# Contents

Preface ................................................................................................................................................................ vii
  Audience...................................................................................................................................................... vii
  Related Documentation ......................................................................................................................... vii
  Documentation Accessibility ................................................................................................................ vii

1 **About Loading Rated Events into the BRM Database**
   How Rated Events are Loaded Into the BRM Database ................................................................. 1-1
      About Running RE Loader ........................................................................................................... 1-2
   About Backing Up RE Loader Files .............................................................................................. 1-2
   About Using RE Loader in a Multischema System ........................................................................ 1-3
   About Configuring Rated Event Loading ...................................................................................... 1-3

2 **Installing Rated Event Loader**
   Installing RE Loader ........................................................................................................................... 2-1
      Granting Execute Permission for dbms_lock.............................................................................. 2-1
      Granting Write Permission to the DM .......................................................................................... 2-1
      Installing the RE Loader Package ............................................................................................... 2-2
      Creating Your RE Loader Database Partitions ......................................................................... 2-2
      Returning DM Permissions to their Original Values .................................................................. 2-3
      Preventing POID Errors in Multischema Systems .................................................................. 2-3
      Uninstalling RE Loader .............................................................................................................. 2-4

3 **Configuring Rated Event Output from ECE**
   Adding a Rated Event Publisher Instance ..................................................................................... 3-1
   Configuring Rated Event Publisher ............................................................................................... 3-2
   Adding a Rated Event Formatter Instance .................................................................................... 3-2
   Configuring Rated Event Formatter ............................................................................................... 3-3
   Adding a BRM CDR Plug-in Instance .............................................................................................. 3-4
   Configuring BRM CDR Plug-in ........................................................................................................ 3-5

4 **Configuring BRM Server Components for Rated Event Loading**
   Enabling a Billing Delay for CDRs ................................................................................................. 4-1
   Disabling Invoice Event Caching ................................................................................................. 4-1
   Setting Up RE Loader for Virtual Column-Enabled Systems ....................................................... 4-2
Configuring Whether to Perform Redo Generation .......................................................... 4-2
Configuring Field Lengths for Input Data Files ................................................................. 4-2
Updating the RE Loader Control Files with ECE Control Data ....................................... 4-3

5 Configuring the RE Loader Infranet.properties File
Configuring the RE Loader Infranet.properties File ......................................................... 5-1
BRM Database Connection ............................................................................................... 5-2
BRM Server Connection ................................................................................................. 5-3
RE Loader Daemon ......................................................................................................... 5-4
RE Loader Processing ...................................................................................................... 5-5
Loading Event Records ................................................................................................. 5-7
Storable Class Configuration ......................................................................................... 5-8
RE Loader Log Files ........................................................................................................ 5-9

6 Configuring RE Loader to Run Automatically by Using Batch Controller
Configuring RE Loader to Run Automatically by Using Batch Controller ......................... 6-1
Configuring the RE Loader Batch Handler ...................................................................... 6-1
Configuring Batch Controller ........................................................................................ 6-2

7 Configuring RE Loader to Run Automatically by Using the RE Loader Daemon
Configuring RE Loader to Run Automatically by Using the RE Loader Daemon ............... 7-1

8 Customizing RE Loader
Customizing RE Loader .................................................................................................. 8-1
Adding New Types of Events for RE Loader to Load ....................................................... 8-2
Creating Custom Error Codes ......................................................................................... 8-3

9 Troubleshooting Rated Event Loading
About Troubleshooting Rated Event Loading ................................................................ 9-1
Checking the RE Loader Log Files for Error Codes ....................................................... 9-2
Checking for Errors that Occurred during the PreUpdate Process .................................. 9-7
Fixing Event Loading Errors .......................................................................................... 9-7
Debugging Mismatches between Data Files and Control Files ....................................... 9-9
Retrieving Data About Events You Load ........................................................................ 9-10

10 Improving RE Loader Performance
Improving RE Loader Performance ................................................................................ 10-1
Increasing the Number of Account Balance and Bill Item Updates .............................. 10-1
Turning Off Index Verification to Improve Database Loading Performance .................. 10-2
Turning Off Database Verification to Improve Processing Performance ...................... 10-3
Pruning Your RE Loader Control and Audit Tables ...................................................... 10-3

Retrieving Data About Events You Load ........................................................................ 9-10
11 Rated Event Loader Utilities
Preface

This guide describes how to load rated events from Elastic Charging Engine (ECE) into the Oracle Communications Billing and Revenue Management (BRM) database by using Rated Event Loader (RE Loader).

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 how to load rated events into the Oracle Communications Billing and Revenue Management (BRM) database from Elastic Charging Engine (ECE) by using Rated Event Loader (RE Loader).

Topics in this document:

- How Rated Events are Loaded Into the BRM Database
- About Backing Up RE Loader Files
- About Using RE Loader in a Multischema System
- About Configuring Rated Event Loading

See also:

- Installing Rated Event Loader
- Configuring the RE Loader Infranet.properties File

How Rated Events are Loaded Into the BRM Database

To load rated events from ECE into the BRM database:

1. In the ECE system, Rated Event Publisher publishes rated events to an Oracle Knuckles database.

2. Rated Event Formatter (RE Formatter) sends the rated events to the BRM server. RE Formatter uses the BRM CDR plugin to write the rated events to a CDR file and send the file to a specified directory.

3. Rated Event Loader (RE Loader) retrieves the CDR file from the directory and loads the rated events into the BRM database. RE Formatter and RE Loader can run on the same system or different systems, as long as they can both access the CDR file directory.

   RE Loader loads rated events directly into the BRM database, bypassing the Connection Manager (CM) and Data Manager (DM). RE Loader then updates account balances, billing items, and journals in the BRM database.

   RE Loader uses a partitioned database and inserts pre-rated events into separate partitions allocated for delayed events. The events are called "delayed" because they are rated before they are loaded, and there is a delay between the two actions.
About Running RE Loader

You can run RE Loader in the following ways:

- By using Batch Controller:
  1. Batch Controller detects when a file is present in the RE Formatter output directory and starts the RE Loader batch handler. See .
  2. RE Loader batch handler starts the RE Loader utility (pin_rel) to load the events. RE Loader batch handler moves the original file to an archive directory if the records are successfully loaded or to a reject directory if the records are not successfully loaded.

- By using the RE Loader daemon. The RE Loader daemon detects when CDR files arrive in the input directory, and then processes them. The RE Loader daemon moves the original file to an archive directory if the records are successfully loaded or to a reject directory if the records are not successfully loaded.

If your CDR files are small, Oracle recommends loading events by using the RE Loader daemon. You should compare the two methods by running them both in a test system.

You can also run RE Loader manually, by running the pin_rel utility. See "pin_rel"

If any errors occur during event loading, all events loaded in that session are deleted from the database. After events are successfully loaded, if any errors occur during the update procedure, you can correct the errors and then update the relevant events by rerunning the RE Loader utility. The utility detects that the events loaded correctly and performs only the update procedure. See "Troubleshooting Rated Event Loading".

About Backing Up RE Loader Files

By default, RE Loader skips redo generation when loading files into the BRM database. This optimizes loading performance, but it can cause you to lose data if your system shuts down ungracefully.

To prevent data loss when your system shuts down:

- Make full backups of the BRM database on a regular basis.
- Archive all successfully loaded files until you make a full database backup.

You can re-enable redo generation, at the cost of loading performance, by modifying the RE Loader control files. See "Configuring Whether to Perform Redo Generation."
About Configuring Rated Event Loading

To retrieve and load rated events:

- Configure ECE components Rated Event Publisher and Rated Event Formatter. See "Configuring Rated Event Output from ECE".

- Configure BRM system components to enable proper loading of rated events. For example, to use RE Loader, you must disable invoice event caching. See "Configuring BRM Server Components for Rated Event Loading."

- Configure the RE Formatter BRM CDR plug-in to transfer events from ECE to RE Loader. See "Configuring BRM CDR Plug-in."

- Configure the RE Loader Infranet.properties file. See "Configuring the RE Loader Infranet.properties File."

- Configure RE Loader to load events automatically:
  - Configuring RE Loader to Run Automatically by Using Batch Controller
  - Configuring RE Loader to Run Automatically by Using the RE Loader Daemon

About Using RE Loader in a Multischema System

If you use a multischema system, you must set up the following for each BRM database schema in your system:

- RE Loader instance. Each instance of RE Loader must also have its own set of RE Loader processing directories.

- (Running RE Loader automatically only) An instance of Batch Controller and the RE Loader batch handler.
This chapter explains how to install the Oracle Communications Billing and Revenue Management (BRM) Rated Event (RE) Loader software.

Topics in this document:
- Installing RE Loader
- Preventing POID Errors in Multischema Systems
- Uninstalling RE Loader

See also:
- About Loading Rated Events into the BRM Database
- Configuring the RE Loader Infranet.properties File

Installing RE Loader

To install RE Loader, perform the procedures in these sections:
1. Granting Execute Permission for dbms_lock
2. Granting Write Permission to the DM
3. Creating Your RE Loader Database Partitions
4. Returning DM Permissions to their Original Values

Granting Execute Permission for dbms_lock

Before you install RE Loader, you must grant execute permission to pin_user for dbms_lock:

1. Log in to your database as the SYS user:
   ```sql
   sqlplus sys/password@databaseAlias
   ```
2. Grant execute privileges to pin_user:
   ```sql
   grant execute on dbms_lock to pin_user
   ```

Granting Write Permission to the DM

When you install RE Loader on a system where BRM is not installed, you must grant the DM write permission before installing RE Loader.

Perform the following on all machines containing a DM:

1. In a text editor, open your DM configuration file:
Installing RE Loader

BRM_home/sys/dm_oracle/pin.conf

BRM_home is the directory in which you installed BRM components.

2. Write down the values of your dd_write_enable_fields, dd_write_enable_objects, dd_write_enable_portal_objects, and dd_mark_as_portal entries.

3. Set the values of the following entries to 1:
   - dm dd_write_enable_fields 1
   - dm dd_write_enable_objects 1
   - dm dd_write_enable_portal_objects 1
   - dm dd_mark_as_portal 1

Note: If any entry is not in the file, add it.

For more information, see the information in the DM pin.conf file.

4. Save and close the file.

5. Stop and restart the DM.

You can now install RE Loader.

Installing the RE Loader Package

To install RE Loader, see “Installing Individual BRM Components” in BRM Installation Guide.

Creating Your RE Loader Database Partitions

Caution: You must perform this step to ensure that the new event tables have the same partitioning layout as your existing event tables. If you install several optional components, perform this step only after installing the last component.

To create partitions for RE Loader events:

1. On the system where BRM is installed, go to the BRM_home/apps/partition_utils directory.

2. Run the partition_utils utility to enable delayed-event partitions:

   perl partition_utils.pl -o enable -t delayed -c storable_class

   where storable_class specifies the event classes for which you want partitioning.

Important: You must create partitions for all subclasses of a specific event that you want to load.

For example, this command creates partitions for /event/delayed/session/telco/gsm delayed events:

   perl partition_utils.pl -o enable -t delayed -c /event/session/telco/gsm

For more information, see:
■ “Enabling Delayed-Event Partitions” in BRM System Administrator’s Guide
■ “partition_utils” in BRM System Administrator’s Guide.

Your RE Loader installation is now complete.

Returning DM Permissions to their Original Values
To return your DM permissions to their original values:

1. In a text editor, open your DM configuration file:

```
BRM_home/sys/dm_oracle/pin.conf
```

2. Restore the following entries to their original values (the values they had before you modified them). The default value for each entry is 0:

- `dm dd_write_enable_fields`
- `dm dd_write_enable_objects`
- `dm dd_write_enable_portal_objects`
- `dm dd_mark_as_portal`

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

Preventing POID Errors in Multischema Systems
BRM multischema systems ensure that all POIDs are unique across all database schemas by using a POID-generation algorithm. This BRM algorithm sets each schema’s starting sequence number to a unique value and then increments each sequence number by a set value. By default, BRM sets the increment value equal to the number of schemas in your system.

