Minerva 10.5.4 LT Back-Office Billing and Reports Generation Guide

2019 May, 30 07:10 AM, PDT

Part number: 12587-00

About This Guide

This document describes how to use the Minerva 10.5.4 LT--Draft-- Billing Extraction Tool and the Java Billing Extraction Tool.

Intended Audience

Anyone who wants to extract billing data from the Minerva 10.5.4 LT--Draft-- Back-Office should use this guide.

User Name and Password Requirements

For User Names and Passwords for the system or procedure, log in to the Minerva Support Site and search for Knowledgebase article #36329.

The following user names and passwords are necessary for the execution of this procedure:

- The user name and password of the **Manager User**.

Billing Extraction

The Billing Extraction tool (a command line script) retrieves line item information about a set of customers' purchases and periodic charges and writes this information to one or more files ("invoice files") that you can read into your own OSS/BSS system, use to generate invoices, or use for whatever purpose you like.

This section contains the following topics:

- *Billing Data* looks at the data extracted by the tool.
- *Syntax* describes the tool's arguments and configuration file.
- *Invoking the Billing Extraction Tool* tells you where to find the tool and how to run it.
- *Invoice File Formats* is a specification, with examples, of the invoice file formats.

This section describes the legacy Minerva Billing Extraction tool.

For information on the Java Billing Extraction tool, refer to section *Java Billing Extraction Tool*.

Important:
Minerva has determined that if the asset-deletion database script runs concurrently with the billing extraction, record locking and high CPU usage from the script can cause the billing extraction process to fail. Minerva has developed workaround that reschedules the database script so that it starts at a later time. This allows the billing extraction process to complete before the database script starts.

To implement the workaround, please contact Minerva Support.

Billing Data

Recurring and Non-recurring Data

To understand how the Billing Extraction tool works, you must understand the data that it's extracting. There are two broad types of billing data (or *line items*), recurring and non-recurring:

- A **recurring** line item is the charge associated with a Service Package, Channel Package, Miscellaneous Service (Web, Email, etc), or SVOD, as well as the taxes and surcharges that you create through the Back-Office Console's **Billing > Setup** module. It's assumed that you (the IPTV Provider) will apply a customer's recurring charges on a regular basis for as long as that customer is subscribed to the particular service.
- A **non-recurring** line item is the debit or credit associated with a PPV purchase (including the subtypes, All Day Movie Ticket, Sports Package, and Pay Per Block), a VOD rental, or any of the invoice adjustments that you can impose through the **Billing > Invoices** modul e.

Event Times

Note:
All events are listed in local server time.

Every item has two "event times" associated with it: A "begin event time" and an "end event time." The begin event time is the moment that the item is available; the end event time is when it will no longer be available. Exactly what this means depends on the type of item:

- For VOD rentals the begin event time is the moment the user rents the VOD; the end event time is the end of the rental period (for that title).
- For Promo Offers, the begin event time is the moment the user purchases the promotional channel package; the end event is the end of the promo period.
- For PPV purchases, the event time is the start time of the PPV event in local server time; the end event time is unspecified.
- For all other non-recurring items (such as invoice corrections), the begin event time and the end event time are the same. It's the moment that the CSR submits the correction through the Back-Office Console.
- For recurring items, the begin event time is the moment that the customer buys (or, more accurately, "subscribes to") the item, and the end event time is the moment the customer unsubscribes.

Important:
To ensure accurate billing reports, Minerva strongly recommends that, for the Preferred Region, you set a Date Style that includes the year, such as one of these:

- MM/dd/yyyy
- MM/dd/yy
- dd/MM/yyyy
- dd/MM/yy
- yyyy/MM/dd
- MMM-ddyyyy

Syntax

The script that drives the Billing Extraction tool, **PREPARE_EXTRACT_SQL.sh**, is stored on the Back-Office database server, in the **/itvmgr/billing** directory. A configuration file, **PREPARE_EXTRACT_SQL.conf**, is stored in the same directory. By default, the tool writes the invoice files to the **/itvmgr/billing/data** directory.

The tool's syntax looks like this:

```
PREPARE_EXTRACT_SQL.sh -r timespan -t customers -p prefix -f format -s sub-operator
```

Only the ***timespan*** argument (**-r**) is required; the others adopt default values. The order of the arguments is not significant. The arguments, which are examined in the following sections, are:

- **-r *timespan***. The timespan that you want to look at.
- **-t *customers***. The set of customers for whom you want billing information.
- **-p *prefix***. A prefix that's added to the output file's name.
- **-f *format***. The format of the output file.
- **-s *sub-operator***. Specifies the sub-operator for which the report is generated.

In addition, you can use the configuration file to specify a location to which the invoice files will be transferred, as explained in *Transfer Location*.

The next sections describe the arguments and configuration file, the form of the name of the invoice file that the Billing Extraction tool generates, how to invoke the tool, and the error codes that the tool returns.

Timespan

The script's **-r** argument declares the timespan across which you want to extract billing information. The argument takes a keyword argument. There are five such keywords:

- **DAILY**, **WEEKLY**, and **MONTHLY** set the timespan to a single day, single week, and single month, respectively. Specifically:

  — **DAILY** means yesterday.
  — **WEEKLY** means the seven days ending yesterday.
  — **MONTHLY** means the span that started on today's day-of-the-month transposed to the previous month, and that ended yesterday. For example, if today's date is April 1st, **MONTHLY** sets the timespan to March 1st through March 31st.

Note:
- When you use the **DAILY**, **WEEKLY**, and **MONTHLY** parameters, you don't have to specify a date or date range. If you do, the dates are ignored. When you use these parameters, the date range is calculated automatically relative to the present time. To specify a specific date range, use the **OTHER** option, described just below.
- Don't be fooled by the adverbial forms of these keywords—they don't command the script to automatically run at a regular frequency. In other words, **MONTHLY** (for example) doesn't tell the script to run once a month. The keywords would be more appropriate as **DAY**, **WEEK**, and **MONTH**.

- **OTHER** lets you define the begin and end dates yourself, through two additional arguments:

-b sets the begin date
-e sets the end date
The date values take the form “YYYYMMDD”. For example, to set the timespan to include all items that “happened” on March 1, 2014 through March 31, 2014 (inclusive), you would configure the arguments thus:

```
PREPARE_EXTRACT_SQL.sh -r OTHER -b 20140301 -e 200070331 ...
```

- **NEARREALTM** (for “near real time”) is categorically different from the other timespan modes: It extracts non-recurring items (only) for yesterday and today, but it ignores items that have already been extracted. (For all other timespan modes, previously extracted items are not ignored.) Also, the contents of the **NEARREALTM** report, when in **TAB** format, lacks information that’s presented in the other modes (as explained in *File Format*).

In all cases, the invoice file will only contain line items that were “active” during the timespan:

- For non-recurring items, “active” means that the item’s “begin event time” (as defined in *Event Times*) must fall within the timespan.
- For recurring items, the timespan defined by the item’s begin event time and end event time must overlap—wholly or partially—with the timespan that you request here.

Important:
Currently, the tool returns all recurring items, even if the item’s event timespan doesn’t overlap the timespan that you request through the **-r** argument.

You must supply a **-r** value. The **-b** and **-e** arguments, if supplied, are ignored if the **-r** value is anything other than **OTHER**.

Customers

The script’s **-t** argument declares the set of customers for whom you want to extract billing information. The set is defined by one or more customer ID substrings. Multiple substrings are separated by whitespace (not commas) and the entire list is enclosed by quotes; for example, this invocation of the script...

```
PREPARE_EXTRACT_SQL.sh -t "ABC M1 R23" ...
```

...will extract information for those customers whose IDs contain “ABC”, “M1”, or “R23”. The comparison is case-insensitive, thus, all of the following example IDs would be found (with the substring shown in **bold**):

ABC_cust1
AbcCust2
L**abC**ustomer214
For**m1**CustomerA
Custome**r23**

If you don’t supply a **-t** value, billing information is extracted for *all* customers.

Filename Prefix

The **-p** argument lets you add a prefix to the name of the invoice file. The argument takes a single, unquoted string. For example, this invocation of the script...

```
PREPARE_EXTRACT_SQL.sh -p TEMP ...
```

...would add the string **TEMP** to the beginning of the invoice filename.

