Oracle® Application Integration Architecture

Oracle Communications Billing and Revenue Management Integration Pack for Oracle E-Business Suite: Revenue Accounting Implementation Guide

Release 11.1

E50338-02

February 2015

ORACLE®

Oracle Application Integration Architecture Oracle Communications Billing and Revenue Management Integration Pack for Oracle E-Business Suite: Revenue Accounting Implementation Guide, Release 11.1

E50338-02

Contents

Preface

This document describes how to implement and use the Oracle Communications Billing and Revenue Management Integration Pack for Oracle E-Business Suite: Revenue Accounting.

Audience

This document is intended for customer service representatives, billing and pricing administrators, and other individuals who are responsible for configuring, managing and maintaining AIA Communications Pre-Built Integrations.

Downloading Oracle Documentation

Product documentation is located on Oracle Technology Network:

http://docs.oracle.com

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

http://edelivery.oracle.com

Documentation Accessibility

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

Access to Oracle Support

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

Related Documents

My Oracle Support Information Centers provide the most recent information about the following:

- Product guides
- Alerts
- Troubleshooting details

- FAQs
- Patches
- Community links

To see the Information Center for the Oracle Communications Billing and Revenue Management Integration Pack for Oracle Business Suite: Revenue Accounting, see My Oracle Support note 1392663.2 at:

http://support.oracle.com/epmos/faces/DocumentDisplay?id=1392663.2

For more information about Oracle AIA concepts, see *Oracle Fusion Middleware Concepts and Technologies Guide for Oracle Application Integration Architecture Foundation Pack*.

For more information about the installation, configuration, deployment, and upgrade processes, see *Oracle Application Integration Architecture Installation and Upgrade Guide for Pre-Built Integrations*.

1

Understanding the Process Integration for Revenue Management

This chapter provides an overview of the process integration for revenue management and the business process flow. It also discusses the assumptions and constraints for the process integration.

Overview of the Process Integration for Revenue Management

The process integration for revenue management consists of the Revenue Management business flow and is part of the Oracle Communications Billing and Revenue Management Integration Pack for Oracle E-Business Suite: Revenue Accounting.

This process integration lets you use Oracle General Ledger as an accounting engine on top of Oracle Communications Billing and Revenue Management (BRM) by moving general ledger (G/L) data from BRM to Oracle General Ledger.

This process integration differs from other Oracle Communications process integrations in that it does not use Enterprise Business Objects (EBOs) or other standard Oracle Application Integration Architecture (Oracle AIA) objects. Instead, the process integration for revenue management uses Oracle Data Integrator (ODI) to pick up BRM G/L data files and move them to Oracle General Ledger.

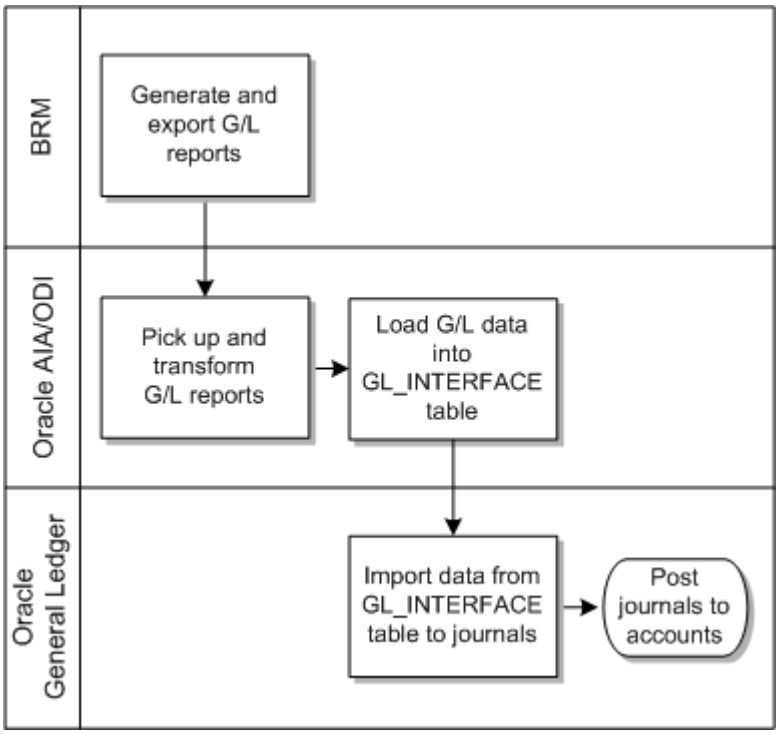
About the Revenue Management Business Process Flow

The Revenue Management business process flow moves G/L data from BRM to Oracle General Ledger as follows:

1. A BRM utility generates G/L reports. The reports are XML files that contain a complete and formatted G/L data set based on summary reports. You can run the utility manually, or schedule it to run automatically.
2. ODI picks up the reports and transforms them into G/L data according to mapping provided by AIA. AIA maps the G/L XML elements from BRM to the columns in the Oracle General Ledger GL_INTERFACE table. You can configure ODI to pick up the reports automatically when BRM generates them, or according to a schedule.
3. ODI loads the G/L data into the Oracle General Ledger GL_INTERFACE table. The GL_INTERFACE table is where Journal Import utility receives accounting data imported to Oracle General Ledger from other systems.
4. In Oracle General Ledger, you run the Journal Import utility. This utility validates the G/L data and converts it into journal entries.
5. In Oracle General Ledger, you post the journals to accounts.

Figure 1–1 illustrates the Revenue Management business process flow.

Figure 1–1 The Revenue Management Business Process Flow



Assumptions and Constraints

This section discusses the solution assumptions and constraints.

Solution Assumptions

The assumptions are:

- Oracle General Ledger accounts IDs are mapped and configured to BRM G/L accounts.
- Oracle recommends that all reports for a single cycle in the scope of one type of report data be processed before the next cycle.

  This action helps prevent picking up out-of-order reports.
- This design addresses revenue, receivables, and liabilities.
- If you need to use exact Oracle General Ledger account IDs, import the Oracle General Ledger chart of accounts into BRM. This implementation is outside the scope of the ready-to-use integration.
- In the event of a configuration error in the account mapping between BRM and Oracle General Ledger, manually reverse (*N* segments * 7 types of reports = M batches) and regenerate the feeds to Oracle General Ledger.

  N can be from tens to hundreds.
- To import nonmonetary journals into Oracle General Ledger (STAT currency), create and use a dummy account. This is because the extraction from BRM G/L

reports creates a debit and credit pair with the same amount in the debit and the credit.

Nonmonetary items increase or decrease in the debit and credit or the converse, depending on the setup, and Oracle General Ledger does not require balanced journals to create journal entries for nonmonetary transactions.

