# Contents

## Preface

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audience</td>
<td>v</td>
</tr>
<tr>
<td>Related Documentation</td>
<td>v</td>
</tr>
<tr>
<td>Documentation Accessibility</td>
<td>v</td>
</tr>
</tbody>
</table>

## 1 About Rerating Events

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reasons for Rerating</td>
<td>1-1</td>
</tr>
<tr>
<td>About the Rerating Process</td>
<td>1-2</td>
</tr>
<tr>
<td>Run-time Options for Rerating</td>
<td>1-4</td>
</tr>
<tr>
<td>How BRM Applies the Balance Impacts of Rerating</td>
<td>1-4</td>
</tr>
<tr>
<td>About Rerating Unbilled Events</td>
<td>1-5</td>
</tr>
<tr>
<td>Adjustments When Billing Follows Rerating</td>
<td>1-5</td>
</tr>
<tr>
<td>About Rerating Billed and Posted Events</td>
<td>1-5</td>
</tr>
<tr>
<td>Determining the G/L Entry for an Event</td>
<td>1-6</td>
</tr>
<tr>
<td>How Rerating Affects Account Migration</td>
<td>1-6</td>
</tr>
<tr>
<td>How BRM Generates Rerated Events</td>
<td>1-6</td>
</tr>
<tr>
<td>How Failed Rerate Jobs Are Processed</td>
<td>1-8</td>
</tr>
</tbody>
</table>

## 2 Configuring Rerating

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>About Configuring Rerating</td>
<td>2-1</td>
</tr>
<tr>
<td>Configuring the Connection Manager for Rerating</td>
<td>2-1</td>
</tr>
<tr>
<td>Creating Rerate Jobs Automatically</td>
<td>2-2</td>
</tr>
<tr>
<td>Creating Rerate Jobs for Backdated Events</td>
<td>2-2</td>
</tr>
<tr>
<td>Rerating Cycle Forward and Cycle Forward Arrears Charges</td>
<td>2-3</td>
</tr>
<tr>
<td>Enabling Rerating and Rollover Correction Due to Delayed Events</td>
<td>2-3</td>
</tr>
<tr>
<td>Enabling and Configuring Automatic Rerate Job Creation</td>
<td>2-4</td>
</tr>
<tr>
<td>Configuring Event Notification for Automatic Rerate Job Creation</td>
<td>2-5</td>
</tr>
</tbody>
</table>

## 3 Advanced Rerating Tasks

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rerating Events when the Rating Conditions Change During the Session</td>
<td>3-1</td>
</tr>
<tr>
<td>Allowing Rerating of Old Events</td>
<td>3-2</td>
</tr>
<tr>
<td>Enabling Correction of First-Usage Validity Dates</td>
<td>3-2</td>
</tr>
<tr>
<td>Configuring Rerating Adjustment Events</td>
<td>3-3</td>
</tr>
<tr>
<td>Configuring How to Allocate Rerating Results</td>
<td>3-5</td>
</tr>
<tr>
<td>Enabling Deferred Tax Calculation during Rerating</td>
<td>3-5</td>
</tr>
</tbody>
</table>
4 Rerating Events

Overview .................................................................................................................................................... 4-1
Running Rerating .................................................................................................................................... 4-2
   Generating Rerating Reports ................................................................................................................ 4-2
   Recycling Records Suspended during Rerating .................................................................................. 4-3
   Processing Failed Rerate Jobs ......................................................................................................... 4-4
   Purging Rerate Jobs ............................................................................................................................ 4-4
Selecting Events for Rerating ............................................................................................................... 4-4
   Rerating Events for a Single Account ............................................................................................... 4-6
   Rerating Events for a List of Accounts .............................................................................................. 4-6
   Rerating Events Associated with Charge Offers ............................................................................... 4-7
   Rerating Events Associated with Specific Event Types .................................................................... 4-7
   Rerating Events Associated with Services ....................................................................................... 4-8
   Rerating Events Associated with a Subscriber Service ID .............................................................. 4-8
   Rerating Events Associated with a Balance Group ........................................................................... 4-8
   Rerating Events Associated with a Custom Event Field .................................................................... 4-9
   Getting an Estimate for the Number of Rerated Events ................................................................ 4-10
   Specifying Events for Rerating ........................................................................................................ 4-10
   Specifying the Event Sequence for Rerating ................................................................................... 4-11
Using Reason Codes to Specify the Rerating Order ........................................................................ 4-12
Rerating Cycle Forward and Cycle Forward Arrears Events ............................................................... 4-14
Running Back-Out-Only Rerating ....................................................................................................... 4-14

5 Tuning Rerating Performance

   Improving pin_rerate Performance ................................................................................................. 5-1
   Setting the Rerating Event Cache Size (Fetch Size) ....................................................................... 5-1
   Configuring the Number of Accounts Per Job and Number of Jobs per Transaction .............. 5-2

6 Rerating Utilities

   load_pin_rerate_flds ............................................................................................................................ 6-2
   pin_rate_change ................................................................................................................................. 6-3
   pin_rerate .......................................................................................................................................... 6-4
Preface

This guide describes how to configure and run the event rerating process in Oracle Communications Billing and Revenue Management.

Audience

This guide is intended for system administrators.

Related Documentation

Before reading this book, read BRM Concepts.

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.
This chapter provides an overview of Oracle Communications Billing and Revenue Management (BRM) rerating using Oracle Communications Billing and Revenue Management Elastic Charging Engine (ECE).

If you are using BRM real-time rating and batch rating engines for usage rating, see “Setting Up Rerating” in BRM Configuring Pipeline Rating and Discounting for information on rerating events.

Topics in this document:
- Reasons for Rerating
- About the Rerating Process
- Run-time Options for Rerating
- How BRM Applies the Balance Impacts of Rerating
- How BRM Generates Rerated Events

See also:
- About Rerating Events
- Configuring Rerating
- Rerating Events
- Advanced Rerating Tasks
- Tuning Rerating Performance

### Reasons for Rerating

You can rerate events for several reasons:

- To correct incorrect charges. For example, if several customers were charged the wrong price for using a service, you can rerate the events for all of those customers.

- To rerate an account when a customer service representative (CSR) backdates a subscriber’s purchase or cancellation. For example, if a customer wants to upgrade his service and purchases a backdated charge offer, events that were charged based on the old charge offer are rerated based on the upgraded charge offer.

- To correct events that should have been rated by using a different charge offer.

- To correct rollover amounts consumed by offline charging. This can occur if a delayed event arrives after the end of the accounting cycle and during the delayed
billing period. The event can borrow against the rollover of the current cycle even when the current rollover is consumed by events of the current cycle.

- To reset first-usage validity dates for recurring charges associated with offers or balances that start on first usage. For example, if the first event rated was not the first event to use the offer’s service, rerating corrects the order of the events and resets the validity period based on the first-usage event.

- To perform back-out-only rerating, which backs out the balance impacts of rating without reapplying new balance impacts.

About the Rerating Process

BRM rerates events in two separate processes:

1. Create rerate jobs that define the events that need rerating.
2. Run rerating to rerate the events.

If you are using ECE for usage rating, the BRM server sends the usage events to ECE to rerate. ECE rerates the events and sends the new balance impacts to the BRM server.

If you are using BRM real-time rating and batch rating engines for usage rating, BRM rerates the events. See “Setting Up Rerating” in BRM Configuring Pipeline Rating and Discounting for configuring rerating.

To rerate events, you use the `pin_rerate` utility to processes rerate jobs. BRM uses rerate jobs for two purposes:

- To manage the processing of many events in an efficient, transactional way. Multiple rerate jobs are created for every rerate request.
- To manage the status of the accounts selected for rerating. For example, if an account is part of a rerate job in progress, it cannot be migrated to another database until rerating has finished.

Creating rerate jobs is a separate process from processing rerate jobs. You can create rerate jobs manually by running the `pin_rerate` utility with event selection criteria. For example, you can create rerate jobs to rerate all the events associated with a single charge offer. This creates the rerate jobs run the next time you run the `pin_rerate` utility.

You can also configure BRM to create rerate jobs automatically. When certain events occur in the BRM system (such as backdated purchase events), customer accounts might require rerating so they are billed accurately. In this case, BRM creates rerate jobs to rerate those events. The next time you run the `pin_rerate` utility, the automatically-created rerate jobs are processed.

To rerate events, you use the `pin_rerate` utility. You run the following commands in this order:

```
pin_rerate -d bundle_rerate.txt -t 07/23/2015 -r
pin_rerate -process jobs ]
pin_rerate -process queue
pin_rerate -rerate
pin_rerate -process queue
pin_rerate -process recycle
```

These commands process rerate jobs created both manually and automatically. The process works as follows:

1. The first step creates rerate jobs for events associated with specific bundles:
pin_rerate -d bundle_rerate.txt -t 01/23/2016 -r

This example creates rerate jobs to rerate all of the events associated with a list of bundles (defined in the bundle_rerate.txt file) that occurred after January 1, 2016

2. The second step processes new rerate jobs:

```
pin_rerate -process jobs
```

The `-process jobs` command finds rerate jobs with the NEW status and sends a prepare-to-rerate message to ECE. Some of the rerate jobs might have been created automatically. The `pin_rerate` utility then updates the status of the rerate jobs to WAITING_ACCOUNT_LOCK.

When ECE finds out which customers are being rerated, ECE sends rated events for those customers to the BRM database to synchronize balances before rerating starts.

3. The third step, the first `-process queue` command, receives acknowledgement from ECE to suspend rating for the specified accounts, and updates the rerate job status to ACCOUNT_LOCKED.

```
pin_rerate -process queue
```

4. The fourth step, the `-rerate` command, sends the events to be rerated to ECE and updates the rerate job status to RERATED.

```
pin_rerate -rerate
```

ECE rerates the events. ECE performs necessary balance updates and backouts and synchronizes the balances for customers in the rerating requests.

After ECE rerates the events, ECE sends the charging results to the BRM database.

5. The fifth step, the second `-process queue` command, receives acknowledgement from ECE to resume processing the accounts being rerated. It updates the rerate job status to READY_TO_RECYCLE.

```
pin_rerate -process queue
```

6. The sixth step, the `-process recycle` command, recycles events suspended during the rerating process, and updates the rerate job status to COMPLETE.

```
pin_rerate -process recycle
```

During rerating, BRM performs the following tasks:

- ECE continues rating usage and applying top-ups while rerating occurs. However, ECE waits until rerating has completed before sending the rated results to the BRM database.

- BRM stores all subscriptions events that occur in the BRM system until rerating is complete. After rerating, BRM rates the subscription events and sends any relevant results to ECE; for example, balance updates and changes to the purchased offers.

