Oracle® Communications Billing and Revenue Management

Collecting General Ledger Data

Release 7.5

E16704-10

August 2016

Oracle Communications Billing and Revenue Management Collecting General Ledger Data, Release 7.5

E16704-10

Contents

Preface vii

Audience vii
Downloading Oracle Communications Documentation vii
Documentation Accessibility vii
Document Revision History vii

1 About Collecting General Ledger Data

About General Ledger (G/L) Data 1-1
How BRM Collects G/L Data 1-2
About A/R and Offset Accounts 1-2
About Setting Up G/L Reporting 1-3
How BRM Stores General Ledger Reports 1-5
About G/L IDs 1-5
How G/L IDs Affect Storage and Reporting of G/L Data 1-6
Planning Your G/L IDs 1-6
About Collecting G/L Data for Brands or Selected Accounts 1-7
Using Multiple pin_glid Files for Multiple Segments 1-7
How Customer Accounts are Assigned to Segments 1-7
Creating Nested Segments 1-7
Setting up G/L IDs for Brands 1-8
About the Chart of Accounts 1-9
About Running G/L Reports 1-10
About Revenue Recognition 1-10
About Reversing G/L Entries 1-11
About Immediate Revenue Recognition 1-11
About Billed and Unbilled Revenue 1-11
Billed and Unbilled Usage Fees 1-12
Billed and Unbilled Purchase and Cancellation Fees 1-12
Billed Cycle Arrears Fees 1-12
Billed and Unbilled Cycle Forward Arrears Fees 1-13
Billed and Unbilled Nonrated Events 1-13
Billed and Unbilled Cycle Forward Fees 1-14
About Unbilled Cycle Forward Fees 1-14
About Accrual–Based Revenue Recognition 1-15
About Earned and Unearned Cycle Forward Arrears Fees 1-15

About Unbilled/Unearned Cycle Forward Fees 1-16
About Previously Billed Earned Revenue 1-16
About Adjustments and G/L Reporting 1-16
About Rounding and G/L Reports 1-17
Rounding G/L Report Data After Billing 1-17
Rounding G/L Report Data Prior to Billing 1-18
How BRM Records the Difference Between Rounded and Unrounded Items 1-18
G/L Report Example 1-19

2 How BRM Calculates Earned and Unearned Revenue

Understanding the Fields and Values Used to Calculate G/L 2-1
About Calculating Earned Revenue 2-1
Calculating Billed Earned Revenue 2-2
Calculating Previously Billed Earned Revenue 2-2
About Calculating Unearned Revenue 2-3

3 G/L Reporting Case Studies

About G/L Case Studies 3-1
Case Study 1 3-2
Case Study 2 3-6
Case Study 3 3-13
Case Study 4 3-19
Case Study 5 3-23
Case Study 6 3-24

4 Creating General Ledger Reports

About Running G/L Reports 4-1
About Exporting G/L Reports to XML Files 4-1
About the pin_ledger_report Utility 4-1
Controlling the Destination of pin_ledger_report Output 4-2
Generating a G/L Report 4-2
Creating an Index for the POID_TYPE Column of the ITEM_T Table 4-2
Running Reports with the pin_ledger_report Utility 4-3
Posting G/L Reports 4-3
Example: Posting a Previously Run G/L Report 4-4
Example: Unposting a G/L report 4-4
Listing Previously-Posted G/L Reports 4-4
Example: Listing G/L Reports for a Specified Time Period 4-4
Example: Listing G/L Reports for a Specific Segment or Brand 4-4
Listing G/L Reports for Specific Types of Revenue 4-4
Exporting G/L Reports to XML files 4-5
Setting the Number of A/R Accounts per G/L Report 4-5
Segregating Unbilled Revenue by G/L Cycle within a Billing Cycle 4-5
Setting Up G/L Calendar Configuration 4-7
Exporting G/L Information 4-8

5 Loading General Ledger Configuration Data

Creating a Chart of Accounts 5-1
Creating General Ledger IDs 5-2
Changing the Default G/L Segment 5-3
Assigning G/L IDs to Nonrated Events 5-4
Assigning G/L IDs to Prerated Events 5-4
Customizing G/L Data 5-5
Enabling Custom Updates to G/L Data 5-5
Customizing the PCM_OP_GL_POL_PRE_UPDATE_JOURNAL Policy Opcode 5-6
Disabling /journal Objects for Non-Currency Resources 5-7
Generating General Ledger Reports Based on the Actual Billing Date 5-8
Disabling General Ledger Collection in BRM 5-10

6 Exporting General Ledger Reports to XML Files

About Exporting G/L Reports 6-1
About Configuring G/L Reports for Export 6-1
About Incremental Reports and Cumulative Revenue 6-2
About G/L Segments and Cumulative Revenue 6-3
About Regenerating and Re-Exporting G/L Reports 6-3
How BRM Exports G/L Reports 6-4
About Exported G/L Report Files 6-4
Sample Output XML File 6-6
Configuring G/L Reports for Export 6-8
Sample G/L Report Configuration File 6-11
Editing the G/L Export Configuration File 6-13
Setting Up G/L Reporting Schedules 6-14
Filtering Noncurrency Resources 6-15
Exporting G/L Reports 6-15
Example: Exporting G/L Reports for a Specific G/L Segment 6-15
Example: Exporting G/L Reports in Parallel 6-15
Example: Regenerating G/L Reports 6-15
Example: Re-Exporting G/L Reports 6-16
Example: Restarting a Failed Export Operation 6-16
Retrieving Audit Data for Exported G/L Reports 6-16
Customizing G/L Reports for Export 6-17
Customizing G/L Export Report Contents 6-18
Mapping BRM G/L Account Names to Third-Party Account Names 6-18
Mapping BRM Resource IDs to Third-Party Resource IDs 6-19
Customizing the G/L Report XML File Names 6-19
Validating and Troubleshooting Exported G/L Reports 6-20
Purging G/L Export Data 6-20

7 General Ledger Utilities

load_pin_config_export_gl 7-2
load_pin_glchartaccts 7-4
load_pin_glid 7-5

pin_ledger_report 7-7

Preface

This book describes how to use and manage general ledger (G/L) data in Oracle Communications Billing and Revenue Management (BRM).

Audience

This document is intended for developers and system administrators.

Downloading Oracle Communications Documentation

Product documentation is located on Oracle Help Center:

`http://docs.oracle.com`

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

`https://edelivery.oracle.com`

Documentation Accessibility

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

Access to Oracle Support

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

Document Revision History

The following table lists the revision history for this book.

Version	Date	Description
E16704-01	November 2011	Initial release.
E16704-02	May 2012	Documentation updates for BRM 7.5 Patch Set 1. ▪ Made minor formatting and text changes.

Version	Date	Description
E16704-03	March 2013	Documentation updates for BRM 7.5 Patch Set 4. - Replaced multidatabase information with multischema information.
E16704-04	October 2013	Documentation updates for BRM 7.5 Patch Set 6. - Made minor formatting and text changes. - Added the "Disabling /journal Objects for Non-Currency Resources" section.
E16704-05	February 2014	Documentation updates for BRM 7.5 Patch Set 7. - Made minor formatting and text changes.
E16704-06	October 2014	Documentation updates for BRM 7.5 Patch Set 10. - Added the "Generating General Ledger Reports Based on the Actual Billing Date" section.
E16704-07	June 2015	Documentation updates for BRM 7.5 Patch Set 12. - Updated the "Generating General Ledger Reports Based on the Actual Billing Date" section.
E16704-08	August 2015	Documentation updates for BRM 7.5 Maintenance Patch Set 1. - Added the "Segregating Unbilled Revenue by G/L Cycle within a Billing Cycle" section.
E16704-09	December 2015	Documentation updates for BRM 7.5 Patch Set 14. - Made minor formatting and text changes.
E16704-10	August 2016	Documentation updates for BRM 7.5 Patch Set 16. - Made minor formatting and text changes.

1

About Collecting General Ledger Data

This chapter describes how to collect general ledger (G/L) data from your Oracle Communications Billing and Revenue Management (BRM) system.

Before reading this chapter, you should be familiar with the BRM price list, and with BRM billing and accounting cycles. See the following documents:

- *BRM Pricing and Rating*
- *BRM Configuring and Running Billing*

To set up G/L reporting, you must be familiar with accounting terms, systems, and data. You need to work with your accountants to find out how your company handles G/L data.

About General Ledger (G/L) Data

Your company's general ledger is the total list of all accounts used by your accountants to track revenue and expenses. For example, a company might use the following general ledger accounts:

- Cash
- Wages
- Accounts receivable

A service-based business typically has general ledger accounts to track revenue from different sources. For example, a company might track revenue in the following general ledger accounts:

- Monthly fees
- Usage fees
- Cancelation fees

When you set up G/L reporting in BRM, your goal is to make sure that data generated in the BRM system is reported to the correct general ledger account; for example, that revenue from monthly fees is recorded in the monthly fees account.

Note: General ledger accounts are not BRM accounts, such as those assigned to customers. A general ledger account is the name of an account used by your company accountants to track revenue.

How BRM Collects G/L Data

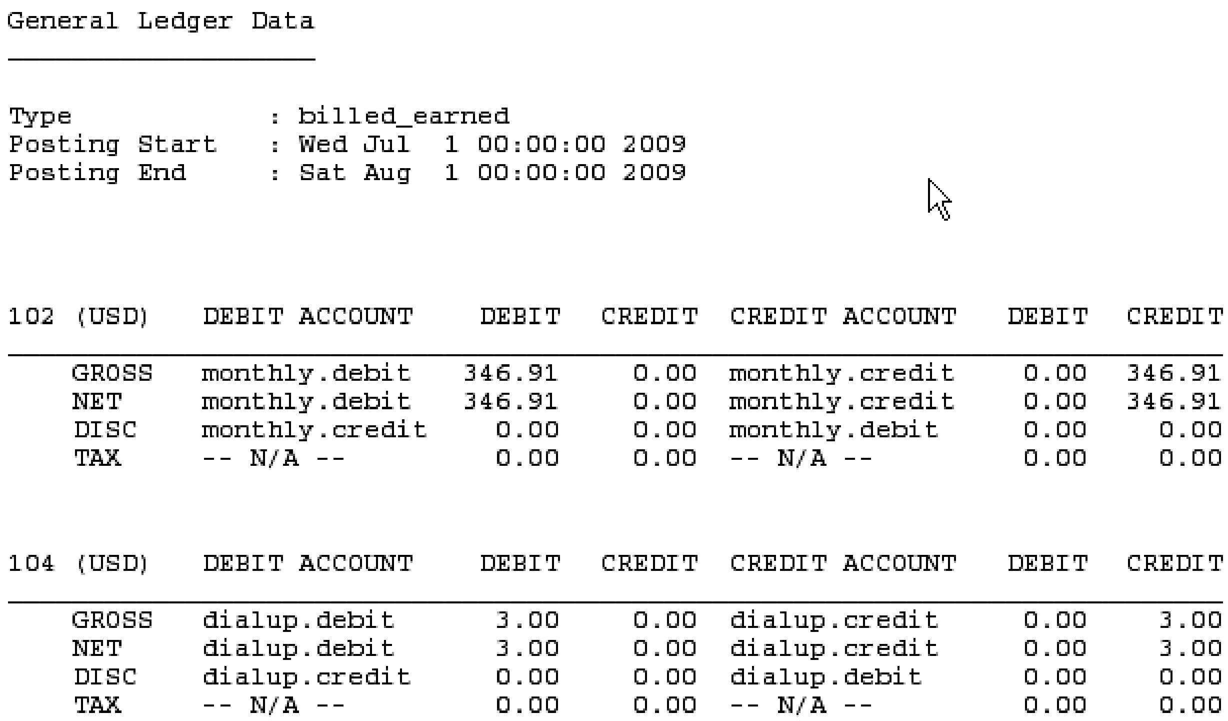
To collect G/L data, you assign a G/L code to each type of balance impact for which you need to track revenue, for example:

- The balance impact for a monthly subscription fee.
- The balance impact of a GPRS usage event.

You run a report to collect the G/L data, typically once a month. When you run the report, BRM looks at the **/journal** objects that contain summaries of revenue from the events for the last month and finds the G/L impact for each event that has a balance impact. The report summarizes the revenue for each type of balance impact.

Figure 1–1 shows a small portion of a G/L report. The account that collects revenue data about monthly fees is called monthly; the account for dialup usage fees is called dialup.

Figure 1–1 G/L Report Portion



```
General Ledger Data
___________________

Type            : billed_earned
Posting Start   : Wed Jul  1 00:00:00 2009
Posting End     : Sat Aug  1 00:00:00 2009


102 (USD)   DEBIT ACCOUNT    DEBIT   CREDIT  CREDIT ACCOUNT   DEBIT   CREDIT
____________________________________________________________________________
    GROSS   monthly.debit   346.91     0.00  monthly.credit    0.00   346.91
    NET     monthly.debit   346.91     0.00  monthly.credit    0.00   346.91
    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      3.00     0.00  dialup.credit     0.00     3.00
    NET     dialup.debit      3.00     0.00  dialup.credit     0.00     3.00
    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
```

About A/R and Offset Accounts

Each general ledger account has two parts, *A/R* and *offset*, each divided into *debit* and *credit* columns. This organization is used for double-entry accounting and is automatically tracked by BRM.

In a G/L report:

- The entries in the column labeled **DEBIT ACCOUNT** represent the A/R accounts and the ones in the column labeled **CREDIT ACCOUNT** represent the Offset accounts.
- The **DEBIT ACCOUNT** entries are always listed first and the **CREDIT ACCOUNT** entries are listed second.

- The G/L Report program always impacts the A/R account (Debit Account in G/L reports) by the impact of an event (same sign as balance impact) and always impacts the Offset account (Credit Account in G/L reports) by the exact opposite (negate the sign) of the balance impact.
- A positive number (+) is always a *debit*, and a negative number (-) is always a *credit*. There are no exceptions, no matter what kind of account it is.

For example:

- A customer purchases a product for $10:
  - The customer account balance has +$10 impact.
  - The A/R Account gets +$10 = $10 debit.
  - The Revenue Account gets -$10 = $10 credit.
- A customer credit card gets charged $10:
  - The customer account balance has -$10 impact.
  - A/R Account gets -$10 = $10 credit.
  - The Cash Account gets +$10 = $10 debit.
- A customer account gets a $10 credit adjustment:
  - The customer account balance has -$10 impact.
  - The A/R Account gets -$10 = $10 credit.
  - The Revenue Account gets +$10 = $10 debit.

About Setting Up G/L Reporting

To set up G/L reporting, you do the following:

- (Optional) Create a chart of accounts (COA) in BRM to verify that G/L accounts in BRM match the G/L accounts in your external G/L tracking system. See "Creating a Chart of Accounts" for more information.
- Create General Ledger IDs. These IDs assign G/L codes to BRM balance impacts. For example, you might assign the G/L ID 1000111 to telephone call usage balance impacts. You create G/L IDs by editing and loading the **pin_glid** file. See "Creating General Ledger IDs" for more information.

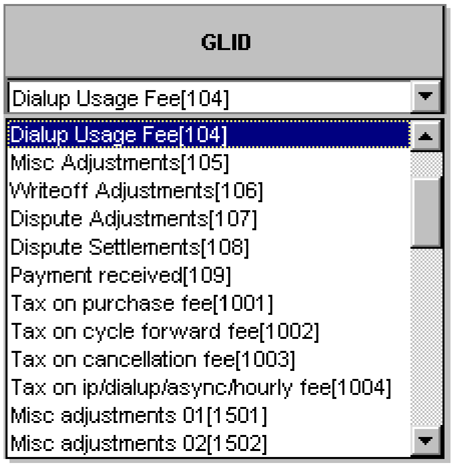
  Some balance impacts, such as payments and adjustments, are generated from nonrated events; that is, the balance impact is not created by rating billable events. You assign G/L IDs to those balance impacts by editing the **reasons**.*locale* file. You can assign a G/L ID to any event that has a balance impact. See "Assigning G/L IDs to Nonrated Events" for more information.

  In addition to assigning G/L IDs to balance impacts from different events, you can create G/L segments to track revenue by brand or by any arbitrary category, for example, by geographic region. See "Creating General Ledger IDs" for more information.
- When you create products in the Pricing Center, assign a G/L ID to each balance impact.

  Within Pricing Center, you choose from a list of G/L IDs to define rates for cycle fees, one-time fees, and usage fees. BRM assigns at least one G/L ID per rate. Within each rate, you can have more than one fee. For example, a total monthly rate can be split into email and IP fees. In this case, you can assign a separate G/L ID for the email fee and IP fee.

Figure 1–2 shows G/L assignments in Pricing Center.

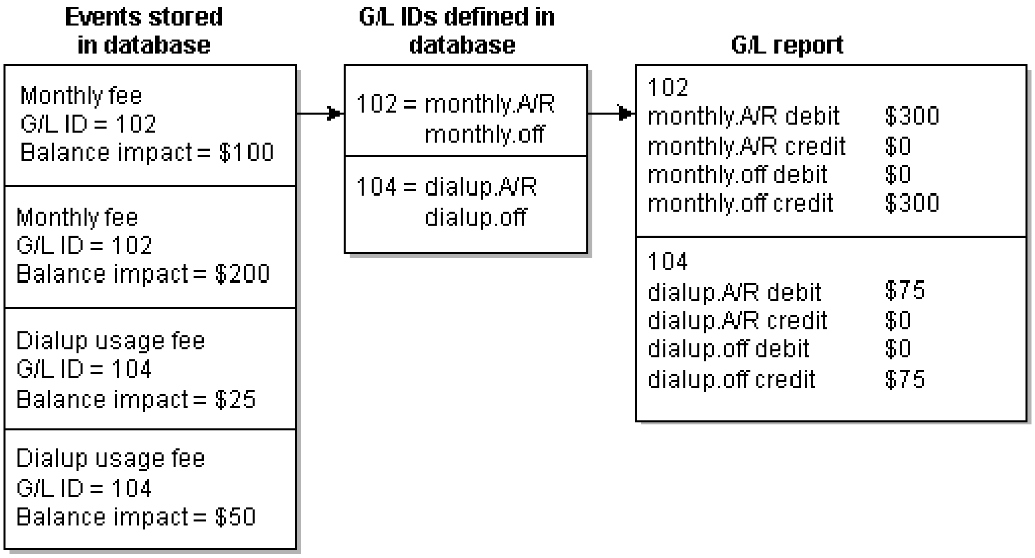
Figure 1–2 Pricing Center G/L Assignments


GLID
Dialup Usage Fee[104]
Dialup Usage Fee[104]
Misc Adjustments[105]
Writeoff Adjustments[106]
Dispute Adjustments[107]
Dispute Settlements[108]
Payment received[109]
Tax on purchase fee[1001]
Tax on cycle forward fee[1002]
Tax on cancellation fee[1003]
Tax on ip/dialup/async/hourly fee[1004]
Misc adjustments 01[1501]
Misc adjustments 02[1502]

As events occur in the BRM system, intermediate G/L results, which are summaries of the revenue, are stored in **/journal** objects in the BRM database. When you run the **pin_ledger_report** utility to create a G/L report, BRM reads the journal object for G/L activity, creates a sum of the balance impacts associated with each type of G/L ID, and records the revenue amount in the corresponding G/L account.

In the example shown in Figure 1–3, BRM finds events, totals the balance impacts for each G/L ID, and compiles the balance impacts in a G/L report.

Figure 1–3 Event G/L Assignments for G/L Report



How BRM Stores General Ledger Reports

The **pin_ledger_report** utility uses PCM_OP_GL_LEDGER_REPORT to store general ledger reports in the BRM database. This opcode calls PCM_OP_CREATE_OBJ to create a **/ledger_report** storable object.

When the PCM_OPFLG_READ_RESULT flag is set, the opcode returns the entire contents of the **/ledger_report** storable object.

About G/L IDs

BRM stores the balance impact of events in the BRM database. To track G/L data, you create a general ledger ID (G/L ID) for each type of balance impact, for example, balance impacts for usage fees.

To create G/L IDs, you edit the **pin_glid** file and use the **load_pin_glid** utility to load the G/L IDs into the BRM database.

Important: If you use Pipeline Manager to rate events, you need to use the same G/L IDs in your price list and in the Pipeline Manager pricing configuration.

Each G/L ID definition contains the following information:

- An identification number.
- A description, which is displayed in Pricing Center.
- (Optional) The tax code used by your tax calculation software.
- The G/L account that collects the revenue data, including the revenue type, for example billed, unbilled, and unearned. (See "About Revenue Recognition".) In addition, you define the following account pair attributes:
  - **gross** – reports the total of net and discounted revenue.
  - **disc** – reports the balance impacts of discounted revenue.
  - **net** – reports the amount of revenue that remains after applying discounts.
  - **tax** – reports the amount of taxes calculated. This data is used for collecting G/L data based on tax codes.

Usually, you post G/L for gross and discount revenue, or you post for net revenue.

For example, if you provide a $5 discount on a $30 cycle forward fee:

- The gross amount is $30
- The discounted amount is $5
- The net amount is $25

This example shows the G/L ID definition for a purchase fee:

```
#===========================================================================
# G/L ID for purchase rate
#===========================================================================
glid
id            10123222
taxcode       PURCHASE_TAXCODE
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       unbilled     gross      purchase.debit       purchase.credit
gl_acct       unbilled     net        purchase.debit       purchase.credit
gl_acct       unbilled     disc       purchase.credit      purchase.debit
```

Note: This organization is used for double-entry accounting, and is tracked by BRM automatically.

How G/L IDs Affect Storage and Reporting of G/L Data

The G/L ID assigned to a balance impact determines whether the balance impact is stored in a **/journal** object and whether it is included in a G/L report as shown in Table 1–1:

Table 1–1 G/L IDs

G/L ID	Description	Stored in Journal Object?	Included in G/L Report?
0	Default G/L ID. If you forget to assign a G/L ID to a balance impact type, BRM automatically assigns G/L ID 0 to it. You should reassign such balance impact types to an appropriate G/L ID.	Yes	No
1 through **99**	Assigned to balance impact types that *are not* included in G/L reports.	No	No
100 and above	Assigned to balance impact types that *are* included in G/L reports.	Yes	Yes

For information on how to control the creation of **/journal** objects for non-currency resources, see "Disabling /journal Objects for Non-Currency Resources".

Planning Your G/L IDs

When planning G/L IDs, you need to know the following information:

- The balance impacts that need revenue reporting, for example, if you need to report revenue from monthly fees, usage fees, and payments.
- The names of G/L accounts used in your company's accounting system. You use these names when assigning G/L IDs to G/L accounts. See "About the Chart of Accounts".
- The nonrated events for which you want to track revenue, for example, payments. See "Assigning G/L IDs to Nonrated Events" for more information.
- The method your company follows when reporting earned and unearned income. See "About Revenue Recognition" for more information.
- Whether your company offers services that require G/L data to be separated by G/L segment. See "About Collecting G/L Data for Brands or Selected Accounts".

> **Important:** When you create products for each brand, you use the brand-specific G/L IDs. Pricing Center displays G/L IDs for all brands. To make sure you know which G/L ID to use for each brand's price list, include the brand name in the G/L ID description.

- If you need to track tax codes. You can modify the **pin_glid** file to assign unique G/L IDs per tax code.

About Collecting G/L Data for Brands or Selected Accounts

You can use G/L segments to create G/L reports that include data for specific brands, or arbitrary sets of customers. For example, G/L reports for different geographic regions.

You specify the segment to run the report on when you run the **pin_ledger_report**.

For information about creating segments, see "Creating General Ledger IDs" for more information.

Using Multiple pin_glid Files for Multiple Segments

You can use one **pin_glid** file for multiple segments, or you can load multiple **pin_glid** files, one for each G/L segment.

- If you use one **pin_glid** file for multiple segments, all segments use the same G/L IDs.
- If you use a different **pin_glid** file for each segment, you can create different G/L IDs for each segment. This is the typical method.

How Customer Accounts are Assigned to Segments

To collect G/L data about a specific brand or a group of accounts, BRM needs to know which brand or G/L segment applies to each customer account.

- Accounts that are assigned to a brand are automatically assigned a G/L segment.
- Your BRM system uses a default *root* segment, which is defined in the Connection Manager **pin.conf** file. By default, all accounts use the root segment. To change the default segment, see "Changing the Default G/L Segment" for more information.
- To assign G/L segments without branding, you must define the G/L segments by editing policy source code. See the PCM_OP_CUST_POL_PREP_BILLINFO opcode for more information.

Creating Nested Segments

You can create nested G/L segments. For example, you can create the following segments:

- northwest
- northwest.washington
- northwest.oregon

This allows you to run reports on all of your Northwest region, including Oregon and Washington, or just on Oregon and Washington specifically.

You can define unlimited levels of nesting.

You can specify if the data for a segment should be included when you run a report against the parent segment. To do so, you use the **no_rollup** entry in the **pin_glid** file to specify that the segment should not be included in the parent report.

```
gl_segment  .
gl_segment  .northwest
gl_segment  .northwest.washington
gl_segment  .northwest.oregon  no_rollup
gl_segment  .southwest     no_rollup
gl_segment  .Central
```

As seen in Table 1–2, .northwest, .northwest.washington, and .central are included in parent reports, but .northwest.oregon and .southwest are not.

Table 1–2 Data Included for Example Reports

Report for This Segment	Includes Data for These Segments
. (root segment)	.northwest, .northwest.washington, and .central
.northwest	.northwest and .northwest.washington
.northwest.washington	.northwest.washington
.northwest.oregon	.northwest.oregon
.southwest	.southwest
.central	.central

Important: Be sure to specify the parent segment in the report or you will get an error message like the following:

```
E 05/26/03 15:39:54  -  load_pin_glid:1960.2460  load_pin_
glid.c:1477
Root Segment .NSL_REGIONAL is missing. Define the root before the
child (.NSL_REGIONAL.BB)
```

Setting up G/L IDs for Brands

To create G/L IDs for separate brands, you create multiple G/L segments and multiple **pin_glid** files.

1. Create a **pin_glid** file for the root segment.
2. Define a separate **pin_glid** file for each brand. Use the brand name as the segment name.
3. After defining the G/L IDs for each brand, prepare a list of G/L ID numbers that are used across all brands.
4. In the **pin_glid** file for the root segment, define all the common G/L IDs. If you do not require ledger reports for the root segment, the general ledger accounts you specify here can be dummy accounts. However, if you need to track revenue for the root segment, define the appropriate general ledger accounts for the G/L IDs you require. For any G/L IDs you do not require, specify dummy general ledger accounts.

Important: When you create products for each brand, you use the brand-specific G/L IDs. Pricing Center displays G/L IDs for all brands. To make sure you know which G/L ID to use for each brand's price list, include the brand name in the G/L ID description.

5. Load the **pin_glid** files, including the **pin_glid** file for the root segment, by using the **load_pin_glid** utility.

Note: The G/L IDs for nonrated events are the same for all brands.

About the Chart of Accounts

You can load a chart of accounts (COA) into the BRM database. The COA verifies that the G/L accounts you enter in the G/L IDs are valid. The COA is optional.

You load the COA before loading G/L IDs. When you load the G/L IDs, the **load_pin_glid** utility reads the COA to verify that the G/L accounts exist. If the utility finds a nonvalid G/L account, it reports an error.

If you use multiple G/L segments, you can load multiple COAs and assign a different COA to each segment.

See "Creating a Chart of Accounts" for information on loading a COA.

The COA includes the following fields:

- **coa_id**: Defines the identification number for the chart of accounts. You use this ID in the **pin_glid** file to specify which COA to use for the G/L IDs in that file.
- **coa_name**: Represents a short name that describes the COA.
- **gl_coa_accts**: Lists the G/L accounts that are part of the COA. Each entry includes a description of each G/L account, such as the account number, name, type, and status of the account.

Shown below is a sample **pin_glchartaccts** file. The sample also indicates the type of the account: asset, revenue, expense, or liability. It also indicates that all the accounts are active.

```
gl_chartaccts (
    coa_id          1000
    coa_name        Primary COA
    gl_coa_acct     0          undefined       revenue    active
    gl_coa_acct     1          undefined       asset      active
    gl_coa_acct     49400      prepaid.off     revenue    active
    gl_coa_acct     49300      monthly.A/R     asset      active
    gl_coa_acct     49200      uncollect.A/R   asset      active
    gl_coa_acct     40800      uncollect.off   revenue    active
    gl_coa_acct     40700      cancel.A/R      asset      active
    gl_coa_acct     40500      prepaid.A/R     asset      active
    gl_coa_acct     40000      purchase.off    revenue    active
    gl_coa_acct     20160      monthly.off     revenue    active
    gl_coa_acct     20150      cancel.off      revenue    active
    gl_coa_acct     11000      purchase.A/R    asset      active
    gl_coa_acct     10600      daily.A/R       asset      active
)
```

About Running G/L Reports

Use the **pin_ledger_report** utility to extract data from BRM and create G/L reports. This utility searches the BRM database for **/journal** objects containing G/L activity and summaries of revenues and either displays the data or writes it to the BRM database. You can also use the **pin_ledger_report** utility to export G/L reports to XML files so they are available to an external G/L system. See "Exporting General Ledger Reports to XML Files" for more information.

Note: **/journal** objects are created and updated by Rated Event (RE) Loader.

Writing the data to the database is called *posting* the G/L report. You typically post G/L reports once a month. When you post a G/L report, you prevent backdating adjustments, write-offs, subscription transactions, or other transactions, prior to the last date you posted the G/L report. This maintains general ledger data integrity. After they are posted, the G/L report updates the **/data/ledger_report** object. This object records the date of the latest posted G/L report and controls transaction backdating.

You can run G/L reports at any time without posting data. When you do not post data, backdating is not affected.

Note: Exporting G/L reports automatically creates them; this is part of the export operation. You do not need to create them first.

About Revenue Recognition

To report general ledger data consistently, you run G/L reports on a regular basis, for example, on the last day of the month. However, customer accounts can be billed on any day of the month. Therefore, at the time the G/L report is run, some accounts have been billed and some have not, so some revenue is recognized as billed and some as unbilled.

Also, you might run a G/L report that includes revenue from a cycle forward fee that is partially earned and partially unearned. For example, if a customer is charged for one month in advance, if you run a G/L report in the middle of the month, the revenue for the first two weeks has been earned, but the revenue for the last two weeks has not been earned yet, even if it has been billed.

BRM recognizes seven revenue types:

- Billed
- Unbilled
- Billed earned
- Unbilled earned
- Billed unearned
- Unbilled unearned
- Previously billed earned

Whereas most companies generate G/L reports for billed and unbilled revenue, it is less common to recognize the difference between earned and unearned revenue. You need to find out if your company uses immediate revenue recognition or accrual-based revenue recognition:

- *Immediate revenue recognition* reports all charges as earned as soon as they are applied. For example, a monthly cycle forward fee is considered earned even if the customer has not used the entire month of service. All revenue is recognized and reported as either billed or unbilled.

  In this case, billed and unbilled revenue include earned and unearned revenue:

  – Billed = billed earned + billed unearned + previously billed earned

  – Unbilled = unbilled earned + unbilled unearned

  See "About Immediate Revenue Recognition".

- *Accrual–based revenue recognition* reports all charges as earned based on when the services are rendered. For example, a monthly cycle forward fee is considered only partially earned if the customer has not used the entire month of service. All revenue is recognized and reported as earned or unearned. See "About Accrual–Based Revenue Recognition".

About Reversing G/L Entries

An amount reported as unbilled should be reversed in your company's general ledger in the period following the one in which it was reported. For example, an entry made for unbilled revenue in the month of January should be reversed if it is billed in February. BRM automatically reverses its general ledger data when it changes from unbilled to billed; however, you might need to reverse the data in your company's accounting software.

About Immediate Revenue Recognition

If your company recognizes immediate revenue, you need to customize BRM to report billed and unbilled revenue for all G/L accounts.

About Billed and Unbilled Revenue

Billed revenue is revenue that was already billed at the time that the G/L report is generated. This includes revenue from charges that have been included in a bill, but not necessarily earned.

Unbilled revenue is revenue that includes charges not included in a bill. For example, usage fees that have accrued after the last bill, but have not been billed. Unbilled items become billed at the end of the customer's billing cycle.

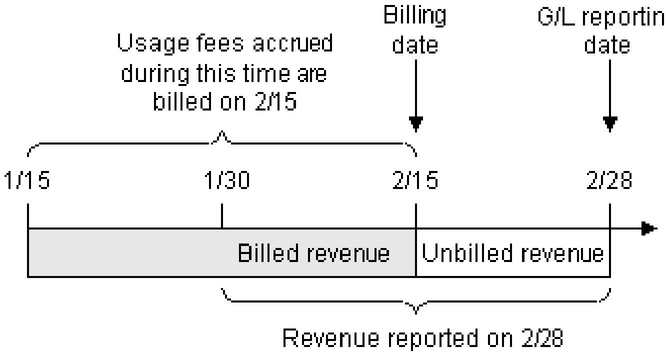
> **Note:** *Unbilled* is not the same as *uncollected*. For example, if a customer uses a credit card to pay a cycle forward fee for the first month of service at registration time, the money is collected from the credit card company almost immediately. The cycle forward fee remains unbilled, however, until you run the BRM billing applications.

The following examples show how different types of rates and events can be billed or unbilled depending on the billing date and the G/L reporting date.

Billed and Unbilled Usage Fees

In the example shown in Figure 1–4, the last billing date for the customer was 2/15, and the G/L report was run on 2/28. Usage fees occurring between 2/15 and 2/28 are reported as unbilled.

Figure 1–4 Billed and Unbilled Usage Fees

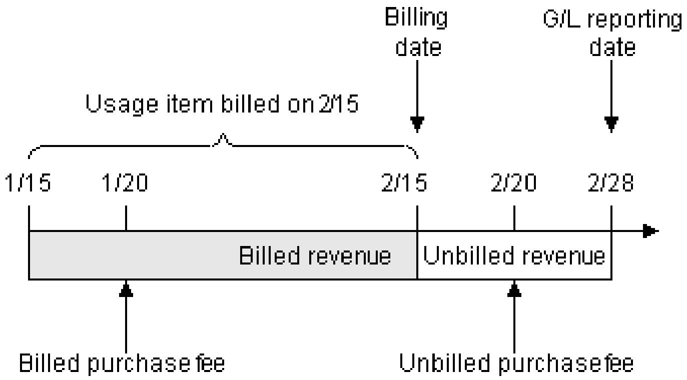


Billed and Unbilled Purchase and Cancellation Fees

If a purchase or cancel event occurs after the billing date but before the G/L reporting date, BRM G/L reports the revenue as unbilled.

In the example shown in Figure 1–5, a purchase fee made on 1/20 was billed during the scheduled billing cycle that ran on 2/15. Another purchase fee was made on 2/20. Since the scheduled billing date (2/15) has passed, the revenue for the second purchase fee is recorded as unbilled.

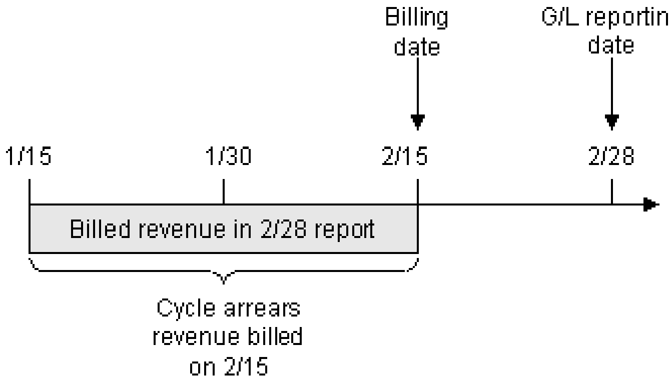
Figure 1–5 Billed and Unbilled Purchase and Cancellation Fees



Billed Cycle Arrears Fees

Cycle arrears fees are always billed. In the example shown in Figure 1–6, the cycle arrears fee billed on 2/15 is included in the 2/28 report as billed.

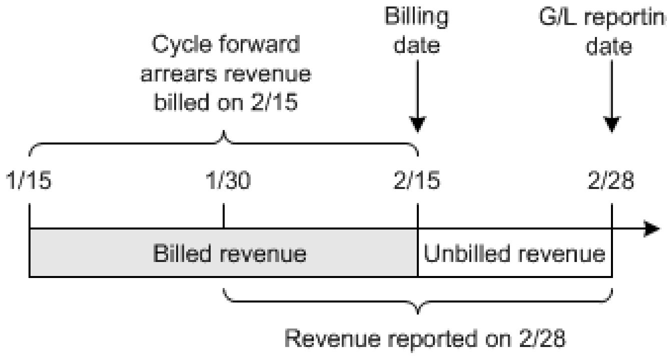
Figure 1–6 Billed Cycle Arrears Fees


Billing date
G/L reporting date
1/15
1/30
2/15
2/28
Billed revenue in 2/28 report
Cycle arrears revenue billed on 2/15

Billed and Unbilled Cycle Forward Arrears Fees

Cycle forward arrears fees can be reported as billed or unbilled, depending on when the G/L report is run. In the example shown in Figure 1–7, the last billing date for the customer was 2/15, and the G/L report was run on 2/28. Cycle forward arrears fees occurring between 2/15 and 2/28 are reported as unbilled:

Figure 1–7 Billed and Unbilled Cycle Forward Arrears Fees

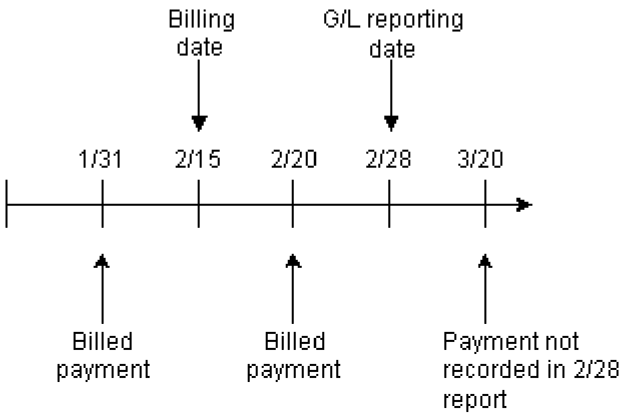


Billed and Unbilled Nonrated Events

Revenue from nonrated events that are not included in bill items, such as payments and refunds, is always recorded as billed if the event occurs before the G/L report. The billing date has no affect on whether a payment or refund is reported as billed. If a payment or refund event occurs after a G/L reporting date, it is not included in the report.

In the example shown in Figure 1–8, the payments made on 1/31 and 2/20 are included as billed revenue in the 2/28 G/L report. The payment made on 3/20 is not included in the 2/28 G/L report.

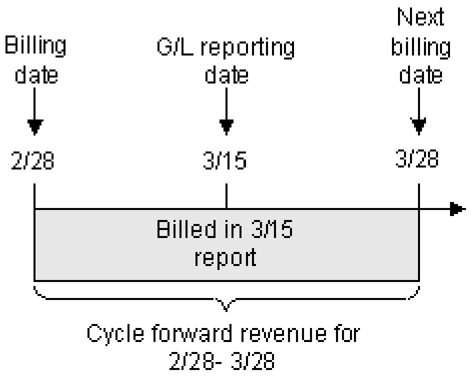
Figure 1–8 Billed and Unbilled Nonrated Events



Billed and Unbilled Cycle Forward Fees

Cycle forward fees are always billed except when the cycle forward fee is for a new account. In the example shown in Figure 1–9, the cycle forward fee billed on 2/28 is reported as billed in the G/L report on 3/15.