However, for this integration you must create a balanced journal entry. For example, if you are importing 100 free minutes into Oracle General Ledger, the Billed Earned report extracts a debit and a credit of 100 free minutes. In Oracle General Ledger, you must set up any Oracle General Ledger account for this; one account must not be used for reporting purposes because it does not provide the actual net amount of 100 free minutes. You create account 7100 for the debit and 7200 (dummy account) for the credit with a classification of expense accounts. The net effect is 100 minutes, but the transaction creates a DR 7100 for 100 and Credit 7200 for 100. The reports must exclude account 7200 to report the 100 minutes. If you are required to reduce the amount to 80 free minutes, the corresponding entry is DR 7200 for 20 and CR 7100 for 20.

- If the same report is loaded multiple times into the Oracle General Ledge GL_ INTERFACE table, it creates the additional accounting (double counting).

  To avoid double counting when reloading a report into Oracle General Ledger, first reverse the original entries created by the first report run and then reload the report.

- One instance of BRM can publish G/L data to one Oracle General Ledger instance.

  To target multiple SOB (with Oracle E-Business Suite R11.5.10 CU2) or Ledgers (with Oracle E-Business Suite R12.1.1) within a single Oracle General Ledger instance, configure the Oracle AIA configuration file per source system and ensure that BRM publishes the report with the appropriate source system information.

  You must specify one SOB (with Oracle E-Business Suite R11.5.10 CU2) or one Ledger ID (with Oracle E-Business Suite R12.1.1) for each source system.

- BRM-generated G/L reports are available in XML format.

- BRM report output values for the four types of revenue accounts within a given GLID are *Gross, Net, Discount*, and *Tax*.

  Choosing to use *Gross, Net*, or both is specific to the implementation. Implementations can also choose to ignore certain revenue account types even if valid G/L accounts are populated. Therefore, the implementation must list in the integration layer all of the account types that must be ignored so that the process does not produce an error condition for those account types.

- The time zone handling solution assumes that the BRM server and the Oracle E-Business Suite corporate time zone are set to the same value. The ODI flow that interfaces G/L data from BRM to Oracle General Ledger does not convert time zones.

Solution Constraints

This design assumes that the following statements are true:

- Multiple BRM instances can contribute to G/L data, but only one target Oracle General Ledger system can exist.

  Each XML file generated by any BRM instance can go to only one Oracle General Ledger system. To support multiple Oracle General Ledger systems, customize the integration so that each integration process instance targets a separate Oracle

General Ledger system. If you use multiple instances for each instance, add a Source SystemID in the configuration file.

- The BRM-generated XML file contains the Source SystemID from which it is generated.
- One BRM instance publishes data to one SOB Id (with Oracle E-Business Suite R11.5.10 CU2) or to one Ledger Id (with Oracle E-Business Suite R12.1.1).

  The same SOB Id or Ledger Id can be used by multiple BRM instances. Therefore, the relationship between a BRM instance and a SOB Id or Ledger Id is many to one.
- You must place the BRM-generated XML file in the defined integration folder.

  If any of the processed XML files fail, manually place those files in the process folder for reprocessing after making the necessary corrections.
- If the agent stops in the middle of processing of a file, and if that file is not in the success or the failure folder, then it is in the ODI InProcess folder.

  Move this file from the ODI InProcess folder back to the input folder for reprocessing.

2

Setting Up the Process Integration for Revenue Management

This chapter describes how to set up the process integration for revenue management and lists the prerequisites. It discusses how to perform setup tasks, configure and generate general ledger (G/L) data in Oracle Communications Billing and Revenue Management (BRM), and how to transform the G/L data.

Prerequisites

Complete these tasks before you set up the process integration for revenue management:

1. Install Oracle Data Integrator (ODI).
2. Install Oracle Communications BRM Integration Pack for Oracle E-Business Suite: Revenue Accounting (the Revenue Accounting pre-built integration).
3. Install BRM and complete the initial setup tasks. See *Oracle Communications Billing and Revenue Management Installation Guide* for more information about installation and initial setup.
4. Install Oracle E-Business Suite and Oracle General Ledger Release 11.5.10 CU2 or 12.1.1 and complete its initial setup tasks. See *Oracle General Ledger Implementation Guide* for more information about initial setup.
5. Install Oracle E-Business Suite General Ledger Release 11.5.10 CU2 Seed Patch 6012471 or Oracle E-Business Suite General Ledger Release 12.1.1 Seed Patch 8344349. If the patch is absent or not available, make the following changes manually in Oracle General Ledger:
   a. Add the Portal G/L Journal Source.
   b. Add the following journal categories:
      - Unbilled unearned
      - Unbilled earned
      - Billed unearned
      - Billed earned
      - Prior billed earned
      - Billed
      - Unbilled

Note: If you are applying Seed Patch to Oracle E-Business Suite General Ledger Release 12.1.1, manually change the journal category *Prior Billed earned* to *Prior billed earned* in Oracle General Ledger.

Journal categories are case-sensitive and must match the BRM configuration.

For more information about installing and setting up the Revenue Accounting pre-built integration and required software, see *Oracle Application Integration Architecture Installation and Upgrade Guide for Pre-Built Integrations*.

Setting Up Oracle General Ledger for the Process Integration for Revenue Management

To set up Oracle General Ledger to support the process integration for revenue management:

1. Define your set of books (SOB) with Oracle E-Business Suite.
2. Load the journal entry source seed data.

   You can also specify whether you want Oracle General Ledger to store journal reference information from BRM for a particular source.

   Seed data is provided using a patch. See "Prerequisites" for the patch number or manual steps if the seed patch is not available.
3. Load the journal entry categories seed data.

   Seed data is provided using a patch. See "Prerequisites" for the patch number or manual steps if the seed patch is not available.
4. If you want Journal Import to assign sequential numbers to your journal entries, enable sequential numbering, specifying **Automatic** as both your numbering method and document generation method.
5. Run the Optimizer program to create indexes for your account segments.
6. Define the concurrent program controls to improve the performance of Journal Import by setting the amount of disk space and memory it uses.

   The Journal Import program requires approximately 1.4 megabytes of memory to run. You can also specify whether to save your Journal Import data each time you run Journal Import. Journal Import runs faster if you do not archive your data.
7. Disable dynamic insertion. Journal Import runs much faster when it is not required to create new account combinations dynamically.
8. Set the status of the accounting period (for which the integration is running) to either **Open** or **Future Enterable**.

Setting Up the Process Integration for Revenue Management

To set up the process integration for revenue management:

1. Configure Oracle General Ledger and its Accounting Flexfield.
2. Map and configure Oracle General Ledger accounts into BRM.
3. Generate G/L data in BRM.

   Rerun reports if the system identifies errors. The system moves G/L data to the directory that you specified in BRM and ODI.

4. Use ODI to pick up the G/L data and load the Oracle General Ledger GL_INTERFACE table with the accounting transactions.

   Handle re-runs. If an error occurs, you must fix the source system and export the XML files again.