File Format

The script’s **-f** argument specifies the format of the output file. The argument takes a keyword argument. There are two format keywords:

- **TAB** produces information about customers, payments, STB accounts, STB devices, and invoice line items as pipe-delimited ASCII data. (“**TAB**” means “table”; it doesn’t mean the tab character).
- **ARB** produces information about PPV and VOD purchases in Arbor/BP format. This includes all flavors of PPV (normal, Pay Per Block, and All Day Movie Ticket) as well as VOD Bundles and VODs that are designated as Free Movies.

If you don’t supply a **-f** argument, **TAB** is used.

Specifications of the formats are given in section *Invoice File Formats* below.

Sub-operators

The script's **-s** argument declares the sub-operator for whom you want to extract billing information. Using the **-s** option, you can specify the sub-operator ID for which the report is generated. Only one report per sub-operator can be generated at a time.

> **Note:**
> The **-s** option can be used alongside all other options.

Example:

```
PREPARE_EXTRACT_SQL.sh -r DAILY -s 1
```

This will produce an output file named **SO_1-DAILY-20190305-20190305-TAB.txt**. The **-t** option is preserved and is working in combination with the **-s** option. The output is a sub-list of customers from the given sub-operator. Here the **SO_1** prefix indicates that this is the file for sub-operator 1.

Example usage:

```
./PREPARE_EXTRACT_SQL.sh -r DAILY -s 1 -t AUT
```

This will produce the following output file:

```
SO_1-AUT-DAILY-20190305-20190305-TAB.txt
```

> **Important:**
> Using the **-s** argument is optional. If you use the script without the **-s** option, you will generate data for all the customers from all sub-operators matching the other conditions.

Invoice Filename Form

For all timespan modes except **NEARREALTM**, the name of the invoice file that's generated by the Billing Extraction tool takes this form:

Sub-operator-prefixCustomer-timespan-startDate-endDate-format.txt

...where the components are:

- ***sub-operator*** is the value of the **-s** argument. Sub-operator is composed of **SO_{sub-operator-id}**.
- ***prefix*** is the value of the **-p** argument.
- ***Customer*** is the value of the **-t** argument. There's a separate file for each substring.
- ***timespan*** is the value of the **-r** argument (timespan).
- ***startData*** and ***endDate*** are the start and end dates of the timespan that's considered, in "YYYYMMDD" format.
- ***format*** is the value of the **-f** argument.

The dashes and "**TAB.txt**" are literal.

For **NEARREALTM** the ***startData*** and ***endDate*** components are replaced with the current date and time:

Sub-operator-prefixCustomer-timespan-YYYYMMDDHHMMSS-format.txt

The table below shows some examples. In all cases, it's assumed that the script is run on August 1, 2014, at 3:14 PM.

Arguments	Filename
-r DAILY -t abc -p PRE	PREabc-DAILY-20140731-20140731-TAB.txt
-r WEEKLY -p PRE -f ARB	PRE-WEEKLY-20140725-20140731-ARB.txt
-r NEARREALTM	NEARREALTM-20140801151400-TAB.txt
-r MONTHLY -t abc def ghi -f ARB	abc-MONTHLY-20140701-20140731-ARB.txt def-MONTHLY-20140701-20140731-ARB.txt ghi-MONTHLY-20140701-20140731-ARB.txt
-r DAILY -s 0 -t AUT	SO_0-AUT-DAILY-20140731-20140731-TAB.txt

Before the tool creates an invoice file, it checks to see if a file of the same name already exists. If it does exist, the tool moves the old file by adding "**.old**" to the name. Note that this doesn't apply to the invoice files that the tool transfers through FTP (as described in section *Transfer Location* below); the tool doesn't append "**.old**" to an existing file before transferring the new file.

> **Important:**

Currently, the Billing Extraction Tool creates an empty file if a file of the same name already exists.

Transfer Location

After the invoice file is generated, the script can automatically transfer it to an FTP site that you specify in the Billing Extraction Configuration file. The file, **PREPARE_EXTRACT_SQL.conf**, lives in the **/itvmgr/billing** directory on the Back-Office database server. The configuration file lets you create correspondences between sets of customers (as specified by the **-t** argument) and FTP sites such that when you run the Billing Extraction script for a particular set of customers, the invoice files for those customers are transferred to the corresponding FTP sites. If a customer set doesn't have an explicit FTP site, the invoice file is sent to a default site (which is also declared in the configuration file).

A default version of the Billing Extraction Configuration file is included with Back-Office.

The Billing Extraction Configuration file contains five types of entries:

- The **ITVMGR** entry is a hard-coded entry; don't change it.
- The **DEFAULT** entry contains FTP login information for the site that's used as the transfer destination for customer sets that don't have explicit entries.
- Another entry, which is optional, specifies FTP login information for specific sets of customers.
- Another optional entry specifies FTP login information for sub-operators.
- Additionally, for each sub-operator a different IP can be assigned for the **-t** option used. For example, if we invoke the script with **-s 2** and **-t AUT**, it will loop up for **SO_2_AUT** config file entry.

The **DEFAULT** entry looks like this:

```
DEFAULT|ftpIP|ftpUsername|ftpPassword|ftpDirectory|ftpTransferMode
```

...where the arguments provide info for the default FTP location:

- ***ftpIP*** is the IP address of the FTP server to which the invoice file will be transferred.
- ***ftpUsername*** and ***ftpPassword*** are the username and password that are used to log into the FTP server.
- ***ftpDirectory*** is the directory into which the invoice file be copied. The directory, which can be a path, is relative to the FTP server's top-level directory.
- ***ftpTransferMode*** lets you declare whether you want to transfer the data in ACII (**ascii**) or binary (**bin**) format.

A **customerSet** entry looks like the **DEFAULT** entry, except it names a customer set in the first field:

```
customerSet|ftpIP|ftpUsername|ftpPassword|ftpDirectory|ftpTransferMode
```

The **customerSet** value must match the value that you passed as the tool's **-t** argument. The comparison is case-insensitive.

Setting a different IP address for sub-operators is achieved by using **SO_{sub-operator id}** or **SO_{sub-operator id}_{keyword from option -t}**.

```
SO_{sub-operator
id}|ftp_ip|ftpUsername|ftpPassword|ftpDirectory|ftpTransferMode
```

The **SO_{sub-operator id}** value must match the value that you passed as the tool's **-s** argument. The comparison is case-insensitive.

```
SO_{sub-operator id}_{keyword from option
-t}|ftp_ip|ftpUsername|ftpPassword|ftpDirectory|ftpTransferMode
```

The **SO_{sub-operator id}_{keyword from option -t}** value must match the value that you passed as the tool's **-s** and **-t** arguments. The comparison is case-insensitive.

Example

The following is an example of a **PREPARE_EXTRACT_SQL.conf** file.

```
ITVMGR|billing|billing|x|x|x
DEFAULT|192.168.31.2|vtc|vtcpass|incoming|ascii
VTC|192.168.31.3|vtc|vtcpass|incoming|bin
ABC|192.168.31.4|abc|abcpass|incoming|ascii
SO_1|172.31.10.153|so|sopass|.|bin
SO_2|172.31.1.11|so|sopass|.|bin
SO_2_AUT|172.31.10.153|so|sopass|.|bin
```

For all billing reports, which do not match the configured **customerSet/SO_{sub-operator id}/SO_{sub-operator id}_{keyword from option -t}**, the DEFAULT option will be used.

Invoking the Billing Extraction Tool

The Billing Extraction tool script, **PREPARE_EXTRACT_SQL.sh**, lives in the **/itvmgr/billing** directory on the Back-Office database server. Although it's expected that you'll invoke the tool through a **cron** job and then use a parsing program to read the file that the tool creates, you can also invoke the tool by hand on the command line.