Figure 1–9 Billed and Unbilled Cycle Forward Fees



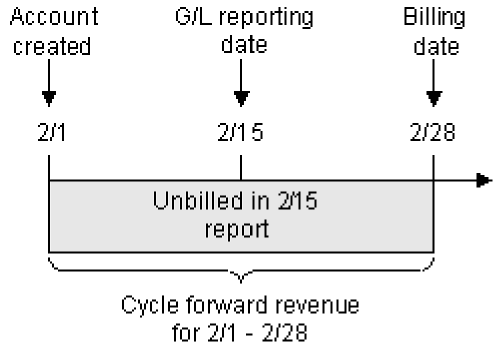
Note: Because cycle forward fees pay for services in advance, the G/L report can include revenue for services that have not yet been earned.

About Unbilled Cycle Forward Fees

Cycle forward fees can be unbilled when the fee is for a new account.

In the example shown in Figure 1–10, the cycle forward fee charged when the account is created is not billed until 2/28. Therefore, the G/L report run on 2/15 reports the revenue as unbilled.

Figure 1–10 Unbilled Cycle Forward Fees



About Accrual–Based Revenue Recognition

When your company recognizes accrual–based revenue, you need to configure your BRM G/L system to report earned and unearned revenue.

Revenue that is earned at the time that a G/L report is run is called *earned* revenue. Fees applied to usage, purchase, and cancellation of items are always accounted for as earned revenue.

Unearned revenue is revenue that is not earned at the time the G/L report is run. Unearned revenue *only* applies to revenue from cycle forward fees and cycle forward arrear fees.

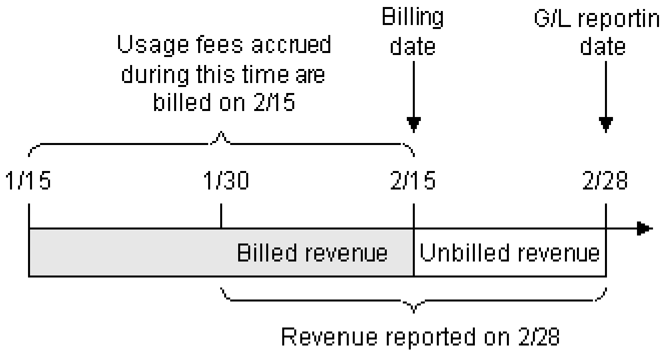
See "How BRM Calculates Earned and Unearned Revenue" for more information on earned and unearned revenue.

About Earned and Unearned Cycle Forward Arrears Fees

Cycle forward arrears fees can be reported as earned or unearned depending on when the G/L report is run. In the example shown in Figure 1–11, the customer’s billing date is 3/31. When the G/L report is run on 3/15, BRM accounts for G/L revenue from cycle forward arrears fees as follows:

- The cycle forward arrears fee from 3/1 - 3/15 is unbilled/earned.
- The cycle forward arrears fee from 3/16 - 3/31 is unbilled/unearned.

Figure 1–11 Earned and Unearned Cycle Forward Arrears Fees



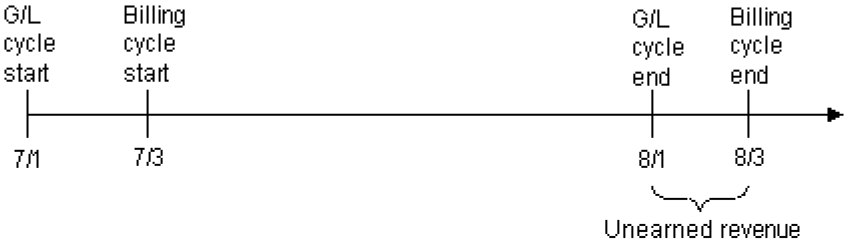
See "How BRM Calculates Earned and Unearned Revenue" for information on how earned and unearned revenue is calculated.

About Unbilled/Unearned Cycle Forward Fees

The first cycle forward fee for a new account can be unbilled and unearned. In the example shown in Figure 1–12, the first cycle forward fee for the new account is not billed until 2/28. When the G/L report is run on 2/15, BRM accounts for G/L revenue from cycle forward fees as follows:

- The cycle forward fee from 2/1 - 2/15 is unbilled/earned.
- The cycle forward fee from 2/16 - 2/28 is unbilled/unearned.

Figure 1–12 Unbilled/Unearned Cycle Forward Fees



About Previously Billed Earned Revenue

Previously–billed revenue is revenue that was billed in the previous billing cycle, but recognized in the current G/L cycle. For example, if a portion of a cycle event, such as a cycle fee, is earned across two G/L cycles, then BRM reports the earned portion of this revenue as previously–billed earned revenue. See "Calculating Previously Billed Earned Revenue" for more information.

About Adjustments and G/L Reporting

By default, adjustments made to an open bill item or a pending bill item are *always* considered billed. The BRM system creates an **/event/billing/adjustment/event** storable object and records the adjustment amount as billed in the adjustment item (**/item/adjustment**). The adjusted amount is reported in the billed G/L report.

In addition, when adjustments are made to a pending bill item, the original amount in the bill item is considered unbilled and is reported in the unbilled G/L report, and the adjustment amount in the adjustment item (**item/adjustment**) is considered billed and reported in the billed G/L report. When the pending bill items are eventually billed, the billed amount is reported in the billed G/L report.

You can change this behavior to create shadow event objects instead of adjustment event objects. Creating shadow event objects has the following advantages:

- For pending bill item adjustments, both the original amount and the adjustment amount are associated with the same bill item and are recorded in the unbilled G/L report until billing is run.
- Adjustment amounts do not show up in customers' bills as line items that modify the total due because the events have already been adjusted. You can specify whether shadow event adjustment details are displayed in invoices. See the discussion on customizing information included in invoices in *BRM Configuring and Running Billing*.

Note: Shadow events are created for event-level adjustments only. BRM creates shadow events only for events that are adjusted before they are billed.

To specify whether to generate a shadow event or adjustment event, you modify the input flist of the PCM_OP_AR_EVENT_ADJUSTMENT opcode. For more information, see the discussion of unbilled event-level adjustments in G/L reports in *BRM Managing Accounts Receivable*.

About Rounding and G/L Reports

To round event balance impacts for billing and G/L reports, you specify a rounding rule for A/R processes in the balance element ID (BEID) configuration file (*BRM_Home***/sys/data/pricing/example/pin_beid**). You then run the **load_pin_beid** utility to load the contents of the **pin_beid** file into the **/config/beid** object in the BRM database. For more information, see the discussion on rounding in *BRM Setting up Pricing and Rating*.

The balance impacts of events are totaled and rounded separately for billing and G/L reports:

- When generating a bill, the event balance impacts in the items are totaled, then the items are rounded, summed, and added to the bill.
- When generating a G/L report, the balance impacts of all events with the same G/L ID are totaled in journal entries, then the journal entries are rounded and posted.

Note: Balance impacts are intermittently sub-totaled in G/L journal entries prior to posting, but are not rounded until a report is generated. For more information about journal entries, see **/journal**.

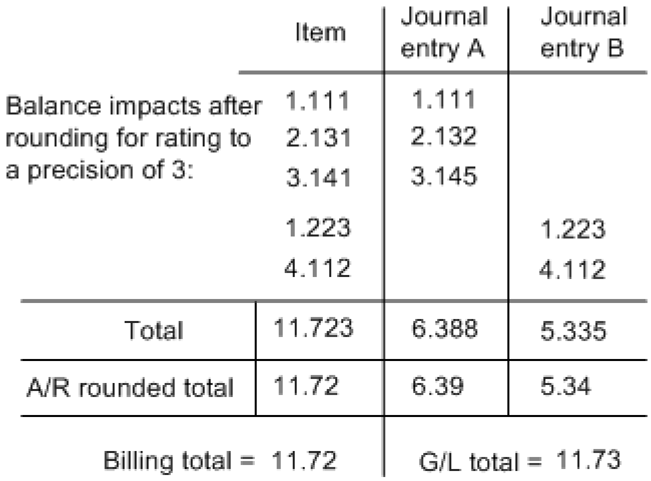
For information about setting up BRM to record rounding differences, see the discussion on resource rounding in *BRM Setting up Pricing and Rating*.

Rounding G/L Report Data After Billing

Events that belong to an item can belong to different journal entries. When there are a great number of items, this can cause minor rounding differences between the billing totals and the G/L report totals.

Figure 1–13 is a simplified example of how billing and G/L totals for the same balance impacts can differ when the balance impacts in an item belong to different journal entries. In this example, the precision is 3 for rating and 2 for A/R, and the mode is round to the nearest:

Figure 1–13 Rounding G/L Report Data after Billing



	Item	Journal entry A	Journal entry B
Balance impacts after rounding for rating to a precision of 3:	1.111 2.131 3.141	1.111 2.132 3.145	
	1.223 4.112		1.223 4.112
Total	11.723	6.388	5.335
A/R rounded total	11.72	6.39	5.34
Billing total = 11.72		G/L total = 11.73	

When you run billing, this rounding difference is recorded in the bill item, and included in G/L reports. A G/L ID is defined for rounding difference so that the G/L report can be accurately reconciled. See "How BRM Records the Difference Between Rounded and Unrounded Items" for more information.

Rounding G/L Report Data Prior to Billing

Because items are not rounded prior to billing, they might have a high precision such as six significant digits. However, journal entries are rounded when G/L reports are run. This can create small rounding discrepancies because rounding differences are not recorded until billing is run. For example, if the sum of all billing items is 100.53009, the pre-billing G/L report will display the rounded amount of 100.53, leaving a difference of 0.00009 undocumented.

How BRM Records the Difference Between Rounded and Unrounded Items

The difference between rounded and unrounded item totals is typically a very small amount referred to as the delta. This delta is recorded and included in G/L reports so that they can be accurately reconciled. The delta rounding difference is recorded in the following ways:

- When billing is run and pending items are rounded, the delta rounding difference is recorded in the **/item** object.
- A corresponding **/journal** entry object is created for each pending item. The delta rounding difference is stored as a credit or debit in the **/journal** object to balance the G/L:
  - If the rounding difference is negative, the balance needs to be credited, so the amount is stored in the PIN_FLD_CR_AR_NET_AMT field.
  - If the rounding difference is positive, the balance needs to be debited, so the amount is stored in the PIN_FLD_DB_AR_NET_AMT field.
- When a G/L report is generated, the journal entry containing the delta rounding difference is included in the report.

The delta rounding difference is calculated as the difference between a rounded pending item total and the sum of the rounded journal entries associated with the pending item:

```
    round(unrounded pending items)
 -  sum(round(journal entries)
------------------------------------------
 =  rounding difference
```

For example, if a rounded pending item total is 11.72 and the sum of rounded journal entries is 11.73, the delta rounding difference is 11.72 - 11.73, or -.01.

G/L Report Example

This example shows how revenue is reported for three customer accounts over several months. The following diagrams show the billing cycles relative to monthly G/L reports for each account. Arrows represent billing cycles. G/L reporting periods begin on the first day of each month and end on the last day of each month.

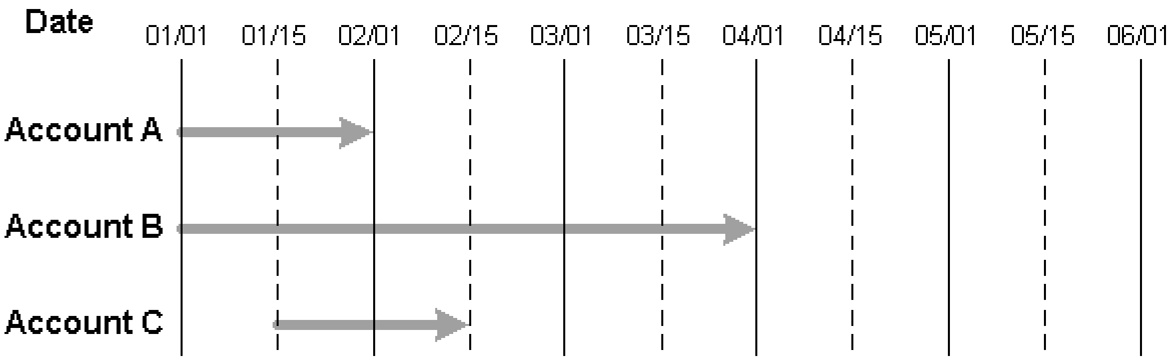
> **Note:** To make these examples clearer, the calculated amounts are approximations. For example, a value of $15.00 might actually be $16.45, if the BRM calculations are used. See "How BRM Calculates Earned and Unearned Revenue" for more information.

G/L results for January 31

- Account A has a monthly billing cycle that starts on 01/01.
- Account B has a quarterly billing cycle that starts on 01/01.
- Account C has a monthly billing cycle that starts on 01/15.

Figure 1–14 shows the G/L results for January 31.

Figure 1–14 Billing Cycles Relative to Monthly G/L Reports for January 31



- Account A incurs a one-time $5 purchase fee for account creation and a recurring $30 monthly cycle forward fee.
- Account B incurs a one-time $5 purchase fee for account creation and a recurring $90 quarterly cycle forward fee.
- Account C incurs a one-time $5 purchase fee for account creation and a recurring $30 monthly cycle forward fee.

Table 1–3 G/L Results as of January 31

January 31 G/L report	A/R Billed	A/R Billed	A/R UnBilled	A/R UnBilled	Billed Earned	Billed Earned	Billed Unearned	Billed Unearned	Previous Billed Earned	Previous Billed Earned	Unbilled Earned	Unbilled Earned	Unbilled Unearned	Unbilled Unearned
Account A	NA	NA	35	NA	NA	NA	NA	NA	NA	NA	NA	35	NA	NA
Account B	NA	NA	95	NA	NA	NA	NA	NA	NA	NA	NA	35	NA	60
Account C	NA	NA	35	NA	NA	NA	NA	NA	NA	NA	NA	20	NA	15
Total period	NA	NA	165	NA	NA	NA	NA	NA	NA	NA	NA	90	NA	75
Total cumulative	NA	NA	165	NA	NA	NA	NA	NA	NA	NA	NA	90	NA	75

Table 1–3 shows as of January 31:

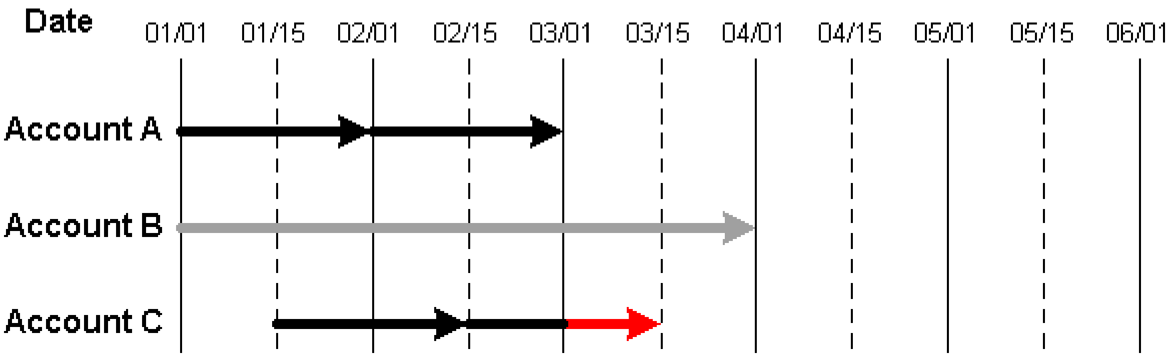
- Revenue from Account A is unbilled because the billing utilities have not yet been run. This revenue appears as a $35 debit to the **AR Unbilled** T-account. In addition, the revenue is earned because service has been rendered for that revenue. Because it is earned as well as unbilled, it appears as a $35 credit to the **Unbilled Earned** T-account.
- Revenue from Account B is also unbilled because the billing utilities have not yet been run. The entire $95 appears as a debit to the **AR Unbilled** T-account. In addition, $35 of the revenue is earned because service has been rendered (account creation and one-third of the quarter.) This $35 appears as a credit to the **Unbilled Earned** T-account. The remaining $60 is unearned because no service has yet been rendered for it. It appears as a $60 credit to the **Unbilled Unearned** T-account.
- Revenue from Account C is unbilled because the billing utilities have not yet been run. It appears as a $35 debit to the **AR Unbilled** T-account. Of Account C's revenue, $20 of it is earned because some service has been rendered (account creation and one-half of the month.) So, this portion appears as a $20 credit to the **Unbilled Earned** T-account. The unearned portion appears as a $15 credit to the **Unbilled Unearned** T-account.

G/L results for February 28

- Account A has been through a complete G/L reporting cycle and a complete billing cycle, and has another monthly billing cycle that starts 02/01.
- Account B has been through a complete G/L reporting cycle, but has not reached the end of its billing cycle.
- Account C has been through the end of both a G/L reporting and a billing cycle, and has a monthly billing cycle starting on 02/15.

Figure 1–15 illustrates the G/L results for February 28.

Figure 1–15 Billing Cycles Relative to Monthly G/L Reports for February 28



- Account A incurs a $30 monthly cycle forward fee on 02/01.
- Account C incurs a $30 monthly cycle forward fee on 02/15.

Table 1–4 G/L Results as of February 28

February 28 G/L report	A/R Billed	A/R Billed	A/R UnBilled	A/R UnBilled	Billed Earned	Billed Earned	Billed Unearned	Billed Unearned	Previous Billed Earned	Previous Billed Earned	Unbilled Earned	Unbilled Earned	Unbilled Unearned	Unbilled Unearned
Account A	65	NA	NA	35	NA	65	NA	NA	NA	NA	35	NA	NA	NA
Account B	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	30	30	NA
Account C	65	NA	NA	35	NA	50	NA	15	NA	NA	20	NA	15	NA
Total period	130	NA	NA	70	NA	115	NA	15	NA	NA	25	NA	45	NA
Total cumulative	130	NA	95	NA	NA	115	NA	15	NA	NA	NA	65	NA	30

Table 1–4 shows as of February 28:

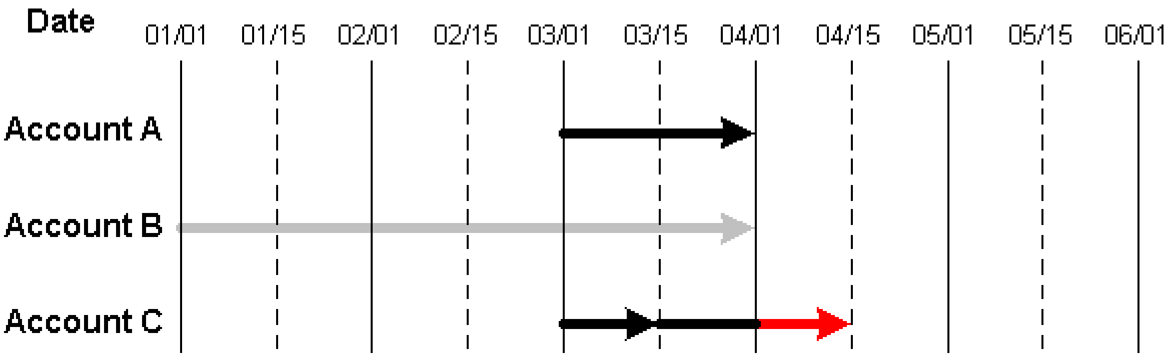
- All revenue from Account A is billed. The $35 that appeared in the **AR Unbilled** T-account on January 31 now appears as a credit to the **AR Unbilled** account and a debit to the **AR Billed** account. New revenue generated by the $30 cycle forward fee during February also appears as a debit to the **AR Billed** T-account. Also, the $35 that appeared as a credit to the **Unbilled Earned** T-account on January 31 is offset by a $35 debit. In addition, a total of $65 ($35 from January and $30 from February) appears as a credit to the **Billed Earned** T-account.
- Revenue from Account B is unbilled because the billing cycle is not over yet. The only change to Account B's T-accounts is a $30 credit to the **Unbilled Earned** T-account and a matching debit to the **Unbilled Unearned** T-account because another one-third of the quarter has elapsed.
- All revenue from Account C is billed. The $35 that appeared in the **AR Unbilled** T-account on January 31 now appears as a credit to the **AR Unbilled** account and as a debit to the **AR Billed** account. New revenue generated by the $30 cycle forward fee during February also appears as a debit to the **AR Billed** T-account. Also, the $20 that appeared as a credit to the **Unbilled Earned** T-account on January 31 is offset by a $20 debit. In addition, a total of $50 ($35 from January and $15 from February) appears as a credit to the **Billed Earned** T-account and $15 appears as a credit to the **Billed Unearned** T-account.

G/L results for March 31

- Account A has a billing cycle that ends 04/01.
- Account B has a billing cycle that ends 04/01.
- Account C has a billing cycle that ends 03/15 and another begins 03/16.

Figure 1–16 illustrates the G/L results for March 31.

Figure 1–16 Billing Cycles Relative to Monthly G/L Reports for March 31



- Account A incurs a $30 monthly cycle forward fee on 03/01.
- Account C incurs a $30 monthly cycle forward fee on 03/15.

Table 1–5 G/L Results as of March 31

March 31 G/L report	A/R Billed	A/R Billed	A/R UnBilled	A/R UnBilled	Billed Earned	Billed Earned	Billed Unearned	Billed Unearned	Previous Billed Earned	Previous Billed Earned	Unbilled Earned	Unbilled Earned	Unbilled Unearned	Unbilled Unearned
Account A	30	NA	NA	NA	NA	30	NA	NA	NA	NA	NA	NA	NA	NA
Account B	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	30	30	NA
Account C	30	NA	NA	NA	NA	15	15	15	NA	15	NA	NA	NA	NA
Total period	60	NA	NA	NA	NA	45	NA	0	NA	15	NA	30	30	NA
Total cumulative	190	NA	95	NA	NA	160	NA	15	NA	15	NA	95	NA	0

Table 1–5 shows as of March 31:

- All revenue from Account A is billed and earned. It appears as a $30 debit to the **AR Billed** T-account and as a $30 credit to the **Billed Earned** T-account.
- Revenue from Account B is unbilled because the billing cycle is not over yet. The only change to Account B's T-accounts is a $30 credit to the **Unbilled Earned** T-account and a matching debit to the **Unbilled Unearned** T-account because the final third of the quarter has elapsed.
- All revenue from Account C is billed. It appears as a $30 debit to the **AR Billed** T-account. Of the $30 billed, $15 appears as a credit to the **Billed Earned** T-account and the other $15 appears as a credit to the **Billed Unearned** T-account. Also, $15

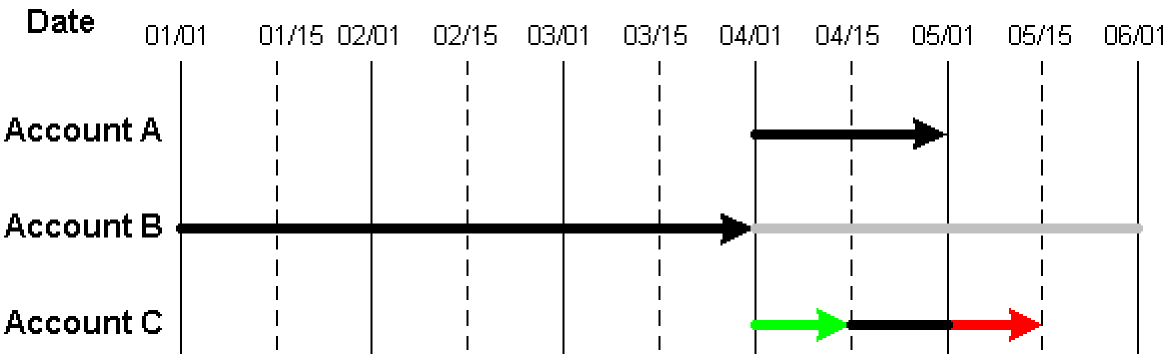
that was considered billed in the February 28 report became earned and appears as a $15 credit to the **Prev Billed Earned** T-account, offset by a $15 debit to the **Billed Unearned** T-account.

G/L reported as of April 30

- Account A has a billing cycle that ends 05/01.
- Account B is in the middle of a billing cycle that ends 07/01.
- Account C has a billing cycle that ends 04/15 and another that begins 04/16.

Figure 1–17 illustrates the G/L results for April 30.

Figure 1–17 Billing Cycles Relative to Monthly G/L Reports for April 30



- Account A incurs a $30 monthly cycle forward fee on 04/01.
- Account C incurs a $30 monthly cycle forward fee on 04/15.

Table 1–6 G/L Results as of April 30

April 30 G/L report	A/R Billed	A/R Billed	A/R UnBilled	A/R UnBilled	Billed Earned	Billed Earned	Billed Unearned	Billed Unearned	Previous Billed Earned	Previous Billed Earned	Unbilled Earned	Unbilled Earned	Unbilled Unearned	Unbilled Unearned
Account A	30	NA	NA	NA	NA	30	NA	NA	NA	NA	NA	NA	NA	NA
Account B	185	NA	NA	95	NA	125	NA	60	NA	NA	95	NA	NA	NA
Account C	30	NA	NA	NA	NA	15	15	15	NA	15	NA	NA	NA	NA
Total period	245	NA	NA	95	NA	175	NA	60	NA	15	95	NA	NA	NA
Total cumulative	435	NA	NA	0	NA	330	NA	75	NA	30	NA	0	NA	NA

Table 1–6 shows as of April 30:

- All revenue from Account A is billed and earned. It appears as a $30 debit to the **AR Billed** T-account and as a $30 credit to the **Billed Earned** T-account.
- All revenue from Account B is billed as follows:
  - $185 debit to **Billed**.
  - $95 credit to **Unbilled** to offset the $95 debit made in January.

- $125 credit to **Billed Earned**, where $95 is for the quarter ending 03/31 and $30 is for the first third of the current billing cycle.
- $60 credit to **Billed Unearned** for the remaining two-thirds of the current billing cycle.
- $95 debit to **Unbilled Earned** to offset the $35 credit made in January and the $30 credits made in February and March.

All revenue from Account C is billed. It appears as a $30 debit to the **AR Billed** T-account. Of the $30 billed, $15 appears as a credit to the **Billed Earned** T-account and the other $15 appears as a credit to the **Billed Unearned** T-account. Also, $15 that was billed in the March 31 report became earned and appears as a $15 credit to the **Prev Billed Earned** T-account and is offset by a $15 debit to the **Billed Unearned** T-account.

2
How BRM Calculates Earned and Unearned Revenue

This chapter provides details about how Oracle Communications Billing and Revenue Management (BRM) calculates earned and unearned revenue.

Before reading this chapter, you should be familiar with the BRM General Ledger (G/L) system and billing cycles. See the following documents:

- "About Billing Customers" in *BRM Configuring and Running Billing*
- About General Ledger (G/L) Data

Understanding the Fields and Values Used to Calculate G/L

Each event in BRM is associated with an item. The status of an item can be pending, open, or closed.

Each item starts in the pending state, gathering events that accrue revenue for the duration of the billing cycle. At the time of billing, the item changes status from pending to open. An open item indicates that payment for the item is due and a bill has been generated. Once payment is received, the status of the item changes to closed.

- Revenue for events associated with a pending item is unbilled revenue.
- Revenue for events associated with an open or closed item is billed revenue.

The PIN_FLD_EFFECTIVE_T field in the **/item** object holds the date on which an item goes from pending (unbilled) to open (billed) status. This date is used by BRM to determine whether revenue is billed or unbilled.

When BRM creates a cycle event (/event/billing/cycle), the cycle dates for the cycle fee are saved in fields called PIN_FLD_EARNED_START_T and PIN_FLD_EARNED_END_T. The equations in this chapter use the values in those fields. The fields are represented in the equations as shown in Table 2–1:

Table 2–1 Fields and Equations

Field Name	Equation Name
PIN_FLD_EARNED_START_T	billing_cycle_start
PIN_FLD_EARNED_END_T	billing_cycle_end

About Calculating Earned Revenue

BRM reports two categories of earned revenue based on when the item was billed:

- Actual earned revenue, or revenue that is billed in the same period as the G/L cycle
- Previously billed earned revenue

Calculating Billed Earned Revenue

BRM uses the equation in Figure 2–1 to calculate billed earned revenue:

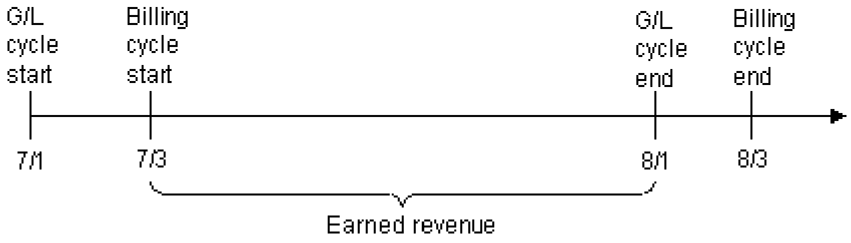
Figure 2–1 Billed Earned Revenue Equation

$$\text{amount} - \left(\frac{\text{billing\_cycle\_end} - \text{greatest(gl\_cycle\_end } \textit{or} \text{ billing\_cycle\_start)}}{\text{billing\_cycle\_end} - \text{billing\_cycle\_start}} \times \text{amount} \right) = \begin{array}{c}\text{earned}\\ \text{revenue}\end{array}$$

For information about values used in the equation, see "Understanding the Fields and Values Used to Calculate G/L".

For example, you run the G/L report on the first of every month. A new customer account is created on 7/3, after the G/L report is run on 7/1. Because the new customer’s billing cycle runs from 7/3 to 8/3, the monthly cycle fee covers the period from 7/3 to 8/3. Since G/L is posted on the first of every month, the report run on 8/1 shows the revenue earned for the monthly cycle fees for the period 7/3 to 8/1 in the **billed_earned** category as shown in Figure 2–2.

Figure 2–2 Calculating Billed Earned Revenue



If the monthly fee is $100, BRM calculates the earned revenue for the period 7/1 to 8/1 as shown in Figure 2–3:

Figure 2–3 Sample Earned Revenue Calculation

$$100 - \left(\frac{2}{31} \times 100 \right) = 93.55$$

Calculating Previously Billed Earned Revenue

BRM uses the equation in Figure 2–4 to calculate previously billed revenue:

Figure 2–4 Previously Billed Revenue Equation

$$\left(\frac{\text{least(billing\_cycle\_end } \textit{or} \text{ gl\_cycle\_end) - greatest(billing\_cycle\_start } \textit{or} \text{ gl\_cycle\_start)}}{\text{billing\_cycle\_end - billing\_cycle\_start}}\right) * \text{amount}$$

For example, if a new customer account is created on 7/3 with a bimonthly cycle fee, then the new customer is billed for the next bimonthly cycle fee on 9/3. Since your company requires you to run the G/L report on the first of every month and the cycle fee was created and billed on 7/3, the G/L report run on 8/1 reports the earnings in the **prev_billed_earned** revenue category. This is because the cycle fee was billed before the G/L run on 8/1.

If the bimonthly fee is $120, BRM calculates the previously billed earned revenue as shown in Figure 2–5:

Figure 2–5 Sample Previously Billed Earned Revenue Calculation

$$\frac{\text{least (9/3 or 9/1) - greatest (7/3 or 8/1)}}{\text{9/3 - 7/3}} = \frac{\text{9/1 – 8/1}}{\text{62 days}} = \frac{\text{31 days}}{\text{62 days}} = \frac{31}{62}$$

$$\frac{31}{62} \text{ x } 120.00 = 60.00$$

About Calculating Unearned Revenue

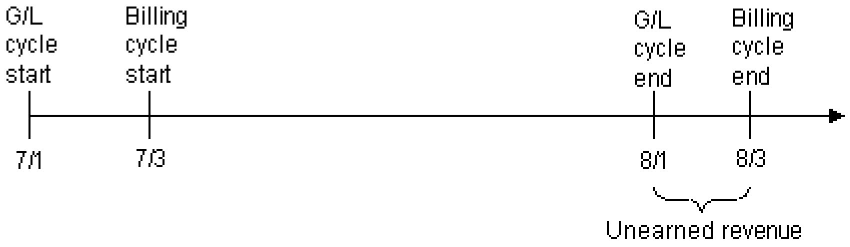
BRM calculates unearned revenue for cycle fees only. BRM uses the equation in Figure 2–6 to calculate unearned revenue:

Figure 2–6 Unearned Revenue Equation

$$\left(\frac{\text{billing\_cycle\_end - greatest(gl\_cycle\_end } \textit{or} \text{ billing\_cycle\_start)}}{\text{billing\_cycle\_end - billing\_cycle\_start}}\right) * \text{amount}$$

For example, if an account is created on 7/3, then the billing cycle will run on the third day of each subsequent month, or on 8/3 as shown in Figure 2–7. If the G/L cycle runs on the first of each month, then the timeline for the billing and G/L cycles is as follows:

Figure 2–7 Calculating Unearned Revenue



For a monthly cycle fee of $19.95, the calculation for unearned revenue is completed as shown in Figure 2–8:

Figure 2–8 Sample Unearned Revenue Calculation

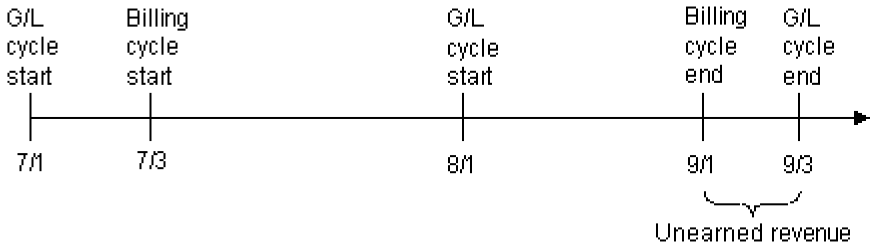
$$\boxed{\frac{8/3 - \text{greatest}(8/1 \text{ or } 7/3)}{8/3 - 7/3} = \frac{8/3 - 8/1}{8/3 - 7/3} = \frac{2 \text{ days}}{31 \text{ days}}}$$

$$\boxed{\frac{2}{31} \times 19.95 = 1.29}$$

BRM always calculates cumulative unearned revenue. For every successive G/L run, BRM continues to decrease the unearned portion and increase the earned portion. This continues until all revenue is earned by the end date.

If a customer starts a service on 7/3 with a bimonthly cycle fee, then the timeline is as shown in Figure 2–9:

Figure 2–9 Cumulative Unearned Revenue Timeline



The unearned revenue for August is also reported as unearned in July. The unearned revenue for July and August is calculated as shown in Figure 2–10:

Figure 2–10 Sample Unearned Revenue Calculations

July's calculation

$$\frac{9/3 - \text{greatest}(8/1 \textit{ or } 7/3)}{9/3 - 7/3} = \frac{9/3 - 8/1}{9/3 - 7/3} = \frac{33 \text{ days}}{62 \text{ days}} * \text{amount}$$

August's calculation

$$\frac{9/3 - \text{greatest}(9/1 \textit{ or } 7/3)}{9/3 - 7/3} = \frac{9/3 - 9/1}{9/3 - 7/3} = \frac{2 \text{ days}}{62 \text{ days}} * \text{amount}$$

3 G/L Reporting Case Studies

This chapter shows an example of how revenue is reported for Oracle Communications Billing and Revenue Management (BRM) general ledger (G/L) reports.

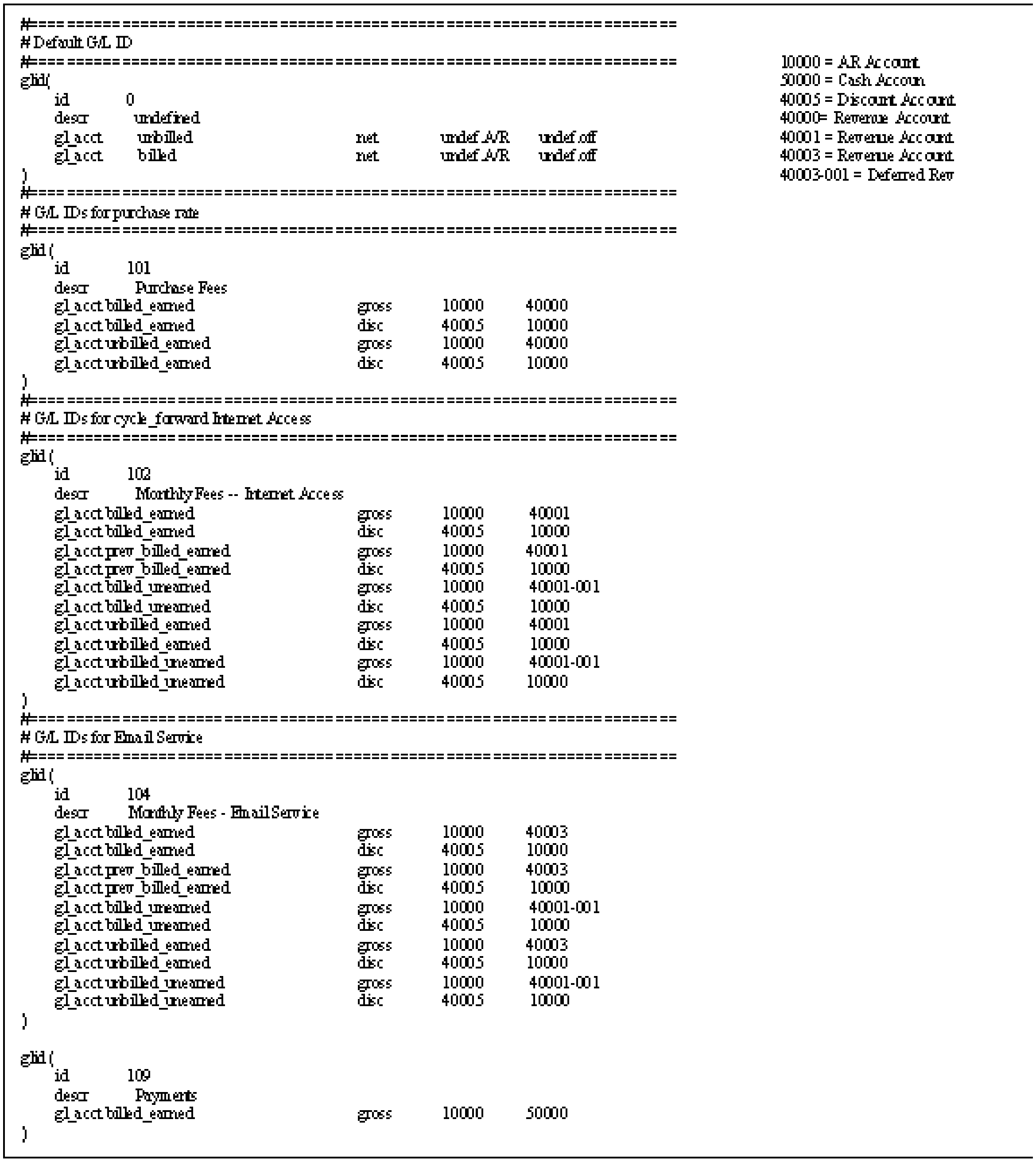
For more information, see the following:

- About Collecting General Ledger Data
- Creating General Ledger Reports

About G/L Case Studies

The following case studies show the results of various plans on customer accounts. Each study contains a short description of the plan followed by actual outputs from running the **pin_ledger_report** utility. The default **pin_glid** file has been modified to contain some G/L accounts as shown in Figure 3–1.