5. Import the journals into Oracle General Ledger.

   Manually or automatically import the reports into journal entries.

   Optionally, you can delete the interface data if the system identifies an error before importing the interface data.

   Use Oracle General Ledger to review and post or delete. With deletion, the system deletes the data from both the interface tables and the journals.

6. Post journals to Oracle General Ledger accounts.

   Manually or automatically post the imported journal entries into the corresponding Oracle General Ledger accounts.

   Optionally, delete the journal entries if the system identifies an error after the batch is imported but before it is posted.

   Optionally, reverse the journal if the system identifies an error after posting.

7. Reconcile or review journal entries.

   Review journal entries and, if errors occurred, create the appropriate adjustment entries.

Setting Up BRM for the Process Integration for Revenue Management

To set up BRM to support the process integration for revenue management:

1. Configure the Oracle General Ledger export configuration file.

   In addition to the standard configuration parameters, you must also specify the Source System Identifier, which is the value of the SOB Id (with Oracle E-Business Suite R11.5.10 CU2) or Ledger Id (with Oracle E-Business Suite R12.1.1) mapping in the integration layer configuration file.

2. Customize the policy opcode PCM_OP_GL_POL_EXPORT_GL to map the BRM G/L account names to the appropriate Oracle General Ledger accounts.

   Mapping is achieved by modifying the relevant fields in the generated ledger report object. The Oracle General Ledger accounts must be the concatenation of the individual Oracle General Ledger segment values separated by the *V* segment separator string. This segment separator string must be the same as the one configured in the integration layer. For example, the system can map an internal BRM G/L account name such as west.california to an Oracle General Ledger account XXX/YYY/ZZZ if the three Oracle General Ledger segments have values of XXX, YYY, ZZZ, and the segment separator.

   Note: Customizing the opcode is only required if the BRM G/L account names are different from the Oracle General Ledger account names (including the segment separator). If the BRM and Oracle General Ledger account names are the same, then you are not required to transform the names using the policy opcode. Also, the segment separator string is not required to be *V*, but both the policy opcode and the integration layer configuration must use the same separator string.

See *Oracle Communications Billing and Revenue Management Collecting General Ledger Data* for more information.

Setting Up the Integration Layer

For the integration layer, you must configure the parameters in the **AIAConfigurationProperties.xml** file, including the mapping of the Source System Identifier to the SOB Id (with Oracle E-Business Suite R11.5.10 CU2) or Ledger Id (with Oracle E-Business Suite R12.1.1).

By just using the Gross, Net, or certain revenue account types when the Oracle General Ledger account configuration contains **--N/A--** for the output values, list the account types that the integration layer must ignore in order not to get an error.

For more information about configuring ODI, see the *Oracle Fusion Middleware Installation and Upgrade Guide for Oracle Application Integration Architecture Foundation Pack*.

Configuring and Generating G/L Data in BRM

This section discusses these topics:

Configuring Data in BRM

Before running any G/L data, you must set up BRM to:

- List all BRM G/L accounts.
- Create G/L IDs that assign G/L codes to BRM balance impacts.
- Configure the summarization of BRM G/L accounts into the Oracle General Ledger accounts.

  You can map BRM revenue into Oracle General Ledger accounts in two ways:

  - Configure the BRM G/L accounts independently of the Oracle General Ledger accounts, and map the Oracle General Ledger accounts within the policy opcode.
  - Configure the Oracle General Ledger accounts within BRM G/L configuration files.

If you cannot configure the exact name of an Oracle General Ledger account, then you must customize BRM to configure the proper account ID. No pattern for configuring the accounts is specified, but typically BRM G/L reporting is more detailed than Oracle General Ledger, where accumulation is done.

Each G/L segment contains its own set of G/L IDs (and accounts). The balance impacts always use the G/L IDs that are dependent on the product rated against, and they are independent of the G/L segment with which the account is associated. The segments do not necessarily match the Oracle General Ledger segments, but they should to simplify the integration.

For more information, see *Oracle Communications Billing and Revenue Management Collecting General Ledger Data*.

Generating Data in BRM

BRM can generate summary and detailed reports. You can use the summary report for this integration to Oracle General Ledger. You can start with detailed report data and customize it to summarize the data, but what is handed to the integration is always a summary report.

You can pick a segment for reporting or, by default, all configured segments are reported.

In BRM, you must configure the G/L account IDs to correspond with the target accounts in Oracle General Ledger.

For more information, see the discussions of loading general ledger configuration data and exporting general ledger reports to XML files in *Oracle Communications Billing and Revenue Management Collecting General Ledger Data*.

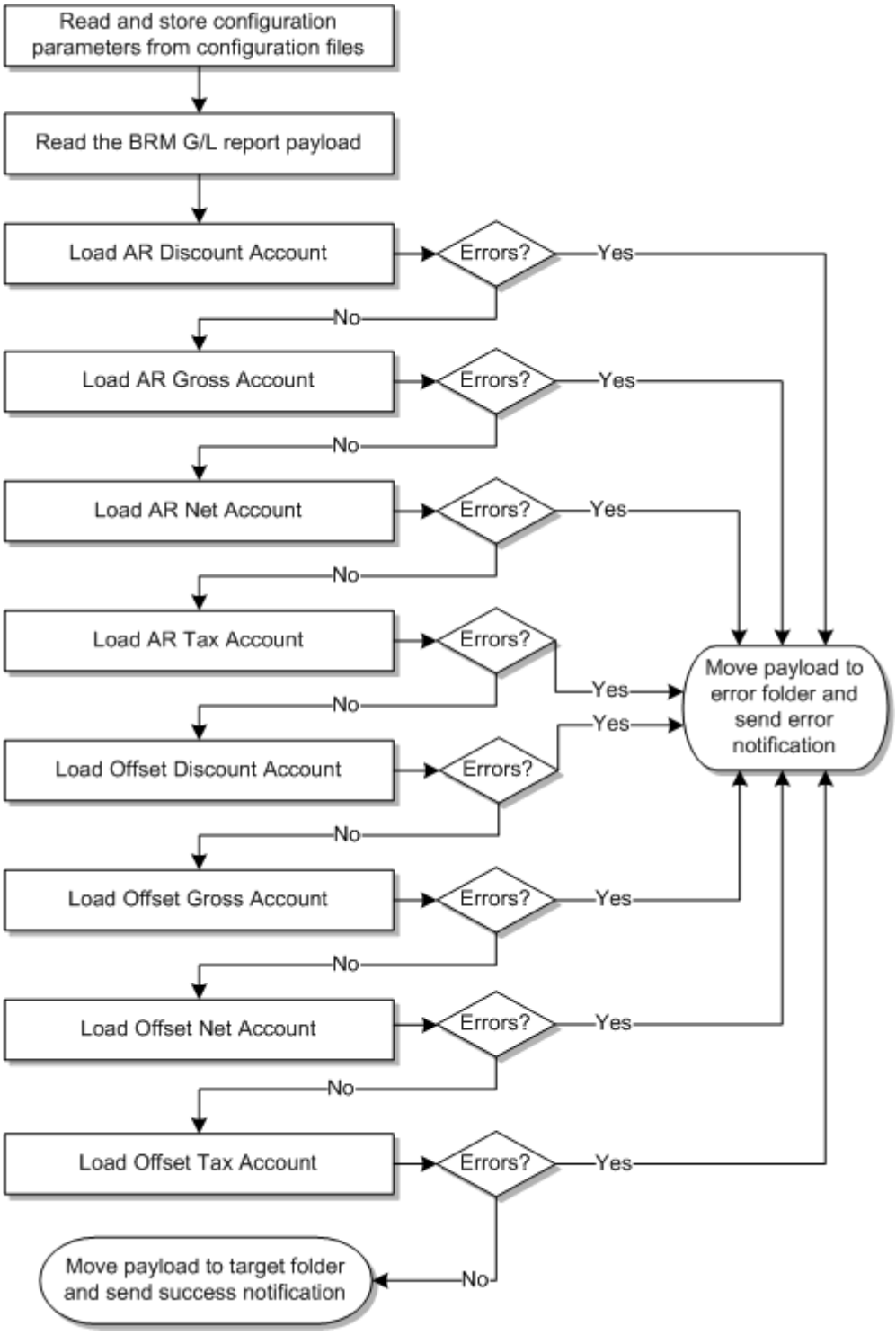
Picking Up and Transforming the G/L Data