For example, if your system contains three schemas:

- Schema 1 uses a starting sequence number of 10000
- Schema 2 uses a starting sequence number of 10001
- Schema 3 uses a starting sequence number of 10002

The incremental value is 3.

This example results in the following POID numbers shown in Table 2–1:

<table>
<thead>
<tr>
<th>Time</th>
<th>POID for Schema 1</th>
<th>POID for Schema 2</th>
<th>POID for Schema 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10000</td>
<td>10001</td>
<td>10002</td>
</tr>
<tr>
<td>2</td>
<td>10003</td>
<td>10004</td>
<td>10005</td>
</tr>
<tr>
<td>3</td>
<td>10006</td>
<td>10007</td>
<td>10008</td>
</tr>
</tbody>
</table>

When RE Loader loads a batch of objects into the BRM database, it reserves a group of POIDs as follows:

1. Changes the increment value by using the following equation:

\[(\text{Number of objects to load}) \times (\text{Current increment value})\]

For example, if RE Loader must load 2,000 objects into the database and the current increment value is 3, it changes the increment value to \(2,000 \times 3 = 6,000\).
2. Allocates POIDs to objects.

3. Returns the increment value to its original value.

However, if a major error occurs during the allocation process, the increment value can remain at the incorrect high value. To catch these situations, you can configure RE Loader to check the database increment value against a specified maximum before it reserves a group of POIDs. When the increment value exceeds the specified maximum, RE Loader exits and logs an error message, notifying your database administrator to manually reset the increment value.

To configure RE Loader to compare the increment value against a specified maximum:

1. Open the BRM_home/apps/pin_rel/Infranet.properties file in a text editor.

2. Set the infranet.rel.max_increment_by entry to the number of database schemas in your system:

   infranet.rel.max_increment_by = 20

   The default is 20.

3. Save and close the file.

Uninstalling RE Loader

To uninstall RE Loader, see “Uninstalling Optional Components” in BRM Installation Guide.
This document describes how to configure Oracle Communications Billing and Revenue Management (BRM) Elastic Charging Engine (ECE) to use Rated Event (RE) Loader to load events.

Topics in this document:

- Adding a Rated Event Publisher Instance
- Configuring Rated Event Publisher
- Adding a Rated Event Formatter Instance
- Configuring Rated Event Formatter
- Adding a BRM CDR Plug-in Instance
- Configuring BRM CDR Plug-in

See also:

- About Loading Rated Events into the BRM Database
- Configuring BRM Server Components for Rated Event Loading
- Configuring the RE Loader Infranet.properties File
- Improving RE Loader Performance
- Troubleshooting Rated Event Loading

### Adding a Rated Event Publisher Instance

To add a Rated Event Publisher instance:

1. Access the ECE configuration MBeans:
   a. Log on to the driver machine.
   b. Start the ECE charging servers (if they are not started).
   c. Connect to the ECE charging server node enabled for JMX management.
      This is the charging server node set to `start CohMgt = true` in the `ECE_home/occeserver/config/ecetopology.conf` file.
   d. Start a JMX editor that enables you to edit MBean attributes, such as JConsole.
   e. In the editor’s MBean hierarchy, find the ECE configuration MBeans.
2. Expand the ECE Configuration node.
4. Expand Operations.

5. Select `addRatedEventPublisherConfiguration`.

6. Enter a value for the instance name parameter.

7. Click `addRatedEventPublisherConfiguration`.

8. Use Elastic Charging Controller (ECC) to start the Rated Event Publisher instance.

**Configuring Rated Event Publisher**

Before you configure Rated Event Publisher, you need to add an instance of it. See "Adding a Rated Event Publisher Instance".

To configure a Rated Publisher Instance:

1. Access the ECE configuration MBeans:
   a. Log on to the driver machine.
   b. Start the ECE charging servers (if they are not started).
   c. Connect to the ECE charging server node enabled for JMX management.
      This is the charging server node set to `start CohMgt = true` in the `ECE_home/occeserver/config/eeetopology.conf` file.
   d. Start a JMX editor that enables you to edit MBean attributes, such as JConsole.
   e. In the editor's MBean hierarchy, find the ECE configuration MBeans.

2. Expand the ECE Configuration node.

3. Expand the `charging.ratedEventPublishers.instance_name` node.

4. Expand Attributes.

5. Specify values for the following attributes:
   - `threadPoolSize`: The number of threads to publish rated events to an Oracle NoSQL database system.
     The valid number is greater than zero. For best performance, it is recommended to set this parameter to the number of Oracle NoSQL database partitions. Threads that you configure higher than the number of partitions are not used.
   - `name`: The data store name to an Oracle NoSQL database system.
   - `noSQLConnectionName`: This parameter configures Rated Event Publisher to connect to the Oracle NoSQL Database; it configures the data store connection to the Oracle NoSQL database system.
     The Oracle NoSQL database connection string uses the format `hostname:port` for connecting to a pre-configured Oracle NoSQL database system.
     The default is "localhost:5000" for connecting to a standalone Oracle NoSQL database system (KV-Lite).

6. Use Elastic Charging Controller (ECC) to stop and restart Rated Event Publisher.

**Adding a Rated Event Formatter Instance**

To add a Rated Event Formatter instance:

1. Access the ECE configuration MBeans:
a. Log on to the driver machine.
b. Start the ECE charging servers (if they are not started).
c. Connect to the ECE charging server node enabled for JMX management.
   This is the charging server node set to start CohMgt = true in the $ECE_{home}/oceceserver/config/eceTopology.conf$ file.
d. Start a JMX editor that enables you to edit MBean attributes, such as JConsole.
e. In the editor's MBean hierarchy, find the ECE configuration MBeans.

2. Expand the ECE Configuration node.
3. Expand charging.ratedEventFormatters.
4. Expand Operations.
5. Select addRatedEventFormatterConfiguration.
6. Enter a value for the name parameter.
7. Click addRatedEventFormatterConfiguration.
8. Use Elastic Charging Controller (ECC) to start the Rated Event Formatter instance.

### Configuring Rated Event Formatter

To configure RE Formatter, do the following:

1. Access the ECE configuration MBeans:
   a. Log on to the driver machine.
   b. Start the ECE charging servers (if they are not started).
   c. Connect to the ECE charging server node enabled for JMX management.
      This is the charging server node set to start CohMgt = true in the $ECE_{home}/oceceserver/config/eceTopology.conf$ file.
   d. Start a JMX editor that enables you to edit MBean attributes, such as JConsole.
   e. In the editor's MBean hierarchy, find the ECE configuration MBeans.
2. Expand the ECE Configuration node.
3. Expand charging.ratedEventFormatters.instance_name, where Instance_Name is the name of the instance you want to configure.
4. Expand Attributes.
5. Specify values as described in Table 3–1.
6. On the driver machine, change directory to the $ECE_{home}/oceceserver/bin$ directory.
7. Use Elastic Charging Controller (ECC) to stop and restart RE Formatter.

Table 3–1 describes the parameters you must set to configure a RE Formatter instance.
Adding a BRM CDR Plug-in Instance

To add a BRM Plug-in instance:

1. Access the ECE configuration MBeans:
   a. Log on to the driver machine.
   b. Start the ECE charging servers (if they are not started).

### Table 3–1 RE Formatter Instance Configuration Attributes and Values

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Default Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>partition</td>
<td>&quot;1&quot;</td>
<td>The number of the BRM database schema to which the ECE rated events are exported. For example, enter 1 for BRM schema 0.0.0.1, to 2 for BRM schema 0.0.0.2, and so on.</td>
</tr>
<tr>
<td>threadPoolSize</td>
<td>4</td>
<td>The number of threads used by the RE Formatter instance to process a set of rated events for each time range defined by <strong>checkPointInterval</strong>. Valid values are greater than zero and up to any number the system resources allow. Tune this value to the expected workload in the deployed environment.</td>
</tr>
<tr>
<td>pluginPath</td>
<td>n/a</td>
<td>The JAR library path that contains the reader plug-in implementation.</td>
</tr>
<tr>
<td>pluginName</td>
<td>n/a</td>
<td>This attribute must be set to <strong>brmCdrPlugin</strong>.</td>
</tr>
<tr>
<td>name</td>
<td>&quot;formatter&quot;</td>
<td>The name of the RE Formatter instance, for example, <strong>ratedEventFormatter1</strong>, <strong>ratedEventFormatter2</strong>, and so on.</td>
</tr>
<tr>
<td>retainDuration</td>
<td>0</td>
<td>The time in seconds that rated events must be retained in the Oracle NoSQL database after being processed before they can be purged. The default is 0, which means that rated events are purged immediately after being processed.</td>
</tr>
<tr>
<td>ripeDuration</td>
<td>60</td>
<td>The time in seconds that rated events must exist in the Oracle NoSQL database before they can be processed. This value must be greater than the time it takes for ECE charging servers to fully recover after failure. Delaying the processing of rated events allows ECE to resolve duplicate rated events persisted in the Oracle NoSQL database.</td>
</tr>
</tbody>
</table>
| checkPointInterval | 4             | The interval of time in seconds that RE Formatter reads rated event data. For example, if you enter 4, RE Formatter reads event data every 4 seconds. If a batch of rated data has not existed for the time period defined in the **ripeDuration** attribute, RE Formatter does not read it. Valid values must be the following:  
  - Less than or equal to the value of the **ripeDuration** attribute  
  - Evenly divisible by the number of threads configured for the **threadPoolSize** attribute |
| logFormatterWorker | false         | Enables or disables logging for the worker thread pool. |
| dataStoreConnection| "localhost:5000" | The Oracle NoSQL database host name and port number. |
| dataStoreName      | "kvstore"     | The Oracle NoSQL system data store name. |
To configure RE Formatter output, you configure the BRM CDR plug-in run by RE Formatter:

1. Access the ECE configuration MBeans:
   a. Log on to the driver machine.
   b. Start the ECE charging servers (if they are not started).
   c. Connect to the ECE charging server node enabled for JMX management.
      This is the charging server node set to `start CohMgt = true` in the `ECE_home/oceceserver/config/ecdTopology.conf` file.
   d. Start a JMX editor that enables you to edit MBean attributes, such as JConsole.
   e. In the editor’s MBean hierarchy, find the ECE configuration MBeans.

2. Expand the ECE Configuration node.
3. Expand `charging.brmCdrPlugins`.
4. Expand `Operations`.
5. Select `addBrmCdrPluginConfiguration`.
6. Add configuration parameters.
7. Click `addBrmCdrPluginConfiguration`.
8. Use Elastic Charging Controller (ECC) to start the instance of Rated Event Formatter that the BRM CDR plug-in is associated with. This is defined in the Rated Event Formatter MBean (`charging.ratedEventFormatters` `pluginName`) attribute.

### Configuring BRM CDR Plug-in

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>tempDirectoryPath</code></td>
<td><code>/tmp/tmp</code></td>
<td>The directory path for the BRM CDR Plug-in to store temporary files while processing the rated events.</td>
</tr>
<tr>
<td><code>doneDirectoryPath</code></td>
<td><code>/tmp/done</code></td>
<td>The directory path for the BRM CDR Plug-in to store completed CDR files from processing the rated events.</td>
</tr>
<tr>
<td><code>doneFileExtension</code></td>
<td><code>.done</code></td>
<td>The file extension for the completed CDR files created by the BRM CDR Plug-in.</td>
</tr>
</tbody>
</table>
6. Stop and start the Rated Event Formatter instance associated with the BRM CDR Plug-in instance.
This document describes how to set up Oracle Communications Billing and Revenue Management (BRM) server components to use Rated Event (RE) Loader to load events.