Invoking Billing Through a cron Job

To invoke the Billing Extraction tool through a **cron** job, you edit the **itvmgr crontab** file on the Back-Office database server. To edit the file, do the following:

1. Log into the Back-Office database server as the **Manager User**.
2. Open the file for editing:

```
$ crontab -e
```

3. As a convenience, the crontab file contains entries for daily, weekly, monthly, and near-real time invocation:

```
#DAILY billing report
#10 1 * * * sudo -u oracle /itvmgr/billing/PREPARE_EXTRACT_SQL.sh -r
DAILY

#WEEKLY billing report - day of week (0..6); 0 represents Sunday
#15 1 * * 0 sudo -u oracle /itvmgr/billing/PREPARE_EXTRACT_SQL.sh -r
WEEKLY

#MONTHLY billing report- day of month (01..31)
#20 1 01 * * sudo -u oracle /itvmgr/billing/PREPARE_EXTRACT_SQL.sh -r
MONTHLY

#Near-Real-Time billing report
#5 * * * * sudo -u oracle /itvmgr/billing/PREPARE_EXTRACT_SQL.sh -r
NEARREALTM
```

   To enable an entry, remove the # at the beginning of the line.
   Note that the default **crontab** entries create files in **TAB** format for all customers, with no file prefix. If you want a different format, a restricted range of customers, or a file prefix, you must edit the script's invocation.
4. When you've finished editing the file, close the **crontab** file.

Invoking the Tool from the Command Line

To invoke the tool from the command line, do this:

1. Log into the Back-Office database server as the **Manager User**.
2. Navigate to the script's directory:

```
$ cd /itvmgr/billing
```

3. Run the tool. Here, we ask for a week's worth of invoices.

```
$ sudo -u oracle /itvmgr/billing/PREPARE_EXTRACT_SQL.sh -r WEEKLY
report_type=WEEKLY, begin_date=20141128, end_date=20141204
output_file=/itvmgr/billing/data/WEEKLY-20141128-20141204-TAB.txt
...
```

   If the tool runs successfully, it will print statistics as shown above. If it's unsuccessful, it will print an error message (which we'll look at in the next section).

4. After the tool has finished, it sets a return code. To view the code, do this:

```
$ $?
0
```

   0 means success.
5. To see a listing of the file that the script has produced, do this:

```
$ cd data
$ ls -l
```

```
-rwxrwxrwx 1 oracle dba 5705 Sep 7 17:31
MONTHLY-20140807-20140906-TAB.txt
```

> **Note:**
> Minerva recommends that you invoke the Billing Extraction tool with a cron job.

Viewing the Log File

The billing log file is named **tt** and is in directory **/tmp**. View this file with a text editor to determine if any errors occurred during the billing run.

Invoice File Formats

The format of the invoice file that's created by the Billing Extraction tool depends on the value of the tool's **-f** argument:

- **TAB** produces line items as pipe-delimited ASCII data. ("**TAB**" means "table"; it doesn't mean the tab character).
- **ARB** produces information about PPV and VOD purchases Arbor/BP format. This includes all flavors of PPV (normal, Pay Per Block, and All Day Movie Ticket) as well as VOD Bundles and VODs that are designated as Free Movies.

The specifications for these formats are given in the sections, below. General formatting rules are given below:

- The data format, for both invoice file formats, is ASCII. Non-ASCII UTF-8 characters are properly represented as 8-byte character pairs, but since the file itself doesn't have a UTF-8 header, some text editors might not display the characters properly. If you're creating a file-parsing program, make sure you interpret a non-ASCII character (i.e. greater than 0xF7) as the first byte of a UTF-8 byte pair.
- Lines are terminated with **[CR][LF]** (0x0D, 0x0A).

TAB Format

TAB format invoice files contain ASCII data. Each line (or *record*) in an invoice file is a single, complete description of a customer, STB, line item, and so on. Records are never broken across lines—a newline character denotes the beginning of the next record.

Every attribute (or field) within a record, including the last field, is followed by the pipe character "|". For example:

```
CUSTOMER|MD11609|jas||jaswal|20131118-000000|20131118-000000|2150 gold
street|||Alviso|ca|95002||4085976765||Y|||
PAYMENTS|A|0|0|0|20131118-000000|20131118-000000|20131125-000000||||Credit
Card|||||1|1||||
LINEITEM|Monthly Charge|Internet|||
5.95|20131124-000000|20131125-181701|.35||||Y|||||
LINEITEM|Monthly Charge|HD|||
5.95|20131124-000000|20131125-181701|.35||||Y|||||
LINEITEM|Monthly Charge|hls|||
2.00|20131124-000000|20131125-181701|.12||||Y|||||
LINEITEM|Monthly Charge|Caller ID|||
5.95|20131124-000000|20131125-181701|.35||||Y|||||
LINEITEM|Monthly Charge|WHDVR|||
5.95|20131124-000000|20131125-181701|.35||||Y|||||
LINEITEM|Monthly Charge|DVR|||
5.95|20131124-000000|20131125-181701|.35||||Y|||||
LINEITEM|Monthly Charge|Email|||
5.95|20131124-000000|20131125-181701|.35||||Y|||||
LINEITEM|Monthly Charge|isdbt|||
2.00|20131124-000000|20131125-181701|.12||||Y|||||
LINEITEM|Purchase Pay-Per-View Event||Stand Up Guys||
2.00|20131126-010000||2|R|01:35|0/1/0003E63FA97E|||7|PPV|77.7|
LINEITEM|PPHOUR||Pay per hour charge - channel 6||
2.00|20131126-021116|20131126-031116|2|||0/1/0003E63FA97E|||6|PPB|6.6|
```

Note that all fields in a record are positionally represented, even those fields that don't have values. For example, in the record "**LINEITEM|Monthly Charge|2013|||11.95|...**", the fourth and fifth fields don't have values, but the fields themselves are still represented.

Record Types

Every record in the file has a type; the type is declared by the record's first field. There are five record types:

- A **CUSTOMER** records describes a single customer account. There can be more than one **CUSTOMER** record in the invoice file, but each represents a different customer.
- A **PAYMENT** record also contains customer account information as well as information about the Primary STB Account. You should think of it as a continuation of the immediately-preceding **CUSTOMER** record.
- A **LOGININF** record describes a Secondary STB Account. For each customer, there's a **LOGININF** record for each of that customer's STB Accounts.
- The Primary STB Account is not represented by a **LOGININF** record.
- An **STBSINFO** record describes a specific (STB) device. There can be more than one **STBSINFO** record (per customer) in the file.
- A **LINEITEM** record describes a single invoice line item (regardless of whether it's a charge or a credit). There must be at least one **LINEITEM** record per customer in the file. If the customer doesn't have any items that fall within the requested timespan, none of the records associated with that customer are written to the file.

File Organization

The CUSTOMER records also act to organize the file in that all of the other records pertain to the "current" customer; for example:

`CUSTOMER|012|||Mike|...`
`PAYMENTS|A|Mike|...`
`LOGININF|0|NR|NR|...`
`LOGININF|1|PG|TV17|...`
`STBSINFO||00020|...`
`LINEITEM|Monthly Charge|...`
`LINEITEM|Monthly Charge|...`
	Customer "Mike"

`CUSTOMER|M32|||Sue|...`
`PAYMENTS|A|Sue|...`
`LOGININF|0|NR|NR|...`
`STBSINFO||45020|...`
`LINEITEM|Monthly Charge|...`
`LINEITEM|Correction|...`
	Customer "Sue"

`CUSTOMER|B54T|||Hector|...`
`PAYMENTS|A|Hector|...`
`LINEITEM|VOD_RENTAL|...`
`LOGININF|3|NR|NR|...`
`STBSINFO|a123rt23|98080|...`
	Customer "Hector"

Also, keep in mind that if a customer doesn't have any billable events that fall within the requested timespan, none of the records associated with that customer will be written to the file. Thus, for every **CUSTOMER** record, there must be at least one **LINEITEM** record.

Value Formats

- **Date and time**. Date and time values are given in the form *"YYYYMMDD-HHMMSS"*. For example, "**20070301-231500**" represents 11:15 PM on March 1st, 2007.
- **Durations**. Durations are expressed as *"HH:MM"*.
- **Money**. Prices are depicted in the IPTV Provider's decimal format. "10.2" form (major denomination to ten places, minor denomination to two places) without a currency character. For example, if the major and minor denominations are dollars and cents, "**1234.56**" means $1,234.56.
- **NEARREALTM** mode doesn't return **LOGININF** or **STBSINFO** records.

Record Specifications

The tables, below list and describes the fields that are defined by the five record types. Some notes:

- The **#** column is the ordinal position of the field within the record.
- The **Size** column gives the maximum length of an attribute's value, in bytes.
- The **Name** column gives the field's brief description.
- The **Comments** field gives an expanded description of the field. Fields that are guaranteed to contain values are marked **GUARANTEED**. Unmarked fields are optional (the value might be empty).