Figure 3–1 Modified pin_glid File with G/L Entries



```
#=====================================================================
# Default G/L ID
#=====================================================================          10000 = AR Account
glid(                                                                            50000 = Cash Accoun
     id          0                                                               40005 = Discount Account
     descr        undefined                                                      40000= Revenue Account
     gl_acct      unbilled                 net        undef A/R     undef.off    40001 = Revenue Account
     gl_acct      billed                   net        undef A/R     undef.off    40003 = Revenue Account
)                                                                                40003-001 = Deferred Rev
#=====================================================================
# G/L IDs for purchase rate
#=====================================================================
glid (
     id          101
     descr        Purchase Fees
     gl_acct billed_earned                 gross      10000       40000
     gl_acct billed_earned                 disc       40005       10000
     gl_acct unbilled_earned               gross      10000       40000
     gl_acct unbilled_earned               disc       40005       10000
)
#=====================================================================
# G/L IDs for cycle_forward Internet Access
#=====================================================================
glid (
     id          102
     descr        Monthly Fees -- Internet Access
     gl_acct billed_earned                 gross      10000        40001
     gl_acct billed_earned                 disc       40005        10000
     gl_acct prev_billed_earned            gross      10000       40001
     gl_acct prev_billed_earned            disc       40005        10000
     gl_acct billed_unearned               gross      10000        40001-001
     gl_acct billed_unearned               disc       40005        10000
     gl_acct unbilled_earned               gross      10000        40001
     gl_acct unbilled_earned               disc       40005        10000
     gl_acct unbilled_unearned             gross      10000        40001-001
     gl_acct unbilled_unearned             disc       40005       10000
)
#=====================================================================
# G/L IDs for Email Service
#=====================================================================
glid (
     id          104
     descr       Monthly Fees - Email Service
     gl_acct billed_earned                 gross      10000       40003
     gl_acct billed_earned                 disc       40005       10000
     gl_acct prev_billed_earned            gross      10000       40003
     gl_acct prev_billed_earned            disc       40005        10000
     gl_acct billed_unearned               gross      10000        40001-001
     gl_acct billed_unearned               disc       40005        10000
     gl_acct unbilled_earned               gross      10000       40003
     gl_acct unbilled_earned               disc       40005       10000
     gl_acct unbilled_unearned             gross      10000        40001-001
     gl_acct unbilled_unearned             disc       40005        10000
)

glid (
     id          109
     descr        Payments
     gl_acct billed_earned                 gross      10000       50000
)
```

Case Study 1

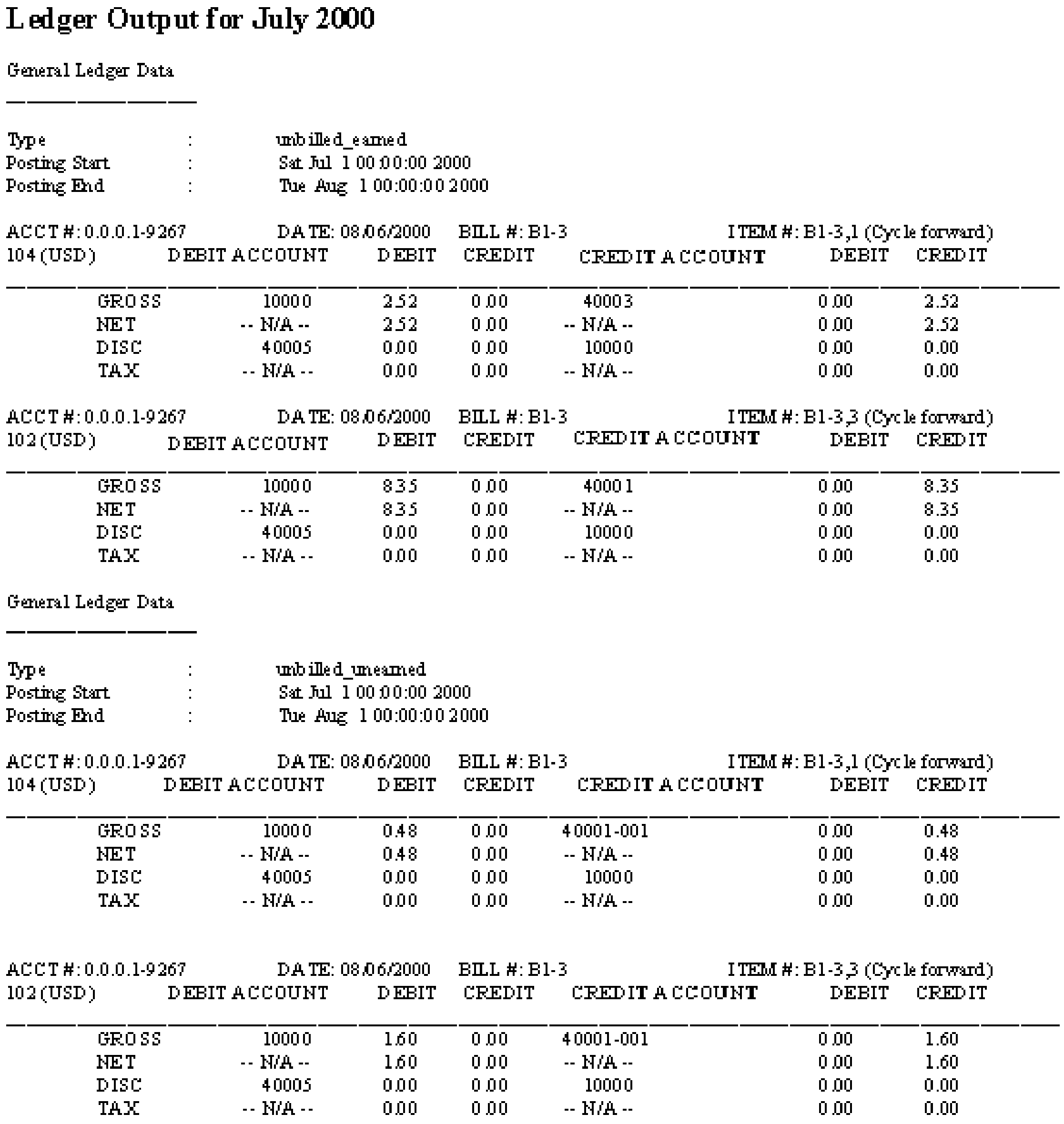
Plan details:

- Customer created: July 6, 2000
- G/L posting: Monthly G/L posting
- Purchase: Plan 1 – Measured Web access with discounts from an example **.ipl** file.
  - Monthly email fee: $3.00 (G/L ID 104)
  - Monthly pro-ratable IP fee: $9.95 (G/L ID 102)

- Accrual-based accounting

The G/L output for July displays as shown in Figure 3–2:

Figure 3–2 July G/L Output

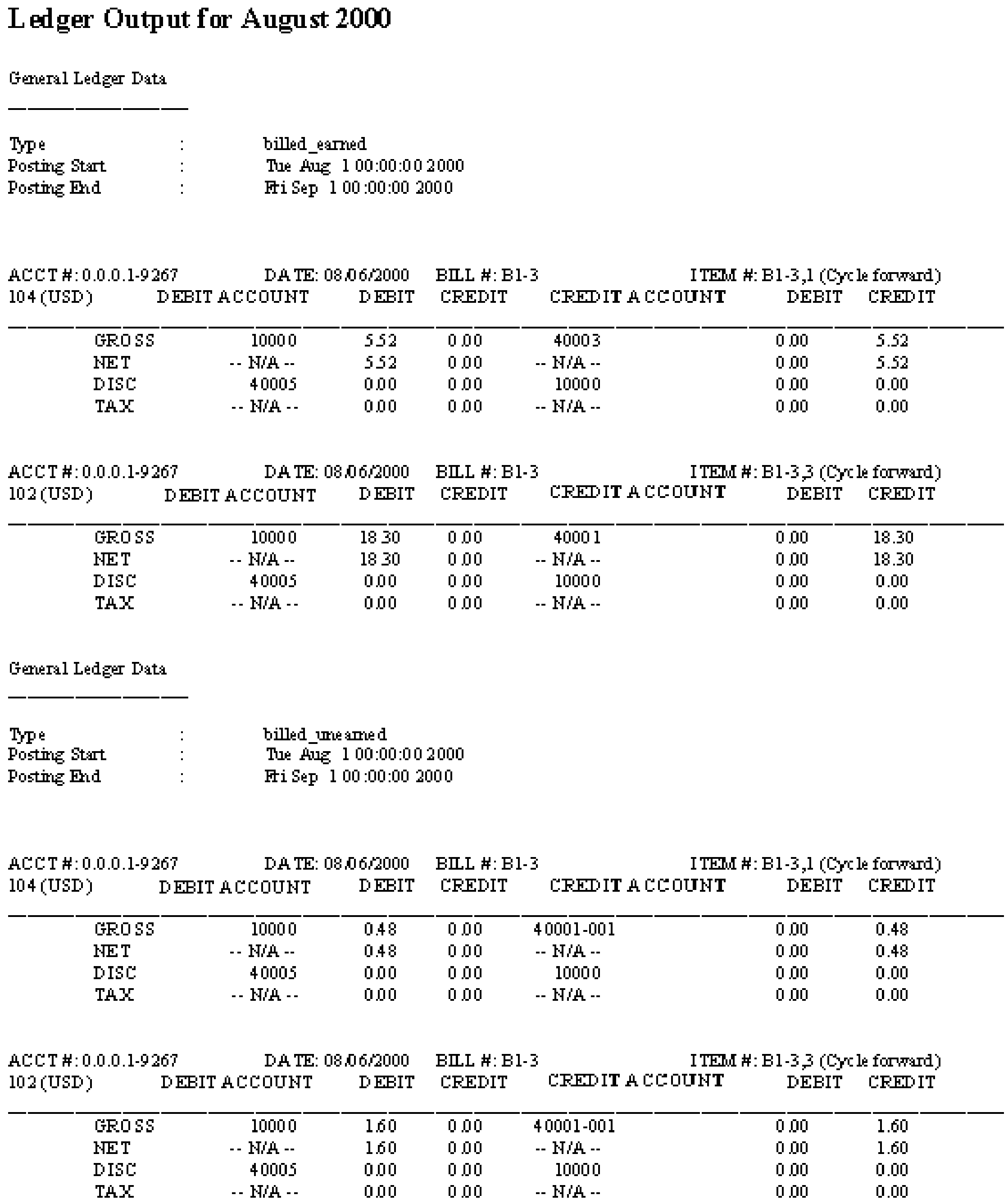


Ledger Output for July 2000

```
General Ledger Data
______________

Type                 :          unbilled_earned
Posting Start        :          Sat Jul  1 00:00:00 2000
Posting End          :          Tue Aug  1 00:00:00 2000

ACCT #: 0.0.0.1-9267        DATE: 08/06/2000    BILL #: B1-3              ITEM #: B1-3,1 (Cycle forward)
104 (USD)          DEBIT ACCOUNT      DEBIT    CREDIT    CREDIT ACCOUNT          DEBIT    CREDIT
___________________________________________________________________________________________________
          GROSS           10000        2.52      0.00        40003                  0.00       2.52
          NET           -- N/A --      2.52      0.00     -- N/A --                 0.00       2.52
          DISC            40005        0.00      0.00        10000                  0.00       0.00
          TAX           -- N/A --      0.00      0.00     -- N/A --                 0.00       0.00

ACCT #: 0.0.0.1-9267        DATE: 08/06/2000    BILL #: B1-3              ITEM #: B1-3,3 (Cycle forward)
102 (USD)          DEBIT ACCOUNT      DEBIT    CREDIT    CREDIT ACCOUNT          DEBIT    CREDIT
___________________________________________________________________________________________________
          GROSS           10000        8.35      0.00        40001                  0.00       8.35
          NET           -- N/A --      8.35      0.00     -- N/A --                 0.00       8.35
          DISC            40005        0.00      0.00        10000                  0.00       0.00
          TAX           -- N/A --      0.00      0.00     -- N/A --                 0.00       0.00

General Ledger Data
______________

Type                 :          unbilled_unearned
Posting Start        :          Sat Jul  1 00:00:00 2000
Posting End          :          Tue Aug  1 00:00:00 2000

ACCT #: 0.0.0.1-9267        DATE: 08/06/2000    BILL #: B1-3              ITEM #: B1-3,1 (Cycle forward)
104 (USD)          DEBIT ACCOUNT      DEBIT    CREDIT    CREDIT ACCOUNT          DEBIT    CREDIT
___________________________________________________________________________________________________
          GROSS           10000        0.48      0.00     40001-001                 0.00       0.48
          NET           -- N/A --      0.48      0.00     -- N/A --                 0.00       0.48
          DISC            40005        0.00      0.00        10000                  0.00       0.00
          TAX           -- N/A --      0.00      0.00     -- N/A --                 0.00       0.00


ACCT #: 0.0.0.1-9267        DATE: 08/06/2000    BILL #: B1-3              ITEM #: B1-3,3 (Cycle forward)
102 (USD)          DEBIT ACCOUNT      DEBIT    CREDIT    CREDIT ACCOUNT          DEBIT    CREDIT
___________________________________________________________________________________________________
          GROSS           10000        1.60      0.00     40001-001                 0.00       1.60
          NET           -- N/A --      1.60      0.00     -- N/A --                 0.00       1.60
          DISC            40005        0.00      0.00        10000                  0.00       0.00
          TAX           -- N/A --      0.00      0.00     -- N/A --                 0.00       0.00
```

The G/L output for August displays as shown in Figure 3–3:

Figure 3–3 August G/L Output

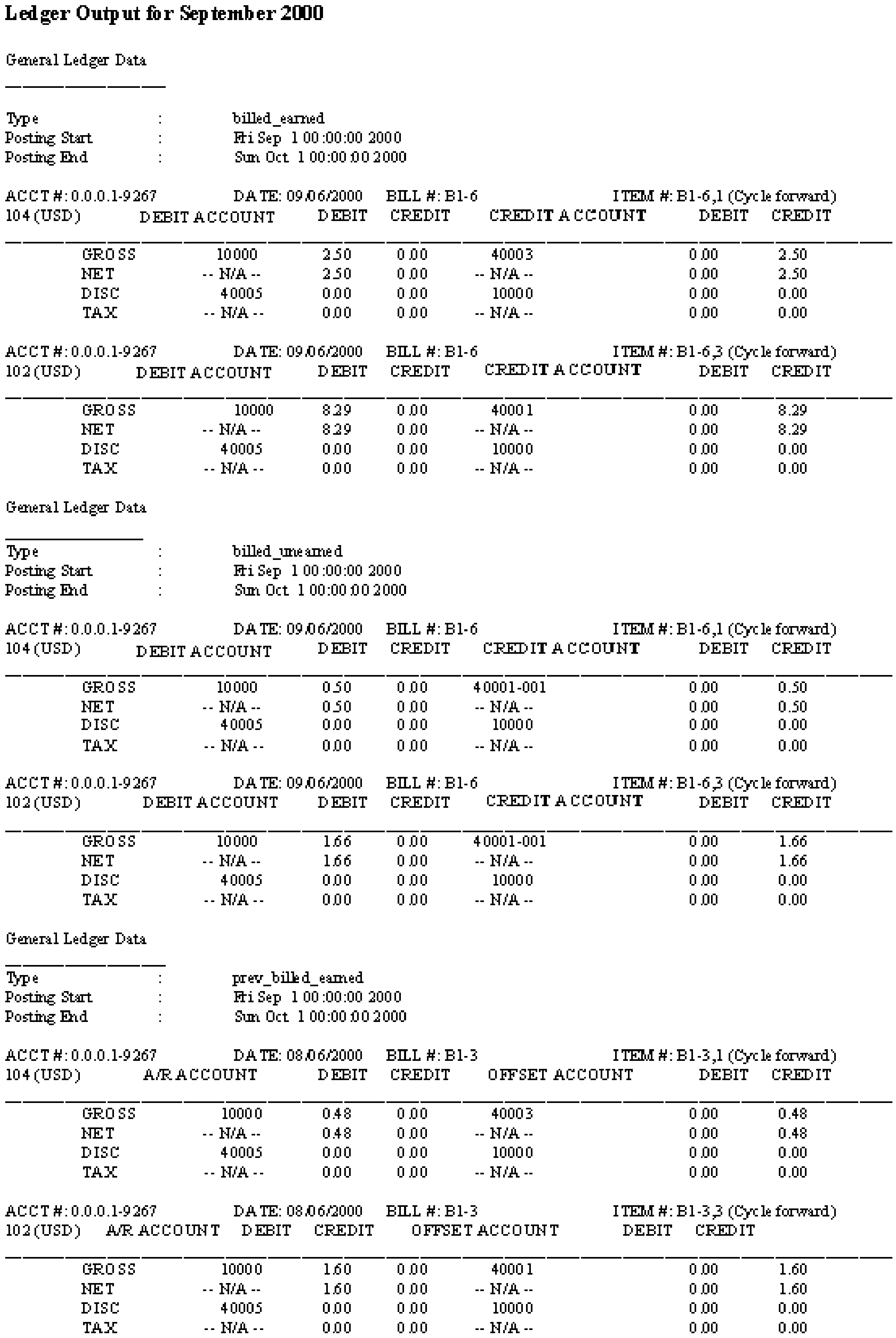


```
Ledger Output for August 2000

General Ledger Data
______________

Type                :        billed_earned
Posting Start       :        Tue Aug  1 00:00:00 2000
Posting End         :        Fri Sep  1 00:00:00 2000


ACCT #: 0.0.0.1-9267        DATE: 08/06/2000    BILL #: B1-3              ITEM #: B1-3,1 (Cycle forward)
104 (USD)          DEBIT ACCOUNT        DEBIT    CREDIT     CREDIT ACCOUNT          DEBIT     CREDIT
______________________________________________________________________________________________________
           GROSS           10000        5.52      0.00       40003                  0.00      5.52
           NET           -- N/A --      5.52      0.00     -- N/A --                0.00      5.52
           DISC            40005        0.00      0.00       10000                  0.00      0.00
           TAX           -- N/A --      0.00      0.00     -- N/A --                0.00      0.00

ACCT #: 0.0.0.1-9267        DATE: 08/06/2000    BILL #: B1-3              ITEM #: B1-3,3 (Cycle forward)
102 (USD)          DEBIT ACCOUNT        DEBIT    CREDIT     CREDIT ACCOUNT          DEBIT     CREDIT
______________________________________________________________________________________________________
           GROSS           10000       18.30      0.00       40001                  0.00     18.30
           NET           -- N/A --     18.30      0.00     -- N/A --                0.00     18.30
           DISC            40005        0.00      0.00       10000                  0.00      0.00
           TAX           -- N/A --      0.00      0.00     -- N/A --                0.00      0.00

General Ledger Data
______________

Type                :        billed_unearned
Posting Start       :        Tue Aug  1 00:00:00 2000
Posting End         :        Fri Sep  1 00:00:00 2000


ACCT #: 0.0.0.1-9267        DATE: 08/06/2000    BILL #: B1-3              ITEM #: B1-3,1 (Cycle forward)
104 (USD)          DEBIT ACCOUNT        DEBIT    CREDIT     CREDIT ACCOUNT          DEBIT     CREDIT
______________________________________________________________________________________________________
           GROSS           10000        0.48      0.00     40001-001                0.00      0.48
           NET           -- N/A --      0.48      0.00     -- N/A --                0.00      0.48
           DISC            40005        0.00      0.00       10000                  0.00      0.00
           TAX           -- N/A --      0.00      0.00     -- N/A --                0.00      0.00

ACCT #: 0.0.0.1-9267        DATE: 08/06/2000    BILL #: B1-3              ITEM #: B1-3,3 (Cycle forward)
102 (USD)          DEBIT ACCOUNT        DEBIT    CREDIT     CREDIT ACCOUNT          DEBIT     CREDIT
______________________________________________________________________________________________________
           GROSS           10000        1.60      0.00     40001-001                0.00      1.60
           NET           -- N/A --      1.60      0.00     -- N/A --                0.00      1.60
           DISC            40005        0.00      0.00       10000                  0.00      0.00
           TAX           -- N/A --      0.00      0.00     -- N/A --                0.00      0.00
```

The unbilled revenue for the duration of July 1 through August 1 appears on the billed report for the duration of August 1 through September 1. To do reversals, you need to reverse the data in your company's accounting software.

Figure 3–4 is an example of the September G/L report:

Figure 3–4 September G/L Report



```
Ledger Output for September 2000

General Ledger Data
___________________

Type                 :          billed_earned
Posting Start        :          Fri Sep  1 00:00:00 2000
Posting End          :          Sun Oct  1 00:00:00 2000

ACCT #: 0.0.0.1-9267          DATE: 09/06/2000    BILL #: B1-6               ITEM #: B1-6,1 (Cycle forward)
104 (USD)        DEBIT ACCOUNT       DEBIT    CREDIT     CREDIT ACCOUNT         DEBIT    CREDIT
______________________________________________________________________________________________________
            GROSS           10000        2.50     0.00       40003                  0.00       2.50
            NET           -- N/A --      2.50     0.00     -- N/A --                0.00       2.50
            DISC            40005        0.00     0.00       10000                  0.00       0.00
            TAX           -- N/A --      0.00     0.00     -- N/A --                0.00       0.00

ACCT #: 0.0.0.1-9267          DATE: 09/06/2000    BILL #: B1-6               ITEM #: B1-6,3 (Cycle forward)
102 (USD)        DEBIT ACCOUNT       DEBIT    CREDIT     CREDIT ACCOUNT         DEBIT    CREDIT
______________________________________________________________________________________________________
            GROSS           10000        8.29     0.00       40001                  0.00       8.29
            NET           -- N/A --      8.29     0.00     -- N/A --                0.00       8.29
            DISC            40005        0.00     0.00       10000                  0.00       0.00
            TAX           -- N/A --      0.00     0.00     -- N/A --                0.00       0.00

General Ledger Data
___________________
Type                 :          billed_unearned
Posting Start        :          Fri Sep  1 00:00:00 2000
Posting End          :          Sun Oct  1 00:00:00 2000

ACCT #: 0.0.0.1-9267          DATE: 09/06/2000    BILL #: B1-6               ITEM #: B1-6,1 (Cycle forward)
104 (USD)        DEBIT ACCOUNT       DEBIT    CREDIT     CREDIT ACCOUNT         DEBIT    CREDIT
______________________________________________________________________________________________________
            GROSS           10000        0.50     0.00     40001-001                0.00       0.50
            NET           -- N/A --      0.50     0.00     -- N/A --                0.00       0.50
            DISC            40005        0.00     0.00       10000                  0.00       0.00
            TAX           -- N/A --      0.00     0.00     -- N/A --                0.00       0.00

ACCT #: 0.0.0.1-9267          DATE: 09/06/2000    BILL #: B1-6               ITEM #: B1-6,3 (Cycle forward)
102 (USD)        DEBIT ACCOUNT       DEBIT    CREDIT     CREDIT ACCOUNT         DEBIT    CREDIT
______________________________________________________________________________________________________
            GROSS           10000        1.66     0.00     40001-001                0.00       1.66
            NET           -- N/A --      1.66     0.00     -- N/A --                0.00       1.66
            DISC            40005        0.00     0.00       10000                  0.00       0.00
            TAX           -- N/A --      0.00     0.00     -- N/A --                0.00       0.00

General Ledger Data
___________________
Type                 :          prev_billed_earned
Posting Start        :          Fri Sep  1 00:00:00 2000
Posting End          :          Sun Oct  1 00:00:00 2000

ACCT #: 0.0.0.1-9267          DATE: 08/06/2000    BILL #: B1-3               ITEM #: B1-3,1 (Cycle forward)
104 (USD)          A/R ACCOUNT       DEBIT    CREDIT     OFFSET ACCOUNT         DEBIT    CREDIT
______________________________________________________________________________________________________
            GROSS           10000        0.48     0.00       40003                  0.00       0.48
            NET           -- N/A --      0.48     0.00     -- N/A --                0.00       0.48
            DISC            40005        0.00     0.00       10000                  0.00       0.00
            TAX           -- N/A --      0.00     0.00     -- N/A --                0.00       0.00

ACCT #: 0.0.0.1-9267          DATE: 08/06/2000    BILL #: B1-3               ITEM #: B1-3,3 (Cycle forward)
102 (USD)    A/R ACCOUNT   DEBIT    CREDIT     OFFSET ACCOUNT       DEBIT    CREDIT
______________________________________________________________________________________________________
            GROSS           10000        1.60     0.00       40001                  0.00       1.60
            NET           -- N/A --      1.60     0.00     -- N/A --                0.00       1.60
            DISC            40005        0.00     0.00       10000                  0.00       0.00
            TAX           -- N/A --      0.00     0.00     -- N/A --                0.00       0.00
```

From this report on, the monthly fees of $9.95 and $3.00 become earned revenue for all successive months. However, because the monthly fee is pro-ratable, the amount depends on the number of days per month. Therefore, the amount earned in September is prorated for 30 days (25/30 * 9.95 = 8.29). The amount that was reported as unearned revenue in August becomes earned revenue in September.

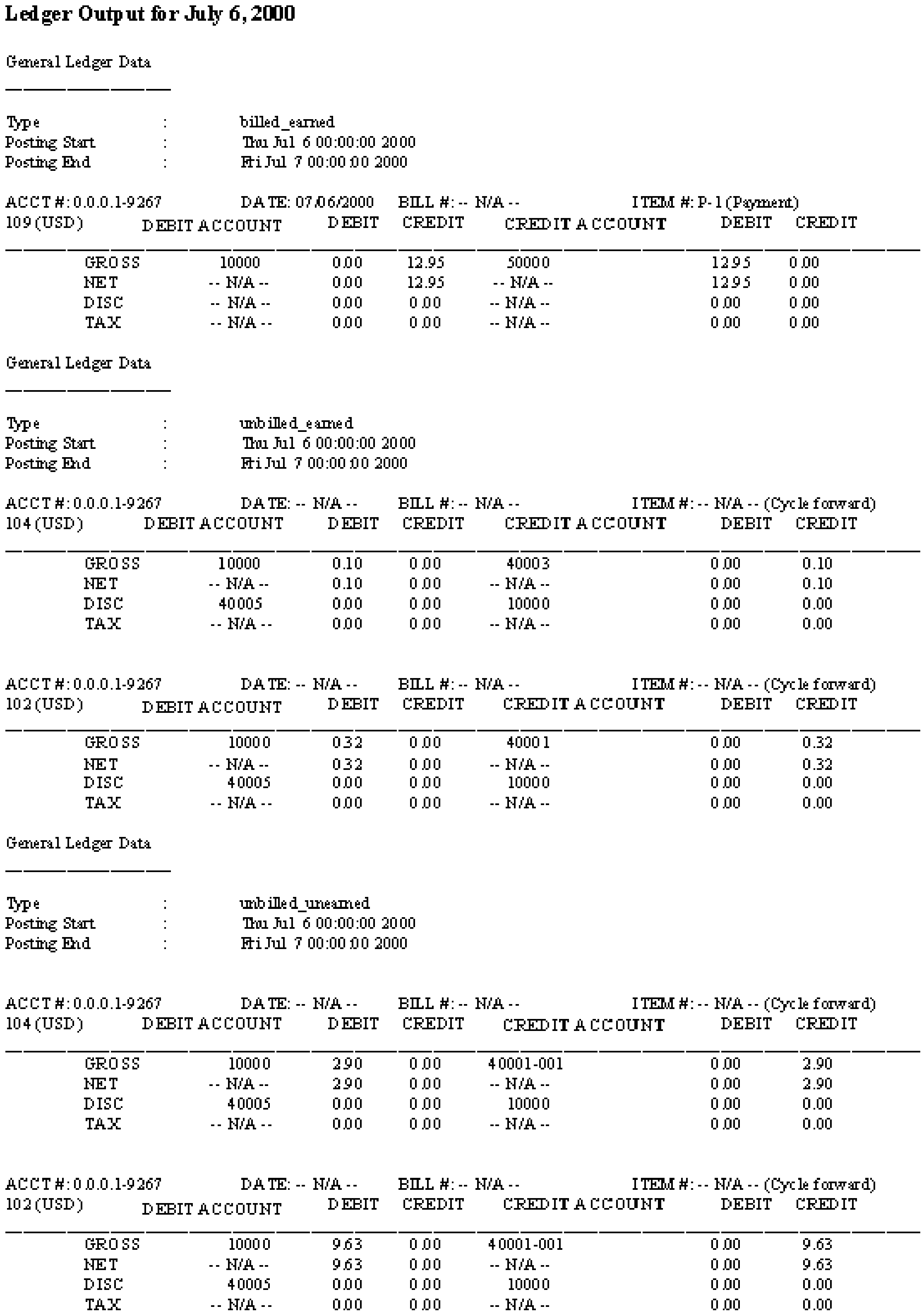
Case Study 2

Plan details:

- Customer created: July 6, 2000
- G/L posting: Daily G/L posting
- Purchase: Plan 1 – Measured Web access with discounts from an example **.ipl** file.
  - Monthly email fee: $3.00 (G/L ID 104)
  - Monthly pro-ratable IP fee: $9.95 (G/L ID 102)
- Accrual-based accounting

The G/L report for July 6 displays as shown in Figure 3–5:

Figure 3–5 July 6 G/L Report



Ledger Output for July 6, 2000

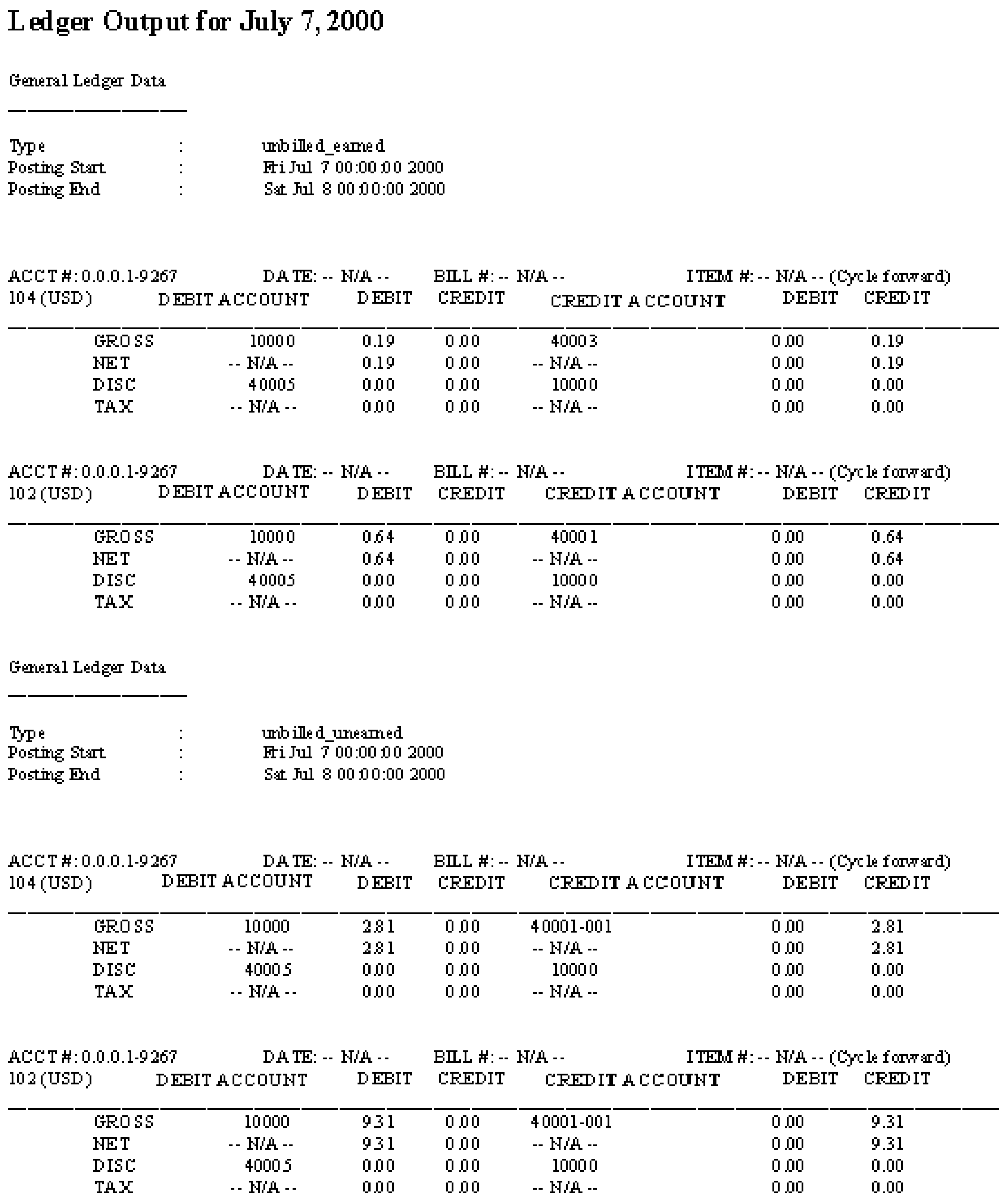
```
General Ledger Data
_______________

Type                    :          billed_earned
Posting Start           :          Thu Jul  6 00:00:00 2000
Posting End             :          Fri Jul  7 00:00:00 2000

ACCT #: 0.0.0.1-9267          DATE: 07/06/2000    BILL #: -- N/A --           ITEM #: P-1 (Payment)
109 (USD)          DEBIT ACCOUNT       DEBIT    CREDIT      CREDIT ACCOUNT        DEBIT    CREDIT
_____________________________________________________________________________________________________
            GROSS         10000        0.00     12.95        50000                12.95     0.00
            NET         -- N/A --      0.00     12.95      -- N/A --              12.95     0.00
            DISC        -- N/A --      0.00      0.00      -- N/A --               0.00     0.00
            TAX         -- N/A --      0.00      0.00      -- N/A --               0.00     0.00

General Ledger Data
_______________

Type                    :          unbilled_earned
Posting Start           :          Thu Jul  6 00:00:00 2000
Posting End             :          Fri Jul  7 00:00:00 2000

ACCT #: 0.0.0.1-9267          DATE: -- N/A --     BILL #: -- N/A --           ITEM #: -- N/A -- (Cycle forward)
104 (USD)          DEBIT ACCOUNT       DEBIT    CREDIT      CREDIT ACCOUNT        DEBIT    CREDIT
_____________________________________________________________________________________________________
            GROSS         10000        0.10      0.00        40003                 0.00      0.10
            NET         -- N/A --      0.10      0.00      -- N/A --               0.00      0.10
            DISC          40005        0.00      0.00        10000                 0.00      0.00
            TAX         -- N/A --      0.00      0.00      -- N/A --               0.00      0.00


ACCT #: 0.0.0.1-9267          DATE: -- N/A --     BILL #: -- N/A --           ITEM #: -- N/A -- (Cycle forward)
102 (USD)          DEBIT ACCOUNT       DEBIT    CREDIT      CREDIT ACCOUNT        DEBIT    CREDIT
_____________________________________________________________________________________________________
            GROSS          10000       0.32      0.00        40001                 0.00      0.32
            NET         -- N/A --      0.32      0.00      -- N/A --               0.00      0.32
            DISC           40005       0.00      0.00        10000                 0.00      0.00
            TAX         -- N/A --      0.00      0.00      -- N/A --               0.00      0.00

General Ledger Data
_______________

Type                    :          unbilled_unearned
Posting Start           :          Thu Jul  6 00:00:00 2000
Posting End             :          Fri Jul  7 00:00:00 2000


ACCT #: 0.0.0.1-9267          DATE: -- N/A --     BILL #: -- N/A --           ITEM #: -- N/A -- (Cycle forward)
104 (USD)          DEBIT ACCOUNT       DEBIT    CREDIT      CREDIT ACCOUNT        DEBIT    CREDIT
_____________________________________________________________________________________________________
            GROSS          10000       2.90      0.00      40001-001               0.00      2.90
            NET         -- N/A --      2.90      0.00      -- N/A --               0.00      2.90
            DISC           40005       0.00      0.00        10000                 0.00      0.00
            TAX         -- N/A --      0.00      0.00      -- N/A --               0.00      0.00


ACCT #: 0.0.0.1-9267          DATE: -- N/A --     BILL #: -- N/A --           ITEM #: -- N/A -- (Cycle forward)
102 (USD)          DEBIT ACCOUNT       DEBIT    CREDIT      CREDIT ACCOUNT        DEBIT    CREDIT
_____________________________________________________________________________________________________
            GROSS          10000       9.63      0.00      40001-001               0.00      9.63
            NET         -- N/A --      9.63      0.00      -- N/A --               0.00      9.63
            DISC           40005       0.00      0.00        10000                 0.00      0.00
            TAX         -- N/A --      0.00      0.00      -- N/A --               0.00      0.00
```

The full payment was received for the cycle forward fees because the customer account was created with the credit card payment method.

The cycle forward fee remains unbilled until the first billing cycle on August 6. In the G/L reports for July 7 and July 15, BRM decreases the **unbilled_unearned** revenue and increases the **unbilled_unearned** revenue amounts.

Figure 3–6 shows the July 7 G/L report.