The process integration for revenue management uses ODI to pick up the BRM data files, transform them from XML, and load the Oracle General Ledger GL_INTERFACE table. After the interface table is loaded, you use Oracle General Ledger to import and post the journals.

In BRM, you configure the OutputDirectory to be the same directory that you specified in the **AIAConfigurationProperties.xml** file. After generating a BRM G/L data file, when the file enters that directory, ODI picks it up and transforms it to the Oracle General Ledger interface table.

Figure 2–1 illustrates the process of loading the data to the GL_INTERFACE table.

Figure 2–1 ODI Data Loading



When this business process initiates, ODI does the following:

1. Reads and stores configuration parameters from the configuration files.
2. Reads the BRM XML file.

3. Checks for the value of the property TargetID from the **AIAConfigurationProperties.xml** file and executes the mapping for either Oracle E-Business Suite Release 11.5.10 CU2 or 12.1.1.
4. Checks to see if the AR discount account element is configured to be filtered out.

   If it is, then the system moves to the next element. Otherwise, it loads the AR discount account element.

   If the load is successful, the system moves to the next element.

   If the load fails, the system moves the BRM XML file to the error folder, stops processing, and sends an email notification that the load failed.
5. Checks to see if the AR gross account element is configured to be filtered out.

   If it is, then the system moves to the next element. Otherwise, it loads the AR gross account element.

   If the load is successful, the system moves to the next element.

   If the load fails, the system moves the BRM XML file to the error folder, stops processing, and sends an email notification that the load failed.
6. Checks to see if the AR net account element is configured to be filtered out.

   If it is, then the system moves to the next element. Otherwise, it loads the AR net account element.

   If the load is successful, the system moves to the next element.

   If the load fails, the system moves the BRM XML file to the error folder, stops processing, and sends an email notification that the load failed.
7. Checks to see if the AR tax account element is configured to be filtered out.

   If it is, then the system moves to the next element. Otherwise, it loads the AR tax account element.

   If the load is successful, the system moves to the next element.

   If the load fails, the system moves the BRM XML file to the error folder, stops processing, and sends an email notification that the load failed.
8. Checks to see if the offset discount account element is configured to be filtered out.

   If it is, then the system moves to the next element. Otherwise, it loads the offset discount account element.

   If the load is successful, the system moves to the next element.

   If the load fails, the system moves the BRM XML file to the error folder, stops processing, and sends an email notification that the load failed.
9. Checks to see if the offset gross account element is configured to be filtered out.

   If it is, then the system moves to the next element. Otherwise, it loads the offset gross account element.

   If the load is successful, the system moves to the next element.

   If the load fails, the system moves the BRM XML file to the error folder, stops processing, and sends an email notification that the load failed.
10. Checks to see if the offset net account element is configured to be filtered out.

    If it is, then the system moves to the next element. Otherwise, it loads the offset net account element.

If the load is successful, the system moves to the next element.

If the load fails, the system moves the BRM XML file to the error folder, stops processing, and sends an email notification that the load failed.

11. Checks to see if the offset tax account element is configured to be filtered out.

    If it is, then the system moves to the next element. Otherwise, it loads the offset tax account element.

    If the load is successful, the system moves to the final step.

    If the load fails, the system moves the BRM XML file to the error folder, stops processing, and sends an email notification that the load failed.

12. Finally, the system moves the successful payload to the target folder and sends an email notification that the load is successful.

3

Configuring and Running the Process Integration for Revenue Management

This chapter discusses how to change the database schema name of Oracle General Ledger, set up a schedule, and configure and run the process integration for revenue management.

Changing the Oracle General Ledger Database Schema Name

The standard Oracle Application Integration Architecture (Oracle AIA) installation process assumes the Oracle General Ledger database schema name is the same as the database user name. If not, then you must run the following steps post installation.

To change the Oracle General Ledger database schema name:

1. Open a command prompt.
2. Source the file **aiaenv.sh** (if UNIX) or **aiaenv.bat** (if Windows) located in *AIA_home*/**aia_instances/AIAHOME/bin**.
3. Navigate to the directory *ODI_home*/**oracledi/client**.
4. Open Topology Manager as follows:
   a. Run one of the following commands:
      – **UNIX:** `./odi.sh`
      – **Windows:** `odi.exe`

      Design Studio for Oracle Data Integrator (ODI) appears.
   b. Click **Connect to Repository**.

      The ODI dialog box appears.
   c. From the **Login Name** list, select **Repository**.
   d. In the **User** field, enter your username.
   e. In the **Password** field, enter your password.
   f. Click **OK**.

      The Topology Manager appears.
5. Click the **Topology** tab.
6. On the Physical Architecture bar, expand the **Technologies** folder, then expand the **Oracle** folder.

7. If the version of Oracle General Ledger is 11.5.10, expand the **Oracle GL 11 DS** folder. (If the version is 12.1.1, skip to step 12.)
8. Right-click the **Oracle GL 11 DS.apps** child element and click **Open**.

   The Physical Schema screen appears.
9. Change the values in the **Schema (Schema)** and **Schema (Work Schema)** fields to your G/L database schema name.

   Note: Both the **Schema (Schema)** and the **Schema (Work Schema)** fields must contain the same value.
10. Click **Apply**.
11. Click **OK**.
12. If the version of Oracle General Ledger is 12.1.1, expand the **Oracle GL Interface DS** folder, right-click the **Oracle GL Interface DS.GL** child element, and click **Open**.
13. In the Physical Schema window, change the values of **Schema (Schema)** and **Schema (Work Schema)** to the actual G/L database schema name.