- For events that include charge sharing, events for customers not in the rerating job, but who share discounts or charges with customers who were in the rerating job, are processed further so that balances are applied correctly.

- If ECE cannot rerate events for a customer, ECE sends a notification to BRM. BRM uses the information in the notification for create a new rerate job for that customer. You rerate the failed rerate jobs using the same procedure described above.
Run-time Options for Rerating

When you rerate events, you have the following options:

- You can select events for rerating based on various criteria such as the time the event occurred, the account number, bundles, charge offers, discount offers, event, service, and so on. See “Selecting Events for Rerating” for information.

- You can control the order in which rerating occurs by creating reason codes. For example, to rerate events that impact future rating, such as volume-based discounting, use reason codes to rerate those events first. See “Using Reason Codes to Specify the Rerating Order”.

- You can specify the order in which events are rated by either the order that they were created in the BRM database, or by the order in which the events occurred. See “Specifying the Event Sequence for Rerating” for information.

- You can run reports during rerating that include statistics such as the original amount, rerated amount, and the difference between them. See “Generating Rerating Reports” for information.

- You can rerate events without reapplying a charge. This is known as back-out rerating. You typically use this only for usage charges that should not have occurred. See “Running Back-Out-Only Rerating” for information.

How BRM Applies the Balance Impacts of Rerating

BRM rerates both billed and unbilled events and can rerate events that belong to multiple billing cycles. Therefore, BRM might need to distribute balance impacts across multiple balances.

When rerated events include noncurrency balance impacts, BRM rerating charges or credits the account accordingly. Financial impacts to accounts receivable (A/R), general ledger (G/L), and taxation are taken into account during rerating.

When an event is rerated, BRM backs out the event from the BRM database by creating an adjustment event that fully negates the original balance impacts. Adjustments are handled differently depending on if the event is billed or unbilled.

- If the event is unbilled, a shadow adjustment event is created. This event is called a shadow event because its balance impact is added to the original event’s bill item rather than an adjustment item. See “About Rerating Unbilled Events”.

- If the event has already been billed or posted to the G/L, a regular adjustment event is created. See “About Rerating Billed and Posted Events”.

When the total rerating adjustment is zero, BRM does not generate a rerating adjustment event if the balance impacts of rerating and previous rating are equivalent. To determine if the balance impacts are equivalent, BRM compares the values of certain balance impact fields for rerating with previous rating, such as the balance element ID and balance group. You can customize how BRM determines whether the balance impacts of rerating and previous rating are equivalent by modifying the event balance impact fields used for comparison. See “Configuring Rerating Adjustment Events”.

ECE moves failed rerating requests to the suspense queue.

You can run the pin_rerate utility at any time, but it’s best to run it on off-peak hours. If you configure BRM to create rerate jobs automatically, you should run the series of pin_rerate commands as part of a cron job.
About Rerating Unbilled Events

When unbilled events are rerated, a shadow adjustment event is added to the original event’s bill item instead of to an adjustment item.

Note: When recording a shadow event, you can customize your invoice to either show the details of rerating or show the result of rerating only in an end balance. See BRM Designing and Generating Invoices for information.

Shadow events are handled differently for usage events and recurring events:

- When unbilled usage events are rerated, the shadow adjustment event fully negates the original balance impacts and applies new balance impacts for the rerated amount.
- When unbilled recurring events are rerated, the shadow adjustment event fully negates the original balance impacts. Then a new recurring event is created that applies the rerated balance impacts. The new recurring event is the same type of event as the original event.

Adjustments When Billing Follows Rerating

When the purchase start time is the same as the current accounting cycle end date, the cycle forward fee balance impact is applied to the next month’s bill instead of the current month’s bill when billing is run after rerating is performed.

For example, suppose an account is created on January 1 with a cycle forward fee of $20, and the purchase start time is deferred by 1 month (set to the accounting cycle end date). When pin-rerate is run on February 1, rerate internally triggers automatic billing because the purchase start time is the same as the accounting cycle end date. The billing process changes the status of the bill item to open. Because the bill item status is now open, rerating applies the rerate adjustments to the next bill. When regular billing is run on February 2, the cycle fee is not applied to the January bill, instead it is applied to the February bill.

About Rerating Billed and Posted Events

When you rerate events that have already been billed, and unbilled events already posted to the G/L, an adjustment event fully negates the original balance impacts and applies new balance impacts for the rerated amount. The results of rerating are applied to the adjustment item in the next bill, which is the bill for the current cycle. For billed cycle fee, fold, rollover, and cycle discount events, a new cycle event is then created that applies the rerated balance impacts.

When billing is run, if a noncurrency balance is zero, no cycle event is generated for that balance. For example, if there are no remaining minutes to roll over at billing time, no rollover event is created. However, if rerating creates a nonzero balance that was previously zero when billed, the rerating process generates the appropriate cycle event.

Determining the G/L Entry for an Event

The general ledger (G/L) entry is determined at the time of rating. If you rerate as a result of pricing changes, the G/L entry could change. Whether you record a shadow event or an adjustment event, you must determine the correct balance to be posted in the G/L.
When recording a shadow event, the G/L ID of the rerating balance impacts have the same G/L ID as the original event’s balance impacts. If you changed the G/L ID after the original event was rated, the balance impacts of rerating uses the new G/L ID.

When recording an adjustment, you can configure the GLID to use. For example, you can use the same GLID as the original event, a new GLID, or an adjustment GLID (a separate GLID bucket to record all adjustments as part of rerating). The adjustment G/L ID can be the same or different than the GLID used for regular (not rerated) event adjustments. For information about setting up G/L IDs, see BRM Collecting General Ledger Data.

**How Rerating Affects Account Migration**

When an account is selected for rerating, the account cannot be migrated to another database until rerating is complete. Otherwise, the account is not rerated. For this reason, the Account Migration Manager (AMM) does not allow migration of an account to another database if the account is being rerated.

However, the account is migrated if the rerate job status is NEW. In this case, the AMM deletes the account from the rerate job in the source database and creates a new rerate job with the account in the destination database.

---

**Note:** After the account migration is complete, you must run `pin_rerate` in the destination database to rerate the account.

---

**How BRM Generates Rerated Events**

BRM rerates events in chronological order:

- **Usage events** are rerated in order starting with the earliest usage event.

- **Rollover, fold, cycle discount, and cycle fee events** are generated in the order in which they logically occur: for example, billing-time discount events are generated before rollover events.

There are three ways in which rerating generates events:

- **Rerate the original event.**
  
  This process passes the event being rerated to ECE for rerating. This is the most basic rerating process.

- **Reapply business logic based on information in the original event.**
  
  This process passes information from the event being rerated to the opcode that generated the event. The opcode may or may not generate a new event. This process is used when the event attributes may be different after rerating. For example, a charge offer’s purchase fee may have been changed since the purchase fee event was generated.

  Usually, purchase events and cancellation events are not directly rerated. Instead, information in the original purchase event and cancel event is used to generate new fee events, if applicable.

- **Reapply business logic independent of the original event.**
  
  This process performs business logic that might generate new events even if there is no existing event. With this process, information in the original event is not used. Instead, the business logic that generated the original event is reapplied and a new event is generated, if appropriate.
For example, when rerating an account for a specific period, if the events selected for rerating are associated with charge offers that have a cycle fee, rerating calls the cycle fee opcode and generates a new cycle fee event based on the charge offer's current rates. This is done independent of any existing cycle fee event. If there is an existing cycle fee event, it is replaced by the corresponding new cycle fee event generated by rerating.

In some cases, the purchase- and cancellation-fee events are directly rerated because the purchase and cancellation events may not be available for rerating: for example, when rerating deferred purchase fees or when using selective rerating in which only purchase or cancellation-fee events are specified.

The following events are not rerated but are reapplied because they were not originally generated by rating:

- /event/billing/adjustment
- /event/billing/debit
- /event/billing/dispute
- /event/billing/item
- /event/billing/settlement
- /event/billing/payment
- /event/billing/reversal
- /event/billing/writeoff
- /event/billing/cycle/tax
- /event/billing/refund
- /event/billing/charge

Table 1–1 shows the rerating process for specific types of events.

<table>
<thead>
<tr>
<th>Event Type</th>
<th>Order of Rerating</th>
<th>How Rerated</th>
</tr>
</thead>
<tbody>
<tr>
<td>Usage</td>
<td>Chronological, based on event time.</td>
<td>Rerate the original event.</td>
</tr>
<tr>
<td>Cycle fee</td>
<td>Generated according to the time that the charge offer is purchased or canceled or when the associated accounting cycle starts or ends, and based on the logical order of events.</td>
<td>Reapply business logic independent of the original event.</td>
</tr>
<tr>
<td>Rollover</td>
<td>Generated according to the time that the associated billing cycle ends, and based on the logical order of events.</td>
<td>Reapply business logic independent of the original event.</td>
</tr>
<tr>
<td>Fold</td>
<td>Generated according to the time that the associated billing cycle ends, and based on the logical order of events.</td>
<td>Reapply business logic independent of the original event.</td>
</tr>
<tr>
<td>Billing-time discount</td>
<td>Generated according to the time that the associated billing cycle ends, and based on the logical order of events.</td>
<td>Reapply business logic independent of the original event.</td>
</tr>
<tr>
<td>Purchase</td>
<td>Chronological, based on event time.</td>
<td>Reapply business logic based in information in the original event.</td>
</tr>
</tbody>
</table>
How Failed Rerate Jobs Are Processed

The accounts in a rerate job are typically processed individually in separate rerate operations. When rerating fails for one or more accounts in a rerate job, the pin_rerate utility sets the status of the accounts in the rerate job batch to FAILED. When the rerating process is complete for the rerate job, pin_rerate creates a new rerate job consisting of only the accounts that failed. The new rerate job is processed the next time pin_rerate is run.

If an account selected for rerating is associated with a subscription service that was transferred during the period for which rerating is specified, then the account to which the service was transferred is included in the rerate job and all those accounts are rerated concurrently in a single rerate operation. When rerating fails for one of these accounts, then rerating fails for all accounts in the rerate request. In this case, pin_rerate creates a new rerate job containing all the accounts in the rerate request.

For example, subscription service X is originally owned by Account A and transferred to Account B on June 15. Later in the month, it is transferred from Account B to Account C. Rerating of Account A from June 1 also results in rerating of accounts B and C. Accounts A, B, and C are grouped together in a single rerate request. If rerating fails for any of these accounts, pin_rerate creates a new rerate job consisting of all three accounts in a single rerate request. The accounts are rerated again the next time pin_rerate is run.