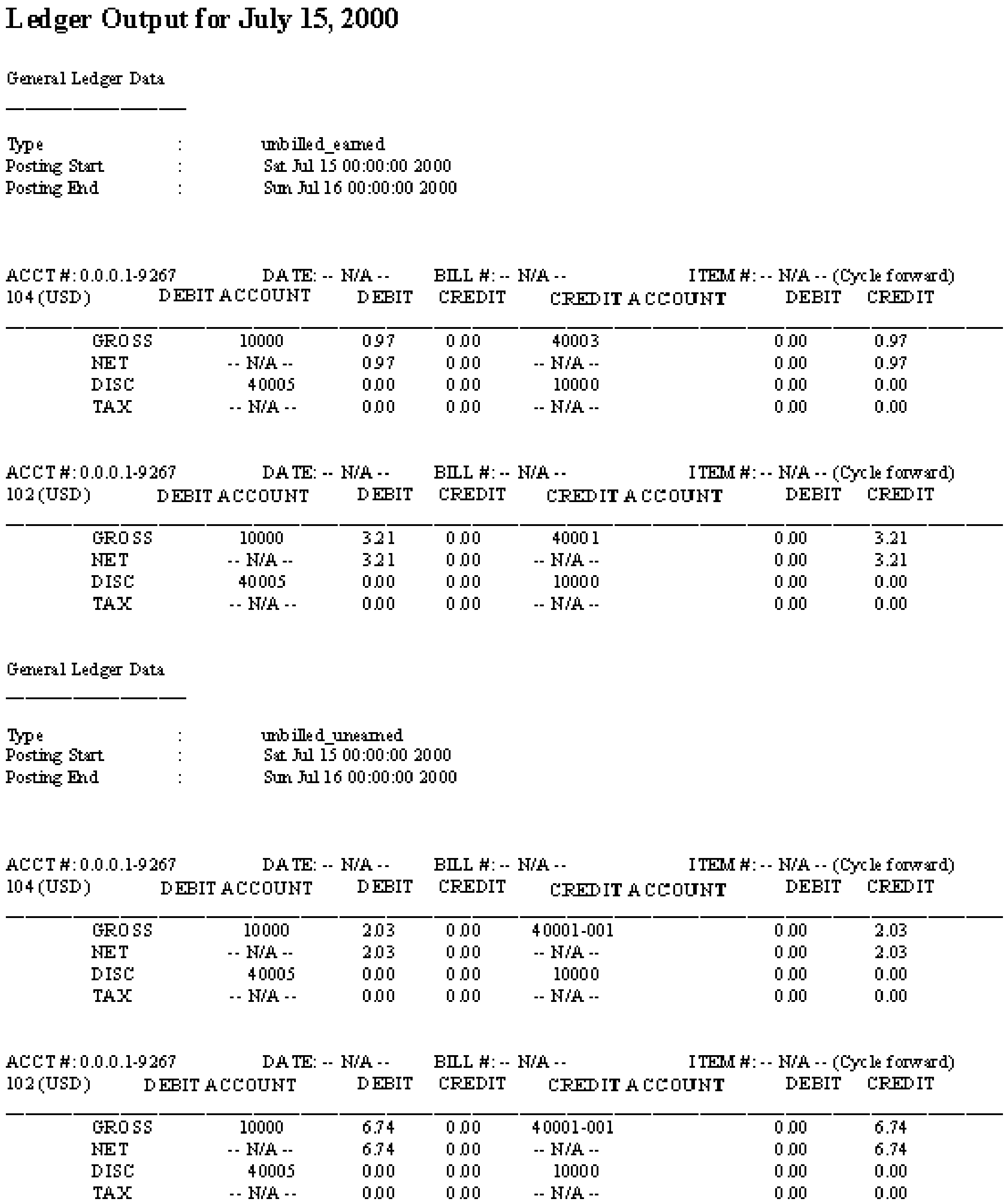
Figure 3–6 July 7 G/L Report



```
Ledger Output for July 7, 2000

General Ledger Data
___________________

Type              :        unbilled_earned
Posting Start     :        Fri Jul  7 00:00:00 2000
Posting End       :        Sat Jul  8 00:00:00 2000


ACCT #: 0.0.0.1-9267      DATE: -- N/A --    BILL #: -- N/A --          ITEM #: -- N/A -- (Cycle forward)
104 (USD)         DEBIT ACCOUNT      DEBIT   CREDIT    CREDIT ACCOUNT          DEBIT    CREDIT
_______________________________________________________________________________________________
         GROSS            10000       0.19    0.00       40003                  0.00     0.19
         NET           -- N/A --      0.19    0.00    -- N/A --                 0.00     0.19
         DISC             40005       0.00    0.00       10000                  0.00     0.00
         TAX           -- N/A --      0.00    0.00    -- N/A --                 0.00     0.00

ACCT #: 0.0.0.1-9267      DATE: -- N/A --    BILL #: -- N/A --          ITEM #: -- N/A -- (Cycle forward)
102 (USD)         DEBIT ACCOUNT      DEBIT   CREDIT    CREDIT ACCOUNT          DEBIT    CREDIT
_______________________________________________________________________________________________
         GROSS            10000       0.64    0.00       40001                  0.00     0.64
         NET           -- N/A --      0.64    0.00    -- N/A --                 0.00     0.64
         DISC             40005       0.00    0.00       10000                  0.00     0.00
         TAX           -- N/A --      0.00    0.00    -- N/A --                 0.00     0.00

General Ledger Data
___________________

Type              :        unbilled_unearned
Posting Start     :        Fri Jul  7 00:00:00 2000
Posting End       :        Sat Jul  8 00:00:00 2000


ACCT #: 0.0.0.1-9267      DATE: -- N/A --    BILL #: -- N/A --          ITEM #: -- N/A -- (Cycle forward)
104 (USD)         DEBIT ACCOUNT      DEBIT   CREDIT    CREDIT ACCOUNT          DEBIT    CREDIT
_______________________________________________________________________________________________
         GROSS           10000        2.81    0.00    40001-001                 0.00     2.81
         NET           -- N/A --      2.81    0.00    -- N/A --                 0.00     2.81
         DISC            40005        0.00    0.00       10000                  0.00     0.00
         TAX           -- N/A --      0.00    0.00    -- N/A --                 0.00     0.00

ACCT #: 0.0.0.1-9267      DATE: -- N/A --    BILL #: -- N/A --          ITEM #: -- N/A -- (Cycle forward)
102 (USD)         DEBIT ACCOUNT      DEBIT   CREDIT    CREDIT ACCOUNT          DEBIT    CREDIT
_______________________________________________________________________________________________
         GROSS           10000        9.31    0.00    40001-001                 0.00     9.31
         NET           -- N/A --      9.31    0.00    -- N/A --                 0.00     9.31
         DISC            40005        0.00    0.00       10000                  0.00     0.00
         TAX           -- N/A --      0.00    0.00    -- N/A --                 0.00     0.00
```

Figure 3–7 shows the July 15 G/L report.

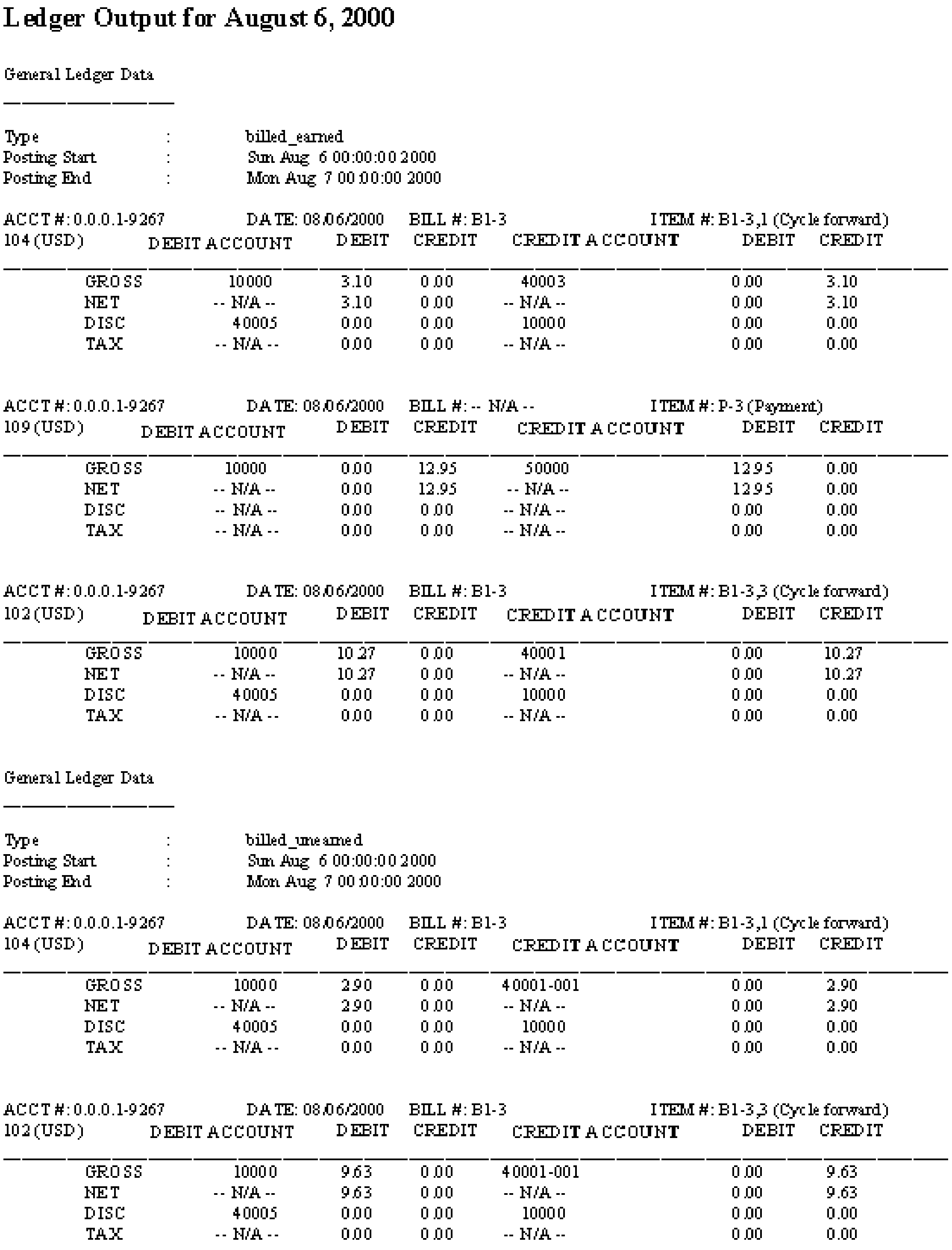
Figure 3–7 July 15 G/L Report



```
Ledger Output for July 15, 2000

General Ledger Data
_______________

Type              :        unbilled_earned
Posting Start     :        Sat Jul 15 00:00:00 2000
Posting End       :        Sun Jul 16 00:00:00 2000


ACCT #: 0.0.0.1-9267        DATE: -- N/A --    BILL #: -- N/A --          ITEM #: -- N/A -- (Cycle forward)
104 (USD)          DEBIT ACCOUNT      DEBIT   CREDIT    CREDIT ACCOUNT        DEBIT    CREDIT
______________________________________________________________________________________________
           GROSS          10000        0.97     0.00        40003               0.00      0.97
           NET         -- N/A --       0.97     0.00     -- N/A --              0.00      0.97
           DISC           40005        0.00     0.00        10000               0.00      0.00
           TAX         -- N/A --       0.00     0.00     -- N/A --              0.00      0.00

ACCT #: 0.0.0.1-9267        DATE: -- N/A --    BILL #: -- N/A --          ITEM #: -- N/A -- (Cycle forward)
102 (USD)          DEBIT ACCOUNT      DEBIT   CREDIT    CREDIT ACCOUNT        DEBIT    CREDIT
______________________________________________________________________________________________
           GROSS          10000        3.21     0.00        40001               0.00      3.21
           NET         -- N/A --       3.21     0.00     -- N/A --              0.00      3.21
           DISC          40005         0.00     0.00        10000               0.00      0.00
           TAX         -- N/A --       0.00     0.00     -- N/A --              0.00      0.00

General Ledger Data
_______________

Type              :        unbilled_unearned
Posting Start     :        Sat Jul 15 00:00:00 2000
Posting End       :        Sun Jul 16 00:00:00 2000


ACCT #: 0.0.0.1-9267        DATE: -- N/A --    BILL #: -- N/A --          ITEM #: -- N/A -- (Cycle forward)
104 (USD)          DEBIT ACCOUNT      DEBIT   CREDIT    CREDIT ACCOUNT        DEBIT    CREDIT
______________________________________________________________________________________________
           GROSS          10000        2.03     0.00     40001-001              0.00      2.03
           NET         -- N/A --       2.03     0.00     -- N/A --              0.00      2.03
           DISC           40005        0.00     0.00        10000               0.00      0.00
           TAX         -- N/A --       0.00     0.00     -- N/A --              0.00      0.00

ACCT #: 0.0.0.1-9267        DATE: -- N/A --    BILL #: -- N/A --          ITEM #: -- N/A -- (Cycle forward)
102 (USD)          DEBIT ACCOUNT      DEBIT   CREDIT    CREDIT ACCOUNT        DEBIT    CREDIT
______________________________________________________________________________________________
           GROSS          10000        6.74     0.00     40001-001              0.00      6.74
           NET         -- N/A --       6.74     0.00     -- N/A --              0.00      6.74
           DISC           40005        0.00     0.00        10000               0.00      0.00
           TAX         -- N/A --       0.00     0.00     -- N/A --              0.00      0.00
```

When billing is run on August 6, the amounts change from unbilled to billed as shown in Figure 3–8. The entire amount is now earned and reported in the **billed_earned** revenue type.

Figure 3–8 August 6 G/L Report

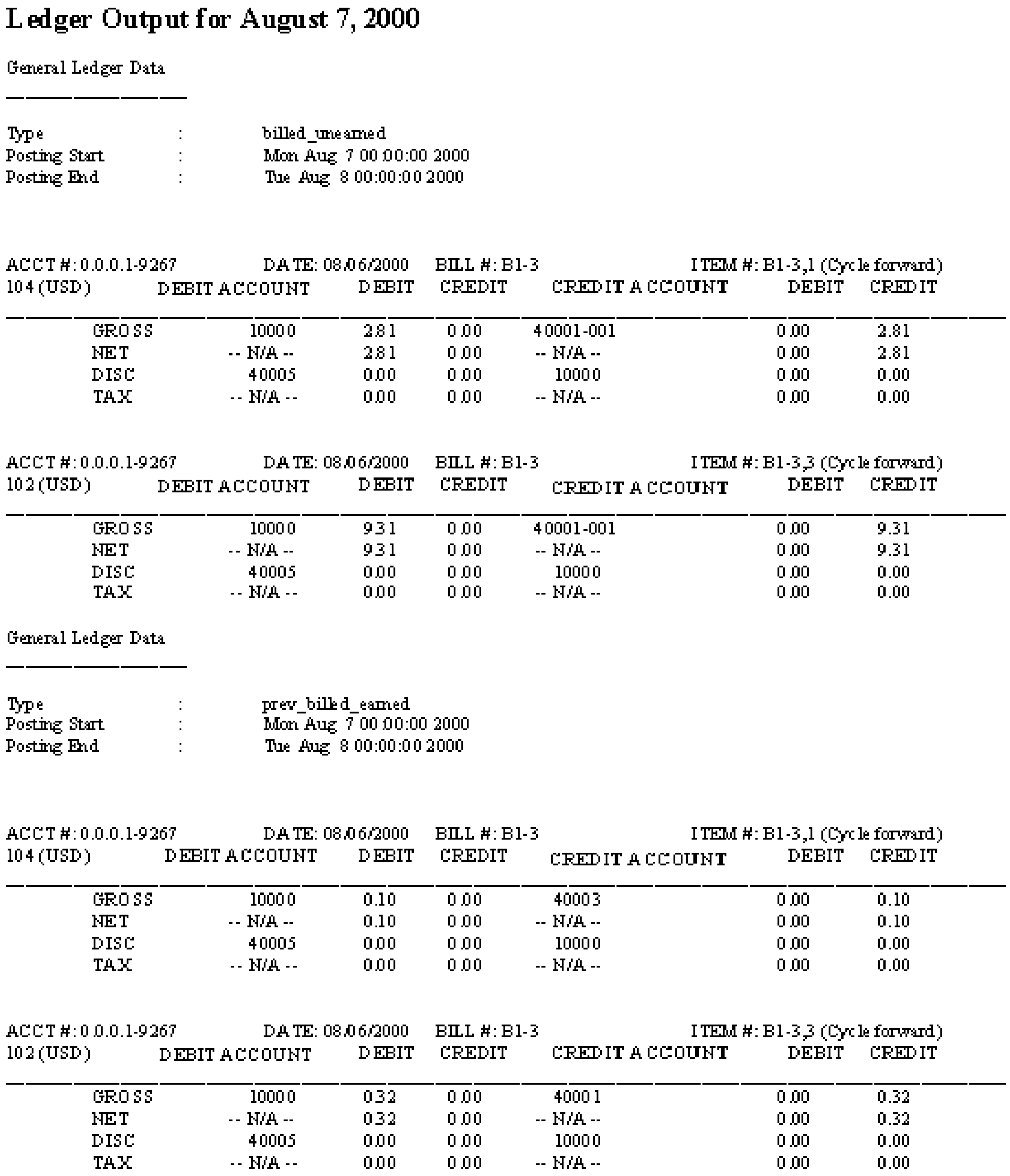


Ledger Output for August 6, 2000

```
General Ledger Data
_______________

Type                 :        billed_earned
Posting Start        :        Sun Aug  6 00:00:00 2000
Posting End          :        Mon Aug  7 00:00:00 2000

ACCT #: 0.0.0.1-9267       DATE: 08/06/2000   BILL #: B1-3          ITEM #: B1-3,1 (Cycle forward)
104 (USD)          DEBIT ACCOUNT       DEBIT    CREDIT    CREDIT ACCOUNT        DEBIT    CREDIT
____________________________________________________________________________________________________
           GROSS        10000          3.10      0.00       40003                0.00      3.10
           NET        -- N/A --        3.10      0.00     -- N/A --              0.00      3.10
           DISC         40005          0.00      0.00       10000                0.00      0.00
           TAX        -- N/A --        0.00      0.00     -- N/A --              0.00      0.00

ACCT #: 0.0.0.1-9267       DATE: 08/06/2000   BILL #: -- N/A --     ITEM #: P-3 (Payment)
109 (USD)          DEBIT ACCOUNT       DEBIT    CREDIT    CREDIT ACCOUNT        DEBIT    CREDIT
____________________________________________________________________________________________________
           GROSS        10000          0.00     12.95       50000               12.95      0.00
           NET        -- N/A --        0.00     12.95     -- N/A --             12.95      0.00
           DISC       -- N/A --        0.00      0.00     -- N/A --              0.00      0.00
           TAX        -- N/A --        0.00      0.00     -- N/A --              0.00      0.00

ACCT #: 0.0.0.1-9267       DATE: 08/06/2000   BILL #: B1-3          ITEM #: B1-3,3 (Cycle forward)
102 (USD)          DEBIT ACCOUNT       DEBIT    CREDIT    CREDIT ACCOUNT        DEBIT    CREDIT
____________________________________________________________________________________________________
           GROSS        10000         10.27      0.00       40001                0.00     10.27
           NET        -- N/A --       10.27      0.00     -- N/A --              0.00     10.27
           DISC         40005          0.00      0.00       10000                0.00      0.00
           TAX        -- N/A --        0.00      0.00     -- N/A --              0.00      0.00

General Ledger Data
_______________

Type                 :        billed_unearned
Posting Start        :        Sun Aug  6 00:00:00 2000
Posting End          :        Mon Aug  7 00:00:00 2000

ACCT #: 0.0.0.1-9267       DATE: 08/06/2000   BILL #: B1-3          ITEM #: B1-3,1 (Cycle forward)
104 (USD)          DEBIT ACCOUNT       DEBIT    CREDIT    CREDIT ACCOUNT        DEBIT    CREDIT
____________________________________________________________________________________________________
           GROSS        10000          2.90      0.00     40001-001              0.00      2.90
           NET        -- N/A --        2.90      0.00     -- N/A --              0.00      2.90
           DISC         40005          0.00      0.00       10000                0.00      0.00
           TAX        -- N/A --        0.00      0.00     -- N/A --              0.00      0.00

ACCT #: 0.0.0.1-9267       DATE: 08/06/2000   BILL #: B1-3          ITEM #: B1-3,3 (Cycle forward)
102 (USD)          DEBIT ACCOUNT       DEBIT    CREDIT    CREDIT ACCOUNT        DEBIT    CREDIT
____________________________________________________________________________________________________
           GROSS        10000          9.63      0.00     40001-001              0.00      9.63
           NET        -- N/A --        9.63      0.00     -- N/A --              0.00      9.63
           DISC         40005          0.00      0.00       10000                0.00      0.00
           TAX        -- N/A --        0.00      0.00     -- N/A --              0.00      0.00
```

From the report run on August 7, the earned revenue is now posted under the **prev_billed_earned** revenue type as shown in Figure 3–9.

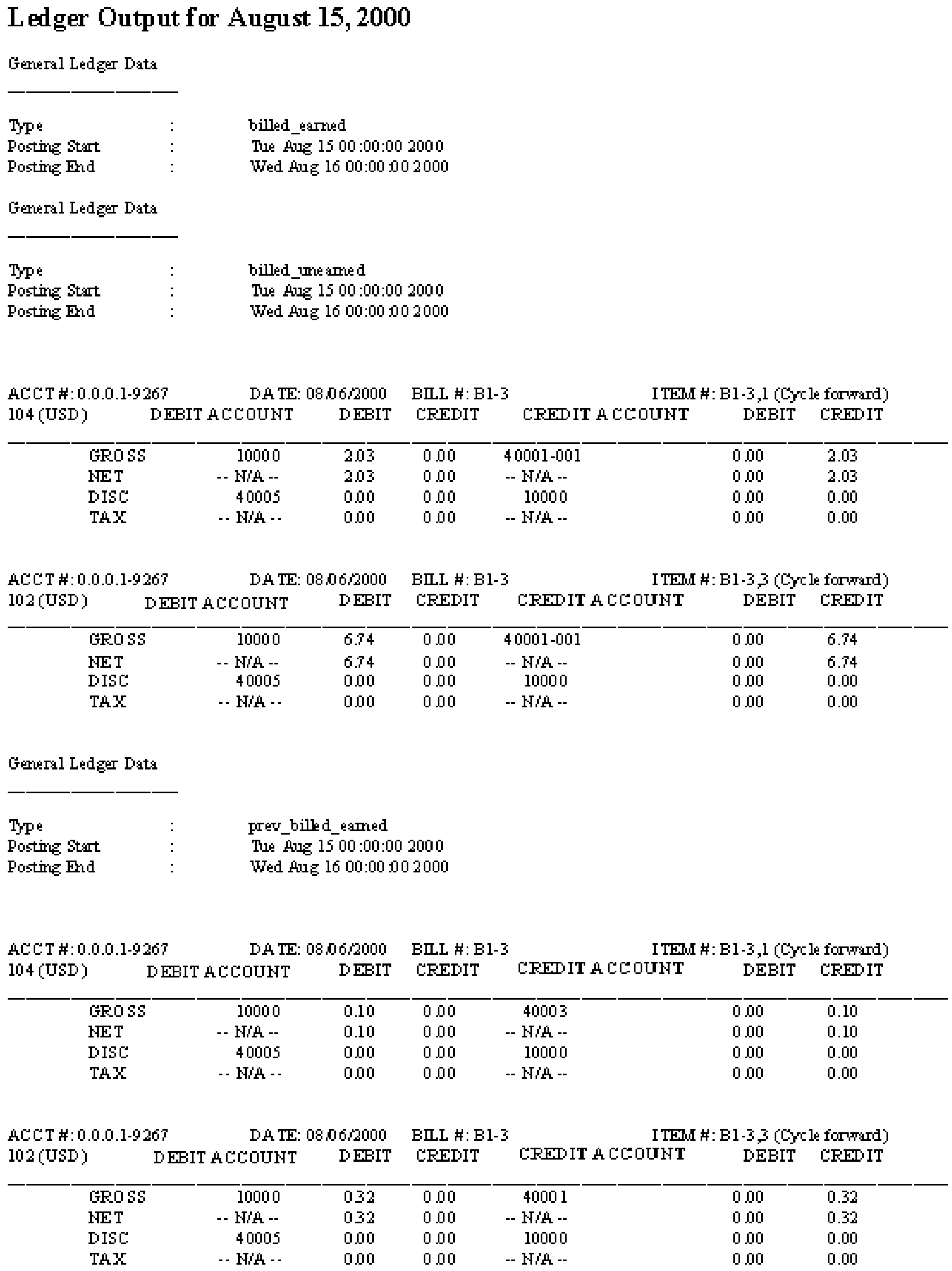
Figure 3–9 August 7 G/L Report



```
Ledger Output for August 7, 2000

General Ledger Data
_______________

Type                 :          billed_unearned
Posting Start        :          Mon Aug 7 00:00:00 2000
Posting End          :          Tue Aug 8 00:00:00 2000


ACCT #: 0.0.0.1-9267          DATE: 08/06/2000    BILL #: B1-3              ITEM #: B1-3,1 (Cycle forward)
104 (USD)          DEBIT ACCOUNT       DEBIT    CREDIT     CREDIT ACCOUNT         DEBIT    CREDIT
__________________________________________________________________________________________________
            GROSS             10000     2.81     0.00     40001-001               0.00      2.81
            NET             -- N/A --   2.81     0.00     -- N/A --               0.00      2.81
            DISC              40005     0.00     0.00       10000                 0.00      0.00
            TAX             -- N/A --   0.00     0.00     -- N/A --               0.00      0.00

ACCT #: 0.0.0.1-9267          DATE: 08/06/2000    BILL #: B1-3              ITEM #: B1-3,3 (Cycle forward)
102 (USD)          DEBIT ACCOUNT       DEBIT    CREDIT     CREDIT ACCOUNT         DEBIT    CREDIT
__________________________________________________________________________________________________
            GROSS             10000     9.31     0.00     40001-001               0.00      9.31
            NET             -- N/A --   9.31     0.00     -- N/A --               0.00      9.31
            DISC              40005     0.00     0.00       10000                 0.00      0.00
            TAX             -- N/A --   0.00     0.00     -- N/A --               0.00      0.00

General Ledger Data
_______________

Type                 :          prev_billed_earned
Posting Start        :          Mon Aug 7 00:00:00 2000
Posting End          :          Tue Aug 8 00:00:00 2000


ACCT #: 0.0.0.1-9267          DATE: 08/06/2000    BILL #: B1-3              ITEM #: B1-3,1 (Cycle forward)
104 (USD)          DEBIT ACCOUNT       DEBIT    CREDIT     CREDIT ACCOUNT         DEBIT    CREDIT
__________________________________________________________________________________________________
            GROSS             10000     0.10     0.00       40003                 0.00      0.10
            NET             -- N/A --   0.10     0.00     -- N/A --               0.00      0.10
            DISC              40005     0.00     0.00       10000                 0.00      0.00
            TAX             -- N/A --   0.00     0.00     -- N/A --               0.00      0.00

ACCT #: 0.0.0.1-9267          DATE: 08/06/2000    BILL #: B1-3              ITEM #: B1-3,3 (Cycle forward)
102 (USD)          DEBIT ACCOUNT       DEBIT    CREDIT     CREDIT ACCOUNT         DEBIT    CREDIT
__________________________________________________________________________________________________
            GROSS             10000     0.32     0.00       40001                 0.00      0.32
            NET             -- N/A --   0.32     0.00     -- N/A --               0.00      0.32
            DISC              40005     0.00     0.00       10000                 0.00      0.00
            TAX             -- N/A --   0.00     0.00     -- N/A --               0.00      0.00
```

Figure 3–10 shows the August 15 G/L report.

Figure 3–10 August 15 G/L Report



```
Ledger Output for August 15, 2000

General Ledger Data
_______________

Type              :        billed_earned
Posting Start     :        Tue Aug 15 00:00:00 2000
Posting End       :        Wed Aug 16 00:00:00 2000

General Ledger Data
_______________

Type              :        billed_unearned
Posting Start     :        Tue Aug 15 00:00:00 2000
Posting End       :        Wed Aug 16 00:00:00 2000


ACCT #: 0.0.0.1-9267        DATE: 08/06/2000    BILL #: B1-3              ITEM #: B1-3,1 (Cycle forward)
104 (USD)         DEBIT ACCOUNT        DEBIT    CREDIT     CREDIT ACCOUNT          DEBIT    CREDIT
______________________________________________________________________________________________________
            GROSS             10000       2.03      0.00    40001-001                 0.00      2.03
            NET             -- N/A --     2.03      0.00   -- N/A --                  0.00      2.03
            DISC              40005       0.00      0.00      10000                   0.00      0.00
            TAX             -- N/A --     0.00      0.00   -- N/A --                  0.00      0.00

ACCT #: 0.0.0.1-9267        DATE: 08/06/2000    BILL #: B1-3              ITEM #: B1-3,3 (Cycle forward)
102 (USD)         DEBIT ACCOUNT        DEBIT    CREDIT     CREDIT ACCOUNT          DEBIT    CREDIT
______________________________________________________________________________________________________
            GROSS             10000       6.74      0.00    40001-001                 0.00      6.74
            NET             -- N/A --     6.74      0.00   -- N/A --                  0.00      6.74
            DISC              40005       0.00      0.00      10000                   0.00      0.00
            TAX             -- N/A --     0.00      0.00   -- N/A --                  0.00      0.00

General Ledger Data
_______________

Type              :        prev_billed_earned
Posting Start     :        Tue Aug 15 00:00:00 2000
Posting End       :        Wed Aug 16 00:00:00 2000


ACCT #: 0.0.0.1-9267        DATE: 08/06/2000    BILL #: B1-3              ITEM #: B1-3,1 (Cycle forward)
104 (USD)         DEBIT ACCOUNT        DEBIT    CREDIT     CREDIT ACCOUNT          DEBIT    CREDIT
______________________________________________________________________________________________________
            GROSS             10000       0.10      0.00      40003                   0.00      0.10
            NET             -- N/A --     0.10      0.00   -- N/A --                  0.00      0.10
            DISC              40005       0.00      0.00      10000                   0.00      0.00
            TAX             -- N/A --     0.00      0.00   -- N/A --                  0.00      0.00

ACCT #: 0.0.0.1-9267        DATE: 08/06/2000    BILL #: B1-3              ITEM #: B1-3,3 (Cycle forward)
102 (USD)         DEBIT ACCOUNT        DEBIT    CREDIT     CREDIT ACCOUNT          DEBIT    CREDIT
______________________________________________________________________________________________________
            GROSS             10000       0.32      0.00      40001                   0.00      0.32
            NET             -- N/A --     0.32      0.00   -- N/A --                  0.00      0.32
            DISC              40005       0.00      0.00      10000                   0.00      0.00
            TAX             -- N/A --     0.00      0.00   -- N/A --                  0.00      0.00
```

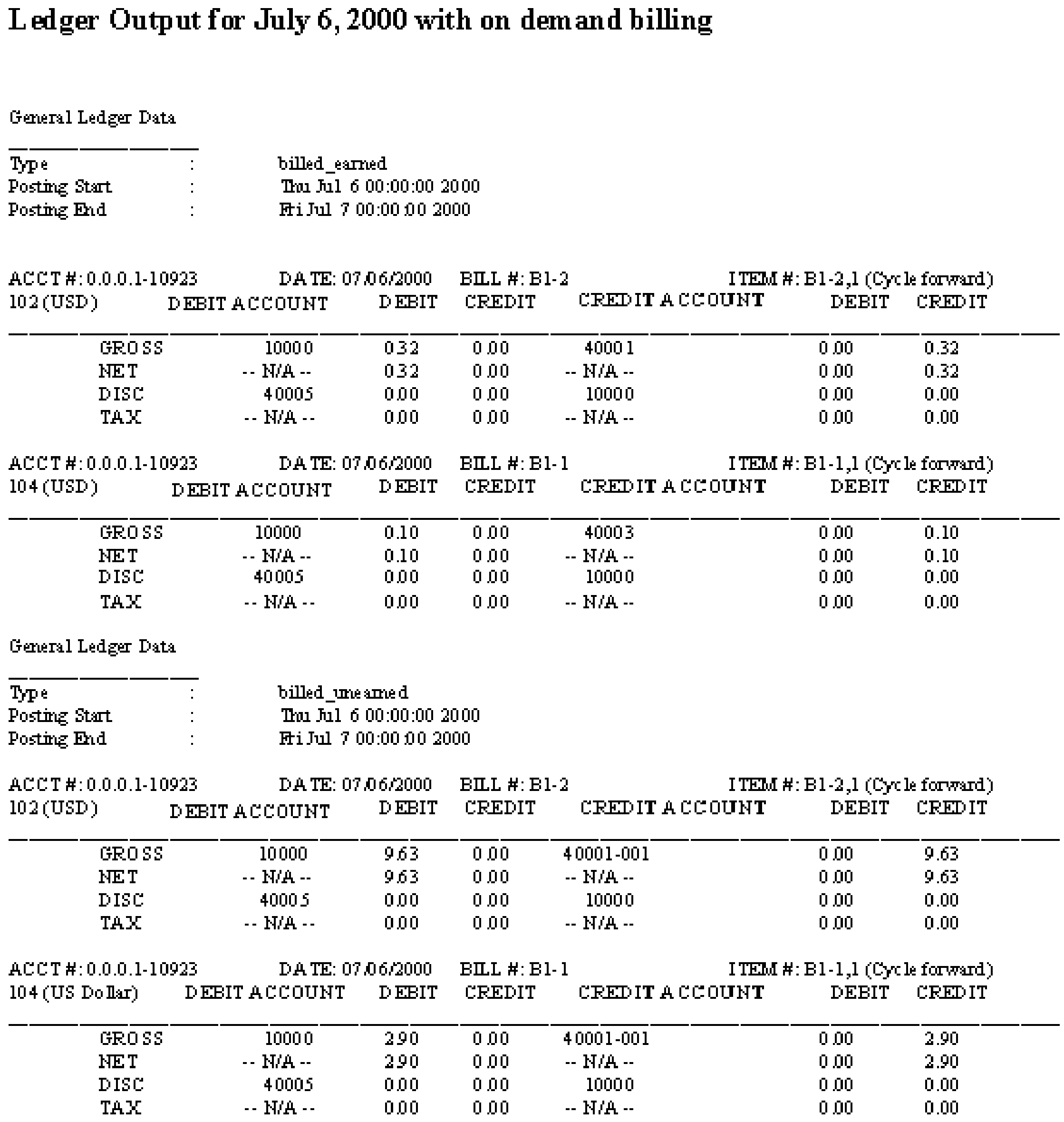
Case Study 3

Plan details:

- Customer created: July 6, 2000
- G/L posting: Daily G/L posting
- Purchase: Plan 1 – Measured Web access with discounts from an example **.ipl** file.
  - Monthly email fee: $3.00 (G/L ID 104)
  - Monthly pro-ratable IP fee: $9.95 (G/L ID 102)
- Accrual-based accounting
- On-demand billing: You generate a bill immediately after the deal is purchased. For more information, see the discussion of on-demand billing in *BRM Configuring and Running Billing*.

The G/L report run for July 6 shows that the revenue is earned on the same day that the deal was purchased as seen in Figure 3–11:

Figure 3–11 July 6 G/L Report

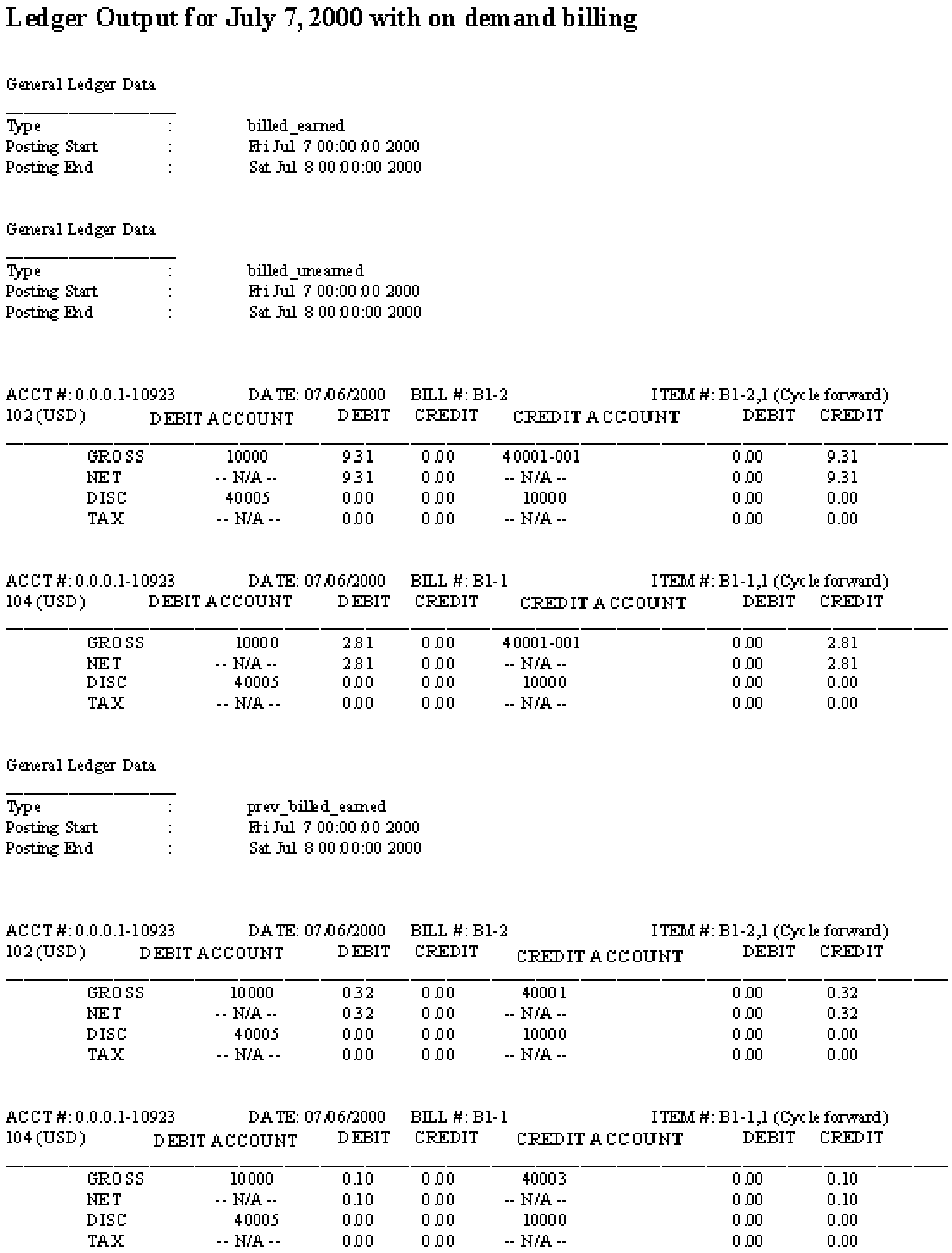


```
Ledger Output for July 6, 2000 with on demand billing


General Ledger Data
_______________
Type                 :          billed_earned
Posting Start        :          Thu Jul  6 00:00:00 2000
Posting End          :          Fri Jul  7 00:00:00 2000

ACCT #: 0.0.0.1-10923        DATE: 07/06/2000    BILL #: B1-2              ITEM #: B1-2,1 (Cycle forward)
102 (USD)          DEBIT ACCOUNT      DEBIT    CREDIT    CREDIT ACCOUNT         DEBIT    CREDIT
______________________________________________________________________________________________________
          GROSS            10000      0.32     0.00       40001                  0.00      0.32
          NET            -- N/A --    0.32     0.00     -- N/A --                0.00      0.32
          DISC             40005      0.00     0.00       10000                  0.00      0.00
          TAX            -- N/A --    0.00     0.00     -- N/A --                0.00      0.00

ACCT #: 0.0.0.1-10923        DATE: 07/06/2000    BILL #: B1-1              ITEM #: B1-1,1 (Cycle forward)
104 (USD)          DEBIT ACCOUNT      DEBIT    CREDIT    CREDIT ACCOUNT         DEBIT    CREDIT
______________________________________________________________________________________________________
          GROSS            10000      0.10     0.00       40003                  0.00      0.10
          NET            -- N/A --    0.10     0.00     -- N/A --                0.00      0.10
          DISC             40005      0.00     0.00       10000                  0.00      0.00
          TAX            -- N/A --    0.00     0.00     -- N/A --                0.00      0.00

General Ledger Data
_______________
Type                 :          billed_unearned
Posting Start        :          Thu Jul  6 00:00:00 2000
Posting End          :          Fri Jul  7 00:00:00 2000

ACCT #: 0.0.0.1-10923        DATE: 07/06/2000    BILL #: B1-2              ITEM #: B1-2,1 (Cycle forward)
102 (USD)          DEBIT ACCOUNT      DEBIT    CREDIT    CREDIT ACCOUNT         DEBIT    CREDIT
______________________________________________________________________________________________________
          GROSS            10000      9.63     0.00     40001-001                0.00      9.63
          NET            -- N/A --    9.63     0.00     -- N/A --                0.00      9.63
          DISC             40005      0.00     0.00       10000                  0.00      0.00
          TAX            -- N/A --    0.00     0.00     -- N/A --                0.00      0.00

ACCT #: 0.0.0.1-10923        DATE: 07/06/2000    BILL #: B1-1              ITEM #: B1-1,1 (Cycle forward)
104 (US Dollar)     DEBIT ACCOUNT     DEBIT    CREDIT    CREDIT ACCOUNT         DEBIT    CREDIT
______________________________________________________________________________________________________
          GROSS            10000      2.90     0.00     40001-001                0.00      2.90
          NET            -- N/A --    2.90     0.00     -- N/A --                0.00      2.90
          DISC             40005      0.00     0.00       10000                  0.00      0.00
          TAX            -- N/A --    0.00     0.00     -- N/A --                0.00      0.00
```

In the following example, BRM generates the bill and reports the revenue as earned when the customer purchases a deal. The following examples show the results of running G/L reports on a bi-monthly cycle with the on-demand billing feature.