    Note: Both the **Schema (Schema)** and the **Schema (Work Schema)** fields must contain the same value.
14. Click **Apply**.
15. Click **OK**.
16. Regenerate the scenario LOAD_PORTAL_DATA_TO_ORACLE_GL_PKG Version 001 from the **Designer** tab.

Setting Up a Schedule

The standard Oracle AIA installation process completes the standard install for the process integration for revenue management. After the standard install, you must complete the following steps.

For more information about the standard installation process, see *Oracle Application Integration Architecture Installation and Upgrade Guide for Pre-Built Integrations*.

1. Open the designer by running the following command in the **oracledi/client** directory:
   - **UNIX:** `./odi.sh`
   - **Windows:** `odi.exe`

   The ODI Login window appears.
2. Enter the appropriate credential to log in to Design Studio. If you have not set up the login, click the **New** button to create a login.

   The Work Repository Connection window appears.
3. Enter the following values:
   - **Login Name**: The name by which you identify this integration at login. For example, Revenue Management.

- **User**: User name of the Designer, for example, SUPERVISOR.
- **Password**: Password for the user, for example, SUNOPSIS.
- **User**: The master repository user, for example, odimaster.
- **Password**: The password for the master repository user, for example, odimaster.
- **Driver List**: Select **Oracle JDBC Driver**.
- **Driver Name**: Enter **oracle.jdbc.driver.OracleDriver**.
- **Url**: Enter **jdbc:oracle:thin:@***host***:***port***:***sid* where *host*, *port*, and *sid* represent the database environment where you created the master repository.
- **Work Repository**: Click the button to the right to display a list of available work repositories.

  The system displays only one work repository. The repository name is the one that you chose during the standard installation.
- **Default Connection**: Select this check box if you have multiple login name.s

4. Click **OK**.

   If the Work Repository Connection asks for a password, enter the Work Repository database user password that you set during the standard installation.

   The Designer window appears.
5. Expand the **Portal to Oracle GL Project** tree to **Portal To Oracle GL Project, Packages, Load Portal Data To Oracle GL Pkg, Scenarios, LOAD PORTAL DATA TO ORACLE _GL_PKG Version 001, Scheduling**.
6. Double-click the existing **RevMgmtAgent** scheduler.

   The Scenario Scheduling window appears.
7. Enter or select the following values:

   Context: Select **Portal to GL Context**.

   Agent: Select **RevMgmtAgent**.

   Log Level: **5**.

   Status: Select the **Active** option.

   Execution: Select the **On startup** option.
8. Click the **Execution Cycle** tab.
9. Enter or select the following values:

   Repetition: Select **Many** times.

   Interval between Repetitions: Enter **30** and choose **Second(s)**.
10. Click **OK**.

    The scheduler is now configured.

Configuring the Process Integration for Revenue Management

After you install the process integration for revenue management, you must configure several parameters. Open the **AIAConfigurationProperties.xml** file that resides in the

aia_instances/**config/** folder. Entries in the **AIAConfigurationProperties.xml** file are case sensitive.

See the discussions of building AIA integration flows and how to set up AIA workstations in *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack* for more information.

Configure the following parameters:

1. Under the moduleName RevMgmtSetOfBooksIDs find the **<Property Name=" Source SystemID ">202</Property>** property tag.

   - Replace the tag value **SourceSystemID** with the value of the SourceSystemID that comes in the in BRM XML GL report. For example, if you have two BRM instances feeding into Oracle General Ledger and the report from one instance has the SourceSystemID value as **abc** and the report from the other instance has the value as **def**, then change the section as follows:

     ```
     <ModuleConfiguration moduleName="RevMgmtSetOfBooksIDs">
       <Property name="abc">101</Property>
       <Property name="def">202</Property>
       ......
     </ModuleConfiguration>
     ```

     You must ensure that the value of the **SourceSystemID** in the BRM report is a value in the **AIAConfigurationProperties.xml** file.

   - Replace the value of the **SourceSystemID**, such as 202, with the SOB ID as defined in E-Business Suite R11.5.10 CU2 or with the Ledger ID defined in E-Business Suite 12.1.1.

2. Under the moduleName RevMgmtExchangeRateType find the **<Property name="ExchangeRateType">DAILY</Property>** property tag.

   The attribute DAILY represents the Exchange Rate Type. Do not change the name attribute value. Replace DAILY with the required exchange rate type.

   The Exchange Rate Type handles currency conversions. Oracle provides a placeholder for the Exchange Rate Type. The Exchange Rate Type is passed to the Oracle GL application and is handled by Oracle General Ledger when loading G/L report data from the interface table to Oracle General Ledger.

3. Under the moduleName RevMgmtParameters are several property tags with Name and Value attributes.

   Do not edit the **Name** attribute value.

   Table 3–1 lists value attributes.

Table 3–1 Value Attributes

Name Attribute Value	Property Tag Value
TargetID	Specify whether the target application is E-Business Suite Release 11.5.10 CU2 or 12.1.1. This property contains the value *EBS_01* for 12.1.1 and *EBS_02* for 11.5.10 CU2. The default value is *EBS_01*. The values entered are case sensitive.
ToMailAddress	Specify the To email address of the person to whom an email is sent, notifying them that the integration process is complete.

Table 3–1 (Cont.) Value Attributes

Name Attribute Value	Property Tag Value
FromMailAddress	Specify the From email address for the person who performs the integration process.
TargetFileLocation	Specify the folder name where you want the process to move the XML file after it is successfully loaded.
ErrorFileTargetLocation	Specify the folder name where you want the process to move the XML file if a failure occurs.
SegmentSeparator	Specify how the BRM XML document separates in representing the Oracle GL segments.
PayloadSourceLocation	Specify the folder name to which the BRM-generated XML files are moved. The staging/ODIInProcess folder cannot be used as the PayloadSourceLocation. See "Solution Constraints" for more information about constraints on where you can place BRM-generated XML files.
Mail Server	Specify the mail server name.
RevMgmtHome	Specify the location of the RevenueManagementHome folder. Normally, this folder is located at <AIAHome>/DISHome/RevenueManagementHome .

4. Under the moduleName RevMgmtAccountFiltering specify what types of accounts you want to filter out (that is, to not load to the Oracle General Ledger interface table), as shown in Table 3–2.

   Possible values are **Yes** and **No**, and they are case sensitive.

   These parameters enable you to filter certain account types. If the value is **No**, then they are not filtered out (they are loaded). If the parameter value is **Yes**, then the corresponding revenue account type entries are filtered out (they are not loaded).

