not support provisioning of telco services.

Workaround: Enable the sample prepaid service life cycle to support the provisioning of telco services by adding the following state transition to that life cycle before associating the life cycle with a telco service type:

- **From state:** Preactive (ID 101)
- **To state:** Suspended (ID 107)

For information about adding state transitions to service life cycles, see the discussion about managing service life cycles in *BRM Managing Customers*.

The SLM Feature Does Not Support Account Billing Relationships

The SLM feature does not support the following account relationships, which enable customers to pay other customers' bills:

- Parent/nonpaying child hierarchies
- Owner/member charge sharing or discount sharing
- Owner/member sponsorship

If a service involved in such a relationship uses a custom life cycle, the SLM feature does not consider the condition of the parent/owner account when an event occurs that triggers a service state change in a child/member account.

For example, Service A uses the sample prepaid service life cycle. For that service, a parent account has a credit limit of 100 euros and its nonpaying child account has a credit limit of 0 euros. In both the parent and the child, the state of Service A is Active. After an account adjustment of 2 euros is applied to the child account, however, the state of Service A in the child account changes to Recharge Only, even though the parent account's available credit limit is still 100 euros.

Therefore, service types involved in account billing relationships must use the default service life cycle (Active, Inactive, Closed). They cannot use custom service life cycles.

Workaround: Customize the `PCM_OP_BAL_POL_CHECK_LIFECYCLE_STATE` policy opcode to consider the condition of parent/owner accounts when a service state change is triggered in child/member accounts.

See the discussion about managing service life cycles in *BRM Managing Customers* for more information.

Settlement Taxes Might Be Incorrectly Calculated

If you adjust only the tax amount in an event (by calling the `PCM_OP_AR_EVENT_ADJUSTMENT` opcode with the `PIN_EVENT_ADJ_TAX_ONLY` flag) and then dispute the item containing the event, the settlement tax is not calculated correctly; therefore the total due amount of the item is incorrect. The settlement is calculated correctly only when the denied amount is 0. This occurs with both real-time and deferred taxation.

Currently, there is no workaround.

stop_eai_js Does Not Work

When you run **stop_eai_js**, the script does not work and the following error message appears:

```
stop_eai_js: ERROR: no pid file.
```

Workaround: Do the following:

1. Open the `BRM_home/bin/stop_eai_js` file in a text editor.
2. Search for the following entry:

```
LOGDIR=$PIN_LOG/dm_eai
```

3. Change this entry to:

```
LOGDIR=$PIN_LOG/eai_js
```

4. Save and close the file.

Summary Invoices Do Not Show Real-Time Discount and Tax Details for Items

Summary invoices display data related to charges at the item level. Details of real-time taxes or real-time discounts are not displayed separately on summary invoices. The charges for the billable items include this data.

Workaround: To view these details on invoices, generate the invoices in one of the following ways:

- Use the detailed format.
- Customize the item assignment configuration to aggregate the discounts and taxes into their own separate items.

The invoices will contain separate lines for the discounts and taxes. For more information, see *BRM Designing and Generating Invoices*.

Tax Not Recalculated after Rerating

When you rerate the bill amount after rerating the changes, the tax amount is not recalculated.

Workaround: manually adjust the tax amount.

Error When Creating a New Data Sequence for BRM

While upgrading from 12.0 Patch Set 3 to Patch Set 8, you may get the following error while creating a new data sequence for BRM:

```
Error: Unable to update the Portal Base database.  
Exiting from the install without finishing successfully.  
Please check for errors in  
and then rerun this program.  
EBUF: errno=<PIN_ERR_DUPLICATE:10> location=<PIN_ERRLOC_DM:4>  
class= field num=<PIN_FLD_POID:7,16> recid= reserved=
```

This occurs because one of the upgrade files, **new_data_sequence_for_ext_id_12PS5.source**, creates sequence number 320, which is a predefined sequence number in BRM.