### Table 1–1 (Cont.) Rerating Process and Event

<table>
<thead>
<tr>
<th>Event Type</th>
<th>Order of Rerating</th>
<th>How Rerated</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cancellation</td>
<td>Chronological, based on event time.</td>
<td>Reapply business logic based in information in the original event.</td>
</tr>
</tbody>
</table>
This chapter describes how to configure the Oracle Communications Billing and Revenue Management (BRM) system to enable rerating.

Topics in this document:

- About Configuring Rerating
- Configuring the Connection Manager for Rerating
- Creating Rerate Jobs Automatically

See also:

- About Rerating Events
- Rerating Events
- Advanced Rerating Tasks
- Tuning Rerating Performance

**About Configuring Rerating**

To configure rerating:

- Configure Connection Managers to enable rerating. See "Configuring the Connection Manager for Rerating."
- Configure automatic creation of rerate jobs. See "Creating Rerate Jobs Automatically" for information.
- Configure optional rerating functions. See "Advanced Rerating Tasks".
- Configure performance tuning settings. See "Tuning Rerating Performance" for information.

**Configuring the Connection Manager for Rerating**

ECE uses ECE EM Gateway to connect a CM with ECE. To configure a CM for ECE rerating:

1. Open the CM configuration file (`BRM_home/sys/cm/pin.conf`) in a text editor.
2. Make sure that this entry is in the file:
   ```bash
   - cm fm_module BRM_home/lib/fm_rerate.so fm_rerate_config - pin
   ```
3. Make sure that this entry is in the file:
   ```bash
   - cm em_group rerating PCM_OP_RATE_ECE_EVENT
   ```
4. Edit the `em_pointer` entry to match your ECE EM Gateway configuration:

   ```
   - cm em_pointer rerating ip em_gateway_host em_gateway_port
   ```

   where:
   - `em_gateway_host` is the IP address of the machine on which the ECE EM Gateway resides.
   - `em_gateway_port` is the port of the machine on which the ECE EM Gateway resides.

5. Save and close the file.

6. Stop and restart the CM.

---

**Creating Rerate Jobs Automatically**

BRM can automatically create rerate jobs when events must be rerated. Automatically creating rerate jobs does not run the `pin_rerate` utility. The rerate jobs are processed the next time you run the `pin_rerate` utility. If you create rerating jobs automatically, you should run the `pin_rerate` utility from a cron job.

By default, BRM supports automatic rerate job creation in the following cases:

- To rerate backdated events. See "Creating Rerate Jobs for Backdated Events" for information.
- To rerate cycle forward and cycle forward arrears charges when pricing changes for those charges. See "Rerating Cycle Forward and Cycle Forward Arrears Charges" for information.
- To correct rollover amounts when delayed events are processed. See "Enabling Rerating and Rollover Correction Due to Delayed Events" for information.

To create rerate jobs automatically, you must enable the BRM server. See "Enabling and Configuring Automatic Rerate Job Creation" and "Configuring Event Notification for Automatic Rerate Job Creation" for information.

---

**Creating Rerate Jobs for Backdated Events**

BRM can automatically create rerate jobs for backdated events in the following cases:

- When purchase or cancellation of a charge offer, discount offer, or bundle is backdated.
- When an extended rating attribute (ERA) modification is backdated. For example, a subscriber changes their service-level agreement from Silver to Gold on July 12. The CSR backdates the change to July 1 to let the subscriber apply the Gold-level benefits to all usage for July. BRM rerates all the relevant events for the account that occurred between July 1 and the current time.
- When an adjustment to a noncurrency balance is backdated.

For example, a subscriber’s billing day is on the first day of the month. On February 15, the included minutes balance is adjusted from 100 to 500 and the adjustment is backdated to January 15. When rerating is run on February 20, BRM rerates all the relevant events from January 15 to February 20. Billing events that previously occurred on February 1, such as billing-time discounts, rollovers, and folds, are recalculated based on the backdated amount of minutes.
By default, BRM creates rerate jobs for backdated events if both of these criteria are met:

- The backdated event time is at least one hour earlier than the current time.
- The backdated event date is not older than one billing cycle.

For example, a subscriber’s billing day is on the first day of the month. The subscriber purchases a charge offer on July 19 at 5:00 a.m. The CSR backdates the purchase start date to July 15. BRM validates the following:

- The backdated purchase start time is at least one hour earlier than the current time, in this case, on or before 4:00 a.m., July 19.
- The backdated purchase start date is not earlier than one billing cycle, which in this case started on June 1.

Because both conditions are met, BRM creates rerate jobs for all the account’s events that occurred after midnight of July 14 to July 20.

You can configure BRM to use different values to determine the backdated event time and backdated event date. See “Enabling and Configuring Automatic Rerate Job Creation” for information.

You can override the backdated event date and rerate older events by running the pin_rerate utility manually. To enable this feature, see “Allowing Rerating of Old Events” for information.

You can customize automatic rerate job creation by specifying which events trigger rerating. For example, you can set up a charge offer cancellation event to automatically create rerate jobs. To customize automatic rerate job creation, you edit the PCM_OP_SUBSCRIPTION_POL_GENERATE_RERATE_REQUEST policy opcode or write your own custom opcode. See BRM Opcode Guide.

Rerating Cycle Forward and Cycle Forward Arrears Charges

**Important:** Rerating recurring events can affect performance.

You can configure BRM to create rerate jobs automatically to rerate cycle forward and cycle forward arrears events after they are charged. See “Rerating Cycle Forward and Cycle Forward Arrears Charges”.

To enable BRM to rerate cycle forward and cycle forward arrears charges:

1. Open the CM) configuration file (BRM_home/sys/cm/pin.conf) in a text editor.
2. Ensure that the following entries are set to 1:
   - fm_subscription rate_change 1
   - fm_price log_price_change event 1
3. Save the file.
4. Stop and restart the CM.

Enabling Rerating and Rollover Correction Due to Delayed Events

If a delayed event arrives after the end of the accounting cycle and during the delayed billing period, the event can borrow against the rollover of the current cycle even when the current rollover is consumed by events of the current cycle.
Creating Rerate Jobs Automatically

If rerating and rollover correction is enabled and delayed events borrow from the rollover of the current cycle, the current cycle events are rerated and the rollover is reallocated so that it comes from the appropriate cycles.

To enable rerating and rollover correction, you must:

- Enable event notification for the feature. Add the following line to the pin_notify file in BRM_home/sys/data/config.

```
3787 0 /event/notification/rollover_correction/rerate
```

See "Configuring Event Notification for Automatic Rerate Job Creation" for information.

- Enable the feature by running the pin_bus_params utility to change the RerateDuringBilling business parameter. For information about this utility, see BRM Developer’s Guide.

To enable rerating and rollover correction:

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 file, change disabled to enabled:

```
<RerateDuringBilling>enabled</RerateDuringBilling>
```

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

```
pin_bus_params bus_params_billing.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.

For information about setting the length of the delayed billing period, see BRM Configuring and Running Billing.

Enabling and Configuring Automatic Rerate Job Creation

To enable and configure automatic rerate job creation:

1. Open the CM configuration file (BRM_home/sys/cm/pin.conf) in a text editor.

2. To turn automatic rerate job creation on or off, set the backdate_trigger_auto_rerate entry: 1 = enabled; 0 = disabled.

```
-fm_subs backdate_trigger_auto_rerate 1
```

3. To specify the minimum time difference necessary to allow an event to be rerated, set the backdate_window entry. This is the amount of time in seconds between the current time and the time that the backdating is applied. The default is 3600 seconds (1 hour).

```
-fm_subs backdate_window 3600
```

4. To specify the maximum number of billing cycles allowed between the current time and the backdated event date for rerating events, set the num_billing_cycles entry. The default is 1 billing cycle.
5. Save and close the file.
6. Stop and restart the CM.

Configuring Event Notification for Automatic Rerate Job Creation

BRM uses event notification to create rerate jobs automatically. Event notification triggers the PCM_OP_SUBSCRIPTION_POL_GENERATE_RERATE_REQUEST policy opcode (opcode number 3787).

To configure notification events for automatic rerating:

1. Edit the `pin_notify` file in `BRM_home/sys/data/config`.

   The default automatic rerate job creation entries are:
   
   - 3787 0 /event/notification/auto_rerate
   - 3787 0 /event/notification/rate_change
   - 3787 0 /event/notification/rollover_correction/rerate

   To rerate recurring events triggered by first usage, add these:
   
   - 9071 0 /event/billing/product/fee/cycle/cycle_arrear
   - 9071 0 /event/billing/product/fee/cycle/cycle_forward_annual
   - 9071 0 event/billing/product/fee/cycle/cycle_forward_arrear
   - 9071 0 /event/billing/product/fee/cycle/cycle_forward_bimonthly
   - 9071 0 /event/billing/product/fee/cycle/cycle_forward_monthly
   - 9071 0 /event/billing/product/fee/cycle/cycle_forward_quarterly
   - 9071 0 /event/billing/product/fee/cycle/cycle_forward_semiannual
   - 3787 0 /event/notification/rollover_correction/rerate

   See "Enabling Correction of First-Usage Validity Dates".

   To rerate cycle forward and cycle forward arrears events when a charge changes mid-cycle, add these:
   
   - 3788 0 /event/audit/price/product_update
   - 3788 0 /event/audit/price/product_complete

   See "Rerating Cycle Forward and Cycle Forward Arrears Charges".

   To enable rerating and rollover correction, added this line:
   
   - 3787 0 /event/notification/rollover_correction/rerate

   See "Enabling Rerating and Rollover Correction Due to Delayed Events".

2. Run the `load_pin_notify` utility. See "Loading the Event Notification List" in BRM Developer's Guide.
Advanced Rerating Tasks

This document describes advanced configuration tasks for Oracle Communications Billing and Revenue Management (BRM) rerating.

Topics in this document:

- Rerating Events when the Rating Conditions Change During the Session
- Allowing Rerating of Old Events
- Enabling Correction of First-Usage Validity Dates
- Configuring Rerating Adjustment Events
- Configuring How to Allocate Rerating Results
- Enabling Deferred Tax Calculation during Rerating

See also:

- About Rerating Events
- Configuring Rerating
- Rerating Events
- Tuning Rerating Performance

Rerating Events when the Rating Conditions Change During the Session

You can configure rerating to account for midsession changes in the rating conditions. For example, a subscriber might initiate a Friends and Family discount during a call, which changes the pricing midsession. When the call is rerated:

- If you enable this option, rerating uses the original pricing from the start time of the call until the called number is added to the Friends and Family list, and then uses the discounted pricing for the remaining session.
- If you do not enable this option, rerating uses the discounted pricing for the entire call.

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

To enable rerating when the rating conditions change during the session:

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

   ```
   pin_bus_params -r BusParamsRerate bus_params_rerate.xml
   ```
Allowing Rerating of Old Events

When you configure automatic rerate job creation, you use the `num_billing_cycles` pin.conf file entry to limit the backdated events that can be rerated to a specified number of billing cycles in the past. This setting also applies to manually rerating events. To manually rerate backdated events older than the allowable date, you enable the `AllowBackdateNoRerate` business parameter.

To enable rerating older events when they are backdated:

- Automatic rerate job creation must be enabled. See "Enabling and Configuring Automatic Rerate Job Creation" for information.
- You must run the `pin_bus_params` utility to change the `AllowBackdateNoRerate` business parameter. For information about this utility, see BRM Developer’s Guide.
- After this option is enabled, run the `pin_rerate` utility manually, including the selection criteria necessary to rerate the events.

To enable rerating of old events:

1. Go to `BRM_home/sys/data/config`.
2. Create an XML file from the `/config/business_params` object:
   ```
   pin_bus_params -r BusParamsSubscription bus_params_subscription.xml
   ```
3. In the file, change `disabled` to `enabled`:
   ```
   <AllowBackdateNoRerate>enabled</AllowBackdateNoRerate>
   ```
4. Save the file as `bus_params_subscription.xml`.
5. Load the XML file into the BRM database:
   ```
   pin_bus_params bus_params_subscription.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.