Topics in this document:

- Enabling a Billing Delay for CDRs
- Disabling Invoice Event Caching
- Setting Up RE Loader for Virtual Column-Enabled Systems
- Configuring Whether to Perform Redo Generation
- Configuring Field Lengths for Input Data Files
- Updating the RE Loader Control Files with ECE Control Data

See also:

- About Loading Rated Events into the BRM Database
- Configuring Rated Event Output from ECE
- Configuring the RE Loader Infranet.properties File
- Improving RE Loader Performance
- Troubleshooting Rated Event Loading

### Enabling a Billing Delay for CDRs

Offline charging often results in events arriving to be rated after the billing date that they belong to. For example, an event that occurred on December 31 might arrive on January 2nd, past the January 1st billing date. To handle this delay, you must configure delayed billing.

To enable delayed billing, see the description on "Setting Up Delayed Billing" in *BRM Configuring and Running Billing*.

### Disabling Invoice Event Caching

If your system uses both RE Loader and invoicing, you must disable invoice event caching to ensure that invoices contain event details.

To disable invoice event caching, disable the `event_cache` entry in the CM pin.conf file. See *BRM Invoicing*. 

---

4

Configuring BRM Server Components for Rated Event Loading
Setting Up RE Loader for Virtual Column-Enabled Systems

If you generate virtual columns on event tables in your BRM installation, you must run the `pin_gen_classid_values.pl` script. Running the script ensures that the proper mapping of BRM object types and their corresponding object IDs is created for your extended event objects in a virtual column-enabled system. For information about enabling virtual columns in the BRM database, see the discussion on virtual columns in *BRM System Administrator’s Guide*.

Configuring Whether to Perform Redo Generation

By default, RE Loader skips redo generation when loading files into the BRM database. This optimizes loading performance, but it can cause you to lose data if your system shuts down ungracefully.

You can re-enable redo generation by removing the UNRECOVERABLE option from each RE Loader control file.

To enable redo generation, do the following for each table's control file:

1. Open the `BRM_home/apps/pin_re/control_file` file in a text editor, where `control_file` can be one of the files shown in Table 4–1.

2. Comment out the UNRECOVERABLE option:

   ```
   # UNRECOVERABLE
   ```

   **Caution:** Removing the UNRECOVERABLE option significantly decreases loading performance.

3. Save and close the file.

Configuring Field Lengths for Input Data Files

Any value in the input data file longer than 255 characters must include its maximum size. If the maximum size is not specified, the value is truncated to 255 characters when it is loaded into the database.

Fields in the input data file that should not be loaded into the database are specified with the label FILLER in the SQL Loader control file. If the input data file contains a

---

**Table 4–1 RE Loader Control Files**

<table>
<thead>
<tr>
<th>File Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>event_bal_impacts_t.ctl</td>
<td>Control file for the EVENT_BAL_IMPACTS_T table.</td>
</tr>
<tr>
<td>event_delayed_act_wap_inter_t.ctl</td>
<td>Control file for the EVENT_DELAYED_ACT_WAP_INTER_T table.</td>
</tr>
<tr>
<td>event_delayed_session_gprs_t.ctl</td>
<td>Control file for the EVENT_DELAYED_SESSION_GPRS_T table.</td>
</tr>
<tr>
<td>event_sub_bals_t.ctl</td>
<td>Control file for the EVENT_SUB_BALS_T table.</td>
</tr>
<tr>
<td>event_sub_bal_imp_t.ctl</td>
<td>Control file for the EVENT_SUB_BAL_IMP_T table.</td>
</tr>
<tr>
<td>event_dlay_sess_tlcs_t.ctl</td>
<td>Control file for the EVENT_DLAY_SESS_TLCS_T table.</td>
</tr>
<tr>
<td>event_dlay_sess_tlcs_svc_cds_t.ctl</td>
<td>Control file for the EVENT_DLAY_SESS_TLCS_SVC_CDS_T table.</td>
</tr>
<tr>
<td>event_t.ctl</td>
<td>Control file for the EVENT_T table.</td>
</tr>
<tr>
<td>event_total_t.ctl</td>
<td>Control file for the EVENT_TOTAL_T table.</td>
</tr>
<tr>
<td>event_dlyd_session_tlco_gsm_t.ctl</td>
<td>Control file for the EVENT_DLYD_SESSION_TLCO_GSM_T table.</td>
</tr>
</tbody>
</table>
FILLER field with a value longer than 255 characters, SQL Loader aborts with an error specifying the field at fault. If this happens, add the maximum field size to the field entry in the SQL Loader control file. Use this syntax:

Field_name  FILLER  CHAR(max_size)

For example:

DISCOUNT_INFO  FILLER  CHAR(2000)

Updating the RE Loader Control Files with ECE Control Data

The following ECE control files are used by the BRM Rated Event (RE) Loader. RE Loader uses these files for processing rated events in the call details records (CDR) coming from ECE.

- **event_t.ctl**: Includes control data used for loading data from the ECE usage request into columns of the EVENT_T table.
- **event_dlay_ses_tlcs_t.ctl**: Includes control data used for loading input and output volume data from the ECE usage request into columns of the EVENT_DLAY SES_TLCS_T table.

To update the RE Loader control files with ECE control data:

1. Go to BRM_home/apps/pin_rel directory.
2. Copy the ECE_home/oceceserver/brm_config/event_t.ctl and ECE_home/oceceserver/brm_config/event_dlay_ses_tlcs_t.ctl control files to each processing directory in BRM_home/apps/pin_rel. For example, if you have a processing directory called GPRS, copy the ECE event_t.ctl and ECE event_dlay_ses_tlcs_t.ctl control files to BRM_home/apps/pin_rel/GPRS.
3. If you have customized control files, modify the ECE control files to load your custom data.
This document describes how to configure the Oracle Communications Billing and Revenue Management (BRM) Rated Event (RE) Loader Infranet.properties file.

Topics in this document:
- Configuring the RE Loader Infranet.properties File
- BRM Database Connection
- BRM Server Connection
- RE Loader Daemon
- RE Loader Processing
- Loading Event Records
- Storable Class Configuration
- RE Loader Log Files

See also:
- About Loading Rated Events into the BRM Database
- Configuring BRM Server Components for Rated Event Loading
- Improving RE Loader Performance
- Troubleshooting Rated Event Loading

**Configuring the RE Loader Infranet.properties File**

To configure RE Loader, you edit the RE Loader Infranet.properties file.

The Infranet.properties file contains configuration information for processing CDR files, such as the location of the RE Loader processing directory, how to connect to the BRM database, and how to process specific events.
To configure your RE Loader Infranet.properties file:
1. Open the BRM_home/apps/pin_rel/Infranet.properties file in a text editor.
2. Edit the file.
3. Save and close the file.

**BRM Database Connection**

Configure the following entries in the RE Loader Infranet.properties file to connect to the BRM database.

**Table 5–1 Database Connection Entries**

<table>
<thead>
<tr>
<th>Entry</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>infranet.rel.dbtype</td>
<td>Specifies the BRM database type. The initial value is <code>oracle</code>. The default is <code>ORACLE</code>.</td>
</tr>
<tr>
<td>infranet.rel.dbname</td>
<td>Specifies the BRM database name. The initial value is the TNSNAMES alias in the Oracle_home/network/admin/tnsnames.ora file. Ensure that this value is correct for your system.</td>
</tr>
<tr>
<td>infranet.rel.userid</td>
<td>Specifies the user ID for connecting to the BRM database. Ensure that this value is correct for your system.</td>
</tr>
<tr>
<td>infranet.rel.password</td>
<td>Specifies the password for connecting to the BRM database. Ensure that this value is correct for your system.</td>
</tr>
<tr>
<td>infranet.rel.dbhost</td>
<td>Specifies the database machine's host name.</td>
</tr>
<tr>
<td>infranet.rel.dbport</td>
<td>Specifies the database port number. The initial value and the default are 1433.</td>
</tr>
<tr>
<td>infranet.rel.partition_set_number</td>
<td>Specifies the partition set number, from 1 through 7. This entry applies only to BRM databases with multiple delayed partition sets. The initial value and the default are 1.</td>
</tr>
<tr>
<td>■ 1 uses delayed partition set P_1D to P_12D.</td>
<td></td>
</tr>
<tr>
<td>■ 2 uses delayed partition set P_1D to P_12D2.</td>
<td></td>
</tr>
<tr>
<td>■ 3 uses delayed partition set P_1D to P_12D3.</td>
<td></td>
</tr>
<tr>
<td>■ 4 uses delayed partition set P_1D to P_12D4.</td>
<td></td>
</tr>
<tr>
<td>■ 5 uses delayed partition set P_1D to P_12D5.</td>
<td></td>
</tr>
<tr>
<td>■ 6 uses delayed partition set P_1D to P_12D6.</td>
<td></td>
</tr>
<tr>
<td>■ 7 uses delayed partition set P_1D to P_12D7.</td>
<td></td>
</tr>
</tbody>
</table>
BRM Server Connection

Configure the following entries in the RE Loader *Infranet.properties* file to configure how log files are created.

**Table 5–2  BRM Server Connection Entries**

<table>
<thead>
<tr>
<th>Entry</th>
<th>Description</th>
</tr>
</thead>
</table>
| **infranet.connection** | Specifies the user login name. For example:  
infranet.connection=pcp://root.0.0.0.1:password@localhost:11960/service/pcm_client  
RE Loader uses this Connection Manager (CM) connection to log audit information.  
**Important:** RE Loader writes audit information to the database specified in this entry. If you use a multischema system, you might want to modify this entry to write audit information to the schema where the records are loaded. |
| **infranet.login.type** | Specifies whether RE Loader requires a login name and password to log in to BRM.  
- 0 specifies that a login name and password *are not* required.  
- 1 specifies that a login name and password *are* required.  
The initial value is 1.  
No default exists. |
| **infranet.failover** | In high-availability systems, specifies the secondary CM connection. For example:  
infranet.failover.1 = pcp://root.0.0.0.db_no:password@failover_host:failover_port/service/pcm_client |
| **infranet.rel.polling_interval** | Specifies the interval, in milliseconds, that RE Loader checks the database to see whether another process is loading. The initial value and the default are 1000.  
The polling interval depends on the number and size of your input files. If you have very large files, make the polling interval longer. If you have many small files, make the interval shorter. |
| **infranet.rel.polling_time_out** | Specifies the time, in milliseconds, that RE Loader waits to load events before exiting. The initial value and the default are 600000.  
The time-out period depends on the number and size of your input files and how many parallel RE Loader processes are running. If you have very large files or many processes, make the time-out period longer. |
| **infranet.rel.updater_threads** | Specifies the number of threads dedicated to the update and preupdate stored procedures. You can specify a fixed number of threads or configure RE Loader to adjust the number of threads based on the number of database objects to update.  
To specify a fixed number of threads, set the entry equal to the desired number of threads.  
To configure RE Loader to automatically adjust the number of threads, set the entry to 0. RE Loader spawns the number of threads shown below:  
Less than 1,000 objects: 2 threads  
Between 1,000 and 200,000 objects: 4 threads  
More than 200,000 objects: 8 threads  
The initial value is 4.  
The default is 1.  
**Note:** Specifying a number of threads that exceeds the number of CPUs in your system may cause deadlock due to a lack of system resources. If you set the **infranet.rel.updater_threads** entry to a value greater than 8, RE Loader returns a warning message and continues processing. |
RE Loader Daemon

To run the RE Loader daemon, add or modify the RE Loader daemon `Infranet.properties` file entries shown in Table 5–3.