Table 1. Customer Info Record, Part 1 (CUSTOMER), TAB Format

#	Size	Name	Comments
1	8	Record tag	**GUARANTEED** **CUSTOMER**
2	40	Customer ID	**GUARANTEED** The customer's unique ID. This is the ID value that the Back-Office database uses to identify the customer—it's not the optional external ID that you're invited to supply when you add the customer to the system. The customer record doesn't return the customer's external ID.
3	512	First name	**GUARANTEED** The customer's first name.
4	512	Middle name	The customer's middle name.
5	512	Last name	**GUARANTEED** The customer's last name.
6	15	Account creation date	**GUARANTEED** The date and time of the creation of the customer's account, in YYYYMMDD-HHMMSS format.
7	15	Duplicate of field #6	**GUARANTEED** Ignore the value in this field; it's an exact duplicate of field #6.
8	80	Address	**GUARANTEED** The customer's street address. The next two fields provide a continuation of the string, if this field isn't large enough.
9	80	Address (cont)	Continuation of field #8.
10	80	Address (cont)	Continuation of field #9.
11	40	City	**GUARANTEED** The customer's city.
12	40	State/Province	**GUARANTEED** The customer's state or province.
13	24	Zip code	**GUARANTEED** The customer's zip code.

14	50	Work phone	The customer's work phone number.
15	50	Home phone	The customer's home phone number.
16	80	Email	The customer's email address.
17	1	Login required	**GUARANTEED** Indicates whether the customer's STB presents the Login Screen. '**Y**' if it does, '**N**' if it doesn't.
18	24	Social security number	The customer's social security number.
19	50	Mobile phone	The customer's mobile phone number.

Table 2. Customer Info Record, Part 2 (PAYMENT), TAB Format

#	Size	Name	Comments
1	8	Record tag	**GUARANTEED PAYMENT**
2	1	Status	**GUARANTEED** The customer's account status, one of '**A**' for active, '**I**' for inactive, or '**D**' for deleted. Deleted means that the customer record has been marked for destruction, but the customer has a balance due. After the balance due is cleared, the customer record will be (completely) destroyed.
3	40	Login name	**GUARANTEED** The login name of the customer's primary STB Account.
4	40	Password	**GUARANTEED** The password of the customer's primary STB Account.
5	20	PIN	**GUARANTEED** The PIN, used to confirm PPV and VOD purchases, of the customer's STB Accounts (there's only one PIN value per customer).
6	15	Account creation date	**GUARANTEED** The date and time of the creation of the customer's primary STB Account, in YYYYMMDD-HHMMSS format.
7	15	Account activation date	The date and time when the customer's primary STB Account will be (or was) activated, in YYYYMMDD-HHMMSS format.
8	15	Account modification date	**GUARANTEED** The date and time of the most recent modification to the customer's primary STB Account, in YYYYMMDD-HHMMSS format.
9	15	Account expiration date	The date and time when the customer's primary STB Account will be (or was) deactivated, in YYYYMMDD-HHMMSS format.
10	-	unused	*unused.*
11	-	unused	*unused.*
12	15	Payment method	The customer's method for paying his or her bill. One of '**Credit Card**', '**Mail**', '**Pre Paid**', or '**Direct Debit**'.
13	20	Credit card type	The customer's credit card type. One of '**American Express**', '**Master Card**', '**Visa**', or '**Discover**'. This field is **GUARANTEED** if the payment method (field #13) is '**Credit Card**'.
14	40	Credit card number	The customer's credit card number. This field is GUARANTEED if the payment method (field #13) is 'Credit Card'.
15	15	Credit card expiration date	The date and time when the customer's credit card will expire deactivated, in YYYYMMDD-HHMMSS format. This field is **GUARANTEED** if the payment method (field #13) is '**Credit Card**'.
16	80	Credit card holder	The name on the customer's credit card account. This field is **GUARANTEED** if the payment method (field #13) is '**Credit Card**'.
17	5	Region ID	Integer value that identifies the customer's region.
18	3	Bank ID	Integer value that identifies the customer's bank. Bank IDs are created through the Back-Office Console's **Billing > Bank Mgmt** module.
19	40	Bank account number	The customer's bank account number.
20	40	Bank account holder	The name on the customer's bank account.
21	40	Bank name	The name of the customer's bank.

Table 3. STB Account Record (LOGININF), TAB Formatt

#	Size	Name	Comments
1	8	Record tag	**GUARANTEED** **LOGININF**
2	40	Login name	**GUARANTEED** The login name of the secondary STB Account.
3	10	Movie rating	**GUARANTEED** The account's Parental Control movie rating, given as a string name.
4	5	TV rating	**GUARANTEED** The account's Parental Control tv rating, given as a string name.
5	1	Has email	**GUARANTEED** '**Y**' if the account has email service, otherwise '**N**'.
6	1	Has internet	**GUARANTEED** '**Y**' if the account has internet service, otherwise '**N**'.
7	16	Allowance	**GUARANTEED** The account's spending limit (for VOD and PPV purchases) given "10.2" form and without a currency symbol.
8	32	Account ID	The STB Account ID.

Important:
NEARREALTM mode doesn't return LOGININF records.

Table 4. STB Info Record (STBSINFO), TAB Format

#	Size	Name	Comments
1	8	Record tag	**GUARANTEED** **STBSINFO**
2	80	Serial number	The device's serial number.
3	80	MAC address	The device's MAC address.
4	64	IP address	The device's IP address.

Important:
NEARREALTM mode doesn't return STBINFO records.

Table 5. Line Item Record (LINEITEM), TAB Format

#	Size	Name	Comments
1	8	Record tag	**GUARANTEED** **LINEITEM**
2	80	Item type	**GUARANTEED** The type of item. There are two recurring item types: • '**Monthly Charge**' is a recurring charge for a service such as a Service Package. Channel Package, SVOD Package, Web or Email access, and so on. • '**Taxes and Surcharges**' is a recurring tax or surcharge. The rest of the types are non-recurring items: • '**VOD_Rental**' is the rental of a single VOD asset. • '**VOD_Bundle**' is the rental of a VOD bundle. • '**Promo_Offer**' is the purchase of a promotional channel package. • '**Purchase Pay-Per-View Event**' is the purchase of a Pay Per View program. • '**PPHOUR**' is the purchase of a Pay Per Block block. • '**Purchase All Day Movie Ticket**' is the purchase of an All Day Movie Ticket. • '**Purchase Sports Package Ticket**' is the purchase of a Sports Package. • '**Purchase Special Offer**' is the purchase of a Special Offer. • '**Get Free Movie**' is a free movie—there should be no charge associated with this item. These non-recurring items represent invoice corrections: • '**Service Adjustment**' • '**Bill Adjustment**' • '**Manager Adjustment**' • '**Late Payment Charge**' • '**VOD Adjustment**'
3	40	Non-VOD/PPV/Promo Offer description	A description of the item (only items which are not VOD, PPV, or promotional channel packages): • For Service Packages, Channel Packages, SVOD packages, and miscellaneous services (email, Web, DVR access, etc), this is the name of service as set through the appropriate Services & Pricing submodule in the Back-Office Console. • For invoice corrections, this is the name of the Back-Office Console account that made the correction. • For VOD, PPV, promotional channel package purchases, this field is empty.