Workaround: In the **new_data_sequence_for_ext_id_12PS5.source** file, locate the following line:

```
0 PIN_FLD_POID POID [0] $PIN_CONF_DB_NO /data/sequence 320 0
```

Replace the number **320** with a unique sequence number other than 320.

Known Problems in Business Operations Center

There are no known problems in Business Operations Center 12.0 patch sets.

Known Problems in Billing Care

This section describes known problems and workarounds for Business Operations Center 12.0 patch sets.

Duplicate Login Request When Using Single Sign-On

When Single Sign-On (SSO) with Oracle Identity and Access Management Lifecycle Management is implemented for your Billing Care environment, users are redirected to the Billing Care login screen after authenticating in the SSO login screen.

Workaround: On each Oracle Platform Security Services (OPSS) JRF-enabled domain where Billing Care is deployed:

1. Comment out the **<variable-definition>** and **<module-override>** entries in the *Middleware_home/setup/Plan.xml* file as shown below, where *Middleware_home* is the WebLogic home directory on the OES Client domain host of the user who installed Billing Care.

```
<?xml version='1.0' encoding='UTF-8'?>
<deployment-plan xmlns="http://xmlns.domain.com/weblogic/deployment-plan"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://
xmlns.domain.com/weblogic/deployment-plan http://xmlns.domain.com/weblogic/
deployment-plan/1.0/deployment-plan.xsd" global-variables="false">
  <application-name>BillingCare.war</application-name>
  <!--<variable-definition>
    <variable>
      <name>CONFIG_SSL_REDIRECT</name>
      <value>CONFIDENTIAL</value>
    </variable>
  </variable-definition>
  <module-override>
    <module-name>BillingCare.war</module-name>
    <module-type>war</module-type>
    <module-descriptor external="true">
      <root-element>web-app</root-element>
      <uri>WEB-INF/web.xml</uri>
      <variable-assignment>
        <name>CONFIG_SSL_REDIRECT</name>
        <xpath>/web-app/security-constraint/user-data-constraint/transport-
guarantee</xpath>
      </variable-assignment>
    </module-descriptor>
  </module-override> -->
</deployment-plan>
```

2. Restart the domain.

Firefox and Internet Explorer Fail to Connect Securely to Billing Care

The **Plan.xml** file deployed by the Billing Care Oracle Universal Installer enables mandatory use of an SSL-enabled port when connecting with all browsers. Firefox and Internet Explorer may fail to connect to Billing Care, producing a secure connection failure error.

Workaround: For Firefox, no workaround exists. For Internet Explorer, start a Command Prompt session with administrative rights, and run the following command before launching Internet Explorer:

```
certutil -setreg chain\EnableWeakSignatureFlags 8
```

Known Problems in PDC

This section describes known problems and workarounds for PDC 12.0 patch sets.

PDC Search Functionality Not Working As Expected

Basic Search

When you perform a basic search in the PDC user interface, clicking a search result does not open it.

Workaround: Double-click a search result to open it.

Advanced Search

If you are using Internet Explorer 11 on Windows 10, the search results do not appear as expected when you perform an Advanced search in the PDC user interface.

Workaround: Use Mozilla Firefox on Windows 10 or use Internet Explorer 11 on Windows 8.1 to perform Advanced search in the PDC user interface.

Migrating Pricing Data Fails in a PDC System with BRE and RRE

When the zone result validity period set in a zone model exceeds the validity period set for the associated rate plan, migration fails with a validation error.

Workaround: Modify the zone result validity in BRM to ensure that the zone result validity overlaps or falls within the associated rate plan validity.

Adding Balance Groups and Setting Credit Limits in a Package Displays Exceptions

Setting Credit Limits Exception

In the Balance Groups section of the Create Package page, when you set the credit limit for a balance element and save it, the changes are saved successfully but PDC displays an exception.

Workaround: Navigate to any other section in the Create Package page and save the changes to avoid exceptions.

Adding Balance Groups Exception

When you add another balance group immediately after setting the credit limit for a balance element, the balance group is created successfully but PDC displays an exception instead of displaying the newly created balance group.