---

**Table 5–3 RE Loader Daemon Entries**

<table>
<thead>
<tr>
<th>Entry</th>
<th>Description</th>
</tr>
</thead>
</table>
| infranet.rel.validate_dbnumber | Specifies whether RE Loader performs an extra validation step to ensure that it is loading a call detail record (CDR) file into the correct database schema. The initial value and the default are **true**.  
**Important:** Use this option only for debugging. In a production environment, set this to **false**. Setting it to **true** degrades performance while loading data into the database.  
See "Turning Off Database Verification to Improve Processing Performance". |
| infranet.rel.validate_indexes | Specifies whether RE Loader verifies that the database indexes are correct before loading data into the database.  
The initial value is **false**.  
The default is **true**.  
**Important:** Use this option only for debugging. In a production environment, set this to **false**. Setting it to **true** degrades performance while loading data into the database. See "Turning Off Index Verification to Improve Database Loading Performance". |
| infranet.rel.max_increment_by | Specifies the number of database schemas in your system. This value is used by the POID generation algorithm to ensure that POIDs are unique across all database schemas in your system.  
The initial value and the default are **20**.  
See "Preventing POID Errors in Multischema Systems". |
| infranet.rel.sort.limit | Defines the maximum number of CDRs that the preprocessing script can sort by account ID. This improves performance later during the balance updating process.  
If the number of CDRs in the input file is greater than the **infranet.rel.sort.limit** value, the preprocessing script does not sort the CDRs.  
The initial value is **100000**.  
The default is **500000**. |

---

**Table 5–2 (Cont.) BRM Server Connection Entries**

<table>
<thead>
<tr>
<th>Entry</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>batch.check.interval</td>
<td>Specifies the time interval, in seconds, to monitor files from the output directory. The default is <strong>5</strong>.</td>
</tr>
<tr>
<td>batch.file.rename.extension</td>
<td>Specifies the file name extension that the RE Loader daemon uses to rename the interim files before processing them. The default is <strong>.bc</strong>.</td>
</tr>
<tr>
<td>batch.start.highload.time</td>
<td>Specifies the start time of your system’s busiest period. Specify the hour, minute, and second, in <strong>hhmmss</strong> format, using the 24-hour clock.</td>
</tr>
<tr>
<td>batch.end.highload.time</td>
<td>Specifies the start time of your system’s slowest period. Specify the hour, minute, and second, in <strong>hhmmss</strong> format, using the 24-hour clock.</td>
</tr>
<tr>
<td>batch.lock.socket.addr</td>
<td>Specifies the port address of the process.</td>
</tr>
<tr>
<td>batch.rel.archiveDir</td>
<td>Specifies the full path to the directory where a successfully processed file is archived. This is the default archive directory for all the event handlers.</td>
</tr>
<tr>
<td>batch.rel.rejectDir</td>
<td>Specifies the full path to the directory where an unsuccessfully processed file is stored. This is the default reject directory for all the event handlers.</td>
</tr>
<tr>
<td>batch.random.events</td>
<td>Specifies the name of the event to process. If you have two or more types of events, separate each with a comma, but no blank space. For example, TEL,SMS,GPRS.</td>
</tr>
</tbody>
</table>
Configuring the RE Loader Infranet.properties File

Configure the following entries in the RE Loader Infranet.properties file to configure how RE Loader retrieves CDR files and loads events. Set the default configuration entries shown in Table 5–4. The configuration information in this section applies to all events except for those defined in the storable class-specific section.

### RE Loader Processing

Table 5–3  (Cont.) RE Loader Daemon Entries

<table>
<thead>
<tr>
<th>Entry</th>
<th>Description</th>
</tr>
</thead>
</table>
| **event.max.at.highload.time** | Specifies the highest number of the RE Loader threads permitted to run simultaneously for this event during the high-load time (that is, from batch.start.highload.time to batch.end.highload.time).  
For example, if event.max.at.highload.time is 2, two threads are permitted to run simultaneously for this event during the high-load time. |
| **event.max.at.lowload.time** | Specifies the highest number of the RE Loader threads permitted to run simultaneously for this event during the low-load time (that is, from batch.end.highload.time to batch.start.highload.time).  
For example, if event.max.at.lowload.time is 2, two threads are permitted to run simultaneously for this event during the low-load time. |
| **event.file.location** | Specifies the full path name of the directory to monitor for the arrival of new files that match the pattern in event.file.pattern.  |
| **event.file.pattern** | Specifies the file name pattern to look for. You can use an asterisk (*) to represent zero or more characters in the file name. No other wildcards are supported. |
| **event.archiveDir** | (Optional) Specifies the full path to the directory where a successfully processed file is archived for a particular event handler. When multiple event handlers are configured, configure this entry to specify the directory that archives files from a particular event handler. |
| **event.rejectDir** | (Optional) Specifies the full path to the directory where an unsuccessfully processed file is stored for a particular event handler. When multiple event handlers are configured, configure this entry to specify the directory that stores files from a particular event handler. |
| **event.file.type** | Specifies the type of input CDR file.  
- **ECE_PRE_SPLIT** specifies that RE Loader uses ECE generated preprocessed control files and data files.  
- **STANDARD** specifies that RE Loader uses pipeline generated input files.  
The default is **STANDARD**.  
Note: You cannot specify both **ECE_PRE_SPLIT** and **STANDARD** in the same Infranet.properties file. |
### Table 5–4 Default Configuration Entries

<table>
<thead>
<tr>
<th>Entry</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>infranet.rel.default.interim_directory</td>
<td>Specifies the RE Loader processing directory. This is the location where RE Formatter loads the CDR files. The initial value is $BRM_home/apps/pin_rel$. The default is ./</td>
</tr>
</tbody>
</table>
| infranet.rel.default.supported_creation_processes       | Specifies which creation processes are supported. As initially configured, RE Loader supports all creation processes:  
  - PIN_REL_TRANSFORM_CDR specifies that the file was last processed by the pin_rel_transform_cdr.pl script and therefore contains discount events.  
  - SUSPENSE_CREATE specifies that the RE Loader process creates new suspense records in the suspended usage table.  
  - SUSPENSE_UPDATE specifies that the RE Loader process updates existing suspense records in the suspended usage table.  
  By default, RE Loader supports only the PIN_REL_TRANSFORM_CDR process. |
| infranet.rel.default.failure_script                     | Specifies the script called when RE Loader attempts to reload events that previously failed to load into the database. The initial value and the default are pin_rel_handle_interim_files.pl. |
| infranet.rel.default.failure_flags                      | Specifies the flag passed to the failure script. You can specify the following flags in the default pin_rel_handle_interim_files.pl failure script:  
  - 0 to do nothing.  
  - 1 to rename the interim files by appending _saved.timestamp to the file name.  
  - 2 to delete the temporary files.  
  - 3 to move the unsuccessfully processed data files generated by ECE to the reject subdirectory.  
  The initial value and the default are 1. |
| infranet.rel.default.preprocess_script                  | Specifies the name of the preprocessing script. The initial value and the default are pin_rel_preprocess_cdr.pl. |
| infranet.rel.default.preprocess_flags                   | Specifies the flag passed to the preprocessing script. The initial value and the default are 0. |
| infranet.rel.default.load_util                          | Specifies the name of the load utility.  
  **Important:** If you use Oracle SQL Loader, you must use conventional-path loading. BRM does not support direct-path loading. You must also use the APPEND option in your RE Loader control files. Do not use the TRUNCATE option. Specify the conventional-path loading mode as follows: sqlldr direct=false  
  The initial value and the default are sqlldr direct=false streamsize=5000000 readsize=1000000. |
| infranet.rel.default.preupdater_sproc                   | Specifies the name of the preupdate stored procedure. The initial value and the default are pin_rel.pin_rel_pre_updater_sp. |
| infranet.rel.default.preupdater_batch_size              | Specifies the size of the preupdate batch. The initial value and the default are 5. |
| infranet.rel.default.preupdater_flags                   | Specifies the flag passed to the preupdate stored procedure. The initial value and the default are 1. |
| infranet.rel.default.updater_sproc                      | Specifies the name of the update stored procedure. The initial value and the default are pin_rel.pin_rel_updater_sp. |
| infranet.rel.default.updater_batch_size                 | Specifies the size of the update batch. The initial value and the default are 5. |
| infranet.rel.default.updater_flags                      | Specifies the flag passed to the update stored procedure. The initial value and the default are 1. |
| infranet.rel.default.success_script                     | Specifies the script called when RE Loader successfully loads a batch of events into the BRM database. The initial value and the default are pin_rel_handle_interim_files.pl. |
Loading Event Records

Configure the following entries in the RE Loader Infranet.properties file to configure how ER Loader handles event records.

Table 5–5 RE Loader Event Handling Configuration Entries

<table>
<thead>
<tr>
<th>Entry</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>infranet.rel.default.success_flags</td>
<td>Specifies the flag passed to the success script. You can specify the following flags in the default pin_rel_handle_interim_files.pl script: 0 to do nothing, 1 to rename the interim files by appending .saved.timestamp to the file name, 2 to delete the temporary files. The initial value and the default are 2.</td>
</tr>
<tr>
<td>infranet.rel.default.storable_class</td>
<td>Specifies the storable class you are loading. The initial value and the default are /event/delayed/session/gprs. Important: If you use Oracle SQL Loader, use the APPEND option in your RE Loader control files. Do not use the TRUNCATE option.</td>
</tr>
<tr>
<td>infranet.rel.default.creation_process</td>
<td>Specifies whether the file contains prerated, rerated, or discount events: PIN_REL_TRANSFORM_CDR specifies that the file was last processed by the pin_rel_transform_cdr.pl script and therefore contains discount events. Important: RE Loader can dynamically source the creation process from the event record header file. Uncomment this entry only if all of your event record files come from the same creation process.</td>
</tr>
<tr>
<td>infranet.rel.default.ece_control_file_directory</td>
<td>Specifies the location of the control files generated by BRM Elastic Charging Engine (ECE). The default is BRM_home/apps/pin_rel.</td>
</tr>
<tr>
<td>infranet.rel.default.ece_data_file_directory</td>
<td>Specifies the location of the data files generated by ECE. The default is BRM_home/apps/pin_rel.</td>
</tr>
<tr>
<td>infranet.rel.ece_preprocessed</td>
<td>Specifies whether RE Loader uses ECE generated preprocessed control files and data files: TRUE specifies that RE Loader uses ECE generated preprocessed control files and data files, FALSE specifies that RE Loader uses pipeline generated input files. The default is FALSE.</td>
</tr>
</tbody>
</table>

Table 5–4 (Cont.) Default Configuration Entries

<table>
<thead>
<tr>
<th>Entry</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>infranet.rel.use_end_time</td>
<td>Specifies whether RE Loader uses the start time or end time of the rated event for deciding the billing cycle. 1 specifies that RE Loader uses the end time of the rated event for deciding the billing cycle. The initial value and the default are 1. 0 specifies that RE Loader uses the start time of the rated event for deciding the billing cycle.</td>
</tr>
<tr>
<td>infranet.rel.default.header.record_type</td>
<td>Specifies the header record type. The initial value and the default are 010.</td>
</tr>
<tr>
<td>infranet.rel.default.detail.record_type</td>
<td>Specifies the detail record type. The initial value and the default are 020.</td>
</tr>
<tr>
<td>infranet.rel.default.trailer.record_type</td>
<td>Specifies the trailer record type. The initial value and the default are 090.</td>
</tr>
<tr>
<td>infranet.rel.field.delimiter</td>
<td>Specifies the delimiter symbol. The initial value and the default are \t for tabs.</td>
</tr>
<tr>
<td>infranet.rel.header.position.storable_class</td>
<td>Specifies which field in the event record file contains the storable class name. The initial value and the default are 20. Note: When you set this field to 0, RE Loader uses the default storable class specified in infranet.rel.default.storable_class.</td>
</tr>
</tbody>
</table>
To override the default settings for the specified storable class, set the storable class-specific Infranet.properties entries shown in Table 5–6.

For each storable class, only the infranet.rel.storable_class.classname.number_of_tables and infranet.rel.storable_class.classname.table.N.name entries are mandatory. RE Loader uses the default settings for any undefined storable class-specific entries.