Figure 3–12 shows the July 7 G/L report.

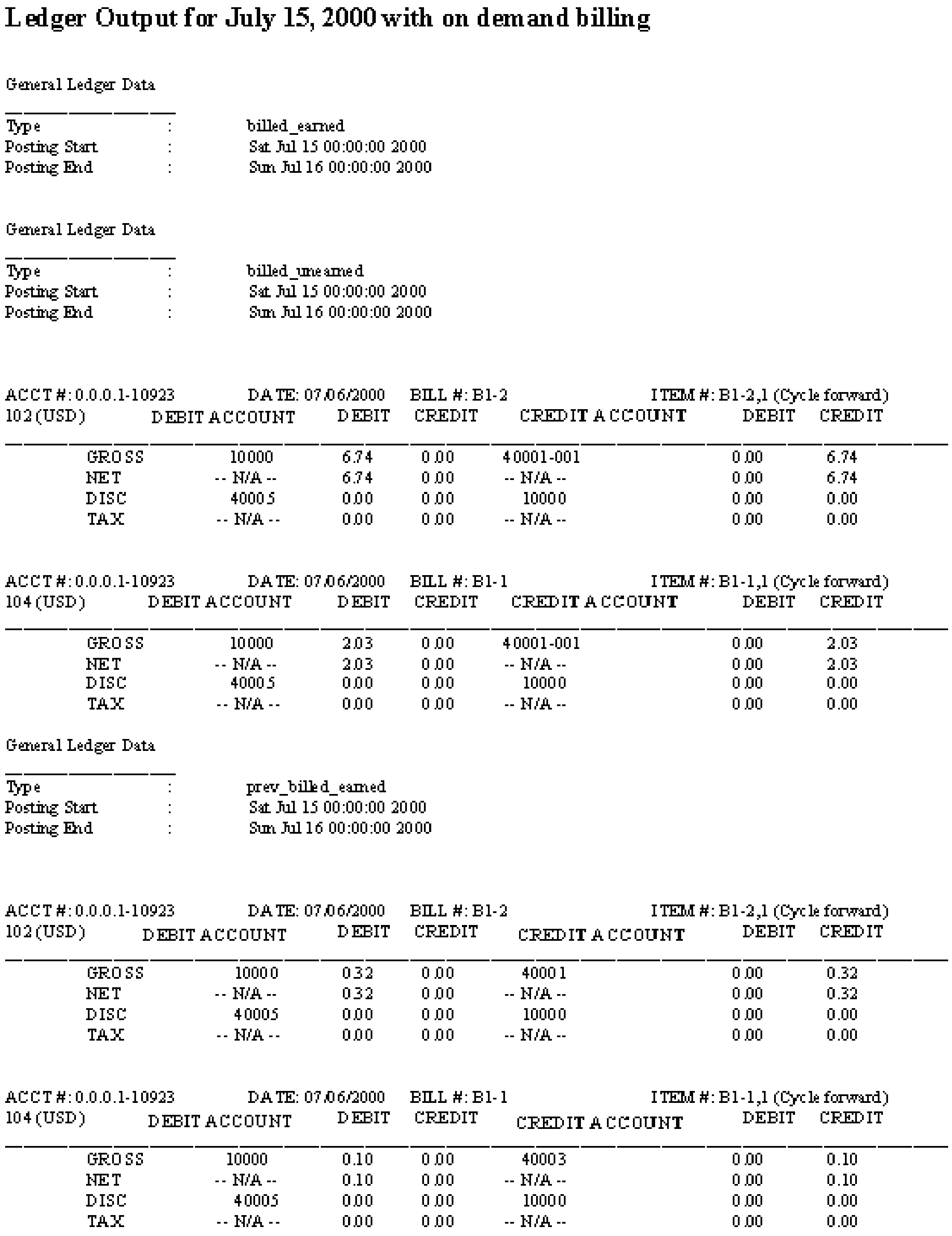
Figure 3–12 July 7 G/L Report



```
Ledger Output for July 7, 2000 with on demand billing

General Ledger Data
___________________
Type                :        billed_earned
Posting Start       :        Fri Jul  7 00:00:00 2000
Posting End         :        Sat Jul  8 00:00:00 2000

General Ledger Data
___________________
Type                :        billed_unearned
Posting Start       :        Fri Jul  7 00:00:00 2000
Posting End         :        Sat Jul  8 00:00:00 2000


ACCT #: 0.0.0.1-10923      DATE: 07/06/2000   BILL #: B1-2            ITEM #: B1-2,1 (Cycle forward)
102 (USD)        DEBIT ACCOUNT       DEBIT    CREDIT    CREDIT ACCOUNT          DEBIT     CREDIT
______________________________________________________________________________________________________
          GROSS          10000        9.31      0.00     40001-001              0.00       9.31
          NET          -- N/A --      9.31      0.00     -- N/A --              0.00       9.31
          DISC           40005        0.00      0.00       10000                0.00       0.00
          TAX          -- N/A --      0.00      0.00     -- N/A --              0.00       0.00

ACCT #: 0.0.0.1-10923      DATE: 07/06/2000   BILL #: B1-1            ITEM #: B1-1,1 (Cycle forward)
104 (USD)        DEBIT ACCOUNT       DEBIT    CREDIT    CREDIT ACCOUNT          DEBIT     CREDIT
______________________________________________________________________________________________________
          GROSS          10000        2.81      0.00     40001-001              0.00       2.81
          NET          -- N/A --      2.81      0.00     -- N/A --              0.00       2.81
          DISC           40005        0.00      0.00       10000                0.00       0.00
          TAX          -- N/A --      0.00      0.00     -- N/A --              0.00       0.00

General Ledger Data
___________________
Type                :        prev_billed_earned
Posting Start       :        Fri Jul  7 00:00:00 2000
Posting End         :        Sat Jul  8 00:00:00 2000


ACCT #: 0.0.0.1-10923      DATE: 07/06/2000   BILL #: B1-2            ITEM #: B1-2,1 (Cycle forward)
102 (USD)        DEBIT ACCOUNT       DEBIT    CREDIT    CREDIT ACCOUNT          DEBIT     CREDIT
______________________________________________________________________________________________________
          GROSS          10000        0.32      0.00       40001                0.00       0.32
          NET          -- N/A --      0.32      0.00     -- N/A --              0.00       0.32
          DISC           40005        0.00      0.00       10000                0.00       0.00
          TAX          -- N/A --      0.00      0.00     -- N/A --              0.00       0.00

ACCT #: 0.0.0.1-10923      DATE: 07/06/2000   BILL #: B1-1            ITEM #: B1-1,1 (Cycle forward)
104 (USD)        DEBIT ACCOUNT       DEBIT    CREDIT    CREDIT ACCOUNT          DEBIT     CREDIT
______________________________________________________________________________________________________
          GROSS          10000        0.10      0.00       40003                0.00       0.10
          NET          -- N/A --      0.10      0.00     -- N/A --              0.00       0.10
          DISC           40005        0.00      0.00       10000                0.00       0.00
          TAX          -- N/A --      0.00      0.00     -- N/A --              0.00       0.00
```

Figure 3–13 shows the July 15 G/L report.

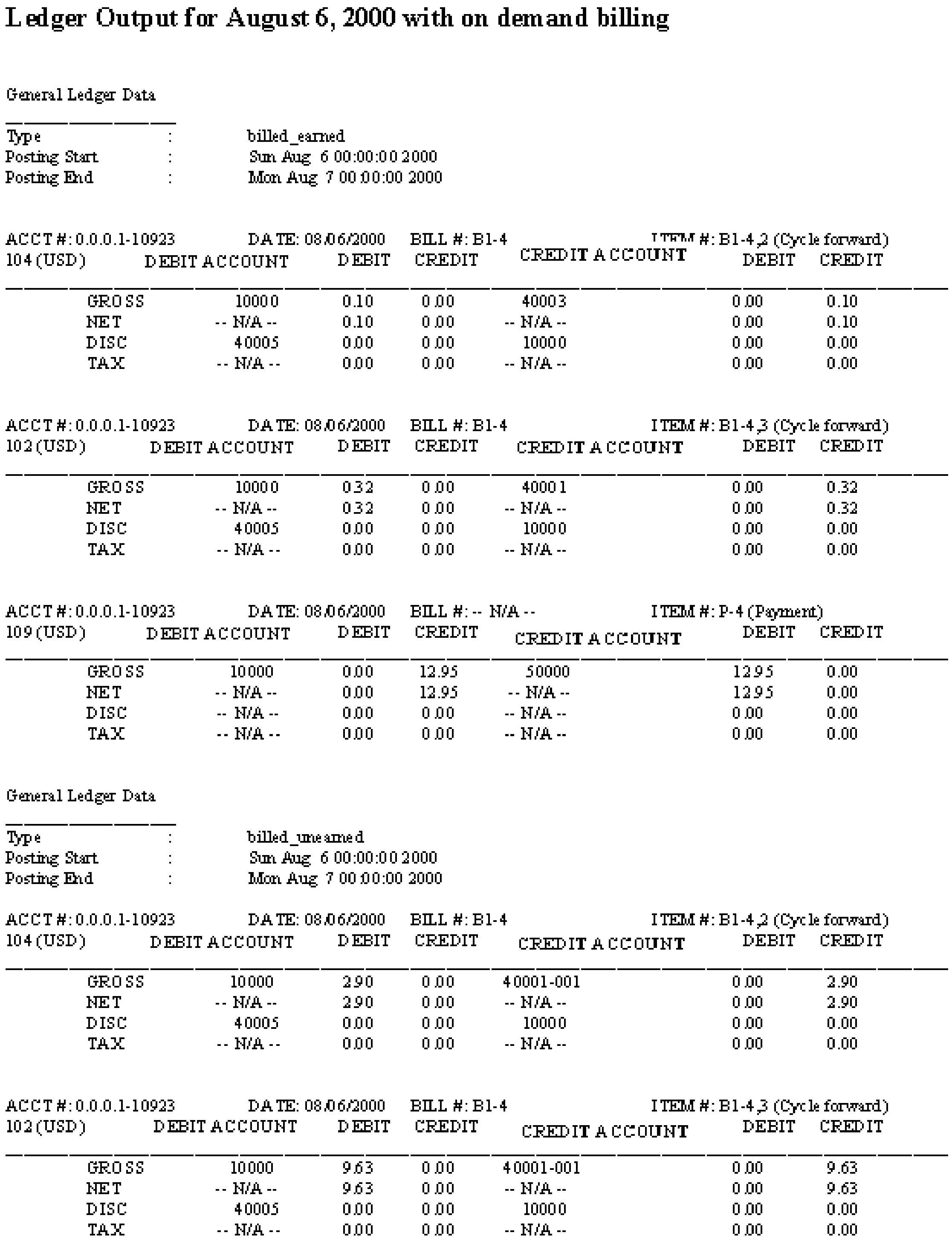
Figure 3–13 July 15 G/L Report



```
Ledger Output for July 15, 2000 with on demand billing

General Ledger Data
________________
Type                :         billed_earned
Posting Start       :         Sat Jul 15 00:00:00 2000
Posting End         :         Sun Jul 16 00:00:00 2000

General Ledger Data
________________
Type                :         billed_unearned
Posting Start       :         Sat Jul 15 00:00:00 2000
Posting End         :         Sun Jul 16 00:00:00 2000


ACCT #: 0.0.0.1-10923       DATE: 07/06/2000    BILL #: B1-2               ITEM #: B1-2,1 (Cycle forward)
102 (USD)           DEBIT ACCOUNT       DEBIT   CREDIT    CREDIT ACCOUNT          DEBIT    CREDIT
_____________________________________________________________________________________________________
         GROSS           10000           6.74    0.00    40001-001                  0.00      6.74
         NET            -- N/A --        6.74    0.00    -- N/A --                  0.00      6.74
         DISC            40005           0.00    0.00      10000                    0.00      0.00
         TAX            -- N/A --        0.00    0.00    -- N/A --                  0.00      0.00

ACCT #: 0.0.0.1-10923       DATE: 07/06/2000    BILL #: B1-1               ITEM #: B1-1,1 (Cycle forward)
104 (USD)           DEBIT ACCOUNT       DEBIT   CREDIT   CREDIT ACCOUNT           DEBIT    CREDIT
_____________________________________________________________________________________________________
         GROSS           10000           2.03    0.00    40001-001                  0.00      2.03
         NET            -- N/A --        2.03    0.00    -- N/A --                  0.00      2.03
         DISC            40005           0.00    0.00      10000                    0.00      0.00
         TAX            -- N/A --        0.00    0.00    -- N/A --                  0.00      0.00

General Ledger Data
________________
Type                :         prev_billed_earned
Posting Start       :         Sat Jul 15 00:00:00 2000
Posting End         :         Sun Jul 16 00:00:00 2000


ACCT #: 0.0.0.1-10923       DATE: 07/06/2000    BILL #: B1-2               ITEM #: B1-2,1 (Cycle forward)
102 (USD)          DEBIT ACCOUNT        DEBIT   CREDIT    CREDIT ACCOUNT          DEBIT    CREDIT
_____________________________________________________________________________________________________
         GROSS            10000          0.32    0.00      40001                    0.00      0.32
         NET            -- N/A --        0.32    0.00    -- N/A --                  0.00      0.32
         DISC             40005          0.00    0.00      10000                    0.00      0.00
         TAX            -- N/A --        0.00    0.00    -- N/A --                  0.00      0.00

ACCT #: 0.0.0.1-10923       DATE: 07/06/2000    BILL #: B1-1               ITEM #: B1-1,1 (Cycle forward)
104 (USD)           DEBIT ACCOUNT       DEBIT   CREDIT    CREDIT ACCOUNT          DEBIT    CREDIT
_____________________________________________________________________________________________________
         GROSS           10000           0.10    0.00      40003                    0.00      0.10
         NET            -- N/A --        0.10    0.00    -- N/A --                  0.00      0.10
         DISC             40005          0.00    0.00      10000                    0.00      0.00
         TAX            -- N/A --        0.00    0.00    -- N/A --                  0.00      0.00
```

Figure 3–14 shows the August 6 G/L report.

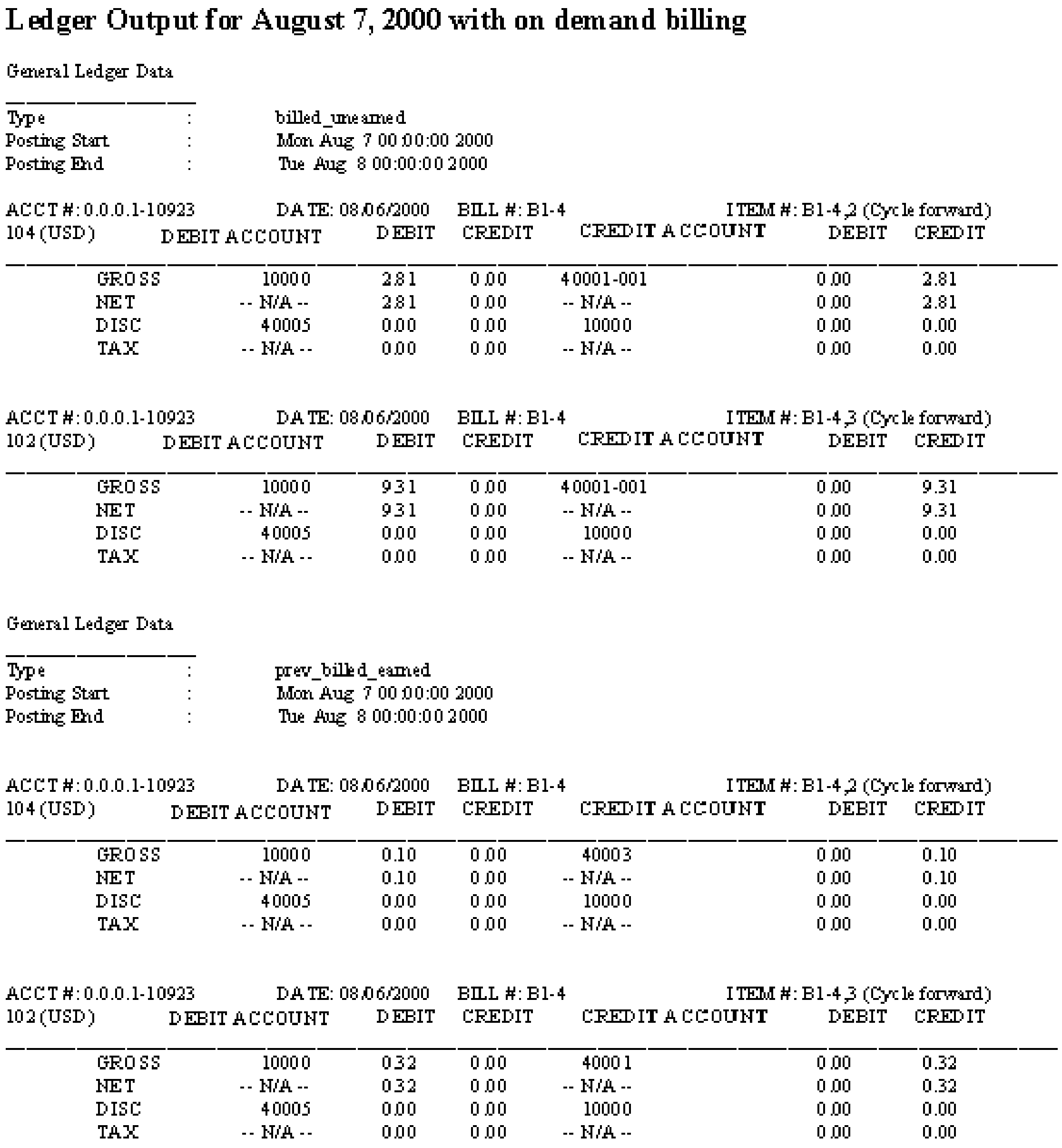
Figure 3–14 August 6 G/L Report



```
Ledger Output for August 6, 2000 with on demand billing

General Ledger Data
_______________
Type             :        billed_earned
Posting Start    :        Sun Aug  6 00:00:00 2000
Posting End      :        Mon Aug  7 00:00:00 2000

ACCT #: 0.0.0.1-10923      DATE: 08/06/2000   BILL #: B1-4                  ITEM #: B1-4,2 (Cycle forward)
104 (USD)          DEBIT ACCOUNT     DEBIT    CREDIT   CREDIT ACCOUNT        DEBIT    CREDIT
______________________________________________________________________________________________
           GROSS        10000        0.10     0.00      40003                0.00     0.10
           NET        -- N/A --      0.10     0.00    -- N/A --              0.00     0.10
           DISC         40005        0.00     0.00      10000                0.00     0.00
           TAX        -- N/A --      0.00     0.00    -- N/A --              0.00     0.00

ACCT #: 0.0.0.1-10923      DATE: 08/06/2000   BILL #: B1-4                  ITEM #: B1-4,3 (Cycle forward)
102 (USD)          DEBIT ACCOUNT     DEBIT    CREDIT   CREDIT ACCOUNT        DEBIT    CREDIT
______________________________________________________________________________________________
           GROSS        10000        0.32     0.00      40001                0.00     0.32
           NET        -- N/A --      0.32     0.00    -- N/A --              0.00     0.32
           DISC         40005        0.00     0.00      10000                0.00     0.00
           TAX        -- N/A --      0.00     0.00    -- N/A --              0.00     0.00

ACCT #: 0.0.0.1-10923      DATE: 08/06/2000   BILL #: -- N/A --             ITEM #: P-4 (Payment)
109 (USD)          DEBIT ACCOUNT     DEBIT    CREDIT   CREDIT ACCOUNT        DEBIT    CREDIT
______________________________________________________________________________________________
           GROSS        10000        0.00     12.95     50000                12.95    0.00
           NET        -- N/A --      0.00     12.95   -- N/A --              12.95    0.00
           DISC       -- N/A --      0.00     0.00    -- N/A --              0.00     0.00
           TAX        -- N/A --      0.00     0.00    -- N/A --              0.00     0.00

General Ledger Data
_______________
Type             :        billed_unearned
Posting Start    :        Sun Aug  6 00:00:00 2000
Posting End      :        Mon Aug  7 00:00:00 2000

ACCT #: 0.0.0.1-10923      DATE: 08/06/2000   BILL #: B1-4                  ITEM #: B1-4,2 (Cycle forward)
104 (USD)          DEBIT ACCOUNT     DEBIT    CREDIT   CREDIT ACCOUNT        DEBIT    CREDIT
______________________________________________________________________________________________
           GROSS        10000        2.90     0.00    40001-001              0.00     2.90
           NET        -- N/A --      2.90     0.00    -- N/A --              0.00     2.90
           DISC         40005        0.00     0.00      10000                0.00     0.00
           TAX        -- N/A --      0.00     0.00    -- N/A --              0.00     0.00

ACCT #: 0.0.0.1-10923      DATE: 08/06/2000   BILL #: B1-4                  ITEM #: B1-4,3 (Cycle forward)
102 (USD)          DEBIT ACCOUNT     DEBIT    CREDIT   CREDIT ACCOUNT        DEBIT    CREDIT
______________________________________________________________________________________________
           GROSS        10000        9.63     0.00    40001-001              0.00     9.63
           NET        -- N/A --      9.63     0.00    -- N/A --              0.00     9.63
           DISC         40005        0.00     0.00      10000                0.00     0.00
           TAX        -- N/A --      0.00     0.00    -- N/A --              0.00     0.00
```

Figure 3–15 shows the August 7 G/L report.

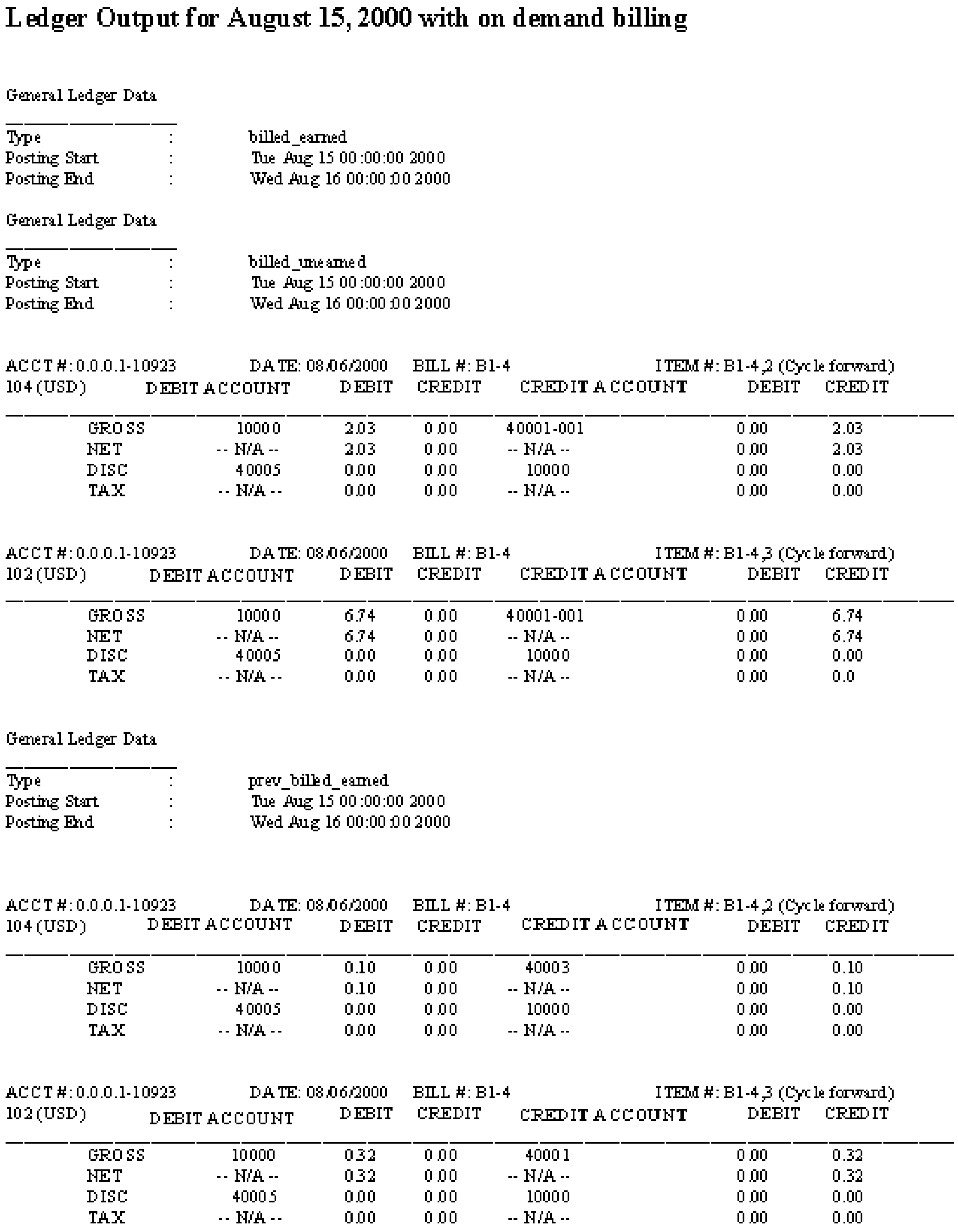
Figure 3–15 August 7 G/L Report



```
Ledger Output for August 7, 2000 with on demand billing

General Ledger Data
_______________
Type                :         billed_unearned
Posting Start       :         Mon Aug  7 00:00:00 2000
Posting End         :         Tue Aug  8 00:00:00 2000

ACCT #: 0.0.0.1-10923      DATE: 08/06/2000   BILL #: B1-4              ITEM #: B1-4,2 (Cycle forward)
104 (USD)       DEBIT ACCOUNT       DEBIT    CREDIT     CREDIT ACCOUNT        DEBIT    CREDIT
_____________________________________________________________________________________________
           GROSS          10000      2.81      0.00    40001-001               0.00      2.81
           NET          -- N/A --    2.81      0.00    -- N/A --               0.00      2.81
           DISC           40005      0.00      0.00      10000                 0.00      0.00
           TAX          -- N/A --    0.00      0.00    -- N/A --               0.00      0.00

ACCT #: 0.0.0.1-10923      DATE: 08/06/2000   BILL #: B1-4              ITEM #: B1-4,3 (Cycle forward)
102 (USD)       DEBIT ACCOUNT       DEBIT    CREDIT     CREDIT ACCOUNT        DEBIT    CREDIT
_____________________________________________________________________________________________
           GROSS          10000      9.31      0.00    40001-001               0.00      9.31
           NET          -- N/A --    9.31      0.00    -- N/A --               0.00      9.31
           DISC           40005      0.00      0.00      10000                 0.00      0.00
           TAX          -- N/A --    0.00      0.00    -- N/A --               0.00      0.00

General Ledger Data
_______________
Type                :         prev_billed_earned
Posting Start       :         Mon Aug  7 00:00:00 2000
Posting End         :         Tue Aug  8 00:00:00 2000

ACCT #: 0.0.0.1-10923      DATE: 08/06/2000   BILL #: B1-4              ITEM #: B1-4,2 (Cycle forward)
104 (USD)       DEBIT ACCOUNT       DEBIT    CREDIT     CREDIT ACCOUNT        DEBIT    CREDIT
_____________________________________________________________________________________________
           GROSS          10000      0.10      0.00      40003                 0.00      0.10
           NET          -- N/A --    0.10      0.00    -- N/A --               0.00      0.10
           DISC           40005      0.00      0.00      10000                 0.00      0.00
           TAX          -- N/A --    0.00      0.00    -- N/A --               0.00      0.00

ACCT #: 0.0.0.1-10923      DATE: 08/06/2000   BILL #: B1-4              ITEM #: B1-4,3 (Cycle forward)
102 (USD)       DEBIT ACCOUNT       DEBIT    CREDIT     CREDIT ACCOUNT        DEBIT    CREDIT
_____________________________________________________________________________________________
           GROSS          10000      0.32      0.00      40001                 0.00      0.32
           NET          -- N/A --    0.32      0.00    -- N/A --               0.00      0.32
           DISC           40005      0.00      0.00      10000                 0.00      0.00
           TAX          -- N/A --    0.00      0.00    -- N/A --               0.00      0.00
```

Figure 3–16 shows the August 15 G/L report.

Figure 3–16 August 15 G/L Report



```
Ledger Output for August 15, 2000 with on demand billing

General Ledger Data
_______________
Type              :        billed_earned
Posting Start     :        Tue Aug 15 00:00:00 2000
Posting End       :        Wed Aug 16 00:00:00 2000

General Ledger Data
_______________
Type              :        billed_unearned
Posting Start     :        Tue Aug 15 00:00:00 2000
Posting End       :        Wed Aug 16 00:00:00 2000

ACCT#: 0.0.0.1-10923       DATE: 08/06/2000    BILL #: B1-4              ITEM#: B1-4,2 (Cycle forward)
104 (USD)        DEBIT ACCOUNT        DEBIT    CREDIT     CREDIT ACCOUNT         DEBIT     CREDIT
_____________________________________________________________________________________________________
          GROSS            10000       2.03      0.00      40001-001               0.00       2.03
          NET            -- N/A --     2.03      0.00      -- N/A --               0.00       2.03
          DISC             40005       0.00      0.00        10000                 0.00       0.00
          TAX            -- N/A --     0.00      0.00      -- N/A --               0.00       0.00

ACCT#: 0.0.0.1-10923       DATE: 08/06/2000    BILL #: B1-4              ITEM#: B1-4,3 (Cycle forward)
102 (USD)        DEBIT ACCOUNT        DEBIT    CREDIT     CREDIT ACCOUNT         DEBIT     CREDIT
_____________________________________________________________________________________________________
          GROSS            10000       6.74      0.00      40001-001               0.00       6.74
          NET            -- N/A --     6.74      0.00      -- N/A --               0.00       6.74
          DISC             40005       0.00      0.00        10000                 0.00       0.00
          TAX            -- N/A --     0.00      0.00      -- N/A --               0.00       0.0

General Ledger Data
_______________
Type              :        prev_billed_earned
Posting Start     :        Tue Aug 15 00:00:00 2000
Posting End       :        Wed Aug 16 00:00:00 2000


ACCT#: 0.0.0.1-10923       DATE: 08/06/2000    BILL #: B1-4              ITEM#: B1-4,2 (Cycle forward)
104 (USD)        DEBIT ACCOUNT        DEBIT    CREDIT     CREDIT ACCOUNT         DEBIT     CREDIT
_____________________________________________________________________________________________________
          GROSS            10000       0.10      0.00        40003                 0.00       0.10
          NET            -- N/A --     0.10      0.00      -- N/A --               0.00       0.10
          DISC             40005       0.00      0.00        10000                 0.00       0.00
          TAX            -- N/A --     0.00      0.00      -- N/A --               0.00       0.00

ACCT#: 0.0.0.1-10923       DATE: 08/06/2000    BILL #: B1-4              ITEM#: B1-4,3 (Cycle forward)
102 (USD)        DEBIT ACCOUNT        DEBIT    CREDIT     CREDIT ACCOUNT         DEBIT     CREDIT
_____________________________________________________________________________________________________
          GROSS           10000        0.32      0.00        40001                 0.00       0.32
          NET            -- N/A --     0.32      0.00      -- N/A --               0.00       0.32
          DISC            40005        0.00      0.00        10000                 0.00       0.00
          TAX            -- N/A --     0.00      0.00      -- N/A --               0.00       0.00
```

Case Study 4

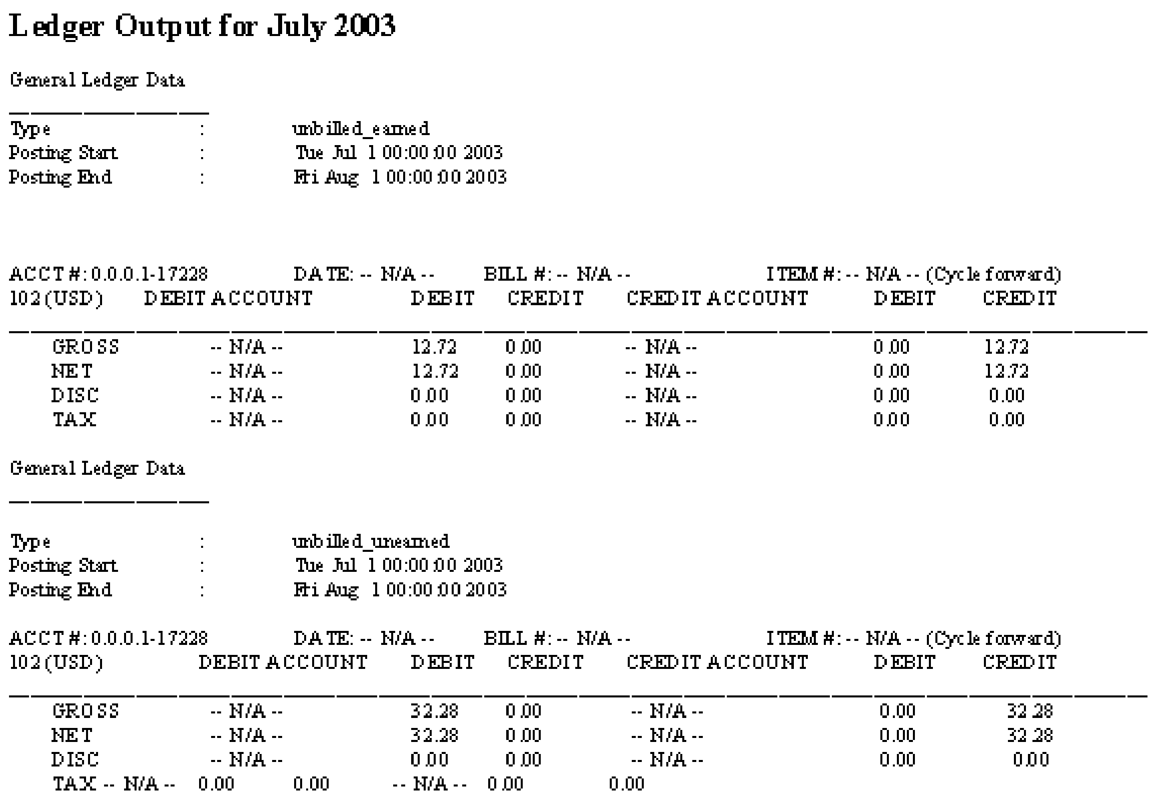
Plan details:

- Customer created: July 6, 2003

- G/L posting: Last day of the month
- Purchase: Plan 1 – Unlimited Web access with recurring discounts.
  - Quarterly email fee: $45.00 (G/L ID 102)
- Accrual-based accounting

This case study shows how BRM handles multi-month cycle fees. In this case, the quarterly fee is $45.00. BRM amortizes the fee over the life of the 92-day multi-month cycle. Because the customer’s first billing date is August 6, 2003, the revenue for July is considered unbilled in the report as shown in Figure 3–17.

Figure 3–17 July G/L Report

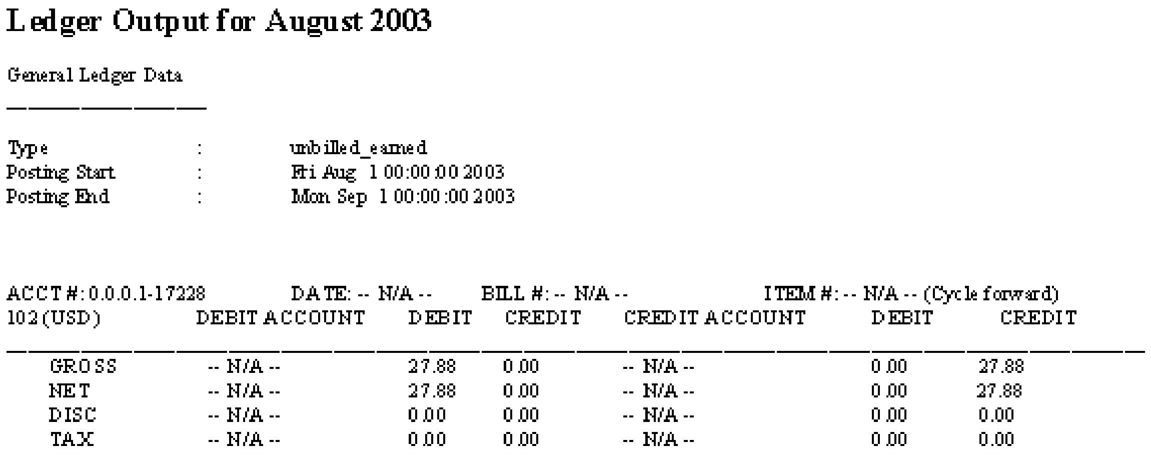


```
Ledger Output for July 2003

General Ledger Data
________________
Type             :        unbilled_earned
Posting Start    :        Tue Jul  1 00:00:00 2003
Posting End      :        Fri Aug  1 00:00:00 2003


ACCT #: 0.0.0.1-17228      DATE: -- N/A --     BILL #: -- N/A --           ITEM #: -- N/A -- (Cycle forward)
102 (USD)    DEBIT ACCOUNT            DEBIT     CREDIT     CREDIT ACCOUNT        DEBIT      CREDIT
___________________________________________________________________________________________________
    GROSS          -- N/A --          12.72     0.00       -- N/A --             0.00       12.72
    NET            -- N/A --          12.72     0.00       -- N/A --             0.00       12.72
    DISC           -- N/A --          0.00      0.00       -- N/A --             0.00        0.00
    TAX            -- N/A --          0.00      0.00       -- N/A --             0.00        0.00

General Ledger Data
________________

Type             :        unbilled_unearned
Posting Start    :        Tue Jul  1 00:00:00 2003
Posting End      :        Fri Aug  1 00:00:00 2003

ACCT #: 0.0.0.1-17228      DATE: -- N/A --     BILL #: -- N/A --           ITEM #: -- N/A -- (Cycle forward)
102 (USD)         DEBIT ACCOUNT       DEBIT     CREDIT     CREDIT ACCOUNT        DEBIT      CREDIT
___________________________________________________________________________________________________
    GROSS          -- N/A --          32.28     0.00        -- N/A --             0.00        32.28
    NET            -- N/A --          32.28     0.00        -- N/A --             0.00        32.28
    DISC           -- N/A --          0.00      0.00        -- N/A --             0.00         0.00
    TAX -- N/A --  0.00       0.00       -- N/A --  0.00         0.00
```

In the August report, you can see that BRM defers revenue based for quarterly fees. Only 26 days of revenue is earned for a quarterly fee that spans 92 days (26/92*45 = 12.71). The remainder is **unbilled_unearned** revenue.