4	120	VOD/PPV/Promo Offer description	The title of the VOD asset, PPV program, or promotional channel package. For all other item types, this field is empty.
5	32	VOD ID	The (internal) ID of the VOD asset or VOD bundle. For non-VOD items, this field is empty.
6	15	Charge amount	**GUARANTEED** The debit or credit amount associated with the item, in “10.2” form without a currency symbol. If the value is positive, it’s a debit; if it’s negative, it’s a credit.
7	15	Begin event time	**GUARANTEED** This is the later of: • The item’s begin event time as defined in the *Event Times* section, or... • the beginning of the timespan that you requested through the **-r** argument. The value is given in *YYYYMMDD-HHMMSS* format.
8	15	End event time	The end event time, as defined in the *Event Times* section. If the end event time is in the future, then this is the date and time that you invoked the script. The value is given in *YYYYMMDDHHMMSS* format. Note that PPV items and Promo Offers don’t have an end time value.
9	-	ignore this field	ignore this field
10	10	PC rating	The movie rating assigned to the VOD asset or TV rating assigned to the PPV program. For all other item types, this field is empty. **Note:** Prior to the 5.7 SP4 HP1 release, the Billing Extraction tool returned movie ratings for PPV movies instead of TV ratings.
11	5	Duration	The duration of the VOD asset or PPV program. For all other item types, this field is empty. The value is given in HH:MM (hours and minutes) format.
12	40	STB account	For STB-initiated VOD and PPV purchases only, this is the identity of the STB account that purchased the item, given as: *stbAccountLogin/stbDeviceID/stbMACAddress* ...where: • ***stbAccountLogin*** is the login name of the account that purchased the item. • ***stbDeviceID*** is the Back-Office device ID of the STB through which the purchase was made. The device ID is a system-wide unique number that’s generated when the device is added to the system. • ***stbMACAddress*** is the MAC address of the STB through which the purchase was made.
13	1	Currently active	For recurring charges only, this field is ‘**Y**’ if the item is still active at the time that the tool is run, and ‘**N**’ otherwise. For Promo Offers, the field is empty.
14	123	Provider’s VOD asset ID	For VOD purchases only, a unique ID that identifies the VOD asset across all VOD providers. The ID is a combination of the VOD provider’s name (or other identifier, such as their website URL) and a 20-character VOD asset ID, in the form “*providerName* + *providerAssetID*” (the plus sign and surrounding spaces are literal). For example: **tvn.com + TVNX2000546781930001**
15	16	Channel number	The number of the channel that the PPV program was viewed on.
16	80	Channel name	The name of the channel that the PPV program was viewed on.
17	16	Channel number (dot notation)	If dot notation is enabled, the channel in dot notation format.
18	30	Provider Name	(5.7 SP4 HP1 and later) The name of the provider, from the ADI.XML metadata.
19	5	Billing ID	(5.7 SP4 HP1 and later) The billing ID, from the ADI.XML metadata.

ARB Format

The Arbor/BP (**ARB**) invoice file contains fixed-length records in ASCII. The format presents information about VOD and PPV purchases only. This includes all types of PPV (normal, Pay Per Block, All Day Movie Ticket, Sports Package), as well as VOD Bundles and VODs that are designated as Free Movies.

Each line in an **ARB** invoice file represents a single record; there are three record types, as explained in the next section. A single record contains some number of fixed-length fields.

The following is an example of an **ARB** invoice file:

```
HDR20131125181356000000000000billingDataExtrc5.7.0.0.21.N.
VS220131126021327No CA ID 2 005020000001Stand Up Guys
000020010080003E63FA97E   S. Num. missingMD11609    0    4085976765
external_customer_id missing      1    2    PPVEVENT
Purchase Pay-Per-View Event   R
67929MV00416726000020131126010000   20131126010000     7
PPV                                                          77.7
VS220131126021116No CA ID 2 005020000001Pay per hour charge - ch
000020010080003E63FA97E S. Num. missingMD11609   0    4085976765
external_customer_id missing        1        1         PPHOUR
Purchase channel by hour          RNA          20131126031116
20131126021116     6    PPB
6.6
TRA20131125181357000000000002billingDataExtrc5.7.0.0.21.N.
~
```

Record Types

In **ARB** format there are three record types:

- The first record, marked in the file as **HDR**, is a header that supplies system information.
- After the header comes some number of line item records, marked as **VS2**. Each line item records represents a single PPV or VOD transaction.
- The final record is a trailer, marked as **TRA**. The trailer contains, in addition to all the information that's in the header, a field that gives the number of line item records in the file.

Value Formats

- **Strings**. String values are left-justified within their fields and right-padded with spaces.
- **Numbers**. Numeric values are left-padded with zeros to fill the field.
- **Date and time**. Date and time values are given in the form "YYYYMMDDHHMMSS". For example, "**20070301231500**" represents 11:15 PM on March 1st, 2007.
- **Money**. Prices are depicted as *"MMMMMnn"*, where *MMMMM* represents the major denomination to five places and *nn* is the minor denomination to two places. Note that there is no decimal point character. For example, if the major and minor denominations are dollars and cents, "**0123456**" means a debit of $1,234.56.
- Record Specifications

The tables below list and describe the fields that are defined by the three record types. Some notes:

- The **Start** column is the ordinal position of the first byte of the field within the statement, starting with byte 0.
- The **Size** column gives the size of the field, in bytes.
- The **Name** column gives the field's brief description.
- The **Size** column gives the maximum length of an attribute's value, in characters.
- The **Comments** field gives an expanded description of the field. If the field is marked **NUMBER**, the value is formatted as a number (left-padded with zeros), otherwise it's formatted as a string (left-justified and right-padded with spaces).

Table 6. Header Record (HDR), ARB Format

Start	Size	Name	Comments
0	3	Record tag	HDR
3	14	Current time	The current date and time of day, in *YYYYMMDDHHMMSS* format.
17	11	Unused	Ignore the value in this field.
28	16	Application name	The name of the Billing Extraction Tool.
44	10	Application version	The version of the Billing Extraction Tool. There is a known issue in which 1) 14 "space" characters are added after the last character of the Application Version string, and 2) the actual length of the Application Version string is can vary from the documented 10 characters. Hence, total header length can vary as the Application Version string length varies.

Table 7. Line Item Record (VS2), ARB Format

Start	Size	Name	Comments
0	3	Record tag	VS2
3	14	Transaction time	The date and time of the transaction, in *YYYYMMDDHHMMSS* format.
17	48	Smart Card ID	The ID of the Smart Card that was on the STB at the time of the purchase. If the STB didn't have a Smart Card at the time of the purchase but does now, the current Smart Card ID is used. In the complete absence of a Smart Card ID, the string "**No CA ID**" or "**SmartCard ID missing**" is used—the two strings mean the same thing.
65	14	Unused	Ignore the value in this field.
79	24	Movie/Program title	The title of the movie, program, or package that the customer purchased.
103	1	Purchase/Credit	If the item is a purchase, this field will be empty. If it's a credit, the field will be a minus sign ("-").
104	7	Price	**NUMBER** The amount of the purchase or credit in *MMMMMnn* format. Note that there's no decimal point character, and the value is left-padded with zeros. For example "**0001295**" means (in dollars) $12.95.
111	4	Unused	Ignore the value in this field.
115	15	MAC address	The MAC address of the STB through which the purchase was made. If the value can't be found, "**MAC ID missing**" is used.
130	15	Serial number	The serial number of the STB through which the purchase was made. If the value can't be found, "**S. Num. missing**" is used.
145	40	Customer ID	The ID of the customer who made the purchase.
185	40	STB login	The login name of the STB account that made the purchase.
225	20	Phone number	The customer's phone number. If this information is missing, "**Home Phone missing**" is used.
245	40	External customer ID	The customer's "external ID". This is an externally-generated ID (a string) that can be attached to the customer's account when the account is created. If the customer doesn't have an external ID, or if the external ID is being used as the customer's main ID, "**external_customer_id missing**" is used.
285	20	Device ID	The device ID of the STB through which the purchase was made. This is the unique ID that's generated by Back-Office when a record of an STB is added to the database. If the ID can't be found, "**device_id missing**" is used.
305	20	Transaction ID	A unique ID, generated by Back-Office, that identifies this purchase. If the ID can't be found, "**purchase_id missing**" is used.
325	20	Asset type	A string that represents the type of asset that was purchased. This is one of: • **VOD** means a normal Video On Demand. • **FREEMOVIE** is a VOD that's specifically offered as a free movie. • **BUNDLE** is a VOD Bundle. • **PPVEVENT** is a normal PPV program. • **PPVBLOCK** is a Pay Per Block program. • **PPHOUR** is also Pay Per Block program. • **ADMT** is an All Day Movie Ticket. • **SEASONPASS** is a Sports Package. • **PROMOCHPKG** is a Promo Offer. If the information is missing, "**transaction_code missing**" is used.
345	30	Unused	Ignore the value in this field.
375	10	PC Rating	The Parental Controls-style rating for the asset. If the rating is missing, "**RNA**" is used.
385	20	VOD asset ID	For VOD purchases only, this is the unique ID, generated by Back-Office, that identifies the VOD asset.
405	20	Rental period end	The date and time when the rental period will (or did) expire, in *YYYYMMDDHHMMSS* format.
425	50	PPV program ID	For PPV purchases only, this is the unique ID, generated by Back-Office, that identifies the specific occurrence of the PPV program that was purchased.
475	14	Rental period start	The date and time when the rental period will (or did) start, in *YYYYMMDDHHMMSS* format.
489	123	Provider's VOD asset ID	For VOD purchases only, a unique ID that identifies the VOD asset across all VOD providers. The ID is a combination of the VOD provider's name (or other identifier, such as their website URL) and a 20-character VOD asset ID, in the form "*providerName* + *providerAssetID*" (the plus sign and surrounding spaces are literal). For example: **tvn.com + TVNX2000546781930001**
612	16	Channel number	The number of the channel that the PPV program was viewed on.