When editing these entries:

- Create a set of entries for each event you want to load.
- Replace classname with the appropriate storable class name. For example, use event_delayed_session_gprs for the /event/delayed/session/gprs storable class.
- Create a set of *.table.N.* entries for each table. For example, if the storable class contains three tables, create a set of *.table.1.* entries, a set of *.table.2.* entries, and a set of *.table.3.* entries.

### Table 5–6 Storable Class-Specific Configuration Entries

<table>
<thead>
<tr>
<th>Entry</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>infranet.rel.storable_class.classname.interim_directory</td>
<td>RE Loader processing directory. This is the location where preprocessed events are temporarily stored before they are loaded.</td>
</tr>
<tr>
<td>infranet.rel.storable_class.classname.supported_creation_processes</td>
<td>Specifies whether the file contains prerated, rerated, or discount events.</td>
</tr>
<tr>
<td>infranet.rel.storable_class.classname.failure_script</td>
<td>Specifies the script to call when RE Loader attempts to load events that previously failed to load into the database.</td>
</tr>
</tbody>
</table>
Configure the following entries in the RE Loader Infranet.properties file to configure how log files are created.

<table>
<thead>
<tr>
<th>Entry</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>infranet.rel.storable_class.classname.failure_flags</td>
<td>Specifies the flag to pass to the failure script.</td>
</tr>
<tr>
<td>infranet.rel.storable_class.classname.preprocess_script</td>
<td>Specifies the name of the preprocessing script.</td>
</tr>
<tr>
<td>infranet.rel.storable_class.classname.preprocess_flags</td>
<td>Specifies the flag to pass to the preprocessing script.</td>
</tr>
<tr>
<td>infranet.rel.storable_class.classname.number_of_tables</td>
<td>Specifies the number of tables in the storable class. <strong>Important:</strong> This entry is mandatory for all types of events.</td>
</tr>
<tr>
<td>infranet.rel.storable_class.classname.table.N.name</td>
<td>Specifies the name of a storable class table. <strong>Important:</strong> This entry is mandatory for all types of events.</td>
</tr>
<tr>
<td>infranet.rel.storable_class.classname.table.N.load_util</td>
<td>Specifies the name of the load utility. <strong>Important:</strong> If you use Oracle SQL Loader, you must use conventional-path loading. BRM does not support direct-path loading. You must also use the APPEND option in your RE Loader control files. Do not use the TRUNCATE option. Specify the conventional-path loading mode as follows: sqlldr direct=false</td>
</tr>
<tr>
<td>infranet.rel.storable_class.classname.table.N.control_file</td>
<td>Specifies the control file to use when loading the data file into the database.</td>
</tr>
<tr>
<td>infranet.rel.storable_class.classname.preupdater_sproc</td>
<td>Specifies the name of the preupdater stored procedure.</td>
</tr>
<tr>
<td>infranet.rel.storable_class.classname.preupdater_batch_size</td>
<td>Specifies the preupdater batch size.</td>
</tr>
<tr>
<td>infranet.rel.storable_class.classname.preupdater_flags</td>
<td>Specifies the flag to pass to the preupdater stored procedure.</td>
</tr>
<tr>
<td>infranet.rel.storable_class.classname.updater_sproc</td>
<td>Specifies the name of the updater stored procedure.</td>
</tr>
<tr>
<td>infranet.rel.storable_class.classname.updater_batch_size</td>
<td>Specifies the updater batch size.</td>
</tr>
<tr>
<td>infranet.rel.storable_class.classname.updater_flags</td>
<td>Specifies the flag to pass to the updater stored procedure.</td>
</tr>
<tr>
<td>infranet.rel.storable_class.classname.success_script</td>
<td>Specifies the script to call when RE Loader successfully loads a data file into the BRM database.</td>
</tr>
<tr>
<td>infranet.rel.storable_class.classname.success_flags</td>
<td>Specifies the flag to pass to the success script when RE Loader successfully loads a data file into the BRM database.</td>
</tr>
</tbody>
</table>
### Table 5–7 RE Loader Log File Configuration Entries

<table>
<thead>
<tr>
<th>Entry</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>infranet.log.file</td>
<td>Specifies the name of the RE Loader log file. The initial value is <em>rel.pinlog</em>. No default exists.</td>
</tr>
<tr>
<td>infranet.log.name</td>
<td>Specifies the name of the application. The initial value is <em>REL</em> for RE Loader. No default exists.</td>
</tr>
</tbody>
</table>
| infranet.log.level   | Specifies the log reporting level:  
  ■ 1 specifies error-level reporting.  
  ■ 2 specifies warning-level reporting.  
  ■ 3 specifies debug-level reporting.  
  The initial value is 1.  
  The default is `String.valueOf(ErrorLog.Error)` (equivalent to 1).  
  See “Setting the Reporting Level for Logging Messages” in *BRM System Administrator’s Guide*.                                                                 |
| infranet.log.logallebuf | Specifies whether RE Loader automatically logs all EbufExceptions. The initial value is `true`. No default exists.                                                                                       |
| infranet.rel.custom_error_codes | Specifies the name of the custom error code file. The initial value is `CustomErrorCodes.properties`. No default exists.  
  To move this file from its default location, you must create a symbolic link between the name of the file and its new location. To create this link, go to the `BRM_home/apps/pin_rel` directory and enter the following at the command prompt:  
  ```bash  
  $ ln -s path_to_where_file_was_moved /CustomErrorCodes.properties  
  ./CustomErrorCodes.properties  
  ``` |
Configuring RE Loader to Run Automatically by Using Batch Controller

This document describes how to set up Oracle Communications Billing and Revenue Management (BRM) Batch Controller to use Rated Event (RE) Loader to load events.

Topics in this document:

- Configuring RE Loader to Run Automatically by Using Batch Controller
- Configuring the RE Loader Batch Handler
- Configuring Batch Controller

See also:

- About Loading Rated Events into the BRM Database
- Configuring the RE Loader Infranet.properties File
- Improving RE Loader Performance
- Troubleshooting Rated Event Loading

Configuring RE Loader to Run Automatically by Using Batch Controller

To run RE Loader automatically, configure Batch Controller and the RE Loader batch handler. When a pre-rated event file is available, Batch Controller automatically starts the RE Loader batch handler, which runs the RE Loader utility (pin_rel).

To configure RE Loader to run automatically, do the following for each instance of RE Loader:

- Configuring the RE Loader Batch Handler
- Configuring Batch Controller

Specifying each RE Loader batch handler and handler settings in the Batch Controller configuration file. See "Handler Identification" in BRM System Administrator’s Guide.

Configuring the RE Loader Batch Handler

To configure the RE Loader batch handler:

1. Give the SampleRelHandler.pl file a unique name. You will configure Batch Controller to call the handler using this name.
2. Create the following subdirectories: An archive subdirectory where successfully processed files can be stored and a reject subdirectory where unsuccessfully processed files can be stored.

3. Open the SampleRelHandler_config.values file and modify the entries shown in Table 6–1.

Table 6–1 Mandatory RE Loader Batch Handler Configuration Entries

<table>
<thead>
<tr>
<th>Entry</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$FILETYPE</td>
<td>Specifies the event record file-name pattern to look for. To load only specific files, change the value of this entry. The default is <em>.dat.bc (any data file processed by Batch Controller). Batch Controller runs the RE Loader batch handler for each file with a name that matches this pattern. Tip: You can use an asterisk (</em>) to represent zero or more characters in the file name. No other wildcards are supported.</td>
</tr>
<tr>
<td>SHANDLER_DIR</td>
<td>Specifies the full path to the directory containing the RE Loader batch handler, which is this processing directory.</td>
</tr>
<tr>
<td>SpinRELDir</td>
<td>Specifies the full path to the directory containing the RE Loader application, which is this processing directory.</td>
</tr>
<tr>
<td>SpinREL</td>
<td>Specifies the full path to the batch application executable.</td>
</tr>
<tr>
<td>$STAGING</td>
<td>Specifies the full path to the event output directory. If you specify a directory other than the event output directory, use the UNIX command that links the event output directory to the input staging directory.</td>
</tr>
<tr>
<td>$PROCESSING</td>
<td>Specifies the full path to the directory from which event record files are processed. The default is defined in the SpinRELDir environment variable. This must be the same directory specified in the following RE Loader Infranet.properties entries: ■ infranet.rel.default.interim_directory ■ infranet.rel.storable_class.className.interim_directory</td>
</tr>
<tr>
<td>$ARCHIVE</td>
<td>Specifies the full path to the directory where a successfully processed file is archived. This is the archive subdirectory created in step 2. Change this value if you used a name other than the default, SpinRELDir/archive.</td>
</tr>
<tr>
<td>$REJECT</td>
<td>Specifies the full path to the directory where an unsuccessfully processed file is stored. This is the reject subdirectory created in step 2. Change this value if you used a name other than the default, SpinRELDir/reject.</td>
</tr>
</tbody>
</table>

4. Save and close the file.

Configuring Batch Controller

The RE Loader package includes Batch Controller. If your system already has Batch Controller installed, the RE Loader installer does not install another. Use the sample Batch Controller properties file (BRM_home/apps/pin_rel/SampleBatchControllerInfranet.properties) to configure Batch Controller.

The default configuration for the sample Batch Controller runs RE Loader batch handler whenever a rated event record file appears in the event output directory. You can change this setting to trigger the RE Loader batch handler at specified times or based on other kinds of occurrences by editing the event entries that trigger the RE Loader batch handler. See “Setting Activity Times and Triggers” in BRM System Administrator’s Guide.

The optimal number of RE Loader processes to configure is at least three. Depending on the size of your files and the time it takes to load events, configuring four or five processes might save time.

To configure Batch Controller:
1. Copy the BRM_home/apps/pin_rel/SampleBatchControllerInfranet.properties file to your BRM_home/apps/batch_controller directory and change its name to Infranet.properties.

2. Open the BRM_home/apps/batch_controller/Infranet.properties file.

3. Edit the BRM connection parameters.


4. Set the relHandler.start.string parameter to the path of the RE Loader batch handler and to the name of the handler script that you gave it when configuring the RE Loader batch handler. See step 1 in "Configuring the RE Loader Batch Handler".

   For example:

   ```
   relHandler.start.string   BRM_home/apps/pin_rel/REL_handler_name.pl
   ```

5. (Optional) To change the number of RE Loader batch handler processes you want to run, set the maximum batch handler entries.

   ```
   relHandler.max.at.highload.time 3
   relHandler.max.at.lowload.time 3
   ```

6. Set the cdrFileEvent.file.location parameter to specify the location of the event output directory:

   ```
   cdrFileEvent.file.location /export/Portal/integRate
   ```

7. Set the cdrFileEvent.file.pattern parameter to which files should be processed by Batch Controller:

   ```
   cdrFileEvent.file.pattern cdr*.dat
   ```

   **Tip:** You can use an asterisk (*) as a wildcard character to represent zero or more characters in the file name.

8. Save and close the file.

9. Stop and restart Batch Controller.

For more information about configuring Batch Controller, see "Controlling Batch Operations" in BRM System Administrator’s Guide.
Configuring RE Loader to Run Automatically by Using the RE Loader Daemon

This document describes how to set up Oracle Communications Billing and Revenue Management (BRM) RE Loader Daemon to use Rated Event (RE) Loader to load events.

See also:
- About Loading Rated Events into the BRM Database
- Configuring the RE Loader Infranet.properties File
- Improving RE Loader Performance
- Troubleshooting Rated Event Loading

Configuring RE Loader to Run Automatically by Using the RE Loader Daemon

The RE Loader daemon detects when CDR files arrive in the input directory, and then processes them. RE Loader daemon moves the original file to an archive directory if the records are successfully loaded or to a reject directory if the records are not successfully loaded.

To run RE Loader automatically by using the RE Loader daemon:

1. Configure the RE Loader Infranet.properties file. See “Configuring the RE Loader Infranet.properties File.”

2. Use the pin_ctl utility to run the BRM_home/bin/start_rel_daemon script. See “Starting a Component by Using the pin_ctl Utility” in BRM System Administrator’s Guide.