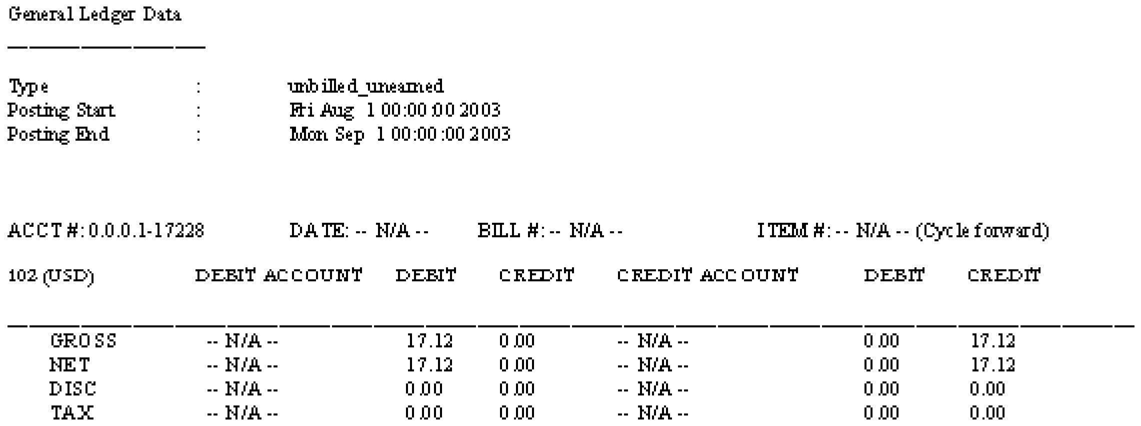
The report generated for August is shown in Figure 3–18 and Figure 3–19:

Figure 3–18 August 1 G/L Report 1



```
Ledger Output for August 2003

General Ledger Data
_______________

Type              :        unbilled_earned
Posting Start     :        Fri Aug  1 00:00:00 2003
Posting End       :        Mon Sep  1 00:00:00 2003


ACCT #: 0.0.0.1-17228      DATE: -- N/A --     BILL #: -- N/A --        ITEM #: -- N/A -- (Cycle forward)
102 (USD)        DEBIT ACCOUNT     DEBIT    CREDIT    CREDIT ACCOUNT      DEBIT      CREDIT
___________________________________________________________________________________________________
    GROSS        -- N/A --         27.88    0.00      -- N/A --           0.00       27.88
    NET          -- N/A --         27.88    0.00      -- N/A --           0.00       27.88
    DISC         -- N/A --         0.00     0.00      -- N/A --           0.00       0.00
    TAX          -- N/A --         0.00     0.00      -- N/A --           0.00       0.00
```

Figure 3–19 August 1 G/L Report 2

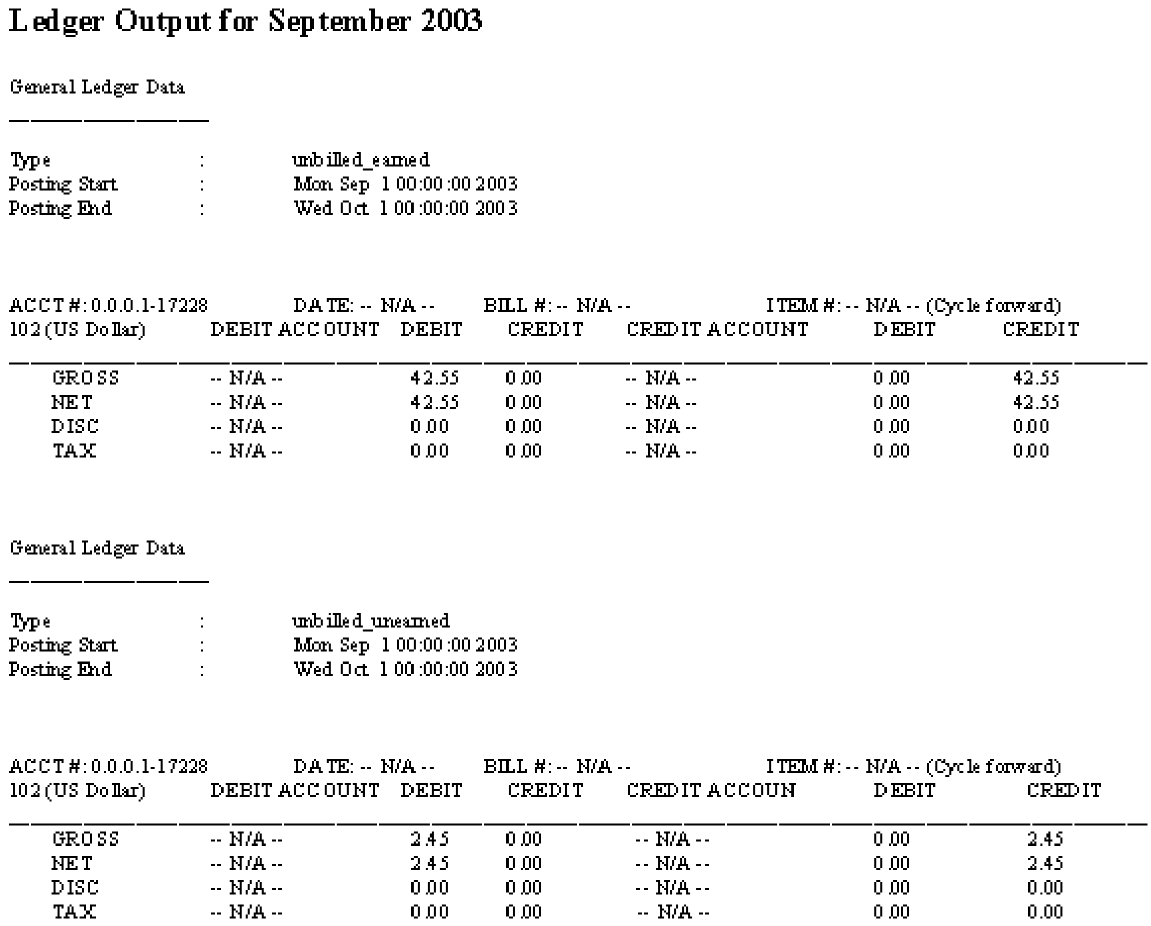


```
General Ledger Data
_______________

Type              :        unbilled_unearned
Posting Start     :        Fri Aug  1 00:00:00 2003
Posting End       :        Mon Sep  1 00:00:00 2003


ACCT #: 0.0.0.1-17228      DATE: -- N/A --     BILL #: -- N/A --        ITEM #: -- N/A -- (Cycle forward)

102 (USD)        DEBIT ACCOUNT     DEBIT    CREDIT    CREDIT ACCOUNT      DEBIT      CREDIT

___________________________________________________________________________________________________
    GROSS        -- N/A --         17.12    0.00      -- N/A --           0.00       17.12
    NET          -- N/A --         17.12    0.00      -- N/A --           0.00       17.12
    DISC         -- N/A --         0.00     0.00      -- N/A --           0.00       0.00
    TAX          -- N/A --         0.00     0.00      -- N/A --           0.00       0.00
```

In this report, the fee is billed and is reported in the **unbilled_earned** revenue type. 52 days out of 92 are shown as earned revenue.

The report generated for September is shown in Figure 3–20:

Figure 3–20 September G/L Report

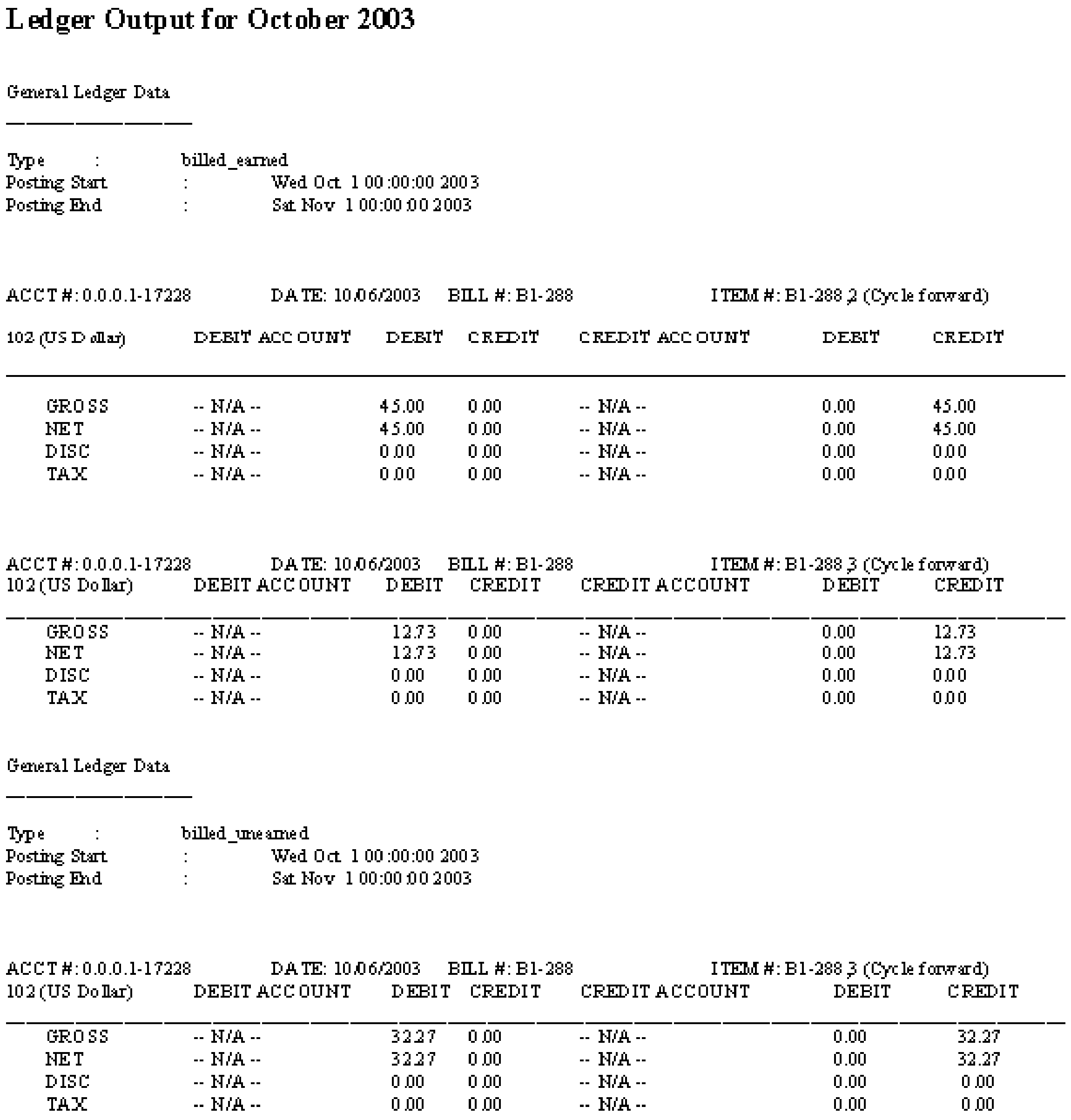


```
Ledger Output for September 2003

General Ledger Data
______________

Type                :         unbilled_earned
Posting Start       :         Mon Sep  1 00:00:00 2003
Posting End         :         Wed Oct  1 00:00:00 2003


ACCT #: 0.0.0.1-17228        DATE: -- N/A --      BILL #: -- N/A --            ITEM #: -- N/A -- (Cycle forward)
102 (US Dollar)        DEBIT ACCOUNT   DEBIT      CREDIT      CREDIT ACCOUNT        DEBIT        CREDIT
__________________________________________________________________________________________________________
     GROSS             -- N/A --         42.55     0.00        -- N/A --              0.00          42.55
     NET               -- N/A --         42.55     0.00        -- N/A --              0.00          42.55
     DISC              -- N/A --          0.00     0.00        -- N/A --              0.00           0.00
     TAX               -- N/A --          0.00     0.00        -- N/A --              0.00           0.00


General Ledger Data
______________

Type                :         unbilled_unearned
Posting Start       :         Mon Sep  1 00:00:00 2003
Posting End         :         Wed Oct  1 00:00:00 2003


ACCT #: 0.0.0.1-17228        DATE: -- N/A --      BILL #: -- N/A --            ITEM #: -- N/A -- (Cycle forward)
102 (US Dollar)        DEBIT ACCOUNT   DEBIT      CREDIT      CREDIT ACCOUN         DEBIT          CREDIT
__________________________________________________________________________________________________________
     GROSS             -- N/A --          2.45     0.00         -- N/A --             0.00            2.45
     NET               -- N/A --          2.45     0.00         -- N/A --             0.00            2.45
     DISC              -- N/A --          0.00     0.00         -- N/A --             0.00            0.00
     TAX               -- N/A --          0.00     0.00          -- N/A --            0.00            0.00
```

The previously billed date, September 6, 2003, falls outside this report, so the **prev_billed_earned** option still reports how much is earned during that month. From the 92-day multi-month cycle, the month of October earned 31 days (31/92*45 = 15.18).

Figure 3–21 shows the October G/L report.

Figure 3–21 October G/L Report



```
Ledger Output for October 2003

General Ledger Data
________________

Type          :          billed_earned
Posting Start            :          Wed Oct  1 00:00:00 2003
Posting End              :          Sat Nov  1 00:00:00 2003


ACCT #: 0.0.0.1-17228        DATE: 10/06/2003     BILL #: B1-288             ITEM #: B1-288,2 (Cycle forward)

102 (US Dollar)      DEBIT ACCOUNT     DEBIT     CREDIT      CREDIT ACCOUNT         DEBIT        CREDIT
_____________________________________________________________________________________________________

     GROSS           -- N/A --         45.00     0.00        -- N/A --              0.00         45.00
     NET             -- N/A --         45.00     0.00        -- N/A --              0.00         45.00
     DISC            -- N/A --         0.00      0.00        -- N/A --              0.00         0.00
     TAX             -- N/A --         0.00      0.00        -- N/A --              0.00         0.00


ACCT #: 0.0.0.1-17228        DATE: 10/06/2003     BILL #: B1-288             ITEM #: B1-288,3 (Cycle forward)
102 (US Dollar)      DEBIT ACCOUNT     DEBIT     CREDIT      CREDIT ACCOUNT         DEBIT        CREDIT
_____________________________________________________________________________________________________
     GROSS           -- N/A --          12.73    0.00        -- N/A --              0.00         12.73
     NET             -- N/A --          12.73    0.00        -- N/A --              0.00         12.73
     DISC            -- N/A --          0.00     0.00        -- N/A --              0.00         0.00
     TAX             -- N/A --          0.00     0.00        -- N/A --              0.00         0.00

General Ledger Data
________________

Type          :          billed_unearned
Posting Start            :          Wed Oct  1 00:00:00 2003
Posting End              :          Sat Nov  1 00:00:00 2003


ACCT #: 0.0.0.1-17228        DATE: 10/06/2003     BILL #: B1-288             ITEM #: B1-288,3 (Cycle forward)
102 (US Dollar)      DEBIT ACCOUNT      DEBIT    CREDIT      CREDIT ACCOUNT           DEBIT        CREDIT
_____________________________________________________________________________________________________
     GROSS           -- N/A --          32.27    0.00        -- N/A --                0.00          32.27
     NET             -- N/A --          32.27    0.00        -- N/A --                0.00          32.27
     DISC            -- N/A --          0.00     0.00        -- N/A --                0.00          0.00
     TAX             -- N/A --          0.00     0.00        -- N/A --                0.00          0.00
```

Case Study 5

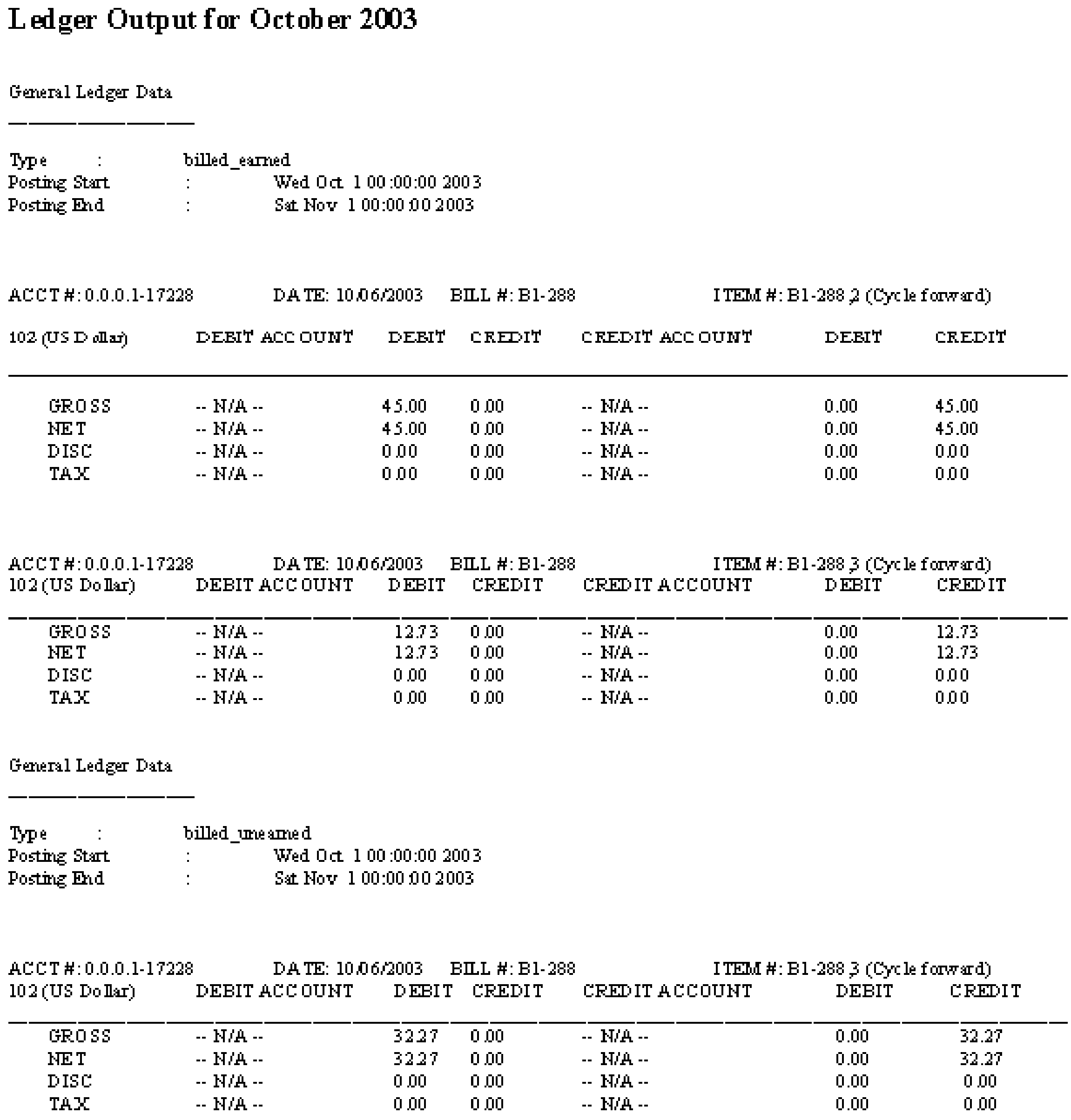
Plan details:

- Customer created: October 6, 2000
- G/L posting: Last day of the month
- Purchase: Plan 1 – Unlimited Web access with discounts. The plan is modified to include a cycle arrears fee instead of a cycle forward fee.
  - Cycle arrears fee: $19.95 (G/L ID 102)
  - Purchase fee: $15.00 (G/L ID 101)
- Accrual-based accounting

The cycle arrears fee, $19.95, is reported as earned when it is billed. The fee was charged on November 6 and the entire amount became billed and earned on that day.

Figure 3–22 shows the October G/L report.

Figure 3–22 October G/L Report



```
Ledger Output for October 2003

General Ledger Data
______________

Type         :            billed_earned
Posting Start             :            Wed Oct  1 00:00:00 2003
Posting End               :            Sat Nov  1 00:00:00 2003


ACCT #: 0.0.0.1-17228       DATE: 10/06/2003    BILL #: B1-288          ITEM #: B1-288 ,2 (Cycle forward)

102 (US Dollar)     DEBIT ACCOUNT    DEBIT    CREDIT    CREDIT ACCOUNT       DEBIT     CREDIT
________________________________________________________________________________________________
     GROSS          -- N/A --        45.00    0.00      -- N/A --            0.00      45.00
     NET            -- N/A --        45.00    0.00      -- N/A --            0.00      45.00
     DISC           -- N/A --        0.00     0.00      -- N/A --            0.00      0.00
     TAX            -- N/A --        0.00     0.00      -- N/A --            0.00      0.00


ACCT #: 0.0.0.1-17228       DATE: 10/06/2003    BILL #: B1-288          ITEM #: B1-288 ,3 (Cycle forward)
102 (US Dollar)     DEBIT ACCOUNT    DEBIT    CREDIT    CREDIT ACCOUNT       DEBIT     CREDIT
________________________________________________________________________________________________
     GROSS          -- N/A --         12.73   0.00      -- N/A --            0.00      12.73
     NET            -- N/A --         12.73   0.00      -- N/A --            0.00      12.73
     DISC           -- N/A --         0.00    0.00      -- N/A --            0.00      0.00
     TAX            -- N/A --         0.00    0.00      -- N/A --            0.00      0.00

General Ledger Data
______________

Type         :            billed_unearned
Posting Start             :            Wed Oct  1 00:00:00 2003
Posting End               :            Sat Nov  1 00:00:00 2003


ACCT #: 0.0.0.1-17228       DATE: 10/06/2003    BILL #: B1-288          ITEM #: B1-288 ,3 (Cycle forward)
102 (US Dollar)     DEBIT ACCOUNT     DEBIT   CREDIT    CREDIT ACCOUNT        DEBIT      CREDIT
________________________________________________________________________________________________
     GROSS          -- N/A --         32.27   0.00      -- N/A --             0.00       32.27
     NET            -- N/A --         32.27   0.00      -- N/A --             0.00       32.27
     DISC           -- N/A --         0.00    0.00      -- N/A --             0.00        0.00
     TAX            -- N/A --         0.00    0.00      -- N/A --             0.00        0.00
```

Case Study 6

Plan details:

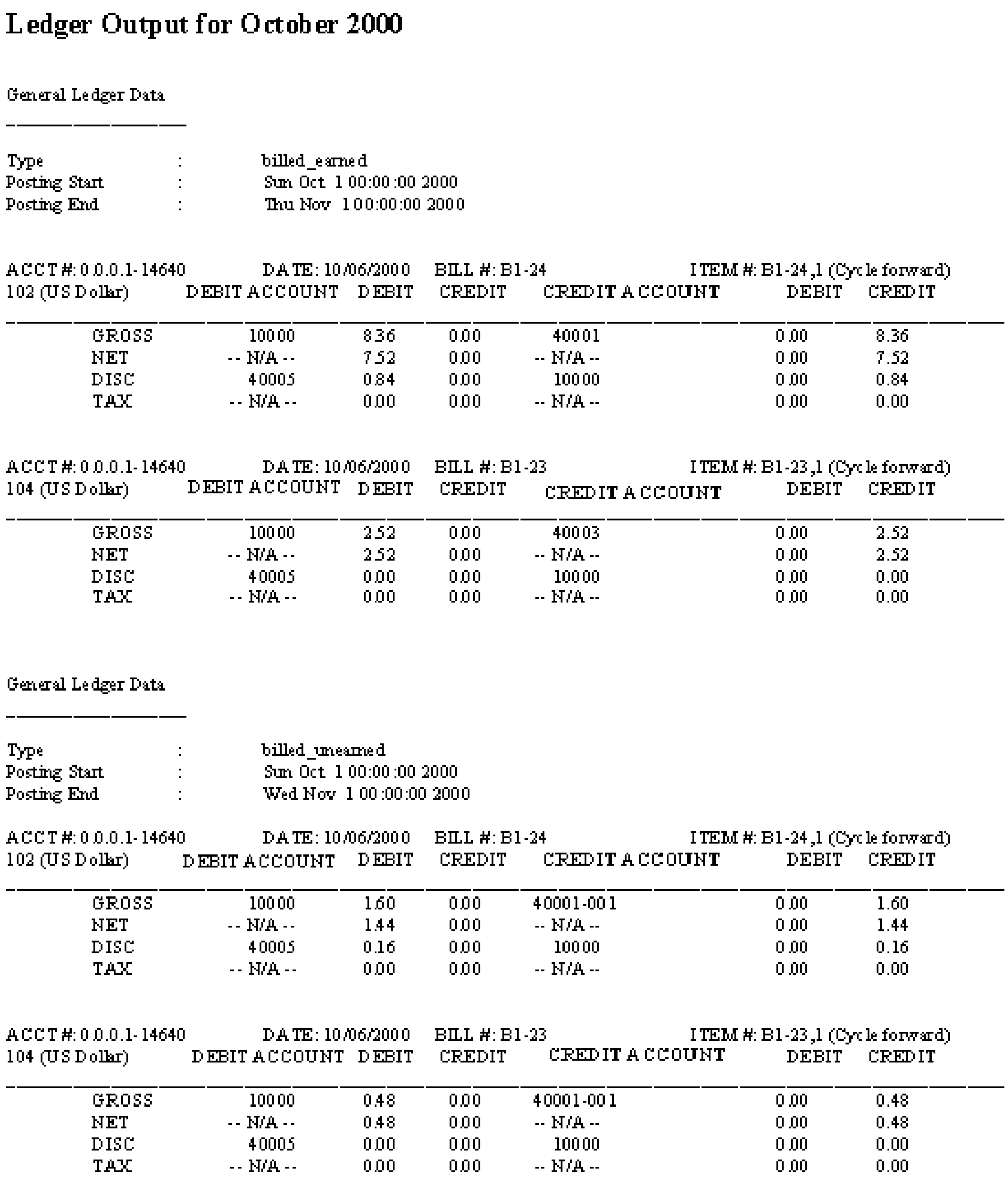
- Customer created: October 6, 2000
- G/L posting: Monthly G/L posting
- Purchase: Plan 1 – Unlimited Web access with discounts. The plan is modified to have a cycle arrears instead of a cycle forward fee.

- Monthly email fee: $19.95 (G/L ID 104), with a 10% discount
- Monthly IP fee: $15.00 (G/L ID 102)

- On-demand billing

Figure 3–23 shows the October G/L report.

Figure 3–23 October G/L Report



```
Ledger Output for October 2000

General Ledger Data
____________________

Type               :          billed_earned
Posting Start      :          Sun Oct  1 00:00:00 2000
Posting End        :          Thu Nov  1 00:00:00 2000

ACCT #: 0.0.0.1-14640        DATE: 10/06/2000    BILL #: B1-24           ITEM #: B1-24,1 (Cycle forward)
102 (US Dollar)       DEBIT ACCOUNT   DEBIT    CREDIT    CREDIT ACCOUNT          DEBIT    CREDIT
______________________________________________________________________________________________________
           GROSS             10000     8.36      0.00       40001                 0.00      8.36
           NET             -- N/A --   7.52      0.00     -- N/A --               0.00      7.52
           DISC              40005     0.84      0.00       10000                 0.00      0.84
           TAX             -- N/A --   0.00      0.00     -- N/A --               0.00      0.00

ACCT #: 0.0.0.1-14640        DATE: 10/06/2000    BILL #: B1-23           ITEM #: B1-23,1 (Cycle forward)
104 (US Dollar)       DEBIT ACCOUNT   DEBIT    CREDIT    CREDIT ACCOUNT          DEBIT    CREDIT
______________________________________________________________________________________________________
           GROSS             10000     2.52      0.00       40003                 0.00      2.52
           NET             -- N/A --   2.52      0.00     -- N/A --               0.00      2.52
           DISC              40005     0.00      0.00       10000                 0.00      0.00
           TAX             -- N/A --   0.00      0.00     -- N/A --               0.00      0.00


General Ledger Data
____________________

Type               :          billed_unearned
Posting Start      :          Sun Oct  1 00:00:00 2000
Posting End        :          Wed Nov  1 00:00:00 2000

ACCT #: 0.0.0.1-14640        DATE: 10/06/2000    BILL #: B1-24           ITEM #: B1-24,1 (Cycle forward)
102 (US Dollar)       DEBIT ACCOUNT   DEBIT    CREDIT    CREDIT ACCOUNT          DEBIT    CREDIT
______________________________________________________________________________________________________
           GROSS             10000     1.60      0.00     40001-001               0.00      1.60
           NET             -- N/A --   1.44      0.00     -- N/A --               0.00      1.44
           DISC              40005     0.16      0.00       10000                 0.00      0.16
           TAX             -- N/A --   0.00      0.00     -- N/A --               0.00      0.00

ACCT #: 0.0.0.1-14640        DATE: 10/06/2000    BILL #: B1-23           ITEM #: B1-23,1 (Cycle forward)
104 (US Dollar)       DEBIT ACCOUNT   DEBIT    CREDIT    CREDIT ACCOUNT          DEBIT    CREDIT
______________________________________________________________________________________________________
           GROSS             10000     0.48      0.00     40001-001               0.00      0.48
           NET             -- N/A --   0.48      0.00     -- N/A --               0.00      0.48
           DISC              40005     0.00      0.00       10000                 0.00      0.00
           TAX             -- N/A --   0.00      0.00     -- N/A --               0.00      0.00
```

4 Creating General Ledger Reports

This chapter describes how to create Oracle Communications Billing and Revenue Management (BRM) general ledger reports.

Before reading this chapter, read "About Collecting General Ledger Data".

About Running G/L Reports

Use the **pin_ledger_report** utility to extract data from BRM and G/L reports. This utility searches the BRM database for **/journal** objects that include summaries of balance impacts for which you want to record G/L data. The report then either displays the data or writes it to the BRM database.

You can generate reports for a variety of options, including those run for specific types of revenue recognition. Your accounting department determines which types of revenue to include in reports (for example, billed, unbilled, earned, or unearned) and how often you run G/L reports. See "About Revenue Recognition" for more information.

About Exporting G/L Reports to XML Files

You use the **pin_ledger_report** utility to export G/L reports to XML files. This enables you to import the G/L reports into an external G/L system. When you run the **pin_ledger_report** utility in **export** mode, G/L reports are automatically generated, posted, and saved to the BRM database before they are exported. You do not need to perform these tasks independently before you export them.

See "Exporting General Ledger Reports to XML Files" for more information.

> **Note:** You can also write a custom program to export your G/L data to a format compatible with your external system. For more information, see "Exporting G/L Information".

About the pin_ledger_report Utility

Use the **pin_ledger_report** utility to generate G/L reports in these modes:

- **run_report**: This mode runs the report. You can choose to display the report or to print it to a file. You can choose to post the report or not.

  See "Generating a G/L Report".

- **post_only**: This mode writes G/L report data to the BRM database. It prevents BRM from backdating events that occurred before the posting date. You can also use it to unpost the last posted report.

  See "Posting G/L Reports".

- **list_previous**: This mode lists previously run G/L reports.

  See "Listing Previously-Posted G/L Reports".

- **export**: This mode generates the report, posts it to the database, and exports it to XML files that can be imported into an external G/L system. If you plan on exporting your G/L reports, you do not need to create them first; the **export** mode does that for you. See "Exporting General Ledger Reports to XML Files" for more information.

The **pin_ledger_report** utility is located in the *BRM_Home*/**bin** directory. For complete syntax information, see "pin_ledger_report".

Controlling the Destination of pin_ledger_report Output

To display or save the output from **pin_ledger_report** you can:

- Save a report as an object in the BRM database by running **pin_ledger_report** without the **-test** option.

  Storing the data in the BRM database can help protect the integrity of your accounting data.

- Save a report to an ASCII file by running **pin_ledger_report** with the **-report** option and redirect the output to a file name.

- Display a report by running **pin_ledger_report** with the **-report** option.

Generating a G/L Report

To generate a G/L report:

- If you are creating a report for billed revenue, create an index on the ITEM_T table's POID_TYPE column prior to running **pin_ledger_report**. See "Creating an Index for the POID_TYPE Column of the ITEM_T Table".

- Run the **pin_ledger_report** utility in **run_report** mode. See "Running Reports with the pin_ledger_report Utility".

Creating an Index for the POID_TYPE Column of the ITEM_T Table

If you are creating a G/L report for billed revenue, you must create an index named **i_item__poid_type** on the ITEM_T table's POID_TYPE column. This index is required for billed revenue G/L reports only and is not used by other applications. The BRM installer does not create this index, so that it does not impact the performance of other applications.

To prevent the index from impacting performance, create the **i_item__poid_type** index prior to running **pin_ledger_report** and then drop the index after the report is generated. If you need to run **pin_ledger_report** multiple times, create the index and drop it after all runs of **pin_ledger_report** have completed.

To create the index, run the following commands:

```
% sqlplus pin/password@databaseAlias

SQL> create index i_item__poid_type on item_t (poid_type) tablespace
```

```
tablespaceName storageParameters partitionInfo
```

Note: Specify [partitionInfo] only if the ITEM_T table is partitioned.

Running Reports with the pin_ledger_report Utility

To create a G/L report, run the following command:

```
pin_ledger_report -mode run_report
                   -start start_date
                  [-end end_date]
                  [-segment gl_segment]
                  [-type billed | unbilled
                         billed_earned | billed_unearned |
                         unbilled_earned | unbilled_unearned |
                         prev_billed_earned]
                  [-report file_name]
                  [-summary | -detail]
                  [-noncurrency]
                  [-verbose]
                  [-test]
                  [-posted]
                  [-help]
```

See "pin_ledger_report" for more information.

Example: Reporting G/L revenue for a specific time period

To report G/L revenue for a specific time period, use the **-start** and **-end** options. For example:

```
pin_ledger_report -mode run_report -start 05/25/03 -end 05/31/03
```

Example: Reporting G/L revenue about a specific type of revenue

To report G/L revenue about a specific type of revenue, use the **-type** options. For example, to generate a report for the unbilled revenue type:

```
pin_ledger_report -mode run_report -start 05/25/03 -end 05/31/03
                  -type unbilled
```

Example: Reporting G/L revenue about single or multiple G/L segments and brands

To run a G/L report for a specific segment, use the **-segment** option. The following example generates a G/L report for the East Coast Enterprises segment:

```
pin_ledger_report -mode run_report -start 05/25/03 -end 05/31/03 -segment
EastCoastEnterprises
```

Posting G/L Reports

To post specific data in a G/L report, run the following command:

```
pin_ledger_report -mode post_only
                   [-verbose]
                   [-segment gl_segment]
                   [-posted posted_date | -unpost]
                   [-help]
```

See "pin_ledger_report" for more information.

Example: Posting a Previously Run G/L Report

To post a previously run G/L report, use the **-posted** and **-segment** options. The example below posts the G/L report run on 6/11/00 for the root segment:

```
pin_ledger_report -mode post_only -posted 06/11/03 -segment
```

Example: Unposting a G/L report

To unpost a previously run G/L report, use the **-unpost** option.

> **Note:** You can only unpost the last previous post. For example, if you post on January 1 and February 1, you can unpost the February 1 post, but not the January 1 post.

The example below unposts the G/L report for the G/L segment EastCoastEnterprises:

```
pin_ledger_report -mode post_only -unpost -segment EastCoastEnterprises
```

Listing Previously-Posted G/L Reports

To display a list of previously run G/L reports, run the following command:

```
pin_ledger_report -mode list_previous
                  -start start_date
                  [-end end_date]
                  [-segment gl_segment]
                  [-type billed | unbilled|
                         billed_earned | billed_unearned |
                         unbilled_earned | unbilled_unearned |
                         prev_billed_earned]
                 [-verbose]
                 [-help]
```

See "pin_ledger_report" for more information.

Example: Listing G/L Reports for a Specified Time Period

To show a list of previously run G/L reports during a specific time period, use the **-start** and **-end** options. The example below lists G/L reports that were run between 5/25/03 and 5/31/03:

```
pin_ledger_report -mode list_previous -start 05/25/03 -end 05/31/03
```

Example: Listing G/L Reports for a Specific Segment or Brand

To show a list of previously run G/L reports for a specific segment of accounts or brand, use the **-segment** option. The following example lists G/L reports that were run for the brand, East Coast Enterprises:

```
pin_ledger_report -mode list_previous -start 05/25/00 -segment
EastCoastEnterprises
```

Listing G/L Reports for Specific Types of Revenue

To show a list of previously run G/L reports for a specific type of revenue, use the **-type** option. By default, the utility generates reports for all types of revenue. The example below lists G/L reports that were run to show unbilled revenue:

```
pin_ledger_report -mode list_previous -start 05/25/00 -type unbilled
```

Exporting G/L Reports to XML files

To export G/L reports to XML files, run the following command:

```
pin_ledger_report  -mode export
                   [-segment gl_segment]
                   [-resend ReportId]
                   [-regenerate ReportId]
                   [-restart]
                   [-verbose]
                   [-help]
```

When you run the **pin_ledger_report** utility in **export** mode, G/L reports are automatically generated, posted, and saved to the BRM database before they are exported. You do not need to perform these tasks independently before you export them.

See "Exporting General Ledger Reports to XML Files" and "pin_ledger_report" for more information.

Setting the Number of A/R Accounts per G/L Report

To make the **pin_ledger_report** utility more scalable, specify the number of accounts receivable (A/R) accounts to include in a G/L report.

To change the default **transaction_grouping** entry for **pin_ledger_report**:

1. Open the (*BRM_home*/**apps/pin_billd**).
2. Edit the following entry:

   ```
   pin_ledger_report transaction_grouping 100000
   ```

3. Save the file.

Segregating Unbilled Revenue by G/L Cycle within a Billing Cycle

By default, the G/L report for unbilled revenue considers the time the journal was created to determine if the revenue was unbilled until the end of the G/L cycles. Because revenue for the same item across multiple ledger cycles is aggregated on the same journal entry, the G/L report shows the unbilled revenue as follows:

- If there is a delay in loading the usage events across multiple ledger reporting cycles, the journal entry for these usage events is created in the new G/L cycle. The unbilled revenue report for the prior G/L cycle does not include these usage events.
- If an item is created in the prior G/L cycle and a usage event for that item occurs in the current G/L cycle, the usage revenue is considered as part of the prior G/L cycle. The unbilled revenue report for the prior G/L cycle includes this usage event.

Segregate unbilled business parameter to segregate revenue for the same item across multiple G/L cycles by creating a separate journal entry for each of these G/L cycles for this item. This ensures the following:

- If there is a delay in loading the usage events across multiple ledger reporting cycles, the journal entry for these usage events is created in the G/L cycle the event occurred in. The unbilled revenue report for the prior G/L cycle includes these usage events.

- If an item is created in the prior G/L cycle and a usage event for that item occurs in the current G/L cycle, the usage revenue is considered as part of the current G/L cycle. The unbilled revenue report for the prior G/L cycle does not include this usage event.

> **Note:** Segregating unbilled revenue increases the number of journal in the BRM system.

BRM makes the journal effective in the G/L cycle in which the event's endtime falls. In the case of delayed and backdated events that are processed after billing or after G/L posting, the event processing determines the G/L period the journal belongs to.

> **Important:** When the **SegregateJournalsByGLPeriod** business parameter is enabled:
>
> - BRM assumes that a single G/L reporting cycle is used for all G/L segments in the system.
> - The future journals will be created based on the G/L calendar but existing journals will remain same.
> - If the existing G/L cycle start dates are modified in the **/config/gl_calendar** object, future journal creation will be based on the new calendar. However, the previously created journals will remain unchanged and do not reflect the new calendar.

To segregate the unbilled revenue report for G/L cycle:

1. Go to the *BRM_Home***/sys/data/config** directory.
2. Run the following command, which creates an editable XML file from the **billing** instance of the **/config/business_params** object:

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

   This command creates the XML file named **bus_params_billing.xml.out** in your working directory. To place this file in a different directory, specify the path as part of the file name.
3. Open the **bus_params_billing.xml.out** file.
4. Search for the following line:

   ```
   <SegregateJournalsByGLPeriod>disabled</SegregateJournalsByGLPeriod>
   ```

   > **Note:** Add this entry in the **bus_params_billing.xml.out** file if it does not already exist.

5. Change **disabled** to **enabled**.
6. Save this file as **bus_params_billing.xml**.
7. Go to the *BRM_Home***/sys/data/config** directory, which includes support files used by the **pin_bus_params** utility.
8. Run the following command, which loads this change into the **/config/business_params** object:

```
pin_bus_params PathToWorkingDirectory/bus_params_billing.xml
```

where *PathToWorkingDirectory* is the directory in which **bus_params_billing.xml** resides.