Table 3–2 Value Attributes: Filtering Account Types

Name Attribute Value	Property Tag Value
FilterARGrossAccount	Specify **No** to load ARGrossAccount entries. Specify **Yes** to filter out this account.
FilterARDiscountAccount	Specify **No** to load FilterARDiscountAccount entries. Specify **Yes** to filter out this account.
FilterARNetAccount	Specify **No** to load ARNetAccount entries. Specify **Yes** to filter out this account.
FilterARTaxAccount	Specify **No** to load ARTaxAccount entries. Specify **Yes** to filter out this account.
FilterOffsetGrossAccount	Specify **No** to load OffsetGrossAccount entries. Specify **Yes** to filter out this account.
FilterOffsetDiscountAccount	Specify **No** to load OffsetDiscountAccount entries. Specify **Yes** to filter out this account.

Table 3–2 (Cont.) Value Attributes: Filtering Account Types

Name Attribute Value	Property Tag Value
FilterOffsetNetAccount	Specify **No** to load OffsetNetAccount entries. Specify **Yes** to filter out this account.
FilterOffsetTaxAccount	Specify **No** to load OffsetTaxAccount entries. Specify **Yes** to filter out this account.

Whenever you change the AIAConfigurationProperties.xml file, you must restart the agent to reflect your changes.

Running the Process Integration for Revenue Management

You can schedule the integration between BRM and Oracle General Ledger to occur automatically. If the scheduler agent is kept running, then whenever BRM generates the report in XML format and writes the XML document into the specified directory, the integration process picks up the XML file, transforms it, and then transfers it to the interface table of Oracle General Ledger.

Note: You can start or stop the scheduler agent whenever you want, which enables you to control the integration process. Therefore, to generate the reports in BRM in the last week of every month, then you can start the scheduler agent at the beginning of the last week and stop it when the week ends.

See "Setting Up a Schedule" for more information.

Starting the Scheduler Agent

A scheduler agent, RevMgmtAgent, is delivered and is running on default port 20910.

To start the schedule agent:

1. Navigate to the **oracledi/agent/bin** directory.
2. Run one of the following commands:
   - **UNIX:** `./agent.sh -NAME=RevMgmtAgent`
   - **Windows:** `agent.bat -NAME=RevMgmtAgent`

This command starts the scheduler agent, which in turn launches a scenario. The scenario waits for a BRM XML report file to arrive in the specified directory. As soon as the report file arrives, the scheduler agent transfers the file to the Oracle General Ledger interface table. You can monitor the status of this integration process using the ODI Operator.

Stopping the Scheduler Agent

You must stop the schedule agent to stop transferring BRM XML reports to the Oracle General Ledger interface table.

To stop the scheduler agent, run one of the following commands:

- **UNIX:** `./agentstop.sh`
- **Windows:** `agentstop.bat`

A

BRM GL Configuration File

This appendix provides a sample Oracle Communications Billing and Revenue Management (BRM) general ledger (G/L) configuration file.

Example A–1 is a sample of the BRM G/L configuration file:

Example A–1 Sample BRM G/L configuration file

```
#
#"@(#)%Portal  Version:   pin_glid:
%";
#
General Ledger (G/L) ID Definitions
#
#    Copyright (c) 1999 - 2006 Oracle. All rights reserved.
#
#    This material is the confidential property of Oracle orporation
#    or its licensors and may be used, reproduced, stored or
#    transmitted only in accordance with a valid Oracle license or
#    sublicense agreement.
#
# Use this file to define G/L IDs that correspond to your accounting
# methods.
#
# After defining the G/L IDs, run the 'load_pin_glid' utility to
# create the new config/glid storable object and load the G/L IDs
# into the Infranet database.
#
# You can have multiple pin_glid files, each defining the G/L IDs
# for some segments of your customer base. Use the "gl_segment"
# entry in this file to specify the segments for which these G/L IDs
# apply.
#
# For more information, see "About Collecting General Ledger
# Information" in the Infranet online documentation.
# gl_segment
#
# Name of the G/L segment described by this configuration file.
# You can define multiple GL segments in this file. All the gl
# segments that are defined shall have the same glids. You have to
# specify at least one gl_segment in this file
#
# You may have different glid files and each glid file may
# have multiple segments. You must ensure that the gl segment
# names specified in different glid files do not conflict. If this
# happens, the system shall overwrite the gl segment with the data
# of the last loaded glid file
```

```
#
# The value you specify should match the value in the
# PIN_FLD_GL_SEGMENT in the account objects for which you use
# this set of G/L IDs.
#
# GLIDs are assigned to accounts through the cm pin.conf parameter
# gl_segment and can be overridden by the policy opcode
# pcm_op_pol_cust_actginfo
# Syntax for G/L Segment Definitions
#
Use this syntax to define a gl_segment:
#
#gl_segment <segment_name> [no_roll-up]
#
#
# segment_name
#
# You specify the segment name. The segment name follows a URL-like
# notation and enables you to nest G/L segments. There is a root
# segment in Infranet and it is named ".".  If your system has
# additional child segments then you define them as <root
# segment>.<child_segment>. You can have unlimited level of nesting.
Segment names are case sensitive and cannot have spaces.
#
# The segment name . is used to identify the root segment. It is
# reserved and you should not change it
#
# no_roll-up
#
# The no_roll-up is used to indicate whether you want the data for
# the gl segment to roll up to its parent. If you do not want the
# data of this gl_segment to roll up to its parent then you must
# specify no_roll-up. If you want the roll-up then you must leave it
# blank. The root segment (.) does not roll up even if you
# specify blank. This is a special case.
#
An example is shown below :
#
# gl_segment .
# gl_segment .A   (Child segment A, rolls up to . )
# gl_segment .A.A1   (Child segment of A, rolls up to A and to .)
# gl_segment .A.A2  no_roll-up (Child segment of A, does not roll up # to A )
# gl_segment .B    no_roll-up (Child segment of . but does not
# roll up)
# gl_segment .C   (Child segment of . rolls up to . )
#
# Based on the previous example, if you prepare the ledger report for
#
#.  The data from .A, .A.A1 and .C is included
#.A The data of .A and .A.A1 is included (.A.A2 is not included)
#.A.A1  The data of .A.A1 is included (no childs of .A.A1 exist)
#.A.A2    The data of .A.A2 is included (no childs of .A.A2 exist)
#.B    The data of .B is included (no childs of .B exist)
#.C    The data of .C is included (no childs of .C exist)
#
# The default file has only the root segment specified in it

gl_segment .
# COA_ID
#
```

```
# Name of the Chart of Accounts (COA) ID that shall be used to
# validate the gl accounts specified under a GL ID
#
# This is an optional parameter. If you want the glids to be
# validated against a COA_ID then you must specify this parameter.
# If you do not want validation against a COA then leave it blank or
# do not specify the
# parameter.
# If specify this parameter and do not specify any value then it is
# as good as not specifying the parameter.
#
# If a COA_ID value is provided then the program checks if it has
# been created. Refer to the Infranet documentation on how
# to create COA's.
# Hence you must load the COA before loading the pin_glid files
#
# The default is no COA_ID
#
# You can specify only one COA_ID in one pin_glid file.
# The same COA_ID can be used in multiple pin_glid files if
# required.
# COA_ID


# Syntax  for G/L  ID  Definitions
#
# Use this   syntax  to  define  a G/L  ID:
#
#glid   (
# id     <uint>
# descr     <str>
# taxcode     <str>
# gl_acct     [type]   [attribute]   [debit  acct]   [credit  acct]
#)
#
# The following sections describe each of these fields.
# id
#
# An identification number you assign to a G/L ID.
#
# This ID can then be referenced by rates.
#
# GLID values has to be assigned according to the following rules:
# 1. GLID from 0 to 1000000 - are reserved for Portal use.
Use values greater than 1000000 when defining your own G/L IDs. The data type must
be "unit".
# 2. GLID 0 - is journalized, but it is excluded from GL reports.
#  This value is considered as undefined and should not be present
#  in production system.
# 3. GLID from 1 to 99 - are excluded from journalizing and GL
#    reports.
# 4. GLID from 100 - are journalized and included in GL reports.
#
# The combined length of the "id" field and the "descr" field must
# be less than 255 characters.
# descr
#
A short description of the G/L ID.
#
# This description is displayed on the Pricing Tool menus.
```

```
#
# The combined length of the "id" field and the "descr" field must
# be less than 255 characters.
taxcode
#
# Type of product being purchased.
#
# This value must match the tax code used for the rates that use
# this G/L ID.
gl_acct
#
# An array that describes a pair of G/L accounts. Each G/L ID can
# have multiple gl_acct arrays.  Each array contains the following
# fields:
#
# ------------------------------------------------------------------
# type - The conditions related to the billing and accounting
#          cycles under which this pair of G/L accounts should be
#          used.  There are seven valid types:
#
#          1. billed             5. unbilled_earned
#          2. unbilled           6. unbilled_unearned
#          3. billed_earned      7. prev_billed_earned
#          4. billed_unearned
#
#          For more information, see "About G/L account pair types"
#          in the Infranet online documentation.
# ------------------------------------------------------------------
#attribute - The conditions related to what the amount of the
#              balance impact represents under which this
#              pair of G/L accounts is necessary.  There are four
#              values:
#
#              1. gross         3. net
#              2. disc          4. tax
#
#              For more information, see "About G/L account
#              attributes" in the Infranet online documentation.
#
#debit acct - The name of one Infranet G/L account that
#            is affected by this G/L ID.  For example, an A/R
#            account appears here for G/L IDs that refer to
#            fees charged to the end user.  The impact on this
#            account is the opposite of the impact on the
#            credit acct.
#
# credit acct - The name of one Infranet G/L account that
#               is affected by this G/L ID.  For example, an
#               revenue account appears here for G/L IDs that
#               refer to fees charged to the end user. The impact
#               on this account is the opposite of the impact
#               on the debit acct.
#
#Validation between the gl account and chart of account type.
#
#
#For a debit general ledger account
#
#gl attribute    |    Chart Of Account Type
#
```

```
#                |
#  Gross                    |    Asset
#  Net                      |    Asset
#  Tax                      |    Asset
#  Disc                     |    Expense or Revenue
#
#
#For a credit general ledger account
#
#gl attribute    |    Chart Of Account Type
#
#                 |
#  Gross                |    Expense or Revenue
#  Net                  |    Expense or Revenue
#  Tax                  |    Liability
#  Disc                 |    Asset
#
#
#Default G/L ID
glid ( id
0
descr undefined
gl_acct unbilled net undef.debit  undef.credit
gl_acct billed net  undef.debit undef.credit
# G/L ID for purchase rate
glid ( id
 101
 descr Purchase Fees
 gl_acct billed  gross purchase.debit purchase.credit gl_acct
 billed net purchase.debit purchase.credit gl_acct billed disc
 purchase.credit purchase.debit gl_acct billed_earned gross purchase.debit
purchase.credit gl_acct billed_earned net purchase.debit purchase.credit gl_acct
billed_earned disc purchase.credit purchase.debit gl_acct unbilled  gross
purchase.debit purchase.credit gl_acct unbilled  net purchase.debit
purchase.credit gl_acct unbilled  disc purchase.credit purchase.debit gl_acct
unbilled_earned gross purchase.debit purchase.credit gl_acct unbilled_earned net
purchase.debit purchase.credit gl_acct unbilled_earned disc purchase.credit
purchase.debit
# G/L ID for cycle_forward rate
glid ( id
 102
 descr Monthly Fees
 gl_acct billed  gross monthly.debit monthly.credit gl_acct
 billed  net monthly.debit monthly.credit gl_acct billed  disc
 monthly.credit monthly.debit gl_acct billed_earned gross
 monthly.debit monthly.credit gl_acct billed_earned net
 monthly.debit monthly.credit gl_acct billed_earned disc
 monthly.credit monthly.debit gl_acct prev_billed_earned gross
 monthly.debit monthly.credit gl_acct prev_billed_earned net
 monthly.debit monthly.credit gl_acct prev_billed_earned disc
 monthly.credit monthly.debit gl_acct billed_unearned gross
 monthly.debit monthly.credit gl_acct billed_unearned net
 monthly.debit monthly.credit gl_acct billed_unearned disc
 monthly.credit monthly.debit gl_acct unbilled  gross
 monthly.debit
 monthly.credit gl_acct unbilled  net
 monthly.debit monthly.credit gl_acct unbilled  disc
 monthly.credit monthly.debit gl_acct unbilled_earned gross
 monthly.debit monthly.credit gl_acct unbilled_earned net
 monthly.debit monthly.credit gl_acct unbilled_earned disc
```

```
 monthly.credit monthly.debit gl_acct unbilled_unearned gross
 monthly.debit monthly.credit gl_acct unbilled_unearned net
 monthly.debit monthly.credit gl_acct unbilled_unearned disc
 monthly.credit monthly.debit
# G/L ID for cancel rate
glid ( id
 103
 descr Cancel Fees gl_acct billed  gross cancel.debit
 cancel.credit
 gl_acct billed  net cancel.debit cancel.credit gl_acct billed
 disc cancel.credit cancel.debit gl_acct billed_earned gross
 cancel.debit cancel.credit gl_acct billed_earned net
 cancel.debit cancel.credit gl_acct billed_earned disc
 cancel.credit cancel.debit gl_acct unbilled  gross
 cancel.debit cancel.credit gl_acct unbilled  net
 cancel.debit cancel.credit gl_acct unbilled  disc
 cancel.credit cancel.debit gl_acct unbilled_earned gross
 cancel.debit cancel.credit gl_acct unbilled_earned net
 cancel.debit cancel.credit gl_acct unbilled_earned disc
 cancel.credit cancel.debit
# G/L ID for usage rate
glid ( id
 104
 descr Dialup Usage Fee
 gl_acct billed  gross dialup.debit dialup.credit gl_acct
 billed  net dialup.debit dialup.credit gl_acct billed  disc
 dialup.credit dialup.debit gl_acct billed_earned gross
 dialup.debit dialup.credit gl_acct billed_earned net
 dialup.debit dialup.credit  gl_acct billed_earned disc
 dialup.credit dialup.debit gl_acct  unbilled  gross
 dialup.debit dialup.credit gl_acct unbilled  net
 dialup.debit dialup.credit gl_acct unbilled  disc
 dialup.credit dialup.debit gl_acct unbilled_earned gross
 dialup.debit dialup.credit gl_acct unbilled_earned net
 dialup.debit
 dialup.credit gl_acct unbilled_earned disc
 dialup.credit dialup.debit

# Your custom G/L IDs below.  Start the G/L ID at 1000000.  You may
# use the G/L IDs defined previously or delete them and create your
# own. However, if you create your own misc G/L, payment,
# adjustment, dispute/settlement and writeoff G/L ID, you must also
# tell Infranet to assign these events with your newly defined G/L
# IDs by using the reasons codes file or modifying
# fm_act_pol_spec_glid.c.
```