Enabling Correction of First-Usage Validity Dates

When rerated events are associated with offers or balances that start on first usage, rerating resets their validity periods, if necessary. For example, if the first event rated, which initiated an offer’s validity period, was not the first event to use the offer’s
Configuring Rerating Adjustment Events

When usage events are rerated, BRM creates an adjustment event to apply the balance impacts of rerating. The balance impact of the adjustment event is the difference between the previous rating results and the rerated results.

When rerating produces the same result as previous rating, creating an adjustment event is not necessary. However, even when the rerated result is the same as the original result, there might be differences in the rating and rerating results in some of the individual balance impacts that contributed to the total charge. Therefore, before it can determine that rerating resulted in no overall change in the balance impact, BRM must ensure that the balance impacts in each rating process are equivalent.

To determine whether the balance impacts of rerating and previous rating are equivalent, BRM checks a list of values in all of the balance impacts. For example, BRM checks to see if the same charge offer was used for all balance impacts. If the balance impact fields have the same values in the event for both rerating and previous rating, and the total rerating adjustment is zero, the results of rerating and previous rating are considered equivalent and no adjustment event is created.

BRM uses the values in the following fields:

- The ID of the impacted balance element
- The ID of the impacted balance group
- The G/L ID
- The tax code
- The charge offer used to rate the event

You can modify the fields that BRM uses to determine whether balance impacts are equivalent. For example, you can specify the impact category field to record the rerating adjustment when calls are made from different locations even though rerating results in the same total charge.

To customize the list of fields, you modify the `pin_rerate_compare_bi.xml` file and then run the `load_pin_rerate_flds` utility to load the contents of the file into the `/config/rerate_flds/compare_bi` object in the BRM database.
To customize the fields used to compare the event balance impacts of rerating with previous rating:

1. Edit the `pin_rerate_compare_bi.xml` file in `BRM_home/sys/data/config`. Add a `RerateCompareBalImpacts` element for each event balance impact field. The following balance impact fields are mandatory:
   - `PIN_FLD_RESOURCE_ID`
   - `PIN_FLD_BAL_GRP_OBJ`
   - `PIN_FLD_GL_ID`
   - `PIN_FLD_TAX_CODE`

   You can specify any additional field from the `/event` object’s `PIN_FLD_BAL_IMPACTS` array.

2. Save and close the file.

3. Use the following command to load the contents of the XML file into the `/config/rerate_flds/compare_bi` object:

   ```bash
   load_pin_rerate_flds pin_rerate_compare_bi.xml
   ```

   If you do not run the utility from the directory in which the XML file is located, you must include the complete path to the file. For example:

   ```bash
   load_pin_rerate_flds BRM_home/sys/data/config/pin_rerate_compare_bi.xml
   ```

   For more information, see "load_pin_rerate_flds".

4. Stop and restart the CM.
To verify that the balance-impact comparison fields were loaded, you can display the `/config/rerate_flds/compare_bi` object by using Object Browser, or use the `robj` command with the `testnap` utility.

## Configuring How to Allocate Rerating Results

By default, BRM creates unallocated items for any adjustment items created as a result of rerating. You can enable BRM to allocate adjustment items to their original bill.

- For open item accounting, the adjustment items are allocated to each bill that included events rerated and resulted in adjustments.
- For balance forward accounting, adjustment items are allocated to the last bill only.

If BRM has rerated a `/bill` object without allocating automatic adjustments (from the rerating) to the original bills, the corrective bill for that `/bill` object does not include the item adjustments generated by that rerating. For such a `/bill` object, if you set the `AllocateReratingAdjustments` business parameter to `enabled` and rerun the rerating process for the bill, BRM does not allocate the adjustments. You must manually allocate the rerated items before you generate the corrective bill for such a `/bill` object.

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

To allocate adjustment items to their original bill during rerating:

1. Go to `BRM_home/sys/data/config`.
2. Create an XML file from the `/config/business_params` object:
   ```
   pin_bus_params -r BusParamsRerate bus_params_rerate.xml
   ```
3. In the file, change `disabled` to `enabled`:
   ```xml
   <AllocateReratingAdjustments>enabled</AllocateReratingAdjustments>
   ```
4. Save the file as `bus_params_rerate.xml`.
5. Load the XML file into the BRM database:
   ```
   pin_bus_params bus_params_rerate.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`.

## Enabling Deferred Tax Calculation during Rerating

By default, BRM calculates taxes on any deferred taxable amount in the rerated events during the subsequent bill run. The rerated tax appears on the invoice for the subsequent bill run.

You can configure BRM to apply deferred taxes during rerating. This enables BRM to include the deferred tax amounts on corrected invoices.

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

To enable calculation of deferred taxes during rerating:
1. Go to BRM_home/sys/data/config.
2. Create an XML file from the /config/business_params object:
   
   ```bash
   pin_bus_params -r BusParamsRerate bus_params_rerate.xml
   ```

3. In the file, change `disabled` to `enabled`:
   
   ```xml
   <ApplyDeferredTaxDuringRerating>enabled</ApplyDeferredTaxDuringRerating>
   ```

4. Save the file as `bus_params_rerate.xml`.
5. Load the XML file into the BRM database:
   
   ```bash
   pin_bus_params bus_params_rerate.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*. 
This chapter describes how to use the Oracle Communications Billing and Revenue Management (BRM) `pin_rerate` utility to rerate events.

Topics in this document:

- Overview
- Running Rerating
- Selecting Events for Rerating
- Using Reason Codes to Specify the Rerating Order
- Rerating Cycle Forward and Cycle Forward Arrears Events
- Running Back-Out-Only Rerating

See also:

- About Rerating Events
- Configuring Rerating
- Advanced Rerating Tasks
- Tuning Rerating Performance

**Overview**

To rerate events, you create rerate jobs, and then run the `pin_rerate` utility several times, each time with a different command. In most cases, you configure automatic rerate job creation. In that case, when you run rerating, you do not need to run the `pin_rerate` utility to create rerating jobs.

This series of commands assumes that rerate jobs have already been created automatically:

```
pin_rerate -process jobs
pin_rerate -process queue
pin_rerate -rerate
pin_rerate -process queue
pin_rerate -process recycle
```

To create rerate jobs and then run rerating, the commands are the same, but preceded by the command that creates the rerate job; for example:

```
pin_rerate -d bundle_rerate.txt -t 07/23/2015 -r
pin_rerate -process jobs
pin_rerate -process queue
pin_rerate -rerate
```
Pin_rerate -process queue
Pin_rerate -process recycle

When events are selected, the pin_rerate utility creates rerate jobs for the related accounts.

The pin_rerate utility includes several options for defining the events to rerate. See "Selecting Events for Rerating" for information.

The rerating process shown in the preceding commands includes processing events for accounts suspended for the rerating process. You can run the recycle command at any time. See "Recycling Records Suspended during Rerating" for information.

The rerating process without event selection shown above is the same process used for processing failed events. See "Processing Failed Rerate Jobs" for information.

You can also run the pin_rerate utility with the following options:

- Use reason codes to specify the order of rerating. See "Using Reason Codes to Specify the Rerating Order" for information.
- Specify the order in which events are rated by event creation time, or when they were loaded into the BRM database. See "Specifying the Event Sequence for Rerating".
- Back out the charges for events, but not rerate. See "Running Back-Out-Only Rerating".

Periodically, you must use the pin_rerate utility to purge completed and failed rerate jobs. See "Purging Rerate Jobs" for information.

**Running Rerating**

To rerate events, run the pin_rerate commands in the following order:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>pin_rerate event_selection -t date -r</td>
<td>(Optional) Select events to rerate</td>
</tr>
<tr>
<td>pin_rerate -process jobs</td>
<td>Process new rerate jobs</td>
</tr>
<tr>
<td>pin_rerate -process queue</td>
<td>Process rerate jobs ready for rerating</td>
</tr>
<tr>
<td>pin_rerate -rerate</td>
<td>Rerate events</td>
</tr>
<tr>
<td>pin_rerate -process queue</td>
<td>Acknowledge rerating results</td>
</tr>
<tr>
<td>pin_rerate -process recycle</td>
<td>Process suspended events</td>
</tr>
</tbody>
</table>

See "About the Rerating Process" for a description of rerating.

You typically run the pin_rerate utility in a cron job. See "pin_rerate" for information about the syntax.

---

**Important:** Do not move accounts to another database schema while rerating events for those accounts.

**Generating Rerating Reports**

To generate reports when you rerate events, you use the -rerate option with the -l parameter. This example creates a summary report:
pin_rerate -rerate -l s

This example creates a detailed report:

pin_rerate -rerate -l d

If you do not use the -l parameter, the rerating process generates both summary and detailed reports by default. You can also generate both types by using this command:

pin_rerate -rerate -l sd

Report generation is resource intensive and can degrade system performance. You can not generate any report by using the -l n option, for example:

pin_rerate -rerate -l n

### Recycling Records Suspended during Rerating

Temporarily suspended records are loaded into the BRM database by the Suspended Event (SE) Loader. The suspended records are stored in the database until they are recycled.

```
Important: Before you can recycle suspended records, they must be loaded into the BRM database by SE Loader. You typically schedule SE Loader to run automatically when you set up standard recycling.
```

To recycle the records suspended during the rerating process, run `pin_rerate` with the -process recycle parameter.

```
Note: Suspended records are typically recycled by running the `pin_recycle` utility. However, `pin_rerate` calls the standard recycling opcode directly so you do not use `pin_recycle` when using `pin_rerate`.
```

`pin_rerate` finds all the rerate jobs with a status of READY_FOR_RECYCLE and calls the standard recycling opcodes to recycle the associated records. Standard recycling uses the recycle key value in the record to identify and retrieve records suspended during the rerating process.