Workaround: Click the existing balance group to view the newly created balance group.

Migration Fails Even When the Migration Process is Enabled to Skip Errors

During migration, when a critical error is encountered while migrating a discount filter or a discount trigger, migration fails with errors even when the migration process is enabled to skip errors and continue migration.

Workaround: Manually fix the errors encountered during migration and restart the migration process.

You Are Not Warned to Save Your Changes

After modifying a setup or pricing component, you are not warned that you have unsaved changes when you do the following:

- Close the tab
- Log out of PDC

If you perform any of the above actions before saving your changes, the changes are lost.

Workaround: Save your changes before closing a tab or logging out of PDC.

Known Problems in PCC

There are no known problems in PCC 12.0 patch sets.

Known Problems in BRM Cloud Native Deployment

This section describes known problems and workarounds for BRM cloud native deployment 12.0 patch sets.

Invoicing Formatter Uses Incorrect XSLT Processor

In BRM cloud native 12.0 Patch Set 4, the invoice formatter Pod uses an incorrect XSLT parser and engine.

Workaround: Set the following keys in your `oc-cn-helm-chart/templates/configmap_infranet_properties_formatter.yaml` file:

```
infranet.pxslt.parser=javax.xml.parsers.SAXParser
infranet.pxslt.engine=com.portal.pxslt.PXSLTEngineXdkImpl
```

Missing Subscriber State Configuration Files

In the BRM cloud native package, these subscriber state configuration files are missing from the Connection Manager (cm) Pod:

- **pin_slm_business_profile.xml**
- **config_lifecycle_states.xml**
- **config_service_state_map.xml**
- **config_service_state_map.xsd**
- **config_lifecycle_states.xsd**
- **config_lifecycle_states.xsl**

Currently, there is no workaround.

syncPDC May Fail After Upgrade to PDC Cloud Native 12.0 Patch Set 3

After upgrading your Pricing Design Center (PDC) cloud native deployment from 12.0 Patch Set 2 to 12.0 Patch Set 3, the **syncPDC** utility may fail due to mismatched events.

Workaround: Enter this SQL*Plus command to drop the duplicate events from the CONFIG_PERMITTED_EVENTS_T table:

```
SQL> select EVENT_TYPE from CONFIG_PERMITTED_EVENTS_T where EVENT_TYPE NOT in  
(select name from dd_objects_t);
```

Error When Upgrading ECE from BRM 12.0 Patch Set 6 to Patch Set 8

When upgrading ECE from BRM 12.0 Patch Set 6 (12.0.0.6.0) to Patch Set 8, you may get the following error:

```
cdrgateway pod is in crashloopbackoff due to incompatibility between  
BRM 12.0 PS6 and Kubernetes 1.25
```

This is because the **ece-cdrgatewayservice.yaml** file does not contain the **publishNotReadyAddresses** entry.

To fix this, manually add the **publishNotReadyAddresses: true** entry to the **oc-cn-ece-helm-chart/ templates/ece-cdrgatewayservice.yaml** file and rerun the upgrade script:

```
clusterIP: None  
selector:  
  application: cdrgateway  
publishNotReadyAddresses: true
```

Known Problems in BRM REST Services Manager

This section describes known problems and workarounds for BRM REST Services Manager 12.0 patch sets.

Must Use Payment Methods that Are Already Associated with Bill Units

When creating a payment using the extended payment object in the BRM REST Services Manager API, you must use the payment method that is associated with the bill unit to which the payment will be allocated.

If you use a payment method that is not associated with the bill unit, one of the following will happen:

- If the payment method in the request is the same type (credit card, debit card, direct debit, or invoice) as the one associated with the bill unit, the payment will be processed using the payment method associated with the bill unit, not the one on the request.
- If the specified payment method is a different type than the one associated with the bill unit, the payment will fail.