628	80	Channel name	The name of the channel that the PPV program was viewed on.
708	16	Channel number (dot notation)	If dot notation is enabled, the channel in dot notation format.
724	20	Promo offer ID	For PROMOCHPKG purchases only, this is the unique ID generated by Back-Office that identifies the specific occurrence of the PROMO that was purchased. It will be reused for other types of PROMO in the future.
744	20	Promo ID	For PROMOCHPKG purchases only, this is the unique ID generated by Back-Office that identifies the specific occurrence of the channel package connected to the PROMOCHPKG. It will be reused for other types of PROMO in the future.
764	20	Promo Code	For PROMOCHPKG purchases only, the promo's code. This is an externally-generated ID (a string) that can be attached to the promo offer. If the promo doesn't have an external code, this field will be empty. It will be reused for other types of PROMO in the future.

Table 8. Trailer Record (TRA), ARB Format

Start	Size	Name	Comments
0	3	Record tag	TRA
3	14	Current time	The current date and time of day, in *YYYYMMDDHHMMSS* format.
17	4	Unused	Ignore the value in this field.
21	7	Number of line items	**NUMBER** The number of line item records in the file. The value is right-justified and padded with leading zeros. For example, if the file contains three line item records, the value will be "0000003".
28	16	Application name	The name of the Billing Extraction Tool.
44	10	Application version	The version of the Billing Extraction Tool. There is a known issue in which 1) 14 "space" characters are added after the last character of the Application Version string, and 2) the actual length of the Application Version string is can vary from the documented 10 characters. Hence, total footer length can vary as the Application Version string length varies.

Java Billing Extraction Tool

The Java Billing Extraction tool is a command line script that retrieves line item information about a set of customers' purchases and periodic charges and writes this information to one or more files ("invoice files") that you can read into your own OSS/BSS system, use to generate invoices, or use for whatever purpose you like.

This chapter contains the following sections:

- *Tool Command Line Syntax and Output Files* describes the Java Billing Extraction tool's arguments, configuration file, and output location.
- *Invoking the Java Billing Extraction Tool* tells you where to find the Java Billing Extraction tool and how to run it.
- *Invoice File Formats* is a specification, with examples, of the invoice file formats.

Note:
For explanation of Billing Data, including the concept of Event Times, refer to section *Billing Data*.

Important:
The Java Billing Extraction tool does not support:

- the Tab file format (for recurring data),
- automatic transfer of reports to an FTP site, and
- scheduled script invocation through cron jobs.

If you require these features, use the legacy Billing Extraction tool described in section *Billing Extraction*.

Tool Command Line Syntax and Output Files

The Java Billing Extraction tool's syntax looks like this:

```
BILLING_EXTRACT.sh -r timespan -t customers -p prefix -f ARB
```

The timespan argument (-r) is required, as well as the format argument (-f ARB); the others adopt default values. The order of the arguments is unimportant. The arguments, which are examined in the following sections, are:

- **-r** *timespan*. The timespan that you want to look at.
- **-t** *customers*. The set of customers for whom you want billing information.
- **-p** *prefix*. A prefix that's added to the output file's name.

The next sections describe the arguments and configuration file, the form of the name of the invoice file that the Java Billing Extraction tool generates, how to invoke the tool, and the error codes that the tool returns.

Timespan

The script's -r argument declares the timespan across which you want to extract billing information. The argument takes a keyword argument. There are five such keywords:

Keywords DAILY, WEEKLY, and MONTHLY

The **DAILY**, **WEEKLY**, and **MONTHLY** keywords set the timespan to a single day, single week, and single month, respectively. Specifically:

- **DAILY** means yesterday.
- **WEEKLY** means the seven days ending yesterday.
- **MONTHLY** means the span that started on today's day-of-the-month transposed to the previous month, and that ended yesterday. For example, if today's date is April 1st, MONTHLY sets the timespan to March 1st through March 31st.

Note:

- When you use the DAY, WEEK, and MONTH parameters, you don't have to specify a date or date range. If you do, the dates are ignored. When you use these parameters, the date range is calculated automatically relative to the present time. To specify a specific date range, use the OTHER option, described just below.
- Don't be fooled by the adverbial forms of these keywords—they don't command the script to automatically run at a regular frequency. In other words, MONTHLY (for example) doesn't tell the script to run once a month. The keywords would be more appropriate as DAY, WEEK, and MONTH.

Keyword OTHER

The OTHER keyword lets you define the begin and end dates yourself, through two additional arguments:

- **-b** sets the begin date
- **-e** sets the end date

The date values take the form "YYYYMMDD". For example, to set the timespan to include all items that "happened" on March 1, 2014 through March 31, 2014 (inclusive), you would configure the arguments thus:

```
BILLING_EXTRACT.sh -r OTHER -b 20140301 -e 200070331 ...
```

Keyword NEARREALTM

The **NEARREALTM** (for "near real time") keyword extracts non-recurring items for yesterday up to current time today.

General Usage Notes

- In all cases, the invoice file will only contain line items that were "active" during the timespan. "Active" means that the item's "begin event time" must fall within the timespan. For more information on event times, refer to section *Event Times*.
- You must supply an -r value. The **-b** and **-e** arguments, if supplied, are ignored if the **-r** value is anything other than OTHER.

Customers

The script's **-t** argument declares the set of customers for whom you want to extract billing information. The set is defined by one or more customer ID substrings. Multiple substrings are separated by white space (not commas) and the entire list is enclosed by quotes; for example, this invocation of the script...

```
BILLING_EXTRACT.sh -t ABC M1 R23 ...
```

...will extract information for those customers whose IDs contain "ABC", "M1", or "R23". The comparison is case-insensitive, thus, all of the following sample IDs would be found (with the substring shown in **bold**):

- **ABC**_cust1
- **Abc**Cust2
- L**abC**ustomer214
- For**m1**CustomerA
- Customer**23**

If you don't supply a **-t** value, billing information is extracted for all customers.

Filename Prefix

The **-p** argument lets you add a prefix to the name of the invoice file. The argument takes a single, unquoted string. For example, this invocation of the script...

```
BILLING_EXTRACT.sh -p TEMP ...
```

...would add the string **TEMP** to the beginning of the invoice filename.

Invoice File Locations

The Java Billing Extraction tool directs report output files to subdirectories under the /itvmgr/billing /data directory that are named according the report's timestamp-based run ID. All reports and logs for each run are placed in the unique directory for that run.

> **Note:**
> The output directories are not automatically maintained. Archive and remove old directories according to your report maintenance policy.

Invoice File Names

For all timespan modes except NEARREALTM, the name of the invoice file that's generated by the Java Billing Extraction tool takes this form:

prefixCustomer-timespan-startDate-endDate-format.txt

where:

- ***prefix*** is the value of the -p argument.
- ***Customer*** is the value of the -t argument. There's a separate file for each substring.
- ***timespan*** is the value of the -r argument (timespan).
- ***startData*** and endDate are the start and end dates of the timespan that's considered, in "YYYYMMDD" format.
- ***format*** is the value of the **-f** argument.

For **NEARREALTM** the ***startData*** and ***endDate*** components are replaced with the current date and time:

prefixCustomer-timespan-YYYYMMDDHHMMSS-format.txt

The table below shows some examples. In all cases, it's assumed that the script is run on August 1, 2014, at 3:14 PM.