To stop the RE Loader daemon, run the BRM_home/bin/stop_rel_daemon script.

You can also run these scripts manually.
This document describes how to customize Oracle Communications Billing and Revenue Management (BRM) Rated Event (RE) Loader to load events.

Topics in this document:
- Customizing RE Loader
- Adding New Types of Events for RE Loader to Load
- Creating Custom Error Codes

See also:
- About Loading Rated Events into the BRM Database
- Configuring the RE Loader Infranet.properties File

**Customizing RE Loader**

Some of the steps required to customize RE Loader should be performed by a programmer and database administrator. To customize RE Loader, you should be familiar with the following topics:

- BRM system architecture. See "BRM System Architecture", in *BRM Concepts*, and "About Loading Rated Events into the BRM Database".
- BRM storable classes. See "Understanding Flists and Storable Classes" in *BRM Developer’s Guide*.
- BRM database configuration. See "Database Configuration and Tuning" in *BRM Installation Guide*.
- SQL and creating SQL control files. See your SQL documentation.

You can customize RE Loader by:

- Adding New Types of Events for RE Loader to Load
- Creating Custom Error Codes

**Important:** Do not modify the `rel_updater_sp.sql` stored procedure or any other stored procedure. Modifying a stored procedure can corrupt data and cause maintenance and upgrade problems. Stored procedures are delivered in source code format due to database limitations and are not designed to be modified. To modify a stored procedure, you must obtain permission to do so from Oracle.
Adding New Types of Events for RE Loader to Load

When you offer a new service, you create a new storable class for the service event.

To use RE Loader to load events from a new service or new service subclass, you must create a delayed event for your new service and configure RE Loader to load it.

It is possible to load a subclass of a preconfigured service event without configuring that subclass. However, BRM will be unaware of the subclass because the subclass events is inserted into the parent class table. To track the activity of the subclass events, configure RE Loader to load the specific subclass.

You must create a new delayed event for RE Loader pre-rated events. The name of the new event storable class must start with `/event/delayed` so that BRM can distinguish it from real-time events. For example, `/event/delayed/session/new_event_type`.

**Important:** Avoid loading pre-rated events by using RE Loader and another application such as an optional component.

To add an event for RE Loader to load:

1. If necessary, add the new event storable class to BRM by using Storable Class Editor. See the Storable Class Editor Help. For information about storable classes, see "About Storable Classes and Objects" in BRM Developer’s Guide.

   **Note:** If you installed GSM Manager, the `/telephony`, `/fax`, `/data`, and `/sms` subclasses of `/event/delayed/session/telco/gsm` already exist in the BRM database and do not need to be created. However, if you want to track activity specific to one of these subclasses, you must perform this entire procedure.

2. Create partitions for the event by running the `partition_utils` utility from the `BRM_home/apps/partition_utils` directory.

   For example, the following command creates partitions for `/event/delayed/session/telco/gsm` delayed events:

   ```
   partition_utils -o enable -t delayed -c /event/session/telco/gsm
   ```

   **Important:** You must create partitions for all subclasses of a specific event that you want to load.

   See "Enabling Delayed-Event Partitions" and "partition_utils" in BRM System Administrator’s Guide.
3. Create a control file for the new event. A control file and format file specifies the format for a single database table (array or struct). If you added new fields to an existing array or struct, modify the control or format file for that table. If you added a new array or struct, create a new control or format file for the new table. For instructions on creating a control or format file, see your Oracle documentation.

4. If you created or modified any control files, modify the RE Loader preprocess script (`BRM_home/apps/pin_rel/pin_rel_preprocess_cdr.pl`) to read the new event fields from the event record data and write the fields to the files loaded by SQL Loader. You can follow the steps used for `/event/session/telco/gsm` in the `pin_rel_preprocess_cdr.pl` file as a guide.

5. Create a new RE Loader directory corresponding to the RE Formatter output directory.

6. Add the following entries to the RE Loader Infranet.properties file in each directory:
   - The new event
   - A new service record type corresponding to the new event
   - The new control file that loads the new event
   - The new event tables that hold the new event

7. If you are running RE Loader automatically, you must also do the following:
   a. Configure the RE Loader batch handler in the new directory to load the new event.

Creating Custom Error Codes

You can create custom error codes for RE Loader scripts and utilities by using the RE Loader CustomErrorCodes.properties file. You use this file to list your custom error codes and messages. All entries should follow the FQEC scheme and be grouped with the correct component. See “Checking the RE Loader Log Files for Error Codes”.

To create custom error codes:

1. Modify the RE Loader script or utility to report the error. For more information, see the comments in the appropriate script or utility.

2. Open the `BRM_home/apps/pin_rel/CustomErrorCodes.properties` file in a text editor.

3. Add your custom error code to the file, making sure you use a minor code in the customer-reserved range.

   For example, the following entry creates a custom error code for the load utility:

   ```plaintext
   5100 = Sample load utility error message for a custom return code of 100.
   ```

4. Save and close the file.
This chapter describes how to troubleshoot the Oracle Communications Billing and Revenue Management (BRM) Rated Event (RE) Loader.

Topics in this document:
- About Troubleshooting Rated Event Loading
- Checking the RE Loader Log Files for Error Codes
- Checking for Errors that Occurred during the PreUpdate Process
- Fixing Event Loading Errors
- Debugging Mismatches between Data Files and Control Files

See also:
- About Loading Rated Events into the BRM Database
- Installing Rated Event Loader
- Configuring BRM Server Components for Rated Event Loading
- Configuring the RE Loader Infranet.properties File

About Troubleshooting Rated Event Loading

There are two distinct error-handling actions that RE Loader takes, depending on when the error occurs:

- If an error occurs while events are being loaded, the process is canceled and all events loaded in the session are deleted from the BRM database. The SQL loader errors are logged in a file (BRM_home/apps/pin_rel/file_name.bad) and a fatal error is recorded in the RE Loader log file (Processing_directory/rel.pinlog).

- If an error occurs while RE Loader is updating account balances, bill items, or journals, the loaded events are left in the database and an error is recorded in the RE Loader log file (Processing_directory/rel.pinlog). If RE Loader stops due to errors while updating account balances, bill items, or journals, correct the problem and run RE Loader again.

Some error messages are sent to standard BRM error handling. Check the rel.pinlog log file. See "Checking the RE Loader Log Files for Error Codes".

RE Loader checks for status in two places:

- The /batch/rel session status object.

  This object stores the status of the last RE Loader process. When you start RE Loader, it checks that status. If you try to reload a file that RE Loader has already
successfully updated, the file is rejected because the session status indicates that the update for that file is complete.

- The REL_SUB_PROCESSES_T table.

This table stores information about loading errors that occurred during the preupdating stage. See “Checking for Errors that Occurred during the PreUpdate Process”.

Checking the RE Loader Log Files for Error Codes

RE Loader uses the SQL Loader utility, sqlldr, to load events into the BRM database. The sqlldr process creates a new log file for each input file so that log files from a previous process are not overwritten.

The log files and the temporary files created during preprocessing incorporate the name of the input file in their file names, making it easier to debug if an error occurs.

Error codes follow the fully qualified error code (FQEC) scheme, which consists of a major code that represents the component and a minor code that represents the error number. All BRM-defined errors use a minor code from 0 through 99, and all custom errors use minor codes 100 and above.

For information on how to create custom error codes for RE Loader scripts and utilities, see “Creating Custom Error Codes”.

Note: Because modifying a stored procedure can corrupt data and cause maintenance and upgrade problems, custom error codes cannot be created for stored procedures.

The major and minor error codes for each RE Loader component are shown in Table 9–1.

Table 9–1  RE Loader Major and Minor Error Codes

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
<th>Major Code</th>
<th>BRM Reserved Minor Codes</th>
<th>Customer Reserved Minor Codes</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>Universal code for success.</td>
<td>0</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>RE Loader driver</td>
<td>pin_rel script and Java driver code.</td>
<td>1000</td>
<td>0 - 999</td>
<td>N/A</td>
</tr>
<tr>
<td>Failure script</td>
<td>Script called when RE Loader attempts to load a data file that previously failed to load into the BRM database.</td>
<td>2000</td>
<td>0 - 99</td>
<td>100 - 255</td>
</tr>
<tr>
<td>Transform script</td>
<td>pin_rel_transform_cdr.pl script, which converts discount files into event record format.</td>
<td>3000</td>
<td>0 - 99</td>
<td>100 - 255</td>
</tr>
<tr>
<td>Preprocess script</td>
<td>pin_rel_preprocess_cdr.pl script, which preprocesses the data files and creates bulk-loadable (.blk) files.</td>
<td>4000</td>
<td>0 - 99</td>
<td>100 - 255</td>
</tr>
<tr>
<td>Load utility</td>
<td>sqlldr utility, which loads data into the BRM database.</td>
<td>5000</td>
<td>0</td>
<td>1 - 999</td>
</tr>
<tr>
<td>Preupdate stored procedure</td>
<td>Stored procedure for updating the loaded data before releasing the partition to other RE Loader sessions.</td>
<td>7000</td>
<td>0 - 99</td>
<td>Not available</td>
</tr>
</tbody>
</table>
Table 9–2 shows the BRM-defined error codes and messages, where \textit{value} is the value returned in the error message:

<table>
<thead>
<tr>
<th>RE Loader Error Number</th>
<th>Error Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>REL encountered an error.</td>
</tr>
<tr>
<td>1002</td>
<td>The \textit{infranet.rel.dbtype} properties value found is not supported: \textit{value}. Supported values are: \textit{value}.</td>
</tr>
<tr>
<td>1003</td>
<td>The \textit{infranet.rel.partition_set_number} properties value found is not valid: \textit{value}. Valid values are between \textit{value} and \textit{value}.</td>
</tr>
<tr>
<td>1004</td>
<td>A table name properties value is missing for the given storable class: \textit{value}.</td>
</tr>
<tr>
<td>1005</td>
<td>A duplicate table name properties value was found: \textit{value}.</td>
</tr>
<tr>
<td>1006</td>
<td>The \textit{load_util} properties value is missing for the given storable class: \textit{value}.</td>
</tr>
<tr>
<td>1007</td>
<td>A control file properties value is missing for the given storable class: \textit{value}.</td>
</tr>
<tr>
<td>1008</td>
<td>The control file name could not be found in the command line.</td>
</tr>
<tr>
<td>1009</td>
<td>REL cannot be executed until the Event Extraction Manager is complete.</td>
</tr>
<tr>
<td>1010</td>
<td>An unexpected SQL exception has occurred.</td>
</tr>
<tr>
<td>1011</td>
<td>An error occurred while attempting to connect to the BRM database.</td>
</tr>
<tr>
<td>1012</td>
<td>An error occurred while attempting to connect to the CM. Please validate the \textit{infranet.connection} property value and ensure the CM is running.</td>
</tr>
<tr>
<td>1013</td>
<td>An error occurred while attempting to perform an opcode call.</td>
</tr>
<tr>
<td>1014</td>
<td>An interrupt has occurred and caused an error.</td>
</tr>
<tr>
<td>1015</td>
<td>The following file was not found: \textit{value}.</td>
</tr>
<tr>
<td>1016</td>
<td>An unexpected I/O error was encountered.</td>
</tr>
<tr>
<td>1017</td>
<td>The POID selected from the database sequence exceeds the maximum supported range of \textit{2^{34}}: \textit{value}.</td>
</tr>
<tr>
<td>1018</td>
<td>REL failed to select the partition name from the database.</td>
</tr>
<tr>
<td>1019</td>
<td>The \textit{poid_db} could not be found in the input file.</td>
</tr>
<tr>
<td>1020</td>
<td>The \textit{poid_db} found in the input file does not match the BRM database number for this CM connection. Found: \textit{value}. Expected: \textit{value}.</td>
</tr>
<tr>
<td>1021</td>
<td>The header record could not be found in the input file.</td>
</tr>
<tr>
<td>1022</td>
<td>The storable class was not defined, or was not found in the header record.</td>
</tr>
<tr>
<td>1023</td>
<td>The time format found in the header record is not valid: \textit{value}.</td>
</tr>
</tbody>
</table>
Checking the RE Loader Log Files for Error Codes