After the records are recycled, `pin_rerate` changes the status of the jobs from READY_FOR_RECYCLE to COMPLETE.

```
Note:

■ In a multischema environment, you must run `pin_rerate` separately on each database schema.

■ If no records were suspended for the accounts being rerated, the job status is still changed to COMPLETE.

■ If an error occurs while recycling records, the job status is not changed; it retains the status of READY_FOR_RECYCLE.
```
Selecting Events for Rerating

Processing Failed Rerate Jobs

If rerating fails, `pin_rerate` creates a report that includes the account numbers and start times for failed rerate jobs. The report file name is `pin_rerate.status_report`, and is in the directory from where you ran the utility.

To process failed rerate jobs, you run `pin_rerate` only with the commands for processing rerate jobs, without specifying selection criteria or rerate reason codes.

---

**Note:** This processes all rerate jobs in the rerate queue, not just failed rerate jobs.

---

For example:

```
pin_rerate -process jobs
pin_rerate -process queue
pin_rerate -rerate
pin_rerate -process queue
pin_rerate -process recycle
```

Purging Rerate Jobs

Purging rerate jobs deletes all rerate jobs that have a rerate job status of COMPLETE or UNSUCCESSFUL.

Use the following command to purge rerate jobs:

```
pin_rerate -purge
```

You can purge rerate jobs that occurred before a specified date and time, for example:

```
pin_rerate -purge -t 05/21/2016
```

Selecting Events for Rerating

To select events for rerating, you run the rerating commands and specify the events to rerate first. For example, this series of commands rerates only the events associated with accounts that used one or more bundles listed in the `bundle_rerate.txt` file:

```
pin_rerate -d bundle_rerate.txt -t 07/23/2015
pin_rerate -process jobs ]
pin_rerate -process queue
pin_rerate -rerate
pin_rerate -process queue
pin_rerate -process recycle
```

The `pin_rerate` utility creates rerate jobs for the events specified in the first command, and then processes them.

To select events for rerating, BRM chooses accounts to rerate based on selection criteria. For example, the following command uses the `-s` parameter to find all accounts that have events related to the list of services defined in the `service_rerate.txt` file:

```
pin_rerate -s service_rerate.txt -t 01/23/2016
```
From those events, BRM selects all events generated by those accounts after January 23. Rerated events do not need to be related to defined services; the only requirement is that the accounts are associated with the specified services.

---

**Note:** Using the `-t` parameter is required for all event selection parameters.

---

In the following example, BRM finds all accounts associated with a list of charge offers. BRM then rerates all the events for those accounts that occurred after January 23, even if the events were not rated by the charge offers listed. This is to ensure that related charges, such as granting minutes, are rerated accordingly.

```
pin_rerate -p charge_offer_rerate.txt -t 01/23/2016
```

---

**Note:** You can use the `-r` parameter to select events based on additional criteria. For example, if you select events based on a service, you can specify that after BRM finds all the associated accounts, it only selects events related to that service, not to any service associated with the account. See "Selecting Events for Rerating" for information.

---

Most of the event selection parameters use an input text file to specify the criteria. For example, the `-p` parameter uses a list of charge offers to select events to rerate.

Table 4–2 shows the `pin_rerate` parameters used for selecting events to rerate.

**Table 4–2  pin_rerate Event Selection Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>-r</code></td>
<td>Specifies that the command rerates only selected events, as opposed to all of the events that apply to the accounts selected for rerating. See &quot;Selecting Events for Rerating&quot; for information.</td>
</tr>
<tr>
<td><code>-t</code></td>
<td>Specifies the time from when to rerate events. This parameter is mandatory when selecting accounts and events to rerate.</td>
</tr>
<tr>
<td><code>-a</code></td>
<td>Rerates events associated with a single account. See &quot;Rerating Events for a Single Account&quot; for information.</td>
</tr>
<tr>
<td><code>-c</code></td>
<td>When rerating events for a single account, this parameter specifies to calculate the rerate amount without updating the BRM database. This parameter works only with the <code>-a</code> parameter. See &quot;Rerating Events for a Single Account&quot; for information.</td>
</tr>
<tr>
<td><code>-m</code></td>
<td>Rerates events associated with one or more specified accounts. See &quot;Rerating Events for a List of Accounts&quot; for information.</td>
</tr>
<tr>
<td><code>-line</code></td>
<td>Rerates events for accounts associated with a single subscriber service ID, such as a phone number. See &quot;Rerating Events Associated with a Subscriber Service ID&quot; for information.</td>
</tr>
<tr>
<td><code>-n</code></td>
<td>Rerates events for accounts associated with one or more types of events. See &quot;Rerating Events Associated with Specific Event Types&quot; for information.</td>
</tr>
<tr>
<td><code>-s</code></td>
<td>Rerates events for accounts associated with one or more services. See &quot;Rerating Events Associated with Services&quot; for information.</td>
</tr>
</tbody>
</table>
Selecting Events for Rerating

Rerating Events for a Single Account

Use the -a parameter to rerate events for a single account. To use this parameter, you enter the account POID object ID. For example, if the account POID is:

```
0.0.0.1 /account 8606 0
```

The POID object ID is 8606. You enter the following:

```
pin_rerate -a 8606 -t 07/23/2007
```

You can run the `pin_rerate` utility in calculate-only mode. This performs rerating but does not update anything in the BRM database. You can only use calculate-only mode when you use the -a parameter.

To use calculate-only mode, use the -c parameter:

```
pin_rerate -a 8606 -c -t 07/23/2007
```

You cannot use the -a parameter with any of the other account selection criteria parameters (-d, -g, -i, -m, -n, -p, -s, -line, and -field_name):

Rerating Events for a List of Accounts

BRM always chooses events based on the accounts that meet the selection criteria. However, you can rerate events generated by specified list of accounts. To do so, you create a list of accounts, and use the -m parameter. For example:

```
pin_rerate -m accounts_to_rerate.txt -t 01/23/2016
```

In the text file, specify each account in a results array that lists account POIDs. For example:

---

Table 4–2 (Cont.) `pin_rerate` Event Selection Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-p</td>
<td>Rerates events for accounts associated with one or more charge offers. See &quot;Rerating Events Associated with Charge Offers&quot; for information.</td>
</tr>
<tr>
<td>-i</td>
<td>Rerates events for accounts associated with one or more discount offers. See &quot;Rerating Events Associated with Charge Offers&quot; for information.</td>
</tr>
<tr>
<td>-d</td>
<td>Rerates events for accounts associated with one or more bundles. See &quot;Rerating Events Associated with Charge Offers&quot; for information.</td>
</tr>
<tr>
<td>-g</td>
<td>Rerates events for accounts associated with one or more bill units or balance groups. See &quot;Rerating Events Associated with a Balance Group&quot; for information.</td>
</tr>
<tr>
<td>-field_name</td>
<td>Rerates events for accounts based on any field in an event. See &quot;Rerating Events Associated with a Custom Event Field&quot; for information.</td>
</tr>
</tbody>
</table>
| -e | Specifies to estimate the number of events that could be impacted. You can only use this parameter when selecting the following:  

- A single account (-a)  
- A list of services (-s)  
- A list of events (-n)  
- A list of charge offers (-p)  
- A list of bundles (-d)  

See "Getting an Estimate for the Number of Rerated Events" for information. |

---

Rerating Events for a Single Account

Use the -a parameter to rerate events for a single account. To use this parameter, you enter the account POID object ID. For example, if the account POID is:

```
0.0.0.1 /account 8606 0
```

The POID object ID is 8606. You enter the following:

```
pin_rerate -a 8606 -t 07/23/2007
```

You can run the `pin_rerate` utility in calculate-only mode. This performs rerating but does not update anything in the BRM database. You can only use calculate-only mode when you use the -a parameter.

To use calculate-only mode, use the -c parameter:

```
pin_rerate -a 8606 -c -t 07/23/2007
```

You cannot use the -a parameter with any of the other account selection criteria parameters (-d, -g, -i, -m, -n, -p, -s, -line, and -field_name):

Rerating Events for a List of Accounts

BRM always chooses events based on the accounts that meet the selection criteria. However, you can rerate events generated by specified list of accounts. To do so, you create a list of accounts, and use the -m parameter. For example:

```
pin_rerate -m accounts_to_rerate.txt -t 01/23/2016
```

In the text file, specify each account in a results array that lists account POIDs. For example:
You cannot use the -m parameter with any of the other account selection criteria parameters (-a, -d, -g, -i, -n, -p, -s, -line, and -field_name).

**Rerating Events Associated with Charge Offers**

You can rerate the following:

- Use the -p parameter to rerate accounts associated with events rated by one or more specified charge offers. For example:
  
  ```
  pin_rerate -p charge_offer_rerate.txt -t 01/23/2016
  ```

- Use the -i parameter to rerate accounts associated with events rated by one or more specified discount offers. For example:
  
  ```
  pin_rerate -i discount_offer_rerate.txt -t 01/23/2016
  ```

- Use the -d parameter to rerate accounts associated with charge offers or discount offers that belong to one or more specified bundles. For example:
  
  ```
  pin_rerate -d bundle_rerate.txt -t 01/23/2016
  ```

  If a charge offer or discount offer is part of multiple bundles, all accounts that purchase the charge offer or discount offer are selected for rerating even if they did not purchase the bundle that was rerated.

  For each parameter, the input text file is a list of offer or bundle names, for example:

  ```
  ChargeOffer_1
  ChargeOffer_2
  ChargeOffer_3
  ```

  The names are case-sensitive and must match the names as defined in PDC.

  The -d, -i, and -p parameters are mutually exclusive. In addition, you cannot use these parameters with any of the other account selection criteria parameters (-a, -g, -n, -s, -line, and -field_name).

**Rerating Events Associated with Specific Event Types**

Use the -n parameter to rerate events for accounts associated with one or more types of events. For example:

```
pin_rerate -n event_rerate.txt -t 01/23/2016
```

The input text file is a list of events, for example:

```
/event/session/gsm/voice
/event/session/gsm/data
```

If you use the -n parameter with the -r parameter to rerate only the events of the specified types of events, all subclasses of the events specified in the input file are also rerated. For example, if you specify /event/delayed/session/telco in the -n parameter input file, events of type /event/delayed/session/telco/gsm that meet the selection criteria are also rerated.
Rerating Events Associated with Services

Use the -s parameter to rerate events for accounts associated with one or more services. For example:

```
pin_rerate -s service_rerate.txt -t 01/23/2016
```

The input text file is a list of services, for example:

```
/service/telco/gsm/data
/service/telco/gsm/telephony
/service/telco/gsm/sms
```

Rerating Events Associated with a Subscriber Service ID

Use the -line parameter to rerate events for all accounts associated with a specific subscriber service ID. For example, you can select only the accounts that have the phone number 6495832245:

```
pin_rerate -line 6495832245 -t 01/23/2016
```

The ID is the caller ID such as a phone number or the MAC address.