Arguments	Filename
`-r DAILY -t abc -p PRE -f ARB`	`PREabc-DAILY-20140731-20140731-ARB.txt`
`-r WEEKLY -p PRE -f ARB`	`PRE-WEEKLY-20140725-20140731-ARB.txt`
`-r NEARREALTM -f ARB`	`NEARREALTM-20140801151400-ARB.txt`

`-r MONTHLY -t abc def ghi -f ARB`	`abc-MONTHLY-20140701-20140731-ARB.txt`
	`def-MONTHLY-20140701-20140731-ARB.txt`
	`ghi-MONTHLY-20140701-20140731-ARB.txt`

Invoking the Java Billing Extraction Tool

The Java Billing Extraction tool script, **BILLING_EXTRACT.sh**, is in the **/itvmgr/billing** directory on the server.

> **Note:**
> The Java Billing Extraction tool can't be invoked through a cron job. Support for scheduled script launch will be added in a later release.

Invoking the Java Billing Extraction Tool

To invoke the Java Billing Extraction tool from the command line, do this:

1. Ensure that IP_PORT property has the correct IP address:
   a. Open the **/itvmgr/billing/conf/billing.properties** file in a text editor such as **vi**.
   b. Locate the **IP_PORT** property.
   c. If required, change the value to the IP address of the database containing transactions to be extracted.

   > **Important:**
   > Note that the port value of '1521' should remain unchanged. Only the IP address portion of the value needs to be verified.

   For example, if the database is located at 192.168.10.10, ensure that IP_PORT property is set as follows:

   ```
   IP_PORT=192.168.10.10:1521
   ```

2. Log in to the Back-Office application server as the **Manager User**, if you aren't logged in as the **Manager User** already.
3. Navigate to the script's directory:

   ```
   $ cd /itvmgr/billing
   ```

4. Run the Java Billing Extraction tool. Here, we ask for a week's worth of invoices.

   ```
   $ sudo -u oracle ./BILLING_EXTRACT.sh -r WEEKLY -f ARB
   ```

   ```
   report_type=WEEKLY
   report_type=WEEKLY, begin_date=20140208, end_date=20140214
   output_file=WEEKLY-20140208-20140214-ARB.txt
   run id: 20140215011215, begin date: 20140208, end date: 20140214
   ```

   > **Important:**
   > You must supply the -f argument and use the ARB as value, as in the example above. Otherwise, the output file might not be correct.

   If the Java Billing Extraction tool runs successfully, it will print statistics as shown above. If it's unsuccessful, it will print an error message (which we'll look at in the next section).
   Note the "run id" shown above. The "run id" is a timestamp-based unique ID assigned each time the utility is invoked. As discussed earlier, each run produces a subdirectory named by the run id under the data directory.
5. View file(s) produced by the Java Billing Extraction tool:

```
$ cd /itvmgr/billing/
$ ll data
...
total 16
$ cd data
$ ll
drwxrwxrwx 2 oracle dba 4096 Sep 20 14:49 20140920144909
drwxrwxrwx 2 oracle dba 4096 Sep 20 14:51 20140920145123
$ cd 20140920144909
$ ll
total 8
$
```

The **BillLog.txt** log file contains logging information specific to this particular run.

6. Archive or remove old subdirectories according to your data retention policy.

Invoice File Formats

Invoice files created by this release of the Java Billing Extraction tool are in the ARB format. This section provides the specifications for the format.

General formatting rules are:

- The data format is ASCII. Non-ASCII UTF-8 characters are properly represented as 8-byte character pairs, but since the file itself doesn't have a UTF-8 header, some text editors might not display the characters properly. If you're creating a file-parsing program, make sure you interpret a non-ASCII character (that is, greater than 0xF7) as the first byte of a UTF-8 byte pair.
- Lines are terminated with [CR][LF] (0x0D, 0x0A).

The ARB format produces information about PPV and VOD purchases and billing adjustments in Arbor/BP format. This includes all flavors of PPV (normal, Pay Per Block, and All Day Movie Ticket) as well as VOD Bundles and VODs that are designated as Free Movies.

ARB Format Specifications

The Arbor/BP (**ARB**) invoice file contains fixed-length records in ASCII. The format presents information about VOD, SVOD, and PPV purchases only. This includes all flavors of PPV (normal, Pay Per Block, All Day Movie Ticket, Sports Package), as well as VOD Bundles and VODs that are designated as Free Movies.

Each line in an **ARB** invoice file represents a single record; there are three record types, as explained in the next section. A single record contains some number of fixed-length fields.

The following is an example of an **ARB** invoice file:

```
HDR20131125181356000000000000billingDataExtrc5.7.0.0.21.N.
VS220131126021327No CA ID 2 005020000001Stand Up Guys
000020010080003E63FA97E S. Num. missingMD11609 0 4085976765
external_customer_id missing 1 2 PPVEVENT
Purchase Pay-Per-View Event R
67929MV00416726000020131126010000 20131126010000 7
PPV 77.7
VS220131126021116No CA ID 2 005020000001Pay per hour charge - ch
000020010080003E63FA97E S. Num. missingMD11609 0 4085976765
external_customer_id missing 1 1 PPHOUR
Purchase channel by hour RNA 20131126031116
20131126021116 6 PPB
6.6
TRA20131125181357000000000002billingDataExtrc5.7.0.0.21.N.
~
```

Record Types

In **ARB** format there are three record types:

- The first record, marked in the file as **HDR**, is a header that supplies system information.

- After the header comes some number of line item records, marked as **VS2**. Each line item records represents a single PPV or VOD transaction.
- The final record is a trailer, marked as **TRA**. The trailer contains, in addition to all the information that's in the header, a field that gives the number of line item records in the file.

Value Formats

- **Strings**. String values are left-justified within their fields and right-padded with spaces.
- **Numbers**. Numeric values are left-padded with zeros to fill the field.
- **Date and time**. Date and time values are given in the form "*YYYYMMDDHHMMSS*". For example, "**20140301231500**" represents 11:15 PM on March 1st, 2014.
- **Money**. Prices are depicted as "MMMMMnn", where MMMMM represents the major denomination to five places and nn is the minor denomination to two places. Note that there is no decimal point character. For example, if the major and minor denominations are dollars and cents, "**0123456**" means a debit of $1,234.56.
- Record Specifications

The tables, below list and describes the fields that are defined by the five record types. Some notes:

- The **Start** column is the ordinal position of the first byte of the field within the statement, starting with byte 0.
- The **Size** column gives the size of the field, in bytes.
- The **Name** column gives the field's brief description.
- The **Size** column gives the maximum length of an attribute's value, in characters.
- The **Comments** field gives an expanded description of the field. If the field is marked **NUMBER**, the value is formatted as a number (left-padded with zeros), otherwise it's formatted as a string (left-justified and right-padded with spaces).

Table 1. Header Record (HDR), ARB Format

Start	Size	Name	Comments
0	3	Record tag	HDR
3	14	Current time	The current date and time of day, in *YYYYMMDDHHMMSS* format.
17	11	Unused	Ignore the value in this field.
28	16	Application name	The name of the Java Billing Extraction tool.
44	10	Application version	The version of the Java Billing Extraction tool.

Table 2. Line Item Record (VS2), ARB Format

Start	Size	Name	Comments
0	3	Record tag	VS2
3	14	Transaction time	The date and time of the transaction, in *YYYYMMDDHHMMSS* format.
17	48	Smart Card ID	The ID of the Smart Card that was on the STB at the time of the purchase. If the STB didn't have a Smart Card at the time of the purchase but does now, the current Smart Card ID is used. In the complete absence of a Smart Card ID, the string "**No CA ID**" or "**SmartCard ID missing**" is used—the two strings mean the same thing.
65	14	Unused	Ignore the value in this field.
79	24	Movie/Program title	The title of the movie, program, or package that the customer purchased.
103	1	Purchase/Credit	If the item is a purchase, this field will be empty. If it's a credit, the field will be a minus sign ("-").
104	7	Price	**NUMBER** The amount of the purchase or credit in *MMMMMnn* format. Note that there's no decimal point character, and the value is left-padded with zeros. For example "**0001295**" means (in dollars) $12.95.
111	4	Unused	Ignore the value in this field.
115	15	MAC address	The MAC address of the STB through which the purchase was made. If the value can't be found, "**MAC ID missing**" is used.
130	15	Serial number	The serial number of the STB through which the purchase was made. If the value can't be found, "**S. Num. missing**" is used.
145	40	Customer ID	The ID of the customer who made the purchase.
185	40	STB login	The login name of the STB account that made the purchase.
225	20	Phone number	The customer's phone number. If this information is missing, "**Home Phone missing**" is used.
245	40	External customer ID	The customer's "external ID". This is an externally-generated ID (a string) that can be attached to the customer's account when the account is created. If the customer doesn't have an external ID, or if the external ID is being used as the customer's main ID, "**external_customer_id missing**" is used.