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

> **Note:** To run this command from a different directory, see "pin_bus_params" in *BRM Developer's Guide*.

9. Read the object with the **testnap** utility or Object Browser to verify that all fields are correct.

   See "Using testnap" in *BRM Developer's Guide* for general instructions on using the **testnap** utility. See "Reading Objects by Using Object Browser" in *BRM Developer's Guide* for information on how to use Object Browser.

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

Setting Up G/L Calendar Configuration

When the **SegregateJournalsByGLPeriod** business parameter is enabled, BRM uses the **/config/gl_calendar** object to get information related to the G/L reporting cycle.

This configuration includes an array of timestamps that indicates the start dates of the G/L cycle. For example, if the G/L reporting cycle starts on the third of every month, the configuration contains 12 timestamps for each year, each indicating the third day of the month.

> **Note:** Ensure that the current G/L cycle start date is part of this configuration. The timestamps must match the appropriate time zone of the BRM server.

To load data for the unbilled earned revenue report, manually configure the **/config/gl_calendar** object using the PCM_OP_CREATE_OBJ opcode.

To create an object using the PCM_OP_CREATE_OBJ opcode, see "Creating Objects" in *BRM Developer's Guide*.

Following is a sample flist to populate monthly G/L cycle start timestamps for three months in the US Pacific time zone.

```
0    PIN_FLD_POID                POID [0] 0.0.0.1 /config/gl_calendar -1 0
0     PIN_FLD_ACCOUNT_OBJ           POID [0] 0.0.0.1 /account 1 0
0     PIN_FLD_HOSTNAME          STR [0] "<any hostname>"
0     PIN_FLD_NAME             STR [0] "GL_CALENDAR"
0     PIN_FLD_PROGRAM_NAME          STR [0] "<any program name>"
0 PIN_FLD_CALENDAR_DATE             ARRAY [1] allocated 1, used 1
```

```
1  PIN_FLD_EFFECTIVE_START_T               TSTAMP [0] (1420099200) 01/01/2015
00:00:00:000 AM
0 PIN_FLD_CALENDAR_DATE           ARRAY [2] allocated 1, used 1
1  PIN_FLD_EFFECTIVE_START_T          TSTAMP [0] (1422777600) 01/02/2015
00:00:00:000 AM
0 PIN_FLD_CALENDAR_DATE             ARRAY [3] allocated 1, used 1
1  PIN_FLD_EFFECTIVE_START_T          TSTAMP [0] (1425196800) 01/03/2015
00:00:00:000 AM
```

After loading, read the configuration object with the **testnap** utility or Object Browser to verify that all fields are correct.

See "Using testnap" in *BRM Developer's Guide* for general instructions on using the **testnap** utility. See "Reading Objects by Using Object Browser" in *BRM Developer's Guide* for information on how to use Object Browser.

Note: If you want to modify the **/config/gl_calendar** object, delete the existing object using the PCM_OP_DELETE_OBJ opcode and reload it using the PCM_OP_CREATE_OBJ opcode. See "Deleting Objects" in *BRM Developer's Guide*.

Exporting G/L Information

You can export G/L information from BRM in the following ways:

- Configure your G/L data for export and run the **pin_ledger_report** utility with the **-export** parameter to generate the XML files. Then write a program to parse the XML output files to a format your accounting system can read. This is the recommended method for exporting G/L information. See "Exporting General Ledger Reports to XML Files" for more information.
- Write a BRM client program to read the **/ledger_report** object and write the output to a file that is loaded into your accounting system. You then run the **pin_ledger_report** utility without the **-test** option to create a **/ledger_report** object in the BRM database.
- Write a program to parse the redirected output from **pin_ledger_report** to a format your accounting system can read.

5

Loading General Ledger Configuration Data

This chapter describes how to create, customize, and load the following Oracle Communications Billing and Revenue Management (BRM) general ledger (G/L) configuration data:

Before reading this chapter, you should be familiar the BRM G/L features. See "About Collecting General Ledger Data".

Creating a Chart of Accounts

For more information about a chart of accounts (COA), see "About the Chart of Accounts".

You create a COA to:

- Verify that your G/L accounts in BRM match the accounts in your external G/L tracking system.
- Verify that the G/L accounts you entered in the G/L IDs are valid.
- Support different G/L segments.

To create a COA:

1. Go to the *BRM_Home*/**sys/data/pricing/example** directory, where *BRM_Home* is the directory in which you installed BRM components.
2. Open and edit the **pin_glchartaccts** file. The **pin_glchartaccts** file includes instructions and examples.

   > **Tip:** You need to enter a COA ID when you create G/L IDs. If you create multiple COAs, note the COA IDs.

3. Save the **pin_glchartaccts** file.
4. Run the following command, which loads this change into the **/config/glchartaccts** storable object:

   ```
   load_pin_glchartaccts PathToWorkingDirectory/pin_glchartaccts_file
   ```

   where

   - *PathToWorkingDirectory* is the directory in which the *pin_glchartaccts_file* file resides.
   - *pin_glchartaccts_file* is the file that contains the COAs.

Important: The **load_pin_glchartaccts** utility requires a configuration (**pin.conf**) file in the directory from which you run the utility. For information on creating a configuration file, see the discussion of using configuration files to connect and configure components in *BRM System Administrator's Guide*.

Caution: The **load_pin_glchartaccts** utility overwrites existing data. If you are updating the COA, you cannot load new data only. You must load new data each time you run the **load_pin_glchartaccts** utility.

See "load_pin_glchartaccts" for more information.

5. To verify that the COA file was loaded, display the **/config/gl_chartaccts** object by using the Object Browser, or use the **robj** command with the **testnap** utility.

   See "Reading an Object and Writing Its Contents to a File" in *BRM Developer's Guide*.

Creating General Ledger IDs

BRM loads a default sample **pin_glid** file during installation. To create G/L IDs, customize this file before reloading it into the BRM database.

Important: In a production system, do *not* remove or modify any existing G/L IDs. Doing so could generate inaccurate general ledger reports.

To create G/L IDs:

Important: If you use a COA in BRM, load the COA before you load G/L IDs. See "Creating a Chart of Accounts".

1. Go to the *BRM_Home*/**sys/data/pricing/example** directory.
2. Open and edit the **pin_glid** file. The **pin_glid** file includes instructions and examples.

   Important: By default, G/L data for balance impacts associated with G/L IDs 1 through 99 is not saved in **/journal** objects and is not included in G/L reports. See "How G/L IDs Affect Storage and Reporting of G/L Data".

3. (Optional) Enter the COA to use for this **pin_glid** file. The default is to *not* use a COA.
4. (Optional) Enter a G/L segment to use for this **pin_glid** file using the following syntax to define a segment.

   ```
   gl_segment  root segment.child_segment  [no_rollup]
   ```

where *child_segment* is a nested segment and **[no_rollup]** includes the segment in your reports.

For more information, see "Creating Nested Segments".

Important: Segment names are case sensitive and cannot have spaces.

5. Save the **pin_glid** file.
6. Run the following command, which loads G/L IDs into the BRM database.

   ```
   load_pin_glid PathToWorkingDirectory/pin_glid_file
   ```

   where

   - *PathToWorkingDirectory* is the directory in which the *pin_glid_file* file resides.
   - *pin_glid_file* is the file that contains the G/L IDs.

   Important: The **load_pin_glid** utility requires a configuration (**pin.conf**) file in the directory from which you run the utility. For information on creating a configuration file, see the discussion of using configuration files to connect and configure components in *BRM System Administrator's Guide*.

   Caution: The **load_pin_glid** utility overwrites existing G/L IDs. If you are updating G/L IDs, you cannot load only new G/L IDs. You must load complete sets of G/L IDs each time you run the **load_pin_glid** utility.

   See "load_pin_glid" for more information.
7. To verify that the G/L IDs file was loaded, display the **/config/gl_chartaccts** object by using the Object Browser, or use the **robj** command with the **testnap** utility.

   See "Reading an Object and Writing Its Contents to a File" in *BRM Developer's Guide*.

Changing the Default G/L Segment

You can assign a G/L segment value to an account you want to be reported in a G/L segment. The default value is the root segment, symbolized by "." (dot).

Accounts that are assigned to a brand are automatically assigned to a G/L segment.

To change the default G/L segment:

1. Go to the *BRM_Home*/**sys/cm/** directory.
2. Open the Connection Manager (CM) configuration **pin.conf** file in a text editor.
3. Search for the following line:

   ```
   - fm_cust_pol gl_segment .
   ```

4. Change it to:

```
- fm_cust_pol gl_segment segment_name
```

where *segment_name* is the name of the G/L segment you want to use as the default.

5. Save the **pin.conf** file.

You do not need to restart the CM to enable this entry.

Customization of the PCM_OP_CUST_POL_PREP_BILLINFO policy opcode is required if you want to use a G/L segment other than the default or the G/L segments used by brands.

Assigning G/L IDs to Nonrated Events

Nonrated events, such as payments and refunds, are included in G/L reports.

To assign G/L IDs to nonrated events:

1. Go to the *BRM_Home***/sys/msgs/reasoncodes** directory.
2. Open the **reasons.en_US** sample file in a text editor.
3. Add your G/L IDs, making sure that they match the G/L IDs in the **pin_glid** file.

   > **Tip:** You can change the existing G/L IDs, such as 109 (payment), to custom G/L IDs.

4. Save the **reasons.en_US** file.
5. Run the following command, which loads this change into the **/config/map_glid** object:

   ```
   load_localized_strings PathToWorkingDirectory/reasons.locale
   ```

   where

   - *PathToWorkingDirectory* is the directory in which the **reasons**.*locale* file resides.
   - *locale* is the locale file extension. For a list of locale file extensions, see "Locale Names" in *BRM Developer's Guide*.

   > **Caution:** The **load_localized_strings** utility overwrites the **/config/map_glid** object. If you are updating this object, you cannot load new G/L ID maps only. You must load complete sets of data each time you run the **load_localized_strings** utility.

   For more information on loading the **reasons.***locale* file or creating new strings for this file, see "Loading Localized or Customized Strings" in *BRM Developer's Guide*.

Assigning G/L IDs to Prerated Events

Use the PCM_OP_ACT_POL_SPEC_GLID policy opcode to assign G/L IDs to prerated events or partially rated events. By default, this policy opcode retrieves an event's G/L ID from the **/config/map_glid** storable object. You can customize it to perform searches on other objects and look at other fields.

For example, you can categorize adjustments by customizing the policy opcode to assign G/L IDs based on the adjustment type.

All customization is done using **pin_glid** and COA files.

> **Note:** You can also assign G/L IDs to prerated or partially rated events by using the **reasons.***locale* file. See "Assigning G/L IDs to Nonrated Events".

Customizing G/L Data

During event processing, BRM retrieves balance impacts from each event and adds it to the revenue summary in the appropriate **/journal** object.

To improve system performance, **/journal** objects can be disabled for non-currency resources. For more information, see "Disabling /journal Objects for Non-Currency Resources".

You can customize the data before it is written into **/journal** objects by using the PCM_OP_GL_POL_PRE_UPDATE_JOURNAL policy opcode.

> **Note:** You can customize G/L data for real-time rated events only.

To customize G/L data:

- Enable G/L data customizations by using the **CustomJournalUpdate** business parameter. See "Enabling Custom Updates to G/L Data".
- Customize the PCM_OP_GL_POL_PRE_UPDATE_JOURNAL policy opcode. See "Customizing the PCM_OP_GL_POL_PRE_UPDATE_JOURNAL Policy Opcode".

Enabling Custom Updates to G/L Data

By default, BRM does not call the PCM_OP_GL_POL_PRE_UPDATE_JOURNAL policy opcode before writing data into **/journal** objects. You configure BRM to call the policy opcode by modifying the **billing** instance of the **/config/business_params** object with the **pin_bus_params** utility.

To enable custom updates to G/L data:

1. Go to the *BRM_Home***/sys/data/config** directory.
2. Run the following command, which creates an editable XML file from the **billing** instance of the **/config/business_params** object:

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

   This command creates the XML file named **bus_params_billing.xml.out** in your working directory. To place this file in a different directory, specify the path as part of the file name.
3. Open the **bus_params_billing.xml.out** file.
4. Search for the following line:

   ```
   <CustomJournalUpdate>disabled</CustomJournalUpdate>
   ```

5. Change **disabled** to **enabled**.
6. Save the file as **bus_params_billing.xml**.
7. Go to the *BRM_Home***/sys/data/config** directory, which includes support files used by the **pin_bus_params** utility.

8. Run the following command, which loads this change into the **/config/business_params** object:

   ```
   pin_bus_params PathToWorkingDirectory/bus_params_billing.xml
   ```

   where *PathToWorkingDirectory* is the directory in which **bus_params_billing.xml** resides.

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

   > **Note:** To run this command from a different directory, see "pin_bus_params" in *BRM Developer's Guide*.

9. Read the object with the **testnap** utility or Object Browser to verify that all fields are correct.

   See "Using testnap" in *BRM Developer's Guide* for general instructions on using the **testnap** utility. See "Reading Objects by Using Object Browser" in *BRM Developer's Guide* for information on how to use Object Browser.

10. Stop and restart the CM. For more information, see "Starting and Stopping the BRM System" in *BRM System Administrator's Guide*.

11. (Multischema systems only) Run the **pin_multidb** script with the **-R CONFIG** parameter. For more information, see "pin_multidb" in *BRM System Administrator's Guide*.

Customizing the PCM_OP_GL_POL_PRE_UPDATE_JOURNAL Policy Opcode

The PCM_OP_GL_POL_PRE_UPDATE_JOURNAL policy opcode receives in the input flist the entries that will be recorded in the **/journal** object. By default, the policy opcode does not perform any processing and returns only the entries passed in the input flist.

You can customize the policy opcode to add, modify, or remove one or more of the entries that will be recorded in the **/journal** object. For information on how to customize policy opcodes, see "Adding and Modifying Policy Facilities Modules" in *BRM Developer's Guide*.

The policy opcode returns the following in the output flist:

- PIN_FLD_POID field set to the **/event** object POID.
- (Optional) PIN_FLD_JOURNAL_INFO array containing the entries that will be recorded in the **/journal** object.

> **Note:** If the PIN_FLD_JOURNAL_INFO array is passed in the output flist, its PIN_FLD_POID field must be set either to **-1** (when creating a new **/journal** object) or to a valid **/journal** POID (when updating an existing **/journal** object).

Disabling /journal Objects for Non-Currency Resources

By default, creation of **/journal** objects for non-currency resources is enabled in BRM.

> **Note:** Creating fewer **/journal** objects improves system performance.

To disable **/journal** objects for non-currency resources:

1. Go to the *BRM_Home***/sys/data/config** directory.
2. Run the following command, which creates an editable XML file from the **billing** instance of the **/config/business_params** object:

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

   This command creates the XML file named **bus_params_billing.xml.out** in your working directory. To place this file in a different directory, specify the path as part of the file name.
3. Open the **bus_params_billing.xml.out** file.
4. Search the XML file for following line:

   ```
   <NonCurrencyResourceJournaling>all</NonCurrencyResourceJournaling>
   ```

5. Do one of the following:

   - To disable creation of all **/journal** objects for non-currency resources, change **all** to **none**.
   - To create **/journal** objects for specific non-currency resources, change **all** to **+:***resource_id1*[**,**-...]

     where *resource_id#* is the non-currency resource ID number.

     Add the *resource_id#* values after the +: entry, separated by commas. Spaces are not allowed. For example,

     ```
     <NonCurrencyResourceJournaling>+:23466,199990,200000</NonCurrencyResourceJo
     urnaling>
     ```

   - To create **/journal** objects for all except specific non-currency resources, change **all** to **-:***resource_id1*[**,**-...]

     where *resource_id#* is the non-currency resource ID number.

     Add the *resource_id#* values after the -: entry, separated by commas. Spaces are not allowed. For example,

     ```
     <NonCurrencyResourceJournaling>-:458745,190000,2000444,77778</NonCurrencyRe
     sourceJournaling>
     ```

   > **Caution:** Caution is required when entering your non-currency resource ID values. Non-currency resource ID values cannot be validated by the **pin_bus_params** utility for the **NonCurrencyResourceJournaling** business parameter. If a value is not valid, when the CM is restarted, the default value **all** will be used.

   > **Note:** Values other than non-currency resource IDs are ignored.

6. Save this file as **bus_params_billing.xml**.
7. Go to the *BRM_Home*/**sys/data/config** directory, which includes support files used by the **pin_bus_params** utility.
8. Run the following command, which loads this change into the **/config/business_params** object:

   ```
   pin_bus_params PathToWorkingDirectory/bus_params_billing.xml
   ```

   where *PathToWorkingDirectory* is the directory in which **bus_params_billing.xml** resides.

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

   > **Note:** To run this command from a different directory, see "pin_bus_params" in *BRM Developer's Guide*.

9. Stop and restart the CM. For more information, see "Starting and Stopping the BRM System" in *BRM System Administrator's Guide*.
10. (Multischema systems only) Run the **pin_multidb** script with the **-R CONFIG** parameter. For more information, see "pin_multidb" in *BRM System Administrator's Guide*.

Generating General Ledger Reports Based on the Actual Billing Date

By default, in the general ledger reports, irrespective of the actual billing date, the revenue is considered billed on the end date of the corresponding bill period. For example, if the bill period is from 1/1 - 2/1 and the actual billing is run on 2/3, by default, when the general ledger report is generated, the revenue is considered billed on 2/1. When you enable the use of the actual billing date and then generate the general ledger report, the revenue is considered to be billed on the actual billing date (2/3).

The option to use the actual billing date is disabled by using the **item_t_synonym** synonym on the ITEM_T table. To use the actual billing date when generating the general ledger reports, create a view on the ITEM_T table and recreate the **item_t_synonym** synonym on the view on the ITEM_T table to enable the use of the actual billing date.

Caution:

- To avoid discrepancies after enabling the use of the actual billing date, make sure, when running the general ledger reports for the first time, that the general ledger reporting period includes both the end date of the bill period and the actual billing date.
- For accounts with skipped billing cycles, the revenue reported as billed in a previous general ledger report period may again be reported as billed in a new general ledger report period. To avoid this discrepancy, before you enable the use of the actual billing date, make sure of the following:

  There are no subordinate accounts with skipped billing cycles.

  The general ledger reports for the subordinate accounts with skipped billing cycles have already been generated for a period past the billing date of accounts receivable (A/R) parent account.
- After enabling the use of the actual billing date, do not disable it.

To generate general ledger reports based on the actual billing date:

Note: In a multischema system, run these steps for each schema.

1. Run the following command, which opens SQL*Plus:

   ```
   sqlplus login/password@database_alias
   ```

   where:

   - *login* is the login name to use for connecting to the BRM database.
   - *password* is the encrypted password for *login*.
   - *database_alias* is the BRM database alias.

2. Run the following command, which creates the **i_item_glseg_eff__id** index on the **gl_segment** and **opened_t** columns of the ITEM_T table:

   ```
   SQL> CREATE INDEX i_item_glseg_eff__id ON item_t (gl_segment, opened_t)
   tablespace tablespace_name storage_clause;
   ```

   where:

   - *tablespace_name* is the tablespace name where you want to create the index.
   - *storage_clause* specifies the storage parameters to use when you create the index; for example, storage (initial 30k next 30k maxextents unlimited pctincrease 0 freelists 2).

3. Create the **item_t_gl_view** view on ITEM_T table with all the columns in the ITEM_T table with the exception of the **effective_t** column of the view pointing to the **opened_t** column of the ITEM_T table.

   For example:

   ```
   SQL> CREATE OR REPLACE VIEW item_t_gl_view
   AS SELECT POID_ID0, ACCOUNT_OBJ_DB, ACCOUNT_OBJ_ID0, ACCOUNT_OBJ_REV,
   ACCOUNT_OBJ_TYPE,  ADJUSTED, ARCHIVE_STATUS,AR_BILLINFO_OBJ_DB,
   AR_BILLINFO_OBJ_ID0,AR_BILLINFO_OBJ_REV,AR_BILLINFO_OBJ_TYPE,
   ```

```
AR_BILL_OBJ_DB,AR_BILL_OBJ_ID0,AR_BILL_OBJ_REV,AR_BILL_OBJ_TYPE,
BAL_GRP_OBJ_DB,BAL_GRP_OBJ_ID0,BAL_GRP_OBJ_REV,BAL_GRP_OBJ_TYPE,
BILLINFO_OBJ_DB,BILLINFO_OBJ_ID0,BILLINFO_OBJ_REV,BILLINFO_OBJ_TYPE,
BILL_OBJ_DB,BILL_OBJ_ID0,BILL_OBJ_REV,BILL_OBJ_TYPE,CLOSED_T,CREATED_T,
CURRENCY, CURRENCY_OPERATOR,CURRENCY_RATE,CURRENCY_SECONDARY,
DELTA_DUE,DISPUTED,DUE,DUE_T, OPENED_T "EFFECTIVE_T",EVENT_POID_LIST,
FLAGS, GL_SEGMENT, ITEM_NO,ITEM_TOTAL, MOD_T, NAME, OPENED_T, POID_DB,
POID_REV, POID_TYPE, READ_ACCESS, RECVD,SERVICE_OBJ_DB, SERVICE_OBJ_ID0,
SERVICE_OBJ_REV, SERVICE_OBJ_TYPE, STATUS, TRANSFERED, WRITEOFF,
WRITE_ACCESS FROM item_t;
```

4. Run the following command, which creates the **item_t_synonym** synonym on the **item_t_gl_view** view on the ITEM_T table:

   ```
   SQL> CREATE OR REPLACE SYNONYM item_t_synonym FOR item_t_gl_view;
   ```

5. Run the following command, which recompiles only the invalid objects:

   ```
   SQL> exec dbms_utility.compile_schema(USER,false);
   ```

6. Reset the value of OPENED_T of the previously billed items for general ledger reports. For more information, see the discussion on resetting the value of OPENED_T of the previously billed items for general ledger reports in *BRM 7.5 Patch Set Installation Guide*.
7. Run the following command, which exits SQL*Plus:

   ```
   SQL> exit
   ```

Disabling General Ledger Collection in BRM

By default, General Ledger collection is enabled in BRM.

Important: When General Ledger reporting is disabled, **/journal** objects are not created.

To disable General Ledger collection:

1. Go to the *BRM_Home*/**sys/data/config** directory.
2. Run the following command, which creates an editable XML file from the **billing** instance of the **/config/business_params** object:

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

   This command creates the XML file named **bus_params_billing.xml.out** in your working directory. To place this file in a different directory, specify the path as part of the file name.
3. Open the **bus_params_billing.xml.out** file.
4. Search for the following line:

   ```
   <GeneralLedgerReporting>enabled</GeneralLedgerReporting>
   ```

5. Change **enabled** to **disabled**.
6. Save this file as **bus_params_billing.xml**.
7. Go to the *BRM_Home*/**sys/data/config** directory, which includes support files used by the **pin_bus_params** utility.

8. Run the following command, which loads this change into the **/config/business_params** object:

   ```
   pin_bus_params PathToWorkingDirectory/bus_params_billing.xml
   ```

   where *PathToWorkingDirectory* is the directory in which **bus_params_billing.xml** resides.

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

   > **Note:** To run this command from a different directory, see "pin_bus_params" in *BRM Developer's Guide*.

9. Read the object with the **testnap** utility or Object Browser to verify that all fields are correct.

   See "Using testnap" in *BRM Developer's Guide* for general instructions on using the **testnap** utility. See "Reading Objects by Using Object Browser" in *BRM Developer's Guide* for information on how to use Object Browser.

10. Stop and restart the CM. For more information, see "Starting and Stopping the BRM System" in *BRM System Administrator's Guide*.

11. (Multischema systems only) Run the **pin_multidb** script with the **-R CONFIG** parameter. For more information, see "pin_multidb" in *BRM System Administrator's Guide*.

6

Exporting General Ledger Reports to XML Files

This chapter describes how to create Oracle Communications Billing and Revenue Management (BRM) general ledger (G/L) reports and export them to XML files that can be loaded into your external G/L system.

Before reading this chapter, you should be familiar with BRM general ledger data. See "About Collecting General Ledger Data".

About Exporting G/L Reports

To export G/L reports to XML files, run **pin_ledger_report** in **export** mode as follows:

```
pin_ledger_report  -mode export
                   [-segment gl_segment]
                   [-resend ReportId]
                   [-regenerate ReportId]
                   [-restart]
                   [-verbose]
                   [-help]
```

This enables you to import the G/L reports into an external G/L system.

When you run the **pin_ledger_report** utility in **export** mode, G/L reports are automatically generated, posted, and saved to the BRM database before they are exported. You do not need to perform these tasks independently before you export them.

About Configuring G/L Reports for Export

To configure G/L reports for export, you use the **pin_config_export_gl.xml** file. This file contains the settings for all reports that get exported to XML files. For more information, see "Configuring G/L Reports for Export".

Depending on your G/L report configuration, BRM generates the following G/L reports (**/ledger_report** objects) in the database for a given reporting period:

- A basic G/L report. See "Generating a G/L Report".
- A customized G/L report, when the PCM_OP_GL_POL_EXPORT_GL policy opcode performs custom processing. See "Customizing G/L Reports for Export".
- An incremental report, when the revenue type is unbilled, unbilled unearned, unbilled earned, or billed unearned. See "About Incremental Reports and Cumulative Revenue".

Note: When you are reporting in **export** mode, the previously earned revenue is reported only if you have configured the report to include the previously billed earned revenue. For more information on setting up your G/L reports for export, see "Configuring G/L Reports for Export".

About Incremental Reports and Cumulative Revenue

When you export G/L reports for the following revenue types, the reports include cumulative revenue across reporting periods:

- Unbilled
- Unbilled earned
- Unbilled unearned
- Billed unearned

For information about generating G/L reports, see "Generating a G/L Report."

G/L report displays cumulative revenue for a specified revenue type accumulated from the start date of the first reporting period to the end date of the current reporting period. Part of the total revenue reported for the current period may be included in the same report for the previous period.

When you run the **pin_ledger_report** utility in **export** mode, it automatically calculates the difference between the revenue reported in the current reporting period for a G/L segment and the revenue reported in the previous reporting period for the same G/L segment. (The end date for the previous report is the start date for the current report.) The difference, or increment, is the amount of revenue accumulated during the current reporting period. Thus, exported G/L reports contain the incremental revenue amounts; you do not need to calculate them manually.

For example, an unbilled report for the period January 1 to February 1 specifies an unbilled amount of $100 as of February 1. This is published to a financial system as $100 of unbilled charges (assuming it is the first ever reporting period).

For the period from February 1 to March 1, a new charge of $50 is applied and $30 of the original charge of $100 got billed. The unbilled report for the period from February 1 to March 1 specifies the amount of unbilled amount as of March 1. When you generate the base report in **-run_report** mode, the total unbilled amount is:

$100 + $50 - $30 = $120

This is published to the financial system as an increment of $20 ($120 - $100), so that the net amount in the financial system is:

$100 + $20 = $120

The reason for publishing incremental amounts is that multiple source systems could be publishing unbilled amount into the same financial system. If you publish the cumulative amount, the net amount in the financial system would be:

$100 + $120 = $220

All source systems should publish the incremental amounts (positive or negative), and not the latest cumulative amounts.

For information on revenue recognition, see "About Revenue Recognition".

About G/L Segments and Cumulative Revenue

When you export G/L reports for nested segments and the revenue type is unbilled, unbilled unearned, unbilled earned, or billed unearned (cumulative revenue), the segment you report on must be the same from reporting period to reporting period. For example, if a January G/L report calculates unbilled revenue for the **.westcoast.california** G/L segment, the February G/L report must calculate unbilled revenue for the **.westcoast.california** G/L segment. If, instead, the February report calculates unbilled revenue for the **.westcoast** G/L segment or the **.westcoast.california.sf** G/L segment, the G/L data overlaps and invalid revenue is reported.

Note: You can specify overlapping segments for snapshot reports; however, it is not recommended.

For information on revenue recognition, see "About Revenue Recognition".

For information on creating G/L segments, see "Creating General Ledger IDs".

About Regenerating and Re-Exporting G/L Reports

To ensure data integrity within the BRM system and between BRM and your external G/L system, you may need to regenerate or re-export G/L reports that you exported previously. For example:

- When you make changes to the G/L data in the BRM database *after* you generated the report for that G/L data

  In such cases, running the **pin_ledger_report** utility in **export** mode with the -**regenerate** parameter:

  - Recreates the G/L report in the database
  - Reposts the G/L report
  - Recreates the XML export files

  If you regenerate a G/L report that was not the last report generated, you must also regenerate all subsequent reports to ensure the data is accurate. For example, if you need to regenerate a report for March, and reports exist for April and May, you should regenerate the April and May reports.

  Important: A regenerated report does not correct G/L data that was previously imported into the external G/L system. You must reverse the original reports in the external G/L system and then re-import the regenerated data to maintain data integrity.

  For more information, see "Example: Regenerating G/L Reports".

- When the G/L data in the BRM database has *not* changed and the XML export files were either created incorrectly or lost before they could be imported into the external G/L system.

  In such cases, running the **pin_ledger_report** utility in **export** mode with the -**resend** parameter recreates the XML export files. It does *not* recreate or repost the G/L report in the database.

  For more information, see "Example: Re-Exporting G/L Reports".

Important: Do not use the **-regenerate** or **-resend** parameter if the **pin_ledger_report** utility exits abnormally. Instead, use the **-restart** parameter. See "Example: Restarting a Failed Export Operation".

How BRM Exports G/L Reports

The **pin_ledger_report** utility performs the following tasks when it runs:

1. Retrieves G/L report configuration information from the **/config/export_gl** object to determine which reports to generate and which G/L segments to report.
2. Creates a new **/process_audit/export_gl** object for this run and sets the status of the **/process_audit/export_gl** object to IN_PROGRESS.
3. Reads the previous **/process_audit/export_gl** object and the **ReportInitialStartDate** value in the **/config/export_gl** object to determine if any G/L reports for previous periods have not been created. If so, creates them.
4. Calls the PCM_OP_GL_LEDGER_REPORT opcode to create the **/ledger_report** object for the current period.
5. Using the POID of the generated **/ledger_report** object, calls the PCM_OP_GL_POL_EXPORT_GL policy opcode and then does one of the following, based on the PIN_FLD_RESULT value:
   - **0**: No customization; generates and exports the original report.
   - **1**: Customization. If the G/L data was updated, creates a new **/ledger_report** object in the database and exports the new report.

   Note: If the report contains cumulative revenue, generates an incremental report by using the G/L report for the previous period, if one exists, and updates the **/ledger_report** object.
6. If specified in the export configuration file, filters noncurrency resources from the generated **/ledger_report** object to be included in or excluded from the XML output file.
7. Generates the XML output file.
8. Updates the **/process_audit/export_gl** object with information about this run.
9. Posts reports in the database.
10. Sets the status of the **/process_audit/export_gl** object to RUN_COMPLETED.

Note: If the **pin_ledger_report** utility encounters errors during processing, it exits without completing the report processing. (One or more prior reports and one or more G/L export reports *may* have been successfully generated.)

About Exported G/L Report Files

When you export G/L reports, the **pin_ledger_report** utility uses the information in the **/config/export_gl** object to extract the necessary G/L data from the database and save it to XML files. One XML file is created for each combination of G/L segment and revenue type defined in the export configuration file.

If you configure your G/L reports for the root segment, G/L data for all segments is reported on, except those that roll up to their parent segments. As many XML files will be created as there are report types, and each one will contain data for the entire root segment.

By default, the XML files are saved with the following naming convention:

RevenueType_ReportEndDate_ReportStartDate_ReportId.**xml**

where:

- *RevenueType* is the abbreviation for the revenue type being reported, for example, **ue** for unbilled earned and **be** for billed earned.
- *ReportEndDate* is the end date of the current report in *YYYYMMDD* format.
- *ReportStartDate* is the start date for the current report in *YYYYMMDD* format.
- *ReportId* is the ID of the generated report. For details, see "ReportId".

For example:

ue_20070731_20070630_0.0.0.1-123456-10.xml

Table 6–1 describes the XML tags in the G/L export files.

Table 6–1 XML Tags in G/L Reports

XML Element	Description
SourceSystemID	The unique ID for the database containing the G/L data.
ReportId	The unique ID of the G/L report in the BRM system. It comprises the following information, separated by dashes: database number, POID of the **/process_audit/export_gl** object, and report number. For example, **0.0.0.1-123456-10**.
RevenueType	Revenue type: - Billed - Billed earned - Billed unearned - Prior billed earned (Prior billed earned revenue is also known as Previously billed earned revenue. See "About Revenue Recognition"). - Unbilled - Unbilled earned - Unbilled unearned
BRM_GL_Segment	The name of the G/L segment being reported on.
ReportCreatedTime	Report run time, which includes the following information: - Year - Month - Day - Hours - Minutes - Seconds

Table 6–1 (Cont.) XML Tags in G/L Reports

XML Element	Description
PeriodStartTime	The start time of the revenue reporting period, which includes the following information: ■ Year ■ Month ■ Day ■ Hours ■ Minutes ■ Seconds
PeriodEndTime	The end time of the revenue reporting period, which includes the following information: ■ Year ■ Month ■ Day ■ Hours ■ Minutes ■ Seconds
RevenueAmounts	The revenue reported for a specific G/L segment. Revenue amounts are specified with the **<element>** attribute. Each element contains: ■ The resource ID and its associated BRM G/L ID. ■ A list of revenue accounts and their credit and debit amounts. For example, **ARGrossAccount name="monthly.debit"**. The revenue accounts are retrieved from the **pin_glid** file. For information on configuring this data, see "Loading General Ledger Configuration Data".

Sample Output XML File

The following XML output file shows an unbilled earned revenue report for the **.westcoast.california.sf** G/L segment. It contains revenue amounts for two balance impacts, one for monthly cycle fees and the other for international airtime minutes.

> **Note:** The G/L account names for each balance impact type are the default BRM G/L accounts (for example, **ARGrossAccount** and **OffsetGrossAccount**). You can use the PCM_OP_GL_POL_EXPORT_GL policy opcode to convert the BRM values to custom G/L account names before the XML files are created. See "Mapping BRM G/L Account Names to Third-Party Account Names".

```
<GeneralLedgerReport>
   <SourceSystemID>123.45.123</SourceSystemID>
   <ReportId>0.0.0.1-1927591-6</ReportId>
   <RevenueType>"Unbilled earned"</RevenueType>
   <BRM_GL_Segment>.west.california.sf</BRM_GL_Segment>
   <ReportCreatedTime>
      <Year>2006</Year>
      <Month>07</Month>
      <Day>01</Day>
   <Hours>18</Hours>
   <Minutes>44</Minutes>
   <Seconds>21</Seconds>
</ReportCreatedTime>
<PeriodStartTime>
   <Year>2006</Year>
   <Month>06</Month>
   <Day>01</Day>
   <Hours>12</Hours>
   <Minutes>00</Minutes>
   <Seconds>00</Seconds>
</PeriodStartTime>
<PeriodEndTime>
   <Year>2006</Year>
   <Month>07</Month>
   <Day>01</Day>
   <Hours>00</Hours>
   <Minutes>00</Minutes>
   <Seconds>00</Seconds>
</PeriodEndTime>

<RevenueAmounts element="1">
   <ResourceId>840</ResourceId>
   <BRM_GLId>102</BRM_GLId>

   <ARGrossAccount name="monthly.debit">
      <Credit>0.0</Credit>
      <Debit>30.0</Debit>
   </ARGrossAccount>

   <ARDiscountAccount name="monthly.credit">
      <Credit>0.0</Credit>
      <Debit>0.0</Debit>
   </ARDiscountAccount>

   <ARNetAccount name="monthly.debit">
      <Credit>0.0</Credit>
      <Debit>30.0</Debit>
   </ARNetAccount>

   <ARTaxAccount name="tax.debit">
      <Credit>0.0</Credit>
      <Debit>0.0</Debit>
   </ARTaxAccount>

   <OffsetGrossAccount name="monthly.credit">
      <Credit>30.0</Credit>
      <Debit>0.0</Debit>
   </OffsetGrossAccount>

   <OffsetDiscountAccount name="monthly.debit">
      <Credit>0.0</Credit>
      <Debit>0.0</Debit>
   </OffsetDiscountAccount>

   <OffsetNetAccount name="monthly.credit">
      <Credit>30.0</Credit>
      <Debit>0.0</Debit>
   </OffsetNetAccount>
```

```
      <OffsetTaxAccount name="tax.credit">
         <Credit>0.0</Credit>
         <Debit>0.0</Debit>
      </OffsetTaxAccount>
   </RevenueAmounts>

   <RevenueAmounts element="2">
      <ResourceId>10002</ResourceId>
      <BRM_GLId>1421</BRM_GLId>

      <ARGrossAccount name="minutes.credit">
         <Credit>0</Credit>
         <Debit>100</Debit>
      </ARGrossAccount>

      <ARDiscountAccount name="minutes.debit">
         <Credit>0</Credit>
         <Debit>0</Debit>
      </ARDiscountAccount>

      <ARNetAccount name="minutes.credit">
         <Credit>0</Credit>
         <Debit>100</Debit>
      </ARNetAccount>

      <ARTaxAccount name="tax.credit">
         <Credit>0</Credit>
         <Debit>0</Debit>
      </ARTaxAccount>

      <OffsetGrossAccount name="minutes.debit">
         <Credit>100</Credit>
         <Debit>0</Debit>
      </OffsetGrossAccount>

      <OffsetDiscountAccount name="minutes.credit">
         <Credit>0</Credit>
         <Debit>0</Debit>
         </OffsetDiscountAccount>

      <OffsetNetAccount name="minutes.debit">
         <Credit>100</Credit>
         <Debit>0</Debit>
      </OffsetNetAccount>

      <OffsetTaxAccount name="tax.debit">
            <Credit>0</Credit>
         <Debit>0</Debit>
      </OffsetTaxAccount>
   </RevenueAmounts>
</GeneralLedgerReport>
```

Configuring G/L Reports for Export

You configure your G/L reports for export by editing the **pin_config_export_gl.xml** file and loading its contents into the **/config/export_gl** object in the database. The configuration file enables you to set up your reporting schedules, revenue types, and other report information. By default, it is located in the *BRM_home***/sys/data/config** directory, where *BRM_home* is the directory in which BRM is installed.

See "load_pin_config_export_gl" for information on loading the export configuration into the database.

Table 6–2 contains the configuration tags you use to set up your G/L reports for export:

Table 6–2 Configuration Tags for G/L Reports (Export)

Use This Tag:	To Do This:
SourceSystemID	Specify a unique ID for the BRM system containing the G/L data. **Note:** This tag is useful when the G/L data is exported from multiple BRM systems and the applications that process the exported XML files need to identify the source of the data.
OutputDirectory	Specify the directory on your system in which to create the exported XML files.
FileNamePrefix	Specify a prefix to append to the beginning of the generated XML file name. **Note:** The **pin_ledger_report** utility uses a specific naming convention to ensure the XML file names are unique within a directory; therefore a prefix is additional and can be left blank if it is not required.