The selected accounts can be further restricted by selecting accounts associated with:

- Charge offers (-p)
- Discount offers (-i)
- Bundles (-d)
- Events (-n)
- Services (-s)
- Accounts (-m)
- Custom fields (-field_name)

Rerating Events Associated with a Balance Group

Use the -g parameter to rerate events associated with the balance groups belonging to a specified account. For example:

```
pin_rerate -g balance_groups.txt -t 01/23/2016
```

The input text file is an array of POIDS for accounts, bill units, and balance groups. To rerate all events for all balance groups for an account’s bill unit, specify a value of NULL as the POID ID for the balance group, as shown in the following example.

```
0 PIN_FLD_RESULTS ARRAY [1]
  1 PIN_FLD_ACCOUNT_OBJ POID [0] $DB_NO /account 12345 0
  1 PIN_FLDBILLINFO_OBJ POID [0] $DB_NO /billinfo 34567 0
  1 PIN_FLD_BAL_GRP_OBJ POID [0] $DB_NO /balance_group 66765 0
0 PIN_FLD_RESULTS ARRAY [1]
  1 PIN_FLD_ACCOUNT_OBJ POID [0] $DB_NO /account 12344 0
  1 PIN_FLDBILLINFO_OBJ POID [0] $DB_NO /billinfo 45654 0
  1 PIN_FLD_BAL_GRP_OBJ POID [0] $DB_NO /balance_group NULL 0
```

You cannot use the -g parameter with any of the other account selection criteria parameters (-a, -d, -i, -m, -n, -p, -s, -line, and field_name).
Rerating Events Associated with a Custom Event Field

You can define custom `pin_rerate` parameters based on any event criteria. Use the `-field_name` parameter to rerate events associated with a field that you define.

For example, if tax was calculated incorrectly, you might define the custom parameter `-tax_supplier` to select events based on the PIN_FLD_TAX_SUPPLIER field. You specify a tax supplier ID in a file named `tax_supplier.txt`. The command to select accounts associated with that tax supplier is:

```
pin_rerate -tax_supplier tax_supplier.txt -t 01/23/2016
```

When you use a custom rerate event selection parameter, rerating searches the base `/event` storable class for field. If the event field associated with a custom parameter is present only in a subclass, you must use the `-n` parameter to specify the subclass event.

For example, the PIN_FLD_ORIGIN_NETWORK field is not in the base `/event` class, so you must use the `-n` parameter to select events based on that field. In that case, you create a text file specifying the `/event/activity/telco` event in a file named `event_rerate.txt`.

```
Important: When using the `-n` parameter with a custom field, the input file can contain only one type of event.
```

To select accounts whose telco events were generated by the specified origin network, run the following command:

```
pin_rerate -n event_rerate.txt -origin origin.txt -t 06/01/2007
```

If the event specified in the `-n` parameter input file has subclasses, all subclass events are also selected. For example, if you specify `/event/activity/telco`, BRM also selects events from the `/event/activity/telco/gsm` event.

To define custom event selection parameters, you create an XML file that maps the event field name to the parameter name that you use when running the `pin_rerate` utility. You then run the `load_pin_rerate_flds` utility to load the mappings into the `/config/rerate_flds` object in the BRM database. When you run the `pin_rerate` utility, it uses the `/config/rerate_flds` object to identify the event field to search for.

Create an XML file containing the parameter-to-event field mappings according to the BRM rerating XML schema file (`BRM_home/xsd/pin_rerate_flds.xsd`). You can map parameter names to fields in the base `/event` storable class or to fields in an `/event` subclass.

```
Important: Ensure that you validate your XML file against the XML schema. The `load_pin_rerate_flds` utility cannot load an invalid XML file.
```

In the file, the `<Name>` tag is event field name, and the `<value>` tag is the custom event selection parameter name. This example shows how to define an event selection parameter named `-tax_supplier`:

```
<PinRerate>
  <Name>EVENT.PIN_FLD_TAX_SUPPLIER</Name>
  <value>tax_supplier</value>
</PinRerate>
```
If you map a custom parameter to a field in an array or subclass, specify the array or substruct that contains the parameter field. For example, to specify PIN_FLD_ORIGIN_NETWORK, which is located in the PIN_FLD_TELCO_INFO substruct:

```xml
<PinRerate>
  <Name>EVENT.PIN_FLD_TELCO_INFO.PIN_FLD_ORIGIN_NETWORK</Name>
  <value>origin_network</value>
</PinRerate>
```

Use the following command to load the custom `pin_rerate` parameters defined in the XML file into the `/config/rerate_fld` object:

```
load_pin_rerate_flds xml_file_name
```

---

**Important:** The `load_pin_rerate_flds` utility uses a configuration file (`pin.conf`) located in the same directory to connect to the BRM database. Edit the configuration file to connect to your BRM database.

---

For more information, see "load_pin_rerate_flds".

### Getting an Estimate for the Number of Rerated Events

Use the `-e` parameter to display an estimated number of events that will be rerated. Running the `pin_rerate` utility with this parameter does not load any data into the database.

You use this parameter with one of the selection parameters. For example, to get an estimate of the number of events that will be rerated for a list of services, use this command:

```
pin_rerate -e -s service_rerate.txt -t 01/23/2016
```

You can use the `-e` parameter when selecting events associated with the following:

- A single account (`-a`)
- Services (`-s`)
- Events (`-n`)
- Charge offers (`-p`)
- Discount offers (`-i`)
- Bundles (`-d`)

You cannot use the `-e` parameter when selecting events associated with the following:

- A list of accounts (`-m`)
- Balance groups (`-g`)

The `pin_rerate` utility displays the output of this option on the screen or writes it to the `pin_rerate.pinlog` file.

### Specifying Events for Rerating

BRM selects events to rerate by selecting accounts, then selecting events based on event creation date. For example, when you use the `-n` parameter to rerate based on the type of event, BRM finds accounts associated with that event, but it rerates all types of events that the accounts generated.
For example, the following command uses the -n parameter to find all accounts that have events related to the list of events defined in the event_rerate.txt file:

```
pin_rerate -s service_rerate.txt -t 01/01/2016
```

In this example, the event_rerate.txt file includes one entry, for purchase events:

```
/event/billing/product/fee/purchase
```

BRM finds all accounts associated with purchase events. From those events, BRM rerates all events generated by those accounts after January 1, 2016. The rerated events are not all purchase events; they include any type of event associated with the account.

In this example, if you want to rerate only purchase events, you use the -r parameter. For example, when you run this command:

```
pin_rerate -s service_rerate.txt -t 01/01/2016 -r
```

BRM does the following:

1. Selects all the accounts associated with purchase events. The accounts found also generated usage events, cycle events, and so on, so many different types of events are selected for rerating.
2. Because the -r parameter is used, from the events generated by those accounts, BRM selects only purchase events.
3. From those events, BRM selects only the purchase events that occurred after January 1, 2016.

If you use the -r parameter to select events, be sure to consider how it might affect rating overall because balances can be impacted by different types of events: a cycle event can grant minutes and a usage event consumes minutes from the same balance. If, for example, you change the pricing for a charge offer that grants minutes, you must rerate all events for the accounts that own the charge offer. If you rely on credit limits or quantity-based charging, rerating only some events might result in incorrect charges.

When rerating cycle fee events, to get the correct rerating results, include the cycle events that occur during billing that are configured in the product, such as cycle discount, rollover, and fold events in the event file.

For example, if a cycle discount is configured to be some percentage of the charge during billing and if the cycle forward arrear fee is modified during the billing cycle, then to rerate the cycle forward arrear event, you need to include both events in the event file:

- `/event/billing/product/fee/cycle/cycle_forward_arrear`
- `/event/billing/cycle/discount`

The -r parameter can be used with any of the account selection parameters.

You can further refine the event selection criteria used for selective rerating by customizing the PCM_OP_SUBSCRIPTION_POL_SPEC_RERATE policy opcode. See BRM Opcode Guide.

**Specifying the Event Sequence for Rerating**

Events are rerated in sequence based on the event time. You can specify one of two times:

- *Event end time* uses the time that the usage occurred. This is the default.
Using Reason Codes to Specify the Rerating Order

- **Event creation time** uses the time that the event was loaded into the BRM database. The event time you specify might depend on how the original events were rated:
  - Events that are rated or loaded into the BRM database as a result of offline charging can have a significant delay between the event end time and creation time. Using the event end time reflects the real-time sequence of the original events. However, because batch events are recorded in order of creation time, this makes it harder to predict the impact of a price configuration change. To compare the original and rerated balance impacts of batch events, use the event creation time.
  - Events rated by online charging, and cycle events and purchase events, have no or very little delay between the event end time and creation time. Both the event end time and creation time reflect the real-time sequence in which the original events occurred and were recorded.

To specify the event time for rerating, use the `-b` parameter when selecting accounts for rerating. The `-b` parameter uses either the `c` option (for event creation time) or the `e` option (for event end time).

For example:

```bash
pin_rerate -b c -t 01/23/2016
```

The preceding example selects events for all accounts that occurred after 01/23/2016, and rerates the events in order of creation time.

Using Reason Codes to Specify the Rerating Order

Some rerating jobs must be processed before others. For example, if a rerate job includes events that have an impact on future rating, such as quantity-based discounting, those events must be rerated first. You can achieve this by assigning rerate reason codes to rerate jobs when you create those jobs. You can then run the pin_rerate utility multiple times, assigning a reason code each time, in the correct order.

For example, to rerate a specific type of event first, and then all other events, you would run rerating twice; first by using a reason code, and then with no reason code:

```bash
pin_rerate -process jobs -reason 99
pin_rerate -process queue
pin_rerate -rerate
pin_rerate -process queue
pin_rerate -process recycle

pin_rerate -process jobs
pin_rerate -process queue
pin_rerate -rerate
pin_rerate -process queue
pin_rerate -process recycle
```

You can use multiple reason codes by separating them with commas in the pin_rerate command, for example:

```bash
pin_rerate -process jobs -reason 97,98,99
```

However, when you list multiple rerate reasons, the rerate jobs associated with them are processed randomly. They are not processed in the order you list them in the command line, nor are they processed in increasing or decreasing numeric order. To
process rerate jobs according to a rerate reason in a particular order, you must run rerating separately for each reason code.