285	20	Device ID	The device ID of the STB through which the purchase was made. This is the unique ID that's generated by the Back-Office when a record of an STB is added to the database. If the ID can't be found, "**device_id missing**" is used.
305	20	Transaction ID	A unique ID, generated by the Back-Office, that identifies this purchase. If the ID can't be found, "**purchase_id missing**" is used.
325	20	Asset type	A string that represents the type of asset that was purchased. This is one of: • **VOD** means a normal Video On Demand. • **FREEMOVIE** is a VOD that's either specifically offered as a free movie, or that has no price. • **BUNDLE** is a VOD Bundle. • **SVOD** is a Subscription VOD. • **PPVEVENT** is a normal PPV program. • **PPVBLOCK** is a Pay Per Block program. • **PPHOUR** is also Pay Per Block program. • **ADMT** is an All Day Movie Ticket. • **SEASONPASS** is a Sports Package. If the information is missing, "**transaction_code missing**" is used.
345	30	Unused	Ignore the value in this field.
375	10	PC Rating	The Parental Controls-style rating for the asset. If the rating is missing, "**RNA**" is used.
385	20	VOD asset ID	For VOD purchases only, this is the unique ID, generated by the Back-Office, that identifies the VOD asset.
405	20	Rental period end	The date and time when the rental period will (or did) expire, in *YYYYMMDDHHMMSS* format.
425	50	PPV program ID	For PPV purchases only, this is the unique ID, generated by the Back-Office, that identifies the specific occurrence of the PPV program that was purchased.
475	14	Rental period start	The date and time when the rental period will (or did) start, in *YYYYMMDDHHMMSS* format.
489	123	Provider's VOD asset ID	For VOD purchases only, a unique ID that identifies the VOD asset across all VOD providers. The ID is a combination of the VOD provider's name (or other identifier, such as their website URL) and a 20-character VOD asset ID, in the form "*provider Name* + *providerAssetID*" (the plus sign and surrounding spaces are literal). For example: **tvn.com + TVNX2000546781930001**
612	16	Channel number	The number of the channel that the PPV program was viewed on.
628	80	Channel name	The name of the channel that the PPV program was viewed on.
708	16	Channel number (dot notation)	If dot notation is enabled, the channel in dot notation format.

Table 3. Trailer Record (TRA), ARB Format

Start	Size	Name	Comments
0	3	Record tag	TRA
3	14	Current time	The current date and time of day, in *YYYYMMDDHHMMSS* format.
17	4	Unused	Ignore the value in this field.
21	7	Number of line items	**NUMBER** The number of line item records in the file. The value is right-justified and padded with leading zeros. For example, if the file contains three line item records, the value will be "0000003".
28	16	Application name	The name of the Java Billing Extraction tool.
44	10	Application version	The version of the Java Billing Extraction tool.

Appendix A: Using the Billing Period/Day Reset Script

You use the Billing Period and Billing Day Reset Script to reset the Back-Office's Billing Period and Billing Day values. Normally you set these values when you initially set up your the Back-Office system. After this time, the settings can be viewed, but not changed, in the Back-Office Console **Billing > Setup** interface:

The script allows you to change these settings.

Note:
Changes to the Billing Period and Billing Day values have no effect on the Billing extraction reports.

Requirements

Software Requirements

The distribution files **SetBilling.sh** and **SetBilling.sql**.

Changing the Billing Period and Billing Day Values

To change the Billing Period and Billing Day values, do the following:

Note:
Changes to the Billing Period and Billing Day values have no effect on the Billing extraction reports.

- If the reset Billing Period and Billing Day values result in requiring additional invoices to be archived, the **SetBilling.sh** script described in this section could take some time to complete.
- The time required for the script to complete is also dependent on customer account. As a rough estimate, plan on approximately 3 minutes for every 10,000 customers.

Important:
Before proceeding, read section *Notes on Using the Script*.

1. If you aren't logged in to the Back-Office database server as the **Manager User**, log in as the **Manager User** now.
2. Run script **SetBilling.sh**:

```
$ sudo -u oracle /home/oracle/orascripts/SetBilling.sh
```

3. When prompted, enter the desired new **Billing Period** and **Billing Day** values.
4. When the script completes, go to the **Billing > Setup** interface in the Back-Office Console and verify that the new values appear in the respective fields. Also, go to **Billing > Invoices** to see the adjusted archive ranges (archivals) in the **Customer Invoice for** field as created by the script. (For an explanation of the creation of the new archivals, refer to section *Notes on Using the Script* below.)

Notes on Using the Script

Overlapping and Missing Days

If you adjust the Billing Day value, one of the following will happen:

There will be an overlap between the days for which the invoice was archived and the current Billing Period, or

There will some days "missing" between the already archived invoice and the current Billing Period. When the script is run, an archival for these invoices is immediately created. (The length of this period will vary according to when the previous Billing Period ended and the reset Billing Day.)

Example 1

- The current day is July 20.
- The last archival is June 15 to July 14.

If on, July 20, the Billing Day is changed to the first of every month, the next archival for the invoices (viewable starting August 1) will be July 1 to July 31. This results in an overlap of invoices archived for the days July 1 to July 14. So on August 1st you will see these archivals:

- **June 15 to July 14** (The latest existing archival before the script was executed.)
- **July 1 to July 31** (Note the 14 day overlap with the previous archival.)

Example 2

- **Case 1:** The latest existing archival is for the month before the current day's month...

  — The current day is July 20
  — The last archival is June 1 to June 30

  If, on July 20, the Billing Day is changed to the 15th of every month, an archival for July 1 to July 14 is created immediately. The next archival will be July 15 to August 14. So on August 15 you will see these archivals:

  — **June 1 to June 30** (The latest existing archival before the script was executed.)
  — **July 1 to July 14** (Created when the script is run on July 20. The length of this period will vary according to when the previous Billing Period ended and the reset Billing Day.)
  — July 15 to August 14

- **Case 2:** The latest existing archival is for several months before the current day's month...

  — The current day is July 20.
  — The last archival is March 1 to March 31.

  If, on July 20, the Billing Day is changed to the 15th of every month, an archival for April 1 to July 14 is created immediately. The next archival will be July 15 to August 14. So on August 15 you will see these archivals:

  — **March 1 to March 31** (The latest existing archival before the script was executed.)
  — **April 1 to July 14** (Created when the script is run on July 20. The length of this period will vary according to when the previous Billing Period ended and the reset Billing Day.)
  — July 15 to August 14

The Effects on Credit Limits and Allowances

After this script is run, the current day will be within the current billing period. Since the Credit Limit and Spending Allowance is for a Billing Period (1, 2 or 3 months), whatever a customer has spent since the beginning of this Billing Period will be reduced from his allowance for the rest of the days in this Billing Period.

Example 1

Consider the following case:

- The current day is July 29, and
- The Billing Period is July15 to Aug14, and
- A customer spends $500 between July 1 and July 14 (in the previous Billing Period), and
- The customer spends only $10 between July15 and today (July 29), and
- The customer's spending allowance is $500.

In this case, the customer has another $490 to spend in current Billing Period. But if the Billing Period is changed to July1 to July 31, he has already exceeded his allowance and he won't be able to purchase VOD or PPV until the next billing period starts.

Example 2

Consider the following case:

- The current day is July 29, and
- The current Billing Period July 1 to July 31, and
- A customer spends $500 between July 1 and July 14, and
- The customer's spending allowance is $500.

In this case, since the customer already spent his allowance, he cannot purchase any more items. But if the Billing Period is changed to July 15 to August 14, his allowance is reset and he will be able to make purchases again.

Change Log

Revision	Date	Change Details
00	2019-05-30	Initial release