Table 9–2 (Cont.) BRM-Defined Error Codes

<table>
<thead>
<tr>
<th>RE Loader Error Number</th>
<th>Error Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>1024</td>
<td>The creation process found in the header record is not supported: <strong>value</strong></td>
</tr>
<tr>
<td></td>
<td>Valid values are: <strong>value</strong></td>
</tr>
<tr>
<td>1026</td>
<td>An invalid command-line was provided.</td>
</tr>
<tr>
<td>1027</td>
<td>The CM and JDBC BRM database connections are not configured to the same database schema.</td>
</tr>
<tr>
<td>1028</td>
<td>The REL session has timed out waiting for another REL session to complete.</td>
</tr>
<tr>
<td>1029</td>
<td>The file has previously completed successfully so it will not be loaded again: <strong>value</strong></td>
</tr>
<tr>
<td>1030</td>
<td>The file is currently being processed by another REL session: <strong>value</strong></td>
</tr>
<tr>
<td>1031</td>
<td>The <strong>value</strong> key is missing from the properties file.</td>
</tr>
<tr>
<td>1032</td>
<td>The <strong>value</strong> value is missing from the properties file.</td>
</tr>
<tr>
<td>1033</td>
<td>The configured number of tables for this storable class does not match the configured tables: <strong>value</strong></td>
</tr>
<tr>
<td>1034</td>
<td>A number formatting error was encountered in the properties value for: <strong>value</strong></td>
</tr>
<tr>
<td>1035</td>
<td>The <strong>infranet.rel.updater_threads</strong> properties value found is not valid: <strong>value</strong></td>
</tr>
<tr>
<td></td>
<td>Valid values are between <strong>value</strong> and <strong>value</strong>.</td>
</tr>
<tr>
<td></td>
<td>To have REL auto-choose an appropriate number of threads, use the value: <strong>value</strong></td>
</tr>
<tr>
<td>1036</td>
<td>An error occurred while attempting to parse a number for: <strong>value</strong></td>
</tr>
<tr>
<td>1038</td>
<td>Cannot have control file with 'TRUNCATE' option when running REL in parallel loading mode between multiple REL processes.</td>
</tr>
</tbody>
</table>

Table 9–3 shows the BRM-defined failure script error codes.

Table 9–3 Failure Script Error Messages

<table>
<thead>
<tr>
<th>Failure Script Error Number</th>
<th>Error Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>2000</td>
<td>The failure script encountered an error.</td>
</tr>
<tr>
<td></td>
<td>The given command-line was: <strong>value</strong></td>
</tr>
<tr>
<td>2001</td>
<td>The failure script command-line given arguments are not supported.</td>
</tr>
<tr>
<td></td>
<td>The given command-line was: <strong>value</strong></td>
</tr>
<tr>
<td>2002</td>
<td>The failure script command-line given flags value provided is not supported.</td>
</tr>
<tr>
<td></td>
<td>The given command-line was: <strong>value</strong></td>
</tr>
<tr>
<td>2003</td>
<td>The failure script command-line given directory could not be read.</td>
</tr>
<tr>
<td></td>
<td>The given command-line was: <strong>value</strong></td>
</tr>
</tbody>
</table>

Table 9–4 shows the BRM-defined transform script error codes.

Table 9–4 Transform Script Error Messages

<table>
<thead>
<tr>
<th>Transform Script Error Number</th>
<th>Error Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>3000</td>
<td>The transform script encountered an error.</td>
</tr>
<tr>
<td>3001</td>
<td>The transform script command-line given arguments are not supported.</td>
</tr>
<tr>
<td></td>
<td>The given command-line was: <strong>value</strong></td>
</tr>
<tr>
<td>3002</td>
<td>The transform script command-line given input file could not be read.</td>
</tr>
<tr>
<td></td>
<td>The given command-line was: <strong>value</strong></td>
</tr>
<tr>
<td>3003</td>
<td>The transform script command-line given output file could not be created.</td>
</tr>
<tr>
<td>3004</td>
<td>The transform script command-line given negative discount carry over value is invalid.</td>
</tr>
<tr>
<td></td>
<td>The given command-line was: <strong>value</strong></td>
</tr>
</tbody>
</table>
Table 9–5 shows the BRM-defined preprocess script error codes.

<table>
<thead>
<tr>
<th>Preprocess Script Error Number</th>
<th>Error Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>4000</td>
<td>The preprocess script encountered an error.</td>
</tr>
<tr>
<td></td>
<td>The given command-line was: value</td>
</tr>
<tr>
<td>4001</td>
<td>The preprocess script command-line given arguments are not supported.</td>
</tr>
<tr>
<td></td>
<td>The given command-line was: value</td>
</tr>
<tr>
<td>4002</td>
<td>The preprocess script failed to open a file.</td>
</tr>
<tr>
<td>4003</td>
<td>The preprocess script found the input file to be missing a balance record.</td>
</tr>
<tr>
<td></td>
<td>The given command-line was: value</td>
</tr>
<tr>
<td>4004</td>
<td>The preprocess script found the input file to be missing a detail record.</td>
</tr>
<tr>
<td></td>
<td>The given command-line was: value</td>
</tr>
<tr>
<td>4005</td>
<td>The preprocess script command-line given tables are not supported.</td>
</tr>
<tr>
<td></td>
<td>The given command-line was: value</td>
</tr>
<tr>
<td>4006</td>
<td>The preprocess script command-line given increment_by value is not valid.</td>
</tr>
<tr>
<td></td>
<td>The given command-line was: value</td>
</tr>
<tr>
<td>4007</td>
<td>The preprocess script did not find the expected number of records in the input file.</td>
</tr>
<tr>
<td></td>
<td>The given command-line was: value</td>
</tr>
<tr>
<td>4008</td>
<td>The preprocess script found the input file to be missing an event record.</td>
</tr>
<tr>
<td></td>
<td>The given command line was: value</td>
</tr>
<tr>
<td></td>
<td>Used by SE Loader.</td>
</tr>
<tr>
<td>4009</td>
<td>The preprocess script did not find the expected size for an event record.</td>
</tr>
<tr>
<td></td>
<td>The given command line was: value</td>
</tr>
<tr>
<td></td>
<td>Used by SE Loader.</td>
</tr>
<tr>
<td>4010</td>
<td>The preprocess script failed to parse fields mapping data for generating the control file.</td>
</tr>
<tr>
<td></td>
<td>The given command line was: value</td>
</tr>
<tr>
<td></td>
<td>Used by SE Loader.</td>
</tr>
</tbody>
</table>

Table 9–6 shows the BRM-defined load utility error codes.

<table>
<thead>
<tr>
<th>Load Utility Error Number</th>
<th>Error Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>5000</td>
<td>The database load utility encountered an error.</td>
</tr>
</tbody>
</table>

Table 9–7 shows the BRM-defined insert stored procedure error codes.

<table>
<thead>
<tr>
<th>Insert Stored Procedure Error Number</th>
<th>Error Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>6000</td>
<td>The insert stored procedure encountered an error.</td>
</tr>
</tbody>
</table>

Table 9–8 shows the BRM-defined preupdate stored procedure error codes.
Checking the RE Loader Log Files for Error Codes

Table 9–8  Preupdate Stored Procedure Error Messages

<table>
<thead>
<tr>
<th>Preupdate Stored Procedure Error Number</th>
<th>Error Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>7000</td>
<td>The preupdate stored procedure encountered an error.</td>
</tr>
<tr>
<td>7001</td>
<td>The preupdate stored procedure encountered an error on a select statement.</td>
</tr>
<tr>
<td>7002</td>
<td>The preupdate stored procedure encountered an error on an insert statement.</td>
</tr>
<tr>
<td>7003</td>
<td>The preupdate stored procedure encountered an error on an update statement.</td>
</tr>
<tr>
<td>7004</td>
<td>The preupdate stored procedure encountered an error on a delete statement.</td>
</tr>
<tr>
<td>7008</td>
<td>The preupdate stored procedure encountered a parsing error.</td>
</tr>
<tr>
<td>7010</td>
<td>The preupdate stored procedure could not find an item for an account.</td>
</tr>
<tr>
<td>7011</td>
<td>The preupdate stored procedure encountered an unexpected error.</td>
</tr>
</tbody>
</table>

Table 9–9 shows the BRM-defined update stored procedure error codes.

Table 9–9  Update Stored Procedure Error Messages

<table>
<thead>
<tr>
<th>Update Stored Procedure Error Number</th>
<th>Error Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>8000</td>
<td>The update stored procedure encountered an error.</td>
</tr>
<tr>
<td>8001</td>
<td>The update stored procedure encountered an error on a select statement.</td>
</tr>
<tr>
<td>8002</td>
<td>The update stored procedure encountered an error on an insert statement.</td>
</tr>
<tr>
<td>8003</td>
<td>The update stored procedure encountered an error on an update statement.</td>
</tr>
<tr>
<td>8004</td>
<td>The update stored procedure encountered an error on a delete statement.</td>
</tr>
<tr>
<td>8008</td>
<td>The update stored procedure encountered a parsing error.</td>
</tr>
<tr>
<td>8009</td>
<td>The update stored procedure found its record is already being processed.</td>
</tr>
<tr>
<td>8010</td>
<td>The update stored procedure could not find an item for an account.</td>
</tr>
<tr>
<td>8011</td>
<td>The update stored procedure encountered an unexpected error.</td>
</tr>
<tr>
<td>8012</td>
<td>The update stored procedure encountered an invalid record count error.</td>
</tr>
<tr>
<td>8013</td>
<td>The update stored procedure encountered an error when updating the account balances.</td>
</tr>
<tr>
<td>8014</td>
<td>The update stored procedure encountered an error when updating the item balances.</td>
</tr>
<tr>
<td>8015</td>
<td>The update stored procedure encountered an error at TREL precommit.</td>
</tr>
<tr>
<td>8016</td>
<td>The update stored procedure encountered an error at TREL postcommit.</td>
</tr>
</tbody>
</table>

Table 9–10 shows the BRM-defined success script error codes.

Table 9–10  Success Script Error Messages

<table>
<thead>
<tr>
<th>Success Script Error Number</th>
<th>Error Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>9000</td>
<td>The success script encountered an error.  The given command-line was: value</td>
</tr>
<tr>
<td>9001</td>
<td>The success script command-line given arguments are not supported.  The given command-line was: value</td>
</tr>
<tr>
<td>9002</td>
<td>The success script command-line given flags value provided is not supported.  The given command-line was: value</td>
</tr>
<tr>
<td>9003</td>
<td>The success script command-line given directory could not be read.  The given command-line was: value</td>
</tr>
</tbody>
</table>

Table 9–11 shows the BRM-defined database consistency check error codes.
Checking for Errors that Occurred during the PreUpdate Process

Errors that occur during the preupdate stage of the loading process are stored in the REL_SUB_PROCESSES_T table. To check for values in the table, run SQL*Plus.

To troubleshoot event loading errors, check the RE Loader log file BRM_home/apps/pin_rel/rel.pinlog, where BRM_home is the directory in which you installed BRM components. See "Checking the RE Loader Log Files for Error Codes".

At times, when RE Loader fails, the rel.pinlog file does not list the error. If this occurs, check the status column in the BATCH_T table in the BRM database for the status of the REL process. Table 9–13 lists the status entries (and the corresponding code attributes).
Fixing Event Loading Errors

The correct troubleshooting effort for an event loading error depends upon the error scenario:

- **RE Loader fails to start:**
  
The RE Loader log file (*rel.pinlog*) displays the error code 107.
  
The error occurs if REL is not running.
  