B

BRM XML G/L Report

This appendix provides a sample Oracle Communications Billing and Revenue Management (BRM) general ledger (G/L) report.

Example B–1 is a sample of the BRM XML G/L report:

Example B–1 Sample BRM G/L report

```
-<GeneralLedgerReport targetNamespace="http://www.portal.com/schemas/GLSync"
xmlns="http://www.portal.com/schemas/GLSync"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.portal.com/schemas/GLSync brm_gl_data.xsd">
   <SourceSystemID>202</SourceSystemID>
   <ReportID>0.0.0.1-73988-6</ReportID>
   <RevenueType>Unbilled earned</RevenueType>
   <BRM_GL_Segment>.</BRM_GL_Segment>
-<ReportCreatedTime>
   <Year>2007</Year>
   <Month>01</Month>
   <Day>02</Day>
   <Hours>4</Hours>
   <Minutes>14</Minutes>
   <Seconds>51</Seconds>
   </ReportCreatedTime>
-<PeriodStartTime>
   <Year>2007</Year>
   <Month>01</Month>
   <Day>04</Day>
   <Hours>0</Hours>
   <Minutes>0</Minutes>
   <Seconds>0</Seconds>
   </PeriodStartTime>
-<PeriodEndTime>
   <Year>2007</Year>
   <Month>01</Month>
   <Day>05</Day>
   <Hours>0</Hours>
   <Minutes>0</Minutes>
   <Seconds>0</Seconds>
   </PeriodEndTime>
-<RevenueAmounts element="0">
   <ResourceId>84 0</ResourceId>
   <BRM_GL_ID>102</BRM_GL_ID>
-<ARGrossAccount name="monthly.debit">
   <Credit>0</Credit>
   <Debit>0.32</Debit> </ARGrossAccount>
-<ARDiscountAccount name="monthly.credit">
```

```
   <Credit>0</Credit>
   <Debit>0</Debit>
   </ARDi scountAccount>
-<ARNetAccount name="monthly.debit">
   <Credit>0</Credit>
   <Debit>0.32</Debit>
   </ARNetAccount>
-<ARTaxAccount name="-- N/A --">
   <Credit>0</Credit>
   <Debit>0</Debit>
   </ARTaxAccount >
-<OffsetGrossAccount name="monthly.credit">
   <Credit>0.32</Credit>
   <Debit>0</Debit>
   </OffsetGrossAccount>
-<OffsetDiscountAccount name="monthly.debit">
   <Credit>0</Credit>
   <Debit>0</Debit>
   </OffsetDiscountAccount>
-<OffsetNetAccount name="monthly.credit">
   <Credit>0.32</Credit>
   <Debit>0</Debit>
   </OffsetNetAccount>
-<OffsetTaxAccount name="-- N/A --">
   <Credit>0</Credit>
   <Debit>0</Debit>
   </OffsetTaxAccount>
   </RevenueAmounts>
   <RevenueAmounts element="1">
   <ResourceId>84 0</ResourceId>
   <BRM_GL_ID>104</BRM_GL_ID>
-<ARGrossAccount name="dialup.debit">
   <Credit>0</Credit>
   <Debit>0.10</Debit>
   </ARGrossAccount>
-<ARDiscountAccount name="dialup.credit">
   <Credit>0</Credit>
   <Debit>0</Debit>
   </ARDi scountAccount>
-<ARNetAccount name="dialup.debit">
   <Credit>0</Credit>
   <Debit>0.10</Debit>
   </ARNetAccount>
-<ARTaxAccount name="-- N/A --">
   <Credit>0</Credit>
   <Debit>0</Debit>
   </ARTaxAccount >
-<OffsetGrossAccount name="dialup.credit">
   <Credit>0.10</Credit>
   <Debit>0</Debit>
   </OffsetGrossAccount>
-<OffsetDiscountAccount name="dialup.debit">
   <Credit>0</Credit>
   <Debit>0</Debit>
   </OffsetDiscountAccount>
-<OffsetNetAccount name="dialup.credit">
   <Credit>0.10</Credit>
   <Debit>0</Debit>
   </OffsetNetAccount>
   <OffsetTaxAccount name="-- N/A --">
```

```
<Credit>0</Credit>
<Debit>0</Debit>
</OffsetTaxAccount>
</RevenueAmounts>
</GeneralLedgerReport>
```

C

BRM G/L Report

This appendix provides a sample Oracle Communications Billing and Revenue Management (BRM) general ledger (G/L) report.

Figure C–1 is a sample of the report created by BRM:

Figure C–1 Sample BRM G/L Report

General Ledger Data

Type: unbilled
Posting Start: Sun Jan 1 00:00:00 2006
Posting End: Fri Jan 5 00:00:00 2007

102 (USD)

DEBIT ACCOUNT		DEBIT	CREDIT	CREDIT ACCOUNT	DEBIT	CREDIT
GROSS	monthly debit	39.80	9.95	monthly credit	9.95	39.80
NET	monthly debit	39.80	9.95	monthly credit	9.95	39.80
DISC	monthly credit	0.00	0.00	monthly debit	0.00	0.00
TAX	-- N/A --	0.00	0.00	-- N/A --	0.00	0.00

104 (USD)

DEBIT ACCOUNT		DEBIT	CREDIT	CREDIT ACCOUNT	DEBIT	CREDIT
GROSS	dialup debit	33.97	7.00	dialup credit	7.00	33.97
NET	dialup debit	33.97	7.00	dialup credit	7.00	33.97
DISC	dialup credit	0.00	0.00	dialup debit	0.00	0.00
TAX	-- N/A --	0.00	0.00	-- N/A --	0.00	0.00