You can use reason codes when you use automatic rerate job creation, and when you select events to rerate manually.

When you use automatic rerate job creation. In this case, you create custom reason codes and use them when you run rerating. By default, BRM does not create reason codes when automatically creating rerate jobs. If you use automatic rerate job creation, you must customize the PCM_OP_SUBSCRIPTION_POL_GENERATE_RERATE_REQUEST opcode, or create your custom opcode, to assign a custom reason code. See BRM Opcode Guide for information.

If you manually run rerating, you can assign rerate jobs when you specify the events to rerate. You combine reason codes with event election parameters. For example, to create a reason code to rerate jobs that rerate events based on charge offers, you enter the following commands:

```
pin_rerate -reason 99 -p charge_offer_rerate.txt -t 01/23/2016
pin_rerate -process jobs -reason 99
pin_rerate -process queue
pin_rerate -rerate
pin_rerate -process queue
pin_rerate -process recycle
```

You can use manually-created reason codes to process events in a specified order, for example:

```
pin_rerate -reason 97 -p charge_offer_rerate.txt -t 01/23/2016
pin_rerate -reason 98 -i discount_offer_rerate.txt -t 01/23/2016
pin_rerate -reason 99 -d bundle_rerate.txt -t 01/23/2016

pin_rerate -process jobs -reason 97
pin_rerate -process queue
pin_rerate -rerate
pin_rerate -process queue
pin_rerate -process recycle

pin_rerate -process jobs -reason 98
pin_rerate -process queue
pin_rerate -rerate
pin_rerate -process queue
pin_rerate -process recycle

pin_rerate -process jobs -reason 99
pin_rerate -process queue
pin_rerate -rerate
pin_rerate -process queue
pin_rerate -process recycle
```

Rerate reason codes can be any integer except 1. 1 is reserved for internal use.

---

**Important:** Ensure that your reason codes are unique because you process rerate jobs associated with them during separate rerating executions.

---
Rerating Cycle Forward and Cycle Forward Arrears Events

Important: Rerating recurring events can affect performance.

Note: Before you can rerate cycle forward and cycle forward arrears charges, you must enable this feature. See "Rerating Cycle Forward and Cycle Forward Arrears Events".

You can configure BRM to create rerate jobs automatically to rerate cycle forward and cycle forward arrears events after they are charged.

For example, suppose a $10 cycle fee is charged to the account for the cycle from April 15 to May 15. You change the fee from $10 to $20 on April 30, the middle of the cycle. When you rerate events, BRM recalculates the cycle fees as follows:

- Refunds the $10 for the old cycle fee.
- Recalculates the cycle fees based on the $10 fee for the first 14 days in the cycle and the $20 fee for the next 16 days in the cycle:

\[
(\$10 \times \frac{14}{30}) + (\$20 \times \frac{16}{30}) = \$15.33
\]

After you change the pricing for a cycle forward or cycle forward arrears event, do the following:

1. Run the `pin_rate_change` utility to analyze charge offers and create rerate jobs.

   `pin_rate_change`

2. Run the `pin_rerate` utility to recalculate the cycle fees.

   See "pin_rate_change".

Running Back-Out-Only Rerating

Back-out-only rerating backs out the balance impacts of rating without rerating events. BRM backs out balance impacts for back-out-only rerating by creating an adjustment event that fully negates the original balance impacts.

Note: Use caution when choosing the events to back out because it can impact your general ledger. For example, it is incorrect to use back-out-only rerating for a cycle event when the customer has already paid the cycle fee or to use back-out-only rerating when pricing is changed. Typically, back-out-only rerating is performed only on usage events where rating should not have occurred.

To back out the balance impacts of rating, run `pin_rerate` with the `-backout` parameter.

Use the `-backout` parameter with other parameters that select the accounts and their events for rerating. This creates rerate jobs set for back-out-only rerating for the selected accounts.

For example, the following command selects accounts whose events were rated by the charge offers specified in the `charge_offer_backout.txt` file and backs out those events that occurred after 01/23/2016:

```
    pin_rerate -backout -p charge_offer_backout.txt -t 01/23/2016 -r
    pin_rerate -process jobs
```
pin_rerate -process queue
pin_rerate -rerate
pin_rerate -process queue
pin_rerate -process recycle
Running Back-Out-Only Rerating

Rerating Events
Tuning Rerating Performance

This document describes how to improve performance for Oracle Communications Billing and Revenue Management (BRM) rerating.

Topics in this document:
- Improving pin_rerate Performance
- Setting the Rerating Event Cache Size (Fetch Size)
- Configuring the Number of Accounts Per Job and Number of Jobs per Transaction

See also:
- About Rerating Events
- Configuring Rerating
- Rerating Events
- Advanced Rerating Tasks

Improving pin_rerate Performance

The following pin_rerate configuration parameters can be used with pin_rerate -rerate option to improve pin_rerate performance:

- pin_rerate rerate_children 5
- pin_rerate rerate_per_step 1000
- pin_rerate rerate_fetch_size 5000

**Note:** These entries can only be used with pin_rerate -rerate option.

Setting the Rerating Event Cache Size (Fetch Size)

By default, BRM rerating caches 10,000 events in system memory for processing. Depending on your system memory, you can set the event_fetch_size in the Connection Manager’s configuration file to specify the number of events retrieved from the database and cached in the system memory for processing.

To set the event cache size:

1. Open the Connection Manager (CM) configuration file (BRM_home/sys/cm/pin.conf) using a text editor.
2. Uncomment the - fm_subscription event_fetch_size entry.
3. Edit the event_fetch_size value. The default is 10000.
Configuring the Number of Accounts Per Job and Number of Jobs per Transaction

To prepare for rerating, BRM assigns accounts to rerate jobs. By default, BRM assigns 10 accounts to each rerate job and creates 2 rerate jobs per transaction. For performance tuning, you can change the default number of accounts per job and the number of rerate jobs created per transaction:

1. Open the `pin_rerate` configuration file (`BRM_home/apps/pin_rerate/pin.conf`).
2. Set the number of accounts assigned to each rerate job by adding the following line:
   
   - `pin_rerate per_job accounts_per_job`

   where `accounts_per_job` is the number of accounts to assign to each job.

3. Set the number of jobs created per transaction by adding the following line:
   
   - `pin_rerate per_transaction jobs_per_transaction`

   where `jobs_per_transaction` is the number of jobs to create in each transaction.

4. Save and close the file.

**Caution:** Setting the `pin_rerate per_job` entry to a small number, for example 1, results in many rerate jobs being created. Too many rerate jobs can affect your system's performance due to the rerate steps performed for each rerate job. Processing multiple accounts in one rerate job reduces the total number of rerate steps performed compared to processing those same accounts in multiple rerate jobs.
This chapter provides reference information for Oracle Communications Billing and Revenue Management (BRM) rerating utilities.

Topics in this document:

- load_pin_rerate_flds
- pin_rate_change
- pin_rerate

See also:

- About Rerating Events
- Rerating Events
load_pin_rerate_flds

Use this utility to load the following information into the BRM database:

- Custom event selection parameters. The data is loaded into the /config/rerate_flds object. See “Rerating Events Associated with a Custom Event Field” for information.

- Balance-impact comparison fields that determine if an adjustment event must be created. The data is loaded into the /config/rerate_flds/compare_bi object. See “Configuring Rerating Adjustment Events”.

Location

BRM_home/sys/data/config

Run load_pin_rerate_flds from this directory.

Syntax

load_pin_rerate_flds -f xml_file_name [-v] [-h]

Parameters

-f xml_file_name
Specifies the name of the XML file that contains the data.

---

**Important:** The file you load cannot include both event selection data and balance-impact comparison data. You must load the files for these configurations separately.

---

-verbose
Displays information about successful or failed processing as the utility runs.

---

**Note:** This parameter is always used with other parameters and commands. It is not position dependent. For example, you can enter -verbose at the beginning or end of a command to initiate the verbose parameter. To redirect the output to a log file, use the following syntax with the verbose parameter. Replace filename.log with the name of the log file:

```
load_pin_rerate_flds any_other_parameter -v > filename.log
```

-help
Displays the syntax and parameters for this utility.

Results

This utility notifies you when it successfully creates the /config/rerate_flds object.

If it cannot create the /config/rerate_flds, it logs an error in the log file (default.pinlog). It creates the log file either in the directory from which the utility was started or in the directory specified in the configuration file.
**pin_rate_change**

Use this utility to create rerate jobs for cycle forward and cycle forward arrears events when the pricing changes. Use this BRM utility after you change pricing in PDC and before running the **pin_rerate** utility.

See "Rerating Cycle Forward and Cycle Forward Arrears Charges".

The **pin.conf** file for this utility is in `BRM_home/apps/pin_rate_change`.

**Location**

`BRM_home/bin`

**Syntax**

```
pin_rate_change [-v] [-d] [-h]
```

**Parameters**

- **-v**
  
  Displays information about successful or failed processing as the utility runs.

  **Note:** This parameter is not position dependent. For example, you can enter `-v` at the beginning or end of a command to initiate the verbose parameter. To redirect the output to a log file, use the following syntax with the verbose parameter. Replace `filename.log` with the name of the log file:

  ```
  pin_rate_change any_other_parameter -v > filename.log
  ```

- **-d**
  
  Creates a log file for debugging purposes. Use this parameter for debugging when the utility appears to have run with no errors, but the data has not been loaded into the database.

- **-h**
  
  Displays the syntax and parameters for this utility.

**Results**

The **pin_rate_change** utility notifies you when it successfully creates the rerate jobs.

If the **pin_rate_change** utility does not notify you that it was successful, run the command again with the `-d` parameter, and 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.
Use this BRM utility to rerate events. See "About Rerating Events".

To rerate events, you run the `pin_rerate` utility multiple times: once to create rerating jobs and several additional times to process the rerate jobs.

### Note:
- `pin_rerate` is a multithreaded application.
- In a multischema environment, you must run `pin_rerate` separately on each database schema.

### Location

`BRM_home/bin`

The `pin.conf` file for this utility is located in `BRM_home/apps/pin_rerate`. Run `pin_rerate` from this directory.