ReportInitialStartDate	**Important:** This value is used as the report start date only the first time a G/L report is exported for the corresponding segment. All subsequent runs of the **pin_ledger_report** utility for that segment use the end date of the previous G/L report as the report's start date value.
	Specify the initial start date for each G/L segment for which data is exported. Use the following format:
	`<Year>`**`YYYY`**`</Year>`
	`<Month>`**`MM`**`</Month>`
	`<Day>`**`DD`**`</Day>`
	For example:
	`<Year>`**`2007`**`</Year>`
	`<Month>`**`01`**`</Month>`
	`<Day>`**`01`**`</Day>`
	The initial start date must be specified for the root G/L segment (**Segment name**="**.**"). All G/L segments to be reported use this date. To set a different start date for a specific G/L segment, use the **Segment** tag and specify a different initial start date for that G/L segment. This date overrides the root segment start date.

SegmentList	Contains the list of G/L segments on which to report.
Segment	Use the **name** attribute to specify a G/L segment to report on. **Note:** If you specify a nested segment, be sure to include the root segment prefix. For example: **<Segment name=.westcoast>** One G/L report is exported for each combination of G/L segment and revenue type. **Note:** The **Segment name** value is the name you assigned to a G/L segment or brand when you created G/L IDs. **Important:** If you specify multiple instances of the same segment, each one must report on a different set of revenue types.

Table 6–2 (Cont.) Configuration Tags for G/L Reports (Export)

Use This Tag:	To Do This:
Frequency	Specify how often G/L reports are generated for the segment: ■ **Daily** ■ **Weekly** ■ **Monthly** ■ **Yearly** ■ **Specific Dates** **Important:** The frequency for your BRM G/L reports must match the frequency of the reporting schedule in your external G/L system. For example, if the BRM G/L calendar is **Monthly**, the export frequency should not be **Weekly**. This creates inaccurate G/L data.

Date	If **Frequency** is **Yearly**, set the **DayofMonth** and **Month** sub-tags under the **Date** tag to specify the date to generate G/L reports.
	For **Month**, specify **01** through **12**; for **DayofMonth**, specify **01** through **31**. If the month has fewer days than specified, the last day of the month is used. For example:
	`<Frequency>`**`Yearly`**`</Frequency>`
	`<Date>`
	`   <DayofMonth>`**`-29`**`</DayofMonth>`
	`   <Month>`**`-02-`**`</Month>`
	`</Date>`

DayofMonth	If **Frequency** is **Monthly**, specify the day of the month to generate G/L reports: **01** through **31**. If the month has fewer days than specified, the last day of the month is used.
Day	If **Frequency** is **Weekly**, specify the day of week to generate G/L reports: **Monday**, **Tuesday**, **Wednesday**, **Thursday**, **Friday**, **Saturday**, **Sunday**.

Date	If **Frequency** is **Specific Dates**, add one **Month**/**DayofMonth** pair for each date on which to generate a G/L report.
	For **Month** specify **01** through **12**; for **DayofMonth** specify **01** through **31**. If the month has fewer days than specified, the last day of the month is used. For example:
	`<Frequency>`**`Specific Dates`**`</Frequency>`
	`<Date>`
	`   <DayofMonth>`**`-29`**`</DayofMonth>`
	`   <Month>`**`-02-`**`</Month>`
	`</Date>`
	`<Date>`
	`   <DayofMonth>`**`-31`**`</DayofMonth>`
	`   <Month>`**`-05-`**`</Month>`
	`</Date>`

RevenueTypeList	Contains the **RevenueType** list on which to report.

Table 6–2 (Cont.) Configuration Tags for G/L Reports (Export)

Use This Tag:	To Do This:
RevenueType	Specify the revenue type for the G/L report: ■ **Billed** ■ **Billed earned** ■ **Billed unearned** ■ **Prior billed earned** (**Prior billed earned** revenue is also known as **Previously billed earned** revenue. See "About Revenue Recognition".) ■ **Unbilled** ■ **Unbilled unearned** ■ **Unbilled earned** **Important:** If you specify multiple instances of the same G/L segment, each one must report on a different set of revenue types.
ReportLevel	Specify the type of G/L report to generate: **Summary** or **Detailed**. Specify **Detailed** only when the report is customized using PCM_OP_GL_EXPORT_GL. See "Customizing G/L Reports for Export".
ResourceType	Specify the resource type to include in the G/L report: ■ **Monetary** ■ **Nonmonetary** ■ **All**
IncludeNonMonetary	If **ResourceType** is **NonMonetary** or **All**, specify the resource ID of the nonmonetary resource to include. Use a separate **ResourceID** tag for each resource. All other nonmonetary resources are excluded from the report. **Note:** The **IncludeNonMonetary** and **ExcludeNonMonetary** tags are mutually exclusive. You can use only one of these tags.
ExcludeNonMonetary	If **ResourceType** is **NonMonetary** or **All**, specify the resource ID of the nonmonetary resource to exclude. Use a separate **ResourceID** tag for each nonmonetary resource. All other nonmonetary resources are included in the report. **Note:** The **IncludeNonMonetary** and **ExcludeNonMonetary** tags are mutually exclusive. You can use only one of these tags.

Sample G/L Report Configuration File

The following example shows the configuration of **pin_config_export_gl.xml** for the root (.) segment:

```
 <GLReportConfiguration>
   <SourceSystemID>Germany</SourceSystemID>
   <OutputDirectory>
/$PINHOME/GL_output
   </OutputDirectory>

   <FileNamePrefix>
      test
   </FileNamePrefix>

   <ReportInitialStartDate>
      <Segment name=".">
          <Year>2011</Year>
          <Month>--07--</Month>
```

```
            <Day>---01</Day>
        </Segment>
        <Segment name=".east">
            <Year>2011</Year>
            <Month>--07--</Month>
            <Day>---01</Day>
        </Segment>
    </ReportInitialStartDate>

    <SegmentList>
        <Segment name=".">
            <Frequency>Daily</Frequency>
            <RevenueTypeList>
                <RevenueType>Billed earned </RevenueType>
                <RevenueType>Unbilled earned</RevenueType>
                <RevenueType>Billed</RevenueType>
                <RevenueType>Unbilled</RevenueType>
                <RevenueType>Previously billed earned</RevenueType>
            </RevenueTypeList>
            <ReportLevel>Detailed</ReportLevel>
            <ResourceType>All</ResourceType>
             <!--ExcludeNonMonetary>
                <ResourceID>111111111</ResourceID>
             </ExcludeNonMonetary-->
        </Segment>
    </SegmentList>
  </GLReportConfiguration>
```

The following example shows a **pin_config_export_gl.xml** file that contains two segments that are exported at different frequencies:

- **eastcoast** is exported monthly.
- **westcoast** is exported on specific dates.

The **eastcoast** segment gets its initial start date from the root segment, but the **westcoast** segment was added at a later date; therefore, it has a different **ReportInitialStartDate** value.

```
 <GLReportConfiguration>

    <SourceSystemID>California</SourceSystemID>

    <OutputDirectory>
        /users/gluser/exported
    </OutputDirectory>

    <FileNamePrefix>
        USA_
    </FileNamePrefix>

    <ReportInitialStartDate>
        <Segment name=".">
            <Year>2006</Year>
            <Month>07</Month>
            <Day>15</Day>
        </Segment>
            <!-- Segment '.westcoast' was introduced on 1/1/2007 -->
        <Segment name=".westcoast">
            <Year>2007</Year>
            <Month>01</Month>
```

```
          <Day>01</Day>
      </Segment>
   </ReportInitialStartDate>

   <SegmentList>
      <Segment name=".eastcoast">
          <Frequency>Monthly</Frequency>
          <DayOfMonth>---01</DayOfMonth>
          <RevenueTypeList>
              <RevenueType>Billed</RevenueType>
              <RevenueType>Billed earned</RevenueType>
          </RevenueTypeList>
          <ReportLevel>Summary</ReportLevel>
          <ResourceType>All</ResourceType>
           <ExcludeNonMonetary>
              <ResourceID>100000</ResourceID>
              <ResourceID>100001</ResourceID>
              <ResourceID>100002</ResourceID>
              <ResourceID>100003</ResourceID>
           </ExcludeNonMonetary>
      </Segment>

      <Segment name=".westcoast">
          <Frequency>Specific Dates</Frequency>
          <Date>
             <DayofMonth>---29</DayofMonth>
             <Month>--02--</Month>
          </Date>
          <Date>
              <DayofMonth>---31</DayofMonth>
              <Month>--05--</Month>
           </Date>
          <Date>
              <DayofMonth>---31</DayofMonth>
              <Month>--08--</Month>
           </Date>
          <Date>
              <DayofMonth>---30</DayofMonth>
              <Month>--11--</Month>
           </Date>
           <RevenueTypeList>
              <RevenueType>Billed</RevenueType>
              <RevenueType>Billed earned</RevenueType>
           </RevenueTypeList>
           <ReportLevel>Detailed</ReportLevel>
           <ResourceType>All</ResourceType>
           <ExcludeNonMonetary>
              <ResourceID>100000</ResourceID>
              <ResourceID>100001</ResourceID>
              <ResourceID>100002</ResourceID>
              <ResourceID>100003</ResourceID>
            </ExcludeNonMonetary>
      </Segment>
   </SegmentList>
  </GLReportConfiguration>
</BusinessConfiguration>
```

Editing the G/L Export Configuration File

To create or modify G/L reports for export:

1. Open the *BRM_home*/**sys/data/config**/**pin_config_export_gl.xml** in an XML editor or a text editor.
2. Enter the appropriate information into the file. See "Configuring G/L Reports for Export".
3. Save and close the file.
4. Run the following command, which loads the G/L configuration information into the **/config/export_gl** object:

   ```
   load_pin_config_export_gl pin_config_export_gl.xml
   ```

> **Important:**
>
> - Run this command from the directory in which the **pin_config_export_gl.xml** file is located, otherwise, include the complete path to the file. For example:
>
>   ```
>   load_pin_config_export_gl BRM_home/sys/data/config/pin_config_
>   export_gl.xml
>   ```
>
> - When you run the utility, the **pin_config_export_gl.xml** and **business_configuration.xsd** files must be in the same directory. By default, both files are in *BRM_home*/**sys/data/config**.
> - The **load_pin_config_export_gl** utility needs a configuration (**pin.conf**) file in the directory from which you run the utility. For information about creating a configuration file, see "Creating Configuration Files for BRM Utilities" in *BRM System Administrator's Guide*.

Setting Up G/L Reporting Schedules

The frequency for your BRM G/L reports must match the frequency of the reporting schedule in your external G/L system. For example, if the external G/L calendar is **Monthly**, the export frequency for your BRM G/L reports should be **Monthly**. If the two frequencies do not match, inaccurate G/L data will be reported.

You configure the export schedule for your G/L segments by setting the **Frequency** value for the G/L segment in the **pin_config_export_gl.xml** file. Each G/L segment you report on can use a different export frequency as long as it matches the reporting frequency for that G/L segment in the external G/L system.

To set up multiple reporting schedules for the *same* G/L segment, you must specify different revenue types for each report. You cannot specify overlapping revenue types to report on.

For example, to report on billed revenue on a monthly basis and unbilled revenue on a weekly basis for the **eastcoast** G/L segment, list multiple instances of the **eastcoast** segment in the **SegmentList** tag and specify the revenue type for each schedule, as shown in the following sample code:

```
<SegmentList>
   <Segment name=".eastcoast">
       <Frequency>Monthly</Frequency>
       <DayOfMonth>01</DayOfMonth>
       <RevenueTypeList>
           <RevenueType>Billed</RevenueType>
       </RevenueTypeList>
       <ReportLevel>Summary</ReportLevel>
```

```
            <ResourceType>Monetary</ResourceType>
        </Segment>
        <Segment name=".eastcoast">
            <Frequency>Weekly</Frequency>
            <Day>Saturday</Day>
            <RevenueTypeList>
                <RevenueType>Unbilled earned</RevenueType>
            </RevenueTypeList>
            <ReportLevel>Summary</ReportLevel>
            <ResourceType>Monetary</ResourceType>
        </Segment>
    </SegmentList>
```

Filtering Noncurrency Resources

If you have a large number of noncurrency resources in your system, you can use the **IncludeNonMonetary** or **ExcludeNonMonetary** tag in the **pin_config_export_gl.xml** file to filter the resources you report on; for example, if your system contains 20 noncurrency resources, but you want to report on only a few. In this case, use **IncludeNonMonetary** and specify the noncurrency resource IDs to include in the report. To report on *all but* a few of the 20 noncurrency resources, use **ExcludeNonMonetary** and specify the noncurrency resource IDs to *exclude*.

Exporting G/L Reports

Run the **pin_ledger_report** utility in **export** mode to generate G/L reports and export them to XML files. You can also regenerate and re-export G/L reports. **Export** mode exports G/L reports according to the configuration defined in the **pin_config_export_gl.xml** file.

> **Note:** When you export G/L reports, they are always posted to the database. To generate a report without posting it, use the without the **-posted** parameter.

A separate G/L report is generated and exported for each combination of G/L segment and revenue specified in the **pin_config_export_gl.xml** file.

Example: Exporting G/L Reports for a Specific G/L Segment

To export a G/L report for a specific segment, run the **pin_ledger_report** utility in -**export** mode with the **-segment** parameter and specify the segment to report. For example:

```
pin_ledger_report -mode export -segment .westcoast
```

Example: Exporting G/L Reports in Parallel

To process multiple G/L reports at the same time, run multiple instances of the **pin_ledger_report** utility. For each instance, use -**export** mode with the **-segment** parameter and specify the segment for the desired report.

Example: Regenerating G/L Reports

To regenerate a previously generated G/L report and re-export it to an XML file, run the **pin_ledger_report** utility in **export** mode with the **-regenerate** parameter:

```
pin_ledger_report -mode export -regenerate ReportId
```

where *ReportId* is a combination of the database number, the Portal object ID (POID) of the **/process_audit/export_gl** object, and the report number all separated by dashes. For example, if the database number is **0.0.0.1**, the POID is **123456**, and the report number is **4**, enter:

```
pin_ledger_report -mode export -regenerate 0.0.0.1-123456-4
```

To find the report ID, look in the previously generated XML report file, if available, or in the export audit data. See "Retrieving Audit Data for Exported G/L Reports".

For more information about regenerating reports, see "About Regenerating and Re-Exporting G/L Reports".

Example: Re-Exporting G/L Reports

To export a previously exported G/L report to an XML file, run the **pin_ledger_report** utility in **export** mode with the **-resend** parameter:

```
pin_ledger_report -mode export -resend ReportId
```

where *reportId* is a combination of the database number, the POID of the **/process_audit/export_gl** object, and the report number. For example, if the database number is **0.0.0.1**, the POID is **123456**, and the report number is **4**, type:

```
pin_ledger_report -mode export -resend 0.0.0.1-123456-4
```

To find the report ID, look in the previously generated XML report file, if available, or in the export audit data. See "Retrieving Audit Data for Exported G/L Reports".

> **Important:** This command does not regenerate the report contained in the **/audit_object/export_gl** object. For information on regenerating an exported G/L report, see "Example: Regenerating G/L Reports".

For more information about re-exporting reports, see "About Regenerating and Re-Exporting G/L Reports".

Example: Restarting a Failed Export Operation

If the **pin_ledger_report** utility exits abnormally, it attempts to clean up any errors from that run and finishes processing. If it is not successful (the status is not set to COMPLETED after the run), subsequent runs will not work.

To clean up the previous run and finish processing successfully, use **export** mode with the **-restart** parameter:

```
pin_ledger_report -mode export -resend ReportId
```

The utility will continue to process the reports from the point at which it left off when it stopped; therefore, no duplicate reports are generated.

Retrieving Audit Data for Exported G/L Reports

Information about each run of the **pin_ledger_report** utility is written to the **/process_audit/export_gl** object in the BRM database. One **/process_audit** object is created for each run of the utility.

To view the data in the **/process_audit** object, read the object with the **testnap** utility or with the Object Browser.

For information on how to use **testnap** or Object Browser, see "Reading an Object and Writing Its Contents to A File" in *BRM Developer's Guide* .

You can use the G/L export audit data to:

- Ensure that reports have been generated for all periods prior to the start date of the next report.
- Generate statistics.
- Troubleshoot reporting errors.

The **/process_audit/export_gl** object contains the following information:

- The status of the last export run: COMPLETED, IN_PROGRESS, or INCOMPLETE.
    - COMPLETED indicates that the application run has exited normally.
    - IN_PROGRESS indicates that the application is currently running.
    - INCOMPLETE indicates that the application had crashed during an earlier application run and the audit entry was marked as such during a subsequent run.
- If the **pin_ledger_report** utility was run in **export** mode with the **-resend** or **-regenerate** parameter, the POID of the ledger report object last prepared for export.
- Report information array. Each array element contains the following information for one report:
    - G/L segment for which report is generated
    - Type of ledger report; for example **4** for unbilled earned
    - Frequency (schedule) of exporting G/L reports
    - End time of the reporting period
    - If nonmonetary resources were reported on, the resource IDs that were included in or excluded from the report
    - A list of up to three POIDs

      The first POID is the default report generated by calling the PCM_OP_GL_LEDGER_REPORT opcode.

      If there are two POIDs, the second POID is either the custom report (billed, billed earned, and previously billed earned) or the cumulative report (unbilled, unbilled earned, unbilled unearned, and billed unearned) generated after execution of the PCM_OP_GL_POL_EXPORT_GL policy opcode.

      If there are three POIDs, the second POID is the custom report and the third POID is the cumulative report.

Customizing G/L Reports for Export

You can customize the data in your exported G/L reports by configuring the PCM_OP_GL_POL_EXPORT_GL policy opcode. The **pin_ledger_report** utility calls this policy opcode in **export** mode after it generates a G/L report but before it exports the G/L report data to an XML file.

For example, you can map the G/L revenue account name in the BRM system to a different account name. To do this, you customize the PCM_OP_GL_POL_EXPORT_GL policy opcode to do the mapping and to return the modified PIN_FLD_GL_ACCTS array from the input flist.

When the **pin_ledger_report** utility runs in **export** mode, it creates a new ledger report object with the custom data in the PIN_FLD_GL_ACCTS array. This customized G/L report is used for further processing and is exported to the G/L report XML files.

The PIN_FLD_RESULT value in the output flist determines whether to create a customized report from the original report used for export:

- **0**: Use the original G/L data to generate the export report. This is the default.
- **1**: Use the G/L data in the output flist to generate a custom export report.

> **Note:** The values returned by the PCM_OP_GL_POL_EXPORT_GL policy opcode must be in summary format even when the input flist data is in detailed format.

Customizing G/L Export Report Contents

Use the PCM_OP_GL_POL_EXPORT_GL policy opcode to customize the contents of your exported G/L reports. For example, the BRM G/L account names could be mapped to the third-party account names, or the BRM resource IDs could be mapped to the third-party resource IDs.

Mapping BRM G/L Account Names to Third-Party Account Names

Use the PCM_OP_GL_POL_EXPORT_GL policy opcode to convert BRM G/L account names to the G/L account names in your external G/L system before XML reports are exported. This makes importing them into the external system easier. The following is an example of the code used to map BRM G/L account names to the third-party account names after the PCM_OP_GL_POL_EXPORT_GL policy code has been called.

```
#***********************************************************
#Section to customize report
#***********************************************************
PCM_OP(ctxp, PCM_OP_READ_OBJ, 0, in_flistp, &r_flistp, ebufp);
PIN_ERR_LOG_FLIST(PIN_ERR_LEVEL_DEBUG,"Flistp after read obj in Custom report", r_flistp);
while ((tmp_flistp = PIN_FLIST_ELEM_TAKE_NEXT(r_flistp, PIN_FLD_GL_ACCTS,&rec_id, 1,
&cookie, ebufp)) != (pin_flist_t *)NULL)
{
         char *segment = NULL;
         PIN_ERR_LOG_FLIST(PIN_ERR_LEVEL_DEBUG, "GL accts from input flist", tmp_flistp);
         tmp1_flistp = PIN_FLIST_ELEM_ADD(out_flistp, PIN_FLD_GL_ACCTS, ctr, ebufp);
         PIN_FLIST_CONCAT(tmp1_flistp, tmp_flistp, ebufp);
         ctr++;
         vp = PIN_FLIST_FLD_GET(r_flistp, PIN_FLD_GL_SEGMENT, 0, ebufp);
         if (vp)
        {
            segment = (char *)vp;
         }
         modify_acct_name(tmp1_flistp, PIN_FLD_AR_GROSS_GL_ACCT, segment, ebufp);
         modify_acct_name(tmp1_flistp, PIN_FLD_OFF_GROSS_GL_ACCT, segment, ebufp);
         modify_acct_name(tmp1_flistp, PIN_FLD_AR_NET_GL_ACCT, segment, ebufp);
         modify_acct_name(tmp1_flistp, PIN_FLD_OFF_NET_GL_ACCT, segment, ebufp);
         modify_acct_name(tmp1_flistp, PIN_FLD_AR_DISC_GL_ACCT, segment, ebufp);
         modify_acct_name(tmp1_flistp, PIN_FLD_OFF_DISC_GL_ACCT, segment, ebufp);
         modify_acct_name(tmp1_flistp, PIN_FLD_AR_TAX_GL_ACCT, segment, ebufp);
         modify_acct_name(tmp1_flistp, PIN_FLD_OFF_TAX_GL_ACCT, segment, ebufp);
         PIN_ERR_LOG_FLIST(PIN_ERR_LEVEL_DEBUG, "GL accts from output flist ", tmp1_flistp);
         PIN_ERR_LOG_FLIST(PIN_ERR_LEVEL_DEBUG, "Intermediate output flist ", out_flistp);
}
#************************************************************
#Set PIN_FLD_RESULT to indicate customization
#***********************************************************/
result = 1;
PIN_FLIST_FLD_SET(out_flistp, PIN_FLD_RESULT, (void *)&result, ebufp);
```

Mapping BRM Resource IDs to Third-Party Resource IDs

If the mapping capabilities provided with the BRM **pin_glid** mapping file are not sufficient, you can use the PCM_OP_GL_POL_EXPORT_GL policy opcode to convert BRM resource IDs to the resource IDs in your external G/L system before XML reports are exported. This makes importing them into the external system easier. The following is an example of the code used to map BRM resource IDs to the third-party resource IDs after the PCM_OP_GL_POL_EXPORT_GL policy code has been called.

```
#************************************************************
#Section to customize report
#****************************************************************
PCM_OP(ctxp, PCM_OP_READ_OBJ, 0, in_flistp, &r_flistp, ebufp);
PIN_ERR_LOG_FLIST(PIN_ERR_LEVEL_DEBUG, "Flistp after read obj in Custom report", r_flistp);
while ((tmp_flistp =PIN_FLIST_ELEM_TAKE_NEXT(r_flistp, PIN_FLD_GL_ACCTS,&rec_id, 1, &cookie,
ebufp)) != (pin_flist_t *)NULL)
{
       PIN_ERR_LOG_FLIST(PIN_ERR_LEVEL_DEBUG, "GL accts from input flist ", tmp_flistp);
       tmp1_flistp = PIN_FLIST_ELEM_ADD(out_flistp, PIN_FLD_GL_ACCTS, ctr, ebufp);
       PIN_FLIST_CONCAT(tmp1_flistp, tmp_flistp, ebufp);
       ctr++;
       vp = PIN_FLIST_FLD_GET(r_flistp, PIN_FLD_RESOURCE_ID, 0, ebufp);
       if (vp)
          {
          resource_id = (int32 *)vp;
          if (resource_id == 840)
              {
              new_resource_id = 1840;
              }
          }
PIN_FLIST_FLD_SET(out_flistp, PIN_FLD_RESOURCE_ID, (void *)&new_resource_id, ebufp);
PIN_ERR_LOG_FLIST(PIN_ERR_LEVEL_DEBUG, "GL accts from output flist ", tmp1_flistp);
PIN_ERR_LOG_FLIST(PIN_ERR_LEVEL_DEBUG, "Intermediate output flist ", out_flistp);
}
#***********************************************************
#Set PIN_FLD_RESULT to indicate customization
#***********************************************************
result = 1;
PIN_FLIST_FLD_SET(out_flistp, PIN_FLD_RESULT, (void *)&result, ebufp);
```

Customizing the G/L Report XML File Names

To customize the XML file names for your G/L reports, use the **FileNamePrefix** tag in the **pin_config_export_gl.xml** configuration file. For example, if your company tracks G/L data separately for multiple divisions of the same company, you can prefix the G/L data for each division with a unique identifier.

Validating and Troubleshooting Exported G/L Reports

Before exporting your G/L reports, validate that your G/L export configuration data was loaded into the database correctly. Do the following:

1. Run the **load_pin_config_export_gl** utility with the **-d** parameter when you load the **pin_config_export_gl.xml** file into the database. This will write any errors to a log file, which you can use for debugging purposes.
2. Run the **pin_ledger_report** utility in **post_only** mode and verify the **/data/ledger_report** object is updated in the database.
3. Run the **pin_ledger_report** utility in **run_report** mode with the **-report** and **-test** parameters. This will display the G/L report without creating the **/ledger_report** object.

   Verify that the revenue types and G/L segments are correct.
4. Run the **pin_ledger_report** utility in **export** mode and verify that the report name, report type, G/L segments, frequency, and revenue types are correct.
5. Verify that the **/process_audit/export_gl** object was created in the database. View the data in this object and verify the data is correct and that the status of the export run is COMPLETED. See "Retrieving Audit Data for Exported G/L Reports".

Purging G/L Export Data

You can purge the following G/L report data:

- Exported XML files on the file system

  Keep the exported XML files for the latest period for each segment and report type until subsequent runs have been made and the XML files have been imported successfully into the external system.
- **/process_audit/export_gl** objects and **/ledger_report** objects in the database

  Keep **/process_audit/export_gl** objects for each G/L segment/report type combination (for example, a billed revenue report for the **.westcoast** G/L segment) until at least two newer sets of objects are in the database. This ensures you can re-export or regenerate previous G/L reports if necessary.

> **Important:** Each **/process_audit/export_gl** object contains a reference to a corresponding **/ledger_report** object; therefore, you should purge them at the same time. Before doing so, make sure they are no longer needed.

For information on purging data, see "About Purging Data" in *BRM System Administrator's Guide*.

7 General Ledger Utilities

This chapter provides reference information for Oracle Communications Billing and Revenue Management (BRM) General Ledger utilities.

load_pin_config_export_gl

Use this utility to load general ledger (G/L) report configuration parameters into the **/config/export_gl** object in the BRM database. You define the configuration parameters in the *BRM_Home***/sys/data/config/pin_config_export_gl.xml** file. See "Configuring G/L Reports for Export" for more information.

> **Note:** The format of the XML file is specified in the **pin_config_export_gl.xsd** schema file in the *BRM_Home***/xsd** directory.

The **pin_ledger_report** utility uses this information to generate GL reports and export them to XML files.

> **Caution:** This utility overwrites all existing configuration data in the **/config/export_gl** object. When updating the data, you cannot load new data only. You must load all G/L configuration parameters every time you run the utility.

> **Important:** To connect to the BRM database, this utility needs a configuration file in the directory from which you run the utility. For information about creating configuration files for BRM utilities, see "Creating Configuration Files for BRM Utilities" in *BRM System Administrator's Guide*.

Location

*BRM_Home***/bin**

Syntax

```
load_pin_config_export_gl pin_config_export_gl.xml [-d] [-v] [-h]
```

Parameters

***pin_config_export_gl*.xml**

The name of the G/L report configuration file that is being loaded into the BRM database.

If you copy the **pin_config_export_gl.xml** file to the same directory from which you run the **load_pin_config_export_gl** utility, you do not have to specify either the path or the file name.

If you run the command in a different directory from where the **pin_config_export_gl.xml** file is located, you must include the entire path for the file.

-d

Creates a log file for debugging purposes. Use this parameter for debugging when the utility appears to have run with no errors but the G/L export configuration parameters have not been loaded into the database.

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

-h
Displays the syntax and parameters for this utility.

Results

The **load_pin_config_export_gl** utility notifies you only if it encounters errors.

load_pin_glchartaccts

Use this utility to load an external chart of accounts (COA) into the **config/glchartaccts** storable object in the BRM database.

See "About the Chart of Accounts" for information.

> **Note:** You cannot load separate **/config/gl_chartaccts** objects for each brand. All brands use the same object.

Location

BRM_Home/**bin**

Syntax

```
load_pin_glchartaccts [-d] [-t] [-v] [-h] pin_glchartaccts_file
```

Parameters

-d
Creates a log file for debugging purposes. Use this parameter for debugging when the utility appears to have run with no errors, but the usage maps do not appear in the database.

-t
Runs the **load_pin_glchartaccts** utility in test mode. If you run the command with this option, the COA is not loaded into the database.

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

-h
Displays the syntax and parameters for this utility.

pin_glchartaccts_file
The name of the COA file that is being loaded into the BRM database.

If you copy the **pin_glchartaccts** file to the same directory from which you run the **load_pin_glchartaccts** utility, you do not have to specify either the path or the file name.

If you run the command in a different directory from where the **pin_glchartaccts** file is located, you must include the entire path for the file.

Results

The **load_pin_glchartaccts** utility notifies you only if it encounters errors.

load_pin_glid

Use this utility to load G/L IDs into the BRM database. You define the G/L IDs in the *BRM_Home*/**sys/data/pricing/example/pin_glid** file.

See "About Collecting General Ledger Data" for information about defining G/L IDs.

> **Note:** You cannot load separate **/config/gl_segment** objects for each brand. All brands use the same object.

> **Caution:** When you run the **load_pin_glid** utility, it overwrites the existing G/L IDs. If you are updating a set of G/L IDs, you cannot load new G/L IDs only. You must load a complete set of G/L IDs each time you run the **load_pin_glid** utility.

> **Important:** To connect to the BRM database, the **load_pin_glid** utility needs a configuration file in the same directory from which you run the utility. For information about creating configuration files for BRM utilities, see "Creating Configuration Files for BRM Utilities" in *BRM System Administrator's Guide*.

Location

BRM_Home/**bin**

Syntax

```
load_pin_glid [-d] [-t] [-v] pin_glid_file |[-h]
```

Parameters

-d
Creates a log file for debugging purposes. Use this parameter for debugging when the utility appears to have run with no errors, but the usage maps do not appear in the database.

-t
Runs the **load_pin_glid** utility in test mode. If you run the command with this option, the **pin_glid** file is not loaded into the database.

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

-h
Displays the syntax and parameters for this utility.

pin_glid_file
The name and location of the file that defines the G/L IDs. The default **pin_glid** file is in *BRM_Home*/**sys/data/pricing/example**.

If you copy the **pin_glid** file to the same directory from which you run the **load_pin_glid** utility, you do not have to specify either the path or the file name.

If you run the command in a different directory from where the **pin_glid** file is located, you must include the entire path for the file.

Results

The **load_pin_glid** utility notifies you when it successfully creates the G/L IDs.

If the utility displays an error message, look in the utility log file (**default.pinlog**) to find any errors. The log file is either in the directory from which the utility was started, or in a directory specified in the configuration file.

Important: You must restart the Connection Manager (CM) for any changes to take effect. See the discussion of starting and stopping BRM in *BRM System Administrator's Guide*.

pin_ledger_report

Use this utility to collect BRM general ledger (G/L) information, generate G/L reports, and export the reports to XML files. By default, it reports on currency resources only.

For more information, see the following:

- About Collecting General Ledger Data
- Creating General Ledger Reports
- Exporting General Ledger Reports to XML Files

> **Note:** To connect to the BRM database, you must have a configuration file in the same directory as the **pin_ledger_report** utility. For information about creating configuration files for BRM utilities, see "Creating Configuration Files for BRM Utilities" in *BRM System Administrator's Guide*.

> **Important:** For multischema systems, you must run the utility separately against each database schema in your system. See "Running Non-MTA Utilities in Multischema Systems" in *BRM System Administrator's Guide*.

Location

BRM_Home/**bin**

Syntax Overview

You can use the **-mode** parameter to run this utility in these modes:

- "run_report Mode Syntax" runs the report. You can choose whether to post the report or not.
- "post_only Mode Syntax" posts the report but does not display the report or write it to a file.
- "list_previous Mode Syntax" lists previously run reports.
- "export Mode Syntax" exports the report to an XML file using the settings defined in the G/L export configuration file. Reports are always posted in this mode. Use this mode if you plan to import G/L data into a third-party financial system.

> **Note:** Before you can export reports, you must configure the G/L export information. See "Configuring G/L Reports for Export".

run_report Mode Syntax

```
pin_ledger_report -mode run_report
                   -start start_date
                  [-end end_date]
                  [-segment gl_segment]
                  [-type billed | unbilled
                         billed_earned | billed_unearned |
                         unbilled_earned | unbilled_unearned |
      prev_billed_earned]
[-report file_name]
[-summary | -detail]
[-noncurrency | -all_resource]
[-test]
[-verbose]
[-help]
[-post]
```

run_report Mode Parameters

-start *mm/dd/yy* or *mm/dd/yyyy*

-end *mm/dd/yy* or *mm/dd/yyyy*

The start and end dates for revenue to be included in the report. The value you supply for the start date is inclusive, but the value you supply for the end date is noninclusive and defaults to the current date.

> **Note:** Billing cycles and G/L reporting cycles are not always aligned. Therefore, the revenue from multiple events that occur in different billing cycles can be included in the same G/L report.

- To create a report for a period between two specific dates, use the **-start** and **-end** parameters.
- To create a report for a period between a past date and the current date, use only the **-start** parameter.

The following example includes 05/25/07 00:00:00 through 05/31/07 23:59:59:

```
pin_ledger_report -mode run_report -start 05/25/07 -end 06/01/07
```

-segment *gl_segment*

Creates a report for a specific G/L segment. *gl_segment* is the name you assigned to a G/L segment or brand when you created G/L IDs for that segment or brand. The default is the root segment.

-type billed | unbilled | billed_earned | billed_unearned | unbilled_earned | unbilled_unearned | prev_billed_earned

Creates a report for the type of revenue you specify. The default is **billed**. See "About Revenue Recognition".

-report *file_name*

Displays the output of the utility. Use the **-report** parameter with the **-test** parameter to display the report without creating the **/ledger_report** object.

-summary | detail

Creates a detailed report or a summary report. By default, summary customer G/L reports are created.

-noncurrency | -all_resource

Reports G/L data for non-currency resources only or for both currency and non-currency resources. If you do not use either of these parameters, the utility reports currency only.

-test

Runs the utility in calculation mode only.

-verbose

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

-help

Displays the syntax and parameters for this utility.

-post

Posts the report using the value in the **-end_date** parameter as the post date.

> **Note:** If you use the **-post** parameter, it must be the last parameter on the command line.

post_only Mode Syntax

```
pin_ledger_report -mode post_only
                  [-segment gl_segment]
                  [-verbose]
                  [-help]
                  [-posted posted_date | -unpost]
```

post_only Mode Parameters

-segment *gl_segment*

Posts the G/L report for a specific G/L segment or brand. See "About Collecting General Ledger Data".

-verbose

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

-help

Displays the syntax and parameters for this utility.

-posted *posted_date* | **-unpost**

Posts or unposts the specified ledger report. You specify the report by entering the post date. To display a list of previously run reports, see "list_previous Mode Syntax".

If you unpost, the last posted date is changed to the date of the previous post. You can only unpost the last previous post. For example, if you post on January 1 and February 1, you can unpost the February 1 post, but not the January 1 post.

> **Note:** If you use the **-post** parameter, it must be the last parameter on the command line.

list_previous Mode Syntax

```
pin_ledger_report -mode list_previous
                  -start start_date
                  [-end end_date]
                  [-segment gl_segment]
                  [-type billed | unbilled
                         billed_earned | billed_unearned |
                         unbilled_earned | unbilled_unearned |
                         prev_billed_earned]
```

```
                [-verbose]
                [-help]
```

list_previous Mode Parameters

-start *start_date*
-end *end_date*
The start and end times for listing the reports. The default for the end date is the current date. The following example includes 05/25/07 00:00:00 through 05/31/07 23:59:59:

```
pin_ledger_report -mode list_previous -start 05/25/07 -end 05/31/07
```

-segment *gl_segment*
Shows the report for a specific G/L segment or brand. The default is to list reports for all segments.

-type billed | unbilled | billed_earned | billed_unearned | unbilled_earned | unbilled_unearned | prev_billed_earned
Lists reports for a specified type of revenue. By default, the utility lists all types.

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

-help
Displays the syntax and parameters for this utility.

export Mode Syntax

```
pin_ledger_report  -mode export
                   [-segment gl_segment]
                   [-resend ReportId]
                   [-regenerate ReportId]
                   [-restart]
                   [-verbose]
                   [-help]
```

Note: Reports are always posted when export mode is used.

export Mode Parameters

-segment *gl_segment*
Exports XML reports for the G/L segment specified. The default is to generate and export reports for all segments.

gl_segment is the name you assigned to a G/L segment or brand when you created G/L IDs for that segment or brand. The value(s) must be defined in the G/L report configuration file. See "Configuring G/L Reports for Export".

Tip: To process reports for two or more G/L segments at the same time, run multiple instances of the utility. For more information, see "Example: Exporting G/L Reports in Parallel".

-resend *ReportID*

Re-creates the XML files for the specified reports without re-creating the **/ledger_report** object. Use this parameter to re-create one or more export XML files.

ReportId is a combination of the database number, the POID of the **/process_audit/export_gl** object containing the G/L reports, and the G/L report number. These values are separated by a hyphen. For example, if the database number is **0.0.0.1,** the POID is **123456**, and the report number is **1**, *ReportId* is **0.0.0.1-123456-1**. If *ReportId* is not specified, all reports listed in the **/process_audit/export_gl** object are re-exported.

Note: The G/L export configuration file is used only to retrieve the file name prefix (if defined) and the output directory location. All other parameters are the same as those used for the report exported originally.

-regenerate *ReportId*

Re-creates the **/ledger_report** storable object and the XML report. Use this parameter when the G/L data in the BRM database changes after the G/L reports have been exported into XML files.

ReportId is a combination of the database number, the POID of the **/process_audit/export_gl** object containing the G/L reports, and the G/L report number. These values are separated by a hyphen. For example, if the database number is **0.0.0.1,** the POID is **123456**, and the report number is **1**, *ReportId* is **0.0.0.1-123456-1**. If *ReportId* is not specified, all reports listed in the **/process_audit/export_gl** object are re-exported.

Important: If you regenerate a G/L report for a period after which subsequent G/L reports have been generated, you must regenerate all subsequent reports. For example, if you generated monthly reports from January through April, and the February report must be regenerated, you must also regenerate the March and April reports.

Note: The G/L export configuration file is used only to retrieve the file name prefix (if defined) and the output directory location. All other parameters are the same as those used for the report exported originally.

-restart

Cleans up a previous abnormal run and finishes processing successfully.

When the **pin_ledger_report** utility runs in **export** mode, it creates a process audit object and sets the object’s status to IN_PROGRESS to lock the object. When it finishes processing successfully, it sets the status to COMPLETED. If the utility exits abnormally before resetting the status, subsequent runs will not work as a means to prevent erroneous G/L reporting. Use the **-restart** parameter in such cases to enable the utility to clean up the previous run and to finish processing successfully.

The utility continues processing from the point at which it left off when it stopped; therefore, no duplicate reports are generated.

-verbose

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

-help
Displays the syntax and parameters for this utility.

Results

If the utility displays an error message, check the utility log file to determine the source of the error and correct it.

If the utility encounters an error in **-export** mode, either rerun the **pin_ledger_report** utility in **-export** mode or allow the next scheduled run of the utility to continue processing the remaining reports. The **pin_ledger_report** utility will correctly process the remaining reports.

If the utility encounters an error during **-export** mode when the **-resend** or **-regenerate** parameter was used, run the utility again after correcting the errors based on the log file messages. If some of the reports were successfully re-exported or regenerated, run the utility again for the failed reports.