Workaround: Before submitting a payment, determine which bill unit the payment will be allocated to, and identify the payment method associated with the bill unit by submitting a GET request to the following endpoint:

```
https://hostname:port/brm/accountManagement/v4/billingCycleSpecification/id?type=
    billingCycleSpecificationOracle&fields=paymentProfile
```

Where:

- *hostname* is the URL for the BRM REST Services Manager server.
- *port* is the port for the BRM REST Services Manager server.
- *id* is the ID of the bill unit you want to find the payment method for.
- Including **fields=paymentProfile** retrieves values for the relevant fields only, so you can quickly find the information you need. Values for other fields will be **null**.

The following is an example of the response in JSON format, with the payment method ID in bold:

```
{
  "id": "0.0.0.1+-billinfo+110189",
  "href": "https://host:port/brm/accountManagement/v4/billingCycleSpecification/
0.0.0.1+-billinfo+110189",
  "billingDateShift": null,
  "billingPeriod": null,
  "chargeDateOffset": null,
  "creditDateOffset": null,
  "description": null,
  "frequency": null,
  "mailingDateOffset": null,
  "name": null,
  "paymentDueDateOffset": null,
  "validFor": null,
  "@baseType": "BillingCycleSpecification",
  "@schemaLocation": null,
  "@type": "BillingCycleSpecificationOracle",
  "status": null,
  "remainingAmount": null,
  "billingAccount": null,
  "paymentProfile": {
    "id": "0.0.0.1+-payinfo-invoice+109165",
    "href": null,
    "name": "Invoice Payment",
```

```

        "@baseType": null,
        "@schemaLocation": null,
        "@type": null,
        "@referredType": null
    },
    "accountingType": null
}

```

In this example, you would use **0.0.0.1+-payinfo-invoice+109165** as the value for the **id** property of the **PaymentMethodRefOrValue** object in the create payment message.

Cannot Allocate Payments to Bills of Different Bill Units

You cannot allocate payments to bills belonging to different bill units (billing cycle specifications) in a single request using the BRM REST Services Manager API. If you try, the allocation will fail.

In the body for POST paymentAllocation requests, you must specify only bills belonging to the same bill unit.

You can find out what bill unit a bill belongs to by submitting a GET request against the following endpoint:

```

https://host:port/brm/customerBillManagement/v4/customerBill/id?
@type=CustomerBillOracle&fields=billingCycleSpecification.id

```

Where:

- *hostname* is the URL for the BRM REST Services Manager server.
- *port* is the port for the BRM REST Services Manager server.
- *id* is the ID of the bill you want to find the bill unit for.
- Including **fields=billingCycleSpecification.id** retrieves values for the relevant fields only so you can quickly find the information you need. Values for other fields will be **null**.

The following is an example of an excerpt from the response in JSON format, with the bill ID and the bill unit ID in bold:

```

{
  "id": "0.0.0.1+-bill+101",
  ...
  "billingCycleSpecification": {
    "name": null,
    "id": "0.0.0.1+-billinfo+133587",
    "Status": null,
    "href": "https://host:port/brm/accountManagement/v4/
billingCycleSpecification/0.0.0.1+-billinfo+133587",
    "accountingType": null,
    "@type": "PartyBillingCycleSpecificationOracle",
    "@baseType": "PartyBillingCycleSpecification",
    "@schemaLocation": null
  },
  ...
}

```

Workaround: To allocate partial payments to bills belonging to two separate bill units, use two separate requests. For example, if the total amount for payment 109 is \$50, you could submit the following as the POST request body to allocate half to bill 101:

```
{
  "bill": [
    {
      "id": "0.0.0.1+-bill+101",
      "allocatedAmount": {
        "unit": USD,
        "value": 25
      }
    }
  ],
  "paymentId": "0.0.0.1+-item-payment+109"
}
```

Then you would submit the following request separately to allocate the other half to bill 102:

```
{
  "bill": [
    {
      "id": "0.0.0.1+-bill+102",
      "allocatedAmount": {
        "unit": USD,
        "value": 25
      }
    }
  ],
  "paymentId": "0.0.0.1+-item-payment+109"
}
```