### Syntax for Selecting Events

```
pin_rerate [-t [ss/mm/hh/] MM/DD/YYYY]
[-a account POID_id][-c]
[[-d | -g | -i | -m | -n | -p | -s] input_file]
[-line subscription_id]
[[-field_name input_file]
[-r]
[-reason reason_code]
-process recycle [-db database_id]
```

### Syntax for Running the `pin_rerate` Utility

```
pin_rerate
-process jobs [-reason reason_code, [reason_code]...]
-process queue
-rerate [-reason reason_code [,reason_code...][-l [d | s | sd | n]]
-process recycle [-db database_id]
[[-backout]
[[-purge [-t [ss/mm/hh/] MM/DD/YYYY]]
[[-b [c | e]]
[[-e [-a | -s | -p | -n | -d]]
[[-l [d | s | sd | n]]
[[-h | help]]
```

### Parameters for Selecting Accounts for Rerating

The following parameters are used to select accounts for rerating based on the described event criteria.

All account selection parameters are optional and mutually exclusive, except for the `-t` parameter, which is mandatory when selecting the events to rerate. If you specify only the time `-t` parameter, BRM selects all accounts for rerating in the period defined by the `-t` parameter.

For more information about selecting account for rerating, see "Selecting Events for Rerating".
-a [account POID_id]
Selects a single account for rerating based on the account POID.
When you specify an account POID, you can use the -c option to rerate events without updating the database.
This parameter is optional.
You cannot use this parameter with the -d, -g, -i, -m, -p, -s, -line, or custom parameters.

-d input_file
Selects accounts for rerating based on the bundle. Accounts that have events associated with the charge offers and discount offers that belong to the bundles specified in input_file are selected for rerating.

input_file format: A text file with one bundle name specified on each line. The bundle names are case sensitive.

You cannot use this parameter with the -a, -g, -i, -m, -n, -p, -s, -line, or field_name parameters.
This parameter is optional.

---

Important: If a charge offer is part of multiple bundles, all accounts that purchase the charge offer are selected for rerating even if they did not purchase the bundle that was rerated.

---

-g input_file
Selects accounts for rerating based on balance groups.

input_file format: A text file containing information for one or more accounts in flist format. Specify an account POID, bill unit POID, and balance group POID in a results array for each account. For example:

```
0 PIN_FLD_RESULTS ARRAY [1]
  1 PIN_FLD_ACCOUNT_OBJ POID [0] $DB_NO /account 12345 0
  1 PIN_FLD_BILLINFO_OBJ POID [0] $DB_NO /billinfo 34567 0
  1 PIN_FLD_BAL_GRP_OBJ POID [0] $DB_NO /balance_group 66765 0
0 PIN_FLD_RESULTS ARRAY [1]
  1 PIN_FLD_ACCOUNT_OBJ POID [0] $DB_NO /account 12344 0
  1 PIN_FLD_BILLINFO_OBJ POID [0] $DB_NO /billinfo 45654 0
  1 PIN_FLD_BAL_GRP_OBJ POID [0] $DB_NO /balance_group NULL 0
...```

To rerate all events for all balance groups for a given account’s bill unit, specify a value of NULL as the POID ID for the balance group object, as shown in the second results array in the above example.

You cannot use this parameter with the -a, -d, -i, -m, -n, -p, -s, -e, -line, or field_name parameters.
This parameter is optional.

-i input_file
Selects accounts for rerating based on discount offers. Accounts that own at least one instance of the discount offers specified in input_file are selected for rerating.

input_file format: A text file with one discount offer name specified on each line. Discount offer names are case sensitive.
You cannot use this parameter with the -a, -d, -g, -m, -n, -p, -s, -line, or field_name parameters.

This parameter is optional.

-m input_file
Selects a set of accounts for rerating based on the provided account POIDs.

_input_file_ format: A text file containing the accounts to rerate in flist format. Specify each account in a results array. For example:

```
0 PIN_FLD_RESULTS ARRAY [1]
   1 PIN_FLD_ACCOUNT_OBJ POID [0] $DB_NO /account 12345 0
0 PIN_FLD_RESULTS ARRAY [2]
   1 PIN_FLD_ACCOUNT_OBJ POID [0] $DB_NO /account 12333 0
...
```

where PIN_FLD_ACCOUNT_OBJ is the POID of the account object.

You cannot use this parameter with the -a, -d, -g, -i, -n, -p, or -s, -e, -line, or field_name parameters.

-n input_file
Selects accounts for rerating based on the types of events. Accounts that have events of the types specified in input_file are selected for rerating.

_input_file_ format: A text file with one type of event specified on each line. Events are case sensitive.

You cannot use this parameter with the -a, -d, -g, -i, -m, -p, -s, -line, or field_name parameters.

This parameter is optional.

---

**Note:**

If you use this parameter with the -r parameter, all subclasses of the event specified in the input file are also rerated. See "Specifying Events for Rerating".

When a custom _pin_rerate_ parameter is used, the -n parameter is mandatory if the custom parameter is based on an event field present only in an event subclass. In this case, the -n parameter input file can contain only one type of event.

---

-p input_file
Selects accounts for rerating based on the charge offer. Accounts that have events associated with the charge offers specified in input_file are selected for rerating.

_input_file_ format: A text file with one charge offer name specified on each line. Charge offer names are case sensitive.

You cannot use this parameter with the -a, -d, -g, -i, -m, -n, -s, -line, or field_name parameters.

This parameter is optional.

-s input_file
Selects accounts for rerating based on the service. Accounts that have events associated with the services specified in input_file are selected for rerating.
**input_file format**: A text file with one service specified on each line. Service names are case sensitive.

You cannot use this parameter with the `-a, -d, -g, -i, -m, -n, -p, -line, or field_name` parameters.

This parameter is optional.

**-line subscription_id**
Selects accounts for rerating based on a subscription service. The subscription service is identified by the PIN_FLD_ALIAS_LIST.PIN_FLD_NAME field, which can specify, for example, the caller ID such as a phone number or the MAC address.

The group of selected accounts can be further restricted by using criteria specified with the `-a, -d, -g, -i, -m, -n, -p, -s, or field_name` parameters.

This parameter is optional.

**-field_name input_file**
field_name is a custom pin_rerate parameter you have defined that maps to a specific event field.

input_file is a file that contains the values of the event field used to select accounts for rerating.

Selects accounts based on the custom parameter. Accounts are selected for rerating that have events with the field identified by field_name whose field value is one of those specified in input_file.

You cannot use this parameter with the `-a, -d, -g, -i, -m, -p, -s, or -line` parameters.

For information about defining custom pin_rerate parameters, see "Rerating Events Associated with a Custom Event Field".

---

**Important:** If the custom parameter maps to a field in an /event subclass, you must specify the subclass event by including the `-n` parameter, and the `-n` parameter input file can include only one event. If you specify more than one type of event, an error is returned.

---

**-t [ss/mm/hh/MM/DD/YYYY]**
Selects accounts for rerating based on the event start time. All accounts with events that occurred between the start time and the current date are selected for rerating.

This parameter is mandatory when you use the `-a, -d, -g, -i, -m, -n, -p, -s, -line, or field_name` parameters.

**-reason reason_code**
Use with other selection parameters to assign a rerate reason code to the jobs created for the selected accounts.

For more information about assigning rerate reason codes, see "Using Reason Codes to Specify the Rerating Order".

---

**Parameters Used for Processing Rerate Jobs**

To process rerate jobs, you run the following commands in the order shown.

```
pin_rerate -process jobs [-reason comma_separated_reason_codes]
pin_rerate -process queue
```

```
pin_rerate -rerate [-l [d | s | sd | n]]
pin_rerate -process queue
```

---

Rerating Utilities 6-7
**pin_rerate -process recycle**

---

**Important:** You must run **pin_rerate** with all of the above parameters to complete the rerating process.

---

The parameters specify the following:

**-process jobs [-reason comma_separated_reason_codes]**
Searches for all new rerate jobs.

If you use the optional **-reason** parameter, performs the same actions but only for the rerate jobs associated with the specified reason codes.

**-process queue**
Processes acknowledgment events from ECE and updates the rerate job status.

This parameter is used twice: before and after the **-rerate** parameter.

**-rerate [-l [d|s|sd|n]]**
Rerates events.

-`l` Specifies to generate a rerating report. Options are:
  -`-ld` Generates a detailed report (**pin_rerate.detail**).
  -`-ls` Generates a summary report (**pin_rerate.summary**).
  -`-lsd` Generates both detailed and summary reports. This is the default.
  -`-ln` Generates no report.

**-process recycle [-db database_id]**
Recycles the event records suspended during the rerating process.

-`db database_id`: Specifies the ID of the database schema that contains the suspended event records in the format `x.x.x.x` (for example, `0.0.0.2`). Use this only if the suspended event records are not stored in the current BRM schema.

---

**Note:** If you do not specify the **-db database_id** parameter, the suspended event records are picked up from the current schema.

---

**-purge [-t [ss/mm/hh/] MM/DD/YYYY] ]**
Purges rerate jobs that have a rerate job status of COMPLETE or UNSUCCESSFUL.

[-`t [ss/mm/hh/] MM/DD/YYYY] ]

Specifies to purge rerate jobs that have had their rerate job status set to COMPLETE or UNSUCCESSFUL before the time specified.

If no time is specified, all rerate jobs that have a rerate job status of COMPLETE or UNSUCCESSFUL are purged up to the current date.

---

**Parameters Affecting Rerating Behavior**

The following are additional, optional parameters that affect rerating behavior and the utility’s output.
-b  [c|e]
Specifies the order in which batch events are rated. The c option rerates events in the order that they were created in the BRM database. The e option rerates events based on the time when the event occurred. The default is e. See “Specifying the Event Sequence for Rerating”.
This parameter is optional.

-c
Specifies calculation of rerating only. This option can be used when using the -a parameter.
This parameter is optional.

-backout
Specifies back-out-only rerating, which backs out the balance impacts of rating without rerating events.
For more information about back-out-only rerating, see “Rerating Events”.
This parameter is optional and can be used with any other parameter.

Note: When choosing the events to back out, ensure that you do not select events that could imbalance your general ledger, such as events for which fees have already been paid and usage events that should be rerated. Typically, back-out-only rerating is performed only on usage events where rating should not have occurred.

-e  [-a | -s | -p | -n | -d]
Returns an estimate of how many events might be affected by rerating based on which accounts are being rerated. Options:
-e -a: A single account
-e -s: Accounts with a specific service
-e -p: Accounts with specific charge offers
-e -n: Accounts with specific types of events
-e -d: Accounts with specific bundles

Note: If you do not specify one of these options when using the -e parameter, the utility returns 0.

The pin_rerate utility displays the estimated events to rerate.
This parameter is optional.

Note: You cannot use this parameter with the -m or -g parameters.

-h  |  help
Displays the syntax and parameters for this utility.
Results

If the **pin_rerate** utility does not notify you that it was successful, look in the utility log file (**pin_rerate.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.

If rerating fails, the utility creates a report that includes the account numbers and start times for failed rerates. The report file name is **pin_rerate.status_report**, and is in the directory from where you ran the utility.