Start REL using the following command:

  `rel<rated event file>`

- **Load failure:**
  
The RE Loader log file (*rel.pinlog*) displays the error code 5000. The status entry for the REL process in the BATCH_T table in the BRM database displays 1 (see Table 9–13).

  In this error scenario, RE Loader failed either before or during the loading of the events in the event file. The events are deleted from the event tables.

  To troubleshoot this error, reload the events normally by using the same command to process the original event file.

---

**Table 9–13 Status Entries in the BATCH_T Table**

<table>
<thead>
<tr>
<th>Status</th>
<th>Code Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>UPDATE_COMPLETE</td>
</tr>
<tr>
<td>1</td>
<td>LOAD_ERROR</td>
</tr>
<tr>
<td>2</td>
<td>UPDATE_ERROR</td>
</tr>
<tr>
<td>4</td>
<td>INSERT_ERROR</td>
</tr>
<tr>
<td>8</td>
<td>PREUPDATE_ERROR</td>
</tr>
<tr>
<td>16</td>
<td>REL_START</td>
</tr>
<tr>
<td>48</td>
<td>PRE_PROCESS</td>
</tr>
<tr>
<td>64</td>
<td>START_LOAD</td>
</tr>
<tr>
<td>80</td>
<td>LOADING</td>
</tr>
<tr>
<td>96</td>
<td>LOAD_COMPLETE</td>
</tr>
<tr>
<td>107</td>
<td>FAIL_TO_START_REL</td>
</tr>
<tr>
<td>240</td>
<td>PROCESS_LOADING</td>
</tr>
<tr>
<td>256</td>
<td>START_INSERT</td>
</tr>
<tr>
<td>512</td>
<td>INSERTING</td>
</tr>
<tr>
<td>768</td>
<td>INSERT_COMPLETE</td>
</tr>
<tr>
<td>1024</td>
<td>START_PREUPDATE</td>
</tr>
<tr>
<td>1280</td>
<td>PREUPDATING</td>
</tr>
<tr>
<td>1536</td>
<td>PREUPDATE_COMPLETE</td>
</tr>
<tr>
<td>3840</td>
<td>PROCESS_PREUPDATING</td>
</tr>
<tr>
<td>4096</td>
<td>START_UPDATE</td>
</tr>
<tr>
<td>8192</td>
<td>UPDATING</td>
</tr>
<tr>
<td>61440</td>
<td>PROCESS_UPDATING</td>
</tr>
</tbody>
</table>

The correct troubleshooting effort for an event loading error depends upon the error scenario:

- **RE Loader fails to start:**
  
The RE Loader log file (*rel.pinlog*) displays the error code 107.
  
The error occurs if REL is not running.
  
Start REL using the following command:

  `rel<rated event file>`

- **Load failure:**
  
The RE Loader log file (*rel.pinlog*) displays the error code 5000. The status entry for the REL process in the BATCH_T table in the BRM database displays 1 (see Table 9–13).

  In this error scenario, RE Loader failed either before or during the loading of the events in the event file. The events are deleted from the event tables.

  To troubleshoot this error, reload the events normally by using the same command to process the original event file.
RE Loader fails during the loading:

The RE Loader log file (rel.pinlog) does not display any error. The status entry for the REL process in the BATCH_T table in the BRM database displays 80 (see Table 9–13).

In this error scenario, RE Loader failed during the loading of the events and RE Loader was unable to update the session status or execute the cleanup process.

Use the `-override` option to start a new process to reload the events. For example:

```
pin_rel -override event_file_name
```

where `event_file_name` is the event file.

Error occurs during the preupdate stored procedure:

The RE Loader log file (rel.pinlog) displays preupdate stored procedure error codes starting at 7000 and below 8000. The status entry for the REL process in the BATCH_T table in the BRM database displays 8 (see Table 9–13).

In this error scenario, RE Loader crashed during the execution of the preupdate stored procedure.

To troubleshoot this error, reload the events normally by using the same command to process the original event file.

RE Loader fails during the updating of events:

The RE Loader log file (rel.pinlog) does not display any error. The status entry for the REL process in the BATCH_T table in the BRM database displays 8192 (see Table 9–13).

In this error scenario, RE Loader crashed during the updating of the events and RE Loader was unable to update the session status or execute the cleanup process.

To troubleshoot this error, reload the events normally by using the same command to process the original event file.

Error occurs during the update stored procedure:

The RE Loader log file (rel.pinlog) displays update stored procedure error codes starting at 8000 and below 9000. The status entry for the REL process in the BATCH_T table in the BRM database displays 2.

In this error scenario, RE Loader successfully loaded the events but failed during the execution of the update stored procedure. The BATCH_REL_SUB_PROCESSES_T table lists the last commit, indicating the point at which the database update failed.

Reload the events normally. The update starts from this point.

### Debugging Mismatches between Data Files and Control Files

RE Loader customizations can sometimes cause data files and control files to become unsynchronized, resulting in SQL Loader failures. To help you debug these situations, use the `pin_rel_enum_blk.pl` script, which enumerates fields in your bulk-loadable files. You can then manually compare the data file entries to the control file.

To debug mismatches between your data files and control files, enter the following commands:

```
% cd BRM_home/apps/pin_rel
% pin_rel_enum_blk.pl file_name [Line_num]
```
where:

- `file_name` specifies the name of the bulk-loadable file. For example, `test2.blk`.
- `Line_num` specifies the line number of the bulk-loadable file that you want to enumerate. The default is 1.

**Retrieving Data About Events You Load**

BRM stores information about events loaded by RE Loader in a `/batch/rel` object. This object contains the input file name, number of records loaded, and other session information.

---

**Note:** If you use multiple database schemas, the `/batch/rel` object is created in the schema specified in the RE Loader `Infranet.properties` file.
This document describes how to improve performance of the Oracle Communications Billing and Revenue Management (BRM) Rated Event (RE) Loader.

Topics in this document:
- Improving RE Loader Performance
- Increasing the Number of Account Balance and Bill Item Updates
- Turning Off Index Verification to Improve Database Loading Performance
- Turning Off Database Verification to Improve Processing Performance
- Pruning Your RE Loader Control and Audit Tables

See also:
- About Loading Rated Events into the BRM Database
- Installing Rated Event Loader
- Configuring BRM Server Components for Rated Event Loading
- Configuring the RE Loader Infranet.properties File

**Improving RE Loader Performance**

You can improve your RE Loader system performance by:
- Increasing the Number of Account Balance and Bill Item Updates
- Turning Off Index Verification to Improve Database Loading Performance
- Turning Off Database Verification to Improve Processing Performance
- Pruning Your RE Loader Control and Audit Tables

**Increasing the Number of Account Balance and Bill Item Updates**

RE Loader performance can be improved by increasing the number of account balance, bill item, and journal updates performed before committing the transaction. You can modify the pre-update batch size and update batch size in the Infranet.properties file to specify how many updates to perform before committing the transaction. For example, if updater_batch_size is set to 5, the stored procedure commits the transaction after every five updates. Increasing the number of updates might increase performance, but the updated account balances, bill items, and journals are not available until the transaction is committed. The default batch_size value is 5.
To change the `preupdater_batch_size` and `updater_batch_size` values:

1. Open the `BRM_home/apps/pin_rel/Infranet.properties` file in a text editor.

   **Note:** If you have already set up your RE Loader processing directories, make sure you edit the `Infranet.properties` file in each directory.

2. If necessary, edit the `infranet.connection` entry to point to the correct database.
   
   For example:
   ```
   infranet.connection=pcp://root.0.0.0.1:password@localhost:37180/service/pcm_client
   ```

3. Specify the preupdater batch size value in the `preupdater_batch_size` entry.
   
   For example:
   ```
   infranet.rel.default.preupdater_batch_size = 8
   ```

4. Specify the updater batch size value in the `updater_batch_size` entry:
   ```
   infranet.rel.default.updater_batch_size = 8
   ```

5. Save and close the file.

---

**Turning Off Index Verification to Improve Database Loading Performance**

By default, RE Loader automatically verifies that your indexes are correct before loading data into the BRM database. This extra step helps you discover configuration errors when testing your system in a development environment.

In production systems, however, you should turn off index verification to improve database loading performance.

When configured to verify indexes, RE Loader performs the following before it runs the SQL Loader utility:

1. Checks whether the indexes to load are partitioned, local, and usable.
2. Performs one of the following:
   - If the indexes are incorrect, RE Loader aborts the loading process and logs which indexes encountered problems.
   - If the indexes are correct, RE Loader runs the SQL Loader utility to load events into the database.

When configured to skip verification, RE Loader automatically runs the SQL Loader utility to load events into the database. When the indexes are incorrect, SQL Loader fails and RE Loader logs only that the database load utility encountered an error.

To turn off index verification:

1. Open the `BRM_home/apps/pin_rel/Infranet.properties` file in a text editor.
2. Set the `Infranet.rel.validate_indexes` entry to `False`:

   ```
   Infranet.rel.validate_indexes = False
   ```

3. Save and close the file.

---

### Turning Off Database Verification to Improve Processing Performance

By default, RE Loader automatically verifies that it is loading events into the correct database schema by validating the database number in the event record file’s first account object with the PCM database number. This extra step helps you discover configuration errors when testing your multischema system in a development environment.

In production systems, however, you should turn off database verification to improve RE Loader database loading performance.

To turn off database verification:

1. Open the `BRM_home/apps/pin_rel/Infranet.properties` file in a text editor.
2. Set the `infranet.rel.validate_dbnumber` entry to `False`:

   ```
   infranet.rel.validate_dbnumber = False
   ```
3. Save and close the file.

---

### Pruning Your RE Loader Control and Audit Tables

RE Loader control and audit tables grow indefinitely, so you should prune them periodically to increase system performance and reduce memory usage. To make pruning easier, you can use the RE Loader `purge_batch_rel_objects` stored procedure, which automatically prunes the tables for you.

To prune your control and audit tables, run the following commands in SQL*Plus:

```sql
sqlplus system/manager@DatabaseAlias
pin_rel.purge_batch_rel_objects(int: Number)
```

where `Number` specifies how many days worth of data to keep in the tables.
This chapter provides reference information for Oracle Communications Billing and Revenue Management (BRM) rated even loading utilities.

Topics in this document:

- pin_rel
pin_rel

Loads batches of rated event records into the BRM database.

There are two ways to use this utility. When you initially run pin_rel, use the command without any options and use the file name as the only command-line parameter. You use the override option when the utility has not successfully completed its process and must be rerun.

The pin_rel utility looks for the event record file in the directory specified in the Infranet.rel.rated_event_file entry in the Infranet.properties file. Before you run the pin_rel utility, make sure the input event record file is in the specified directory and the Infranet.properties file is configured. See "Configuring the RE Loader Infranet.properties File".

Location

BRM_home/apps/pin_rel

Syntax

pin_rel [-override] event_file_name

Parameters

-override
This option starts a new pin_rel process. Use this option to restart pin_rel when it has abnormally stopped or you have stopped it manually.

---

**Important:** Use this option only when you know there are no other RE Loader processes running.

---

RE Loader maintains a status of its operations. Because only one RE Loader process can load events into the same database at the same time, the status must indicate the loading process is complete before another process can start loading. If you manually stop pin_rel, its status may not reflect its true state. The -override parameter overrides the status and permits a new loading process to start, providing one is not already running.

event_file_name
The name of the event file to load, including its extension.

Results

If pin_rel is successful, it returns PROCESSED SUCCESSFULLY in the RE Loader log file (BRM_home/apps/pin_rel/rel.pinlog).

If an error occurs during loading, this utility aborts the loading process. An error is logged in the rel.pinlog file, SQL loader errors are logged in a "bad" file (BRM_home/apps/pin_rel/CDR_file_name.bad), and the records loaded in this session are deleted from the database.

If an error occurs during account updating, the error is logged in the rel.pinlog file. Loaded records are not deleted from the database.