

**Billing Operation Knowledge**

**ICP Billing**

***Billing Manual***

|  |  |
| --- | --- |
| **Prepared By :** | **Hafizul Fadhli Majid** |
| **Date:** | **16 January 2015** |
| **Version** | **0.1** |

**Document History**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Document Title | Version | State | Created by | Date |
| Billing Manual Run | 0.1 | Final | Hafizul | 16 January 2015 |
| Billing Manual |  | Combine with TroubleShoot | Hafizul | 10 December 2015 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Table of Contents**

1 Introduction 4

1.1 Reference document 4

2 Overview 5

3 Bill Run Processes 6

3.1 Creating and Uploading Flist 6

3.2 Run Large Account Tagging 6

3.3 pin\_deferred\_accts 6

3.4 Billing 7

4 pin\_bill\_day\_large 8

4.1 Pin\_bill\_day\_large 8

5 Delay Billing 9

5.1 Set Delay Billing 9

6 Monitoring 10

6.1 Throughput Monitoring 10

6.2 Billing Log 10

7 TroubleShoot 11

7.1 Common issues and recovery steps 11

7.1.1 Connection and DB Management 11

7.1.2 Server Condition 13

7.1.3 Flist Format 13

7.1.4 DB Locking 13

7.1.5 Misc issue 15

# Introduction

This document is to describe how manual ICP Billing bill run using wrapper script and using manual created flist.

## Reference document

References are made based on the following documentation.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Document** | **File Name** | **Ref Type** | **Version No.** |
| 1 | BRM 7.3.1 Documentation |  |  |  |

# Overview

When ICP Billing activity need to be run manual because of delay billing, Billing script need to be triggered manually using wrapper script provided. The billing activity will kick off by running bill according to Flist created.

# Bill Run Processes

## Creating and Uploading Flist

### Refer to Flist Creating and Uploading Manual

## Run Large Account Tagging

### Large account tagging job is a job that will change the status of an account from Normal account to Large account based on number of usages and services

Node 1

Path : /app/brm/base/apps/largeTagging

Command : nohup large\_acct\_tag\_mnl.sh <bill\_date MM/DD/YYYY> &

Example : nohup large\_acct\_tag\_mnl.sh 01/16/2015 &

**This process will include in the pin\_bill\_day\_large**

## pin\_deferred\_accts

### Use this utility as part of your daily billing to execute deferred actions. For example, if a CSR has scheduled an account to become inactive, the pin\_deferred\_act utility performs the status change on the scheduled date. By default, this utility is included in the pin\_bill\_day script.

Node 1

Path : /app/brm/base/bin

Script : nohup pin\_deferred\_accts &

**This process will include in the pin\_bill\_day\_large**

## Billing

### Billing is triggered using Flist created using following command. This activity will triggered the billing process in BRM.

Node 1

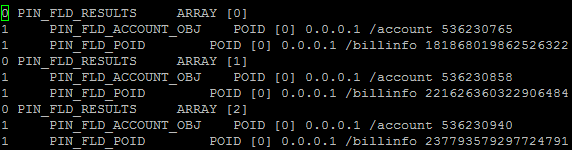
Path : /app/brm/base/apps/pin\_billd

Command : nohup pin\_bill\_accts –from\_file <Flist\_name> -verbose &

Example : nohup pin\_bill\_accts –from\_file bill1 –verbose &

**This process will include in the pin\_bill\_day\_large**

**Structure of a Flist**



# pin\_bill\_day\_large

## Pin\_bill\_day\_large

### Since we got average of 500k account per bill run, one wrapper script created to cater bill run processes for all 500k accounts in one run. When running this script, it will triggered all processed from 3.2 until 3.4 in one run so we don’t need to trigger it again.

Node 1

Path : /app/brm/base/bin

Script : pin\_bill\_day\_large

Command : nohup pin\_bill\_day\_large &

# Delay Billing

If running on manual bill run or predecessor job cannot complete in time, billing activity for ICP need to be delayed. Delay billing setting must be change for delayed billing activity to run.

## Set Delay Billing

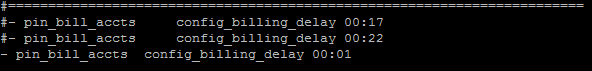
1. By default, delay billing set to 00:01. (Delay 1 hour). Will trigger at 1.30AM (A-Auto)
2. Time format:
   1. 00:01 – 0 Day 1 hour
   2. 01:01 – 1 Day 1 hour
3. After setting delay billing, for manual run can trigger from hh:01 to hh:59
4. Need to change in **pin.conf** at all nodes that will be used for run billing.
5. Login as **pin**. From node 8, ssh to node 1 and others.
6. Path & Nodes:
7. Node 1: /app/brm/base/apps/**pin\_billd**

/app/brm/base/sys/**cm**

/app/brm/base/sys/**cm2**

1. Node 3-6 /app/brm/base/sys/**cm**

/app/brm/base/sys/**cm2**



1. Contact TA Pic for CM/CM2 bounce for Node1 – Node6

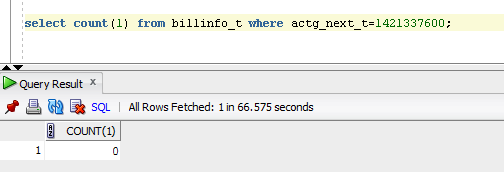
# Monitoring

## Throughput Monitoring

For every bill run, the number of accounts in billinfo\_t will decrease as account successfully billed and a new row will be created in bill\_t table. Use the following query to check the current number of billed account and throughput. When the number query return 0 means all the account has successfully billed.

PIN DB : select count(1) from billinfo\_t where actg\_next\_t =<bill\_date>;

Example : select count(1) from billinfo\_t where actg\_next\_t = 1421337600;



## Billing Log

Log for billing activity located at following location.

Pin\_billd Log : /app/brm/var/pin\_billd/pin\_mta.pinlog

CM Log : /app/brm/var/cm/cm.pinlog

/app/brm/var/cm2/cm.pinlog

# TroubleShoot

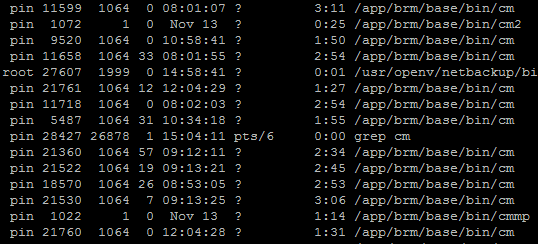
Every billing run need to be monitored the throughput until the process is completed.

## Common issues and recovery steps

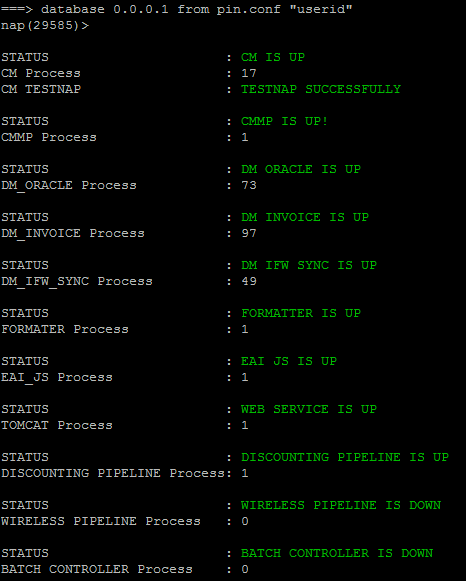
### Connection and DB Management

Make sure CM and DM is up in the server which is running billing. If these two process is down, error will encounter when running billing. To check whether all this task is running, run this command.

Command : ps –ef|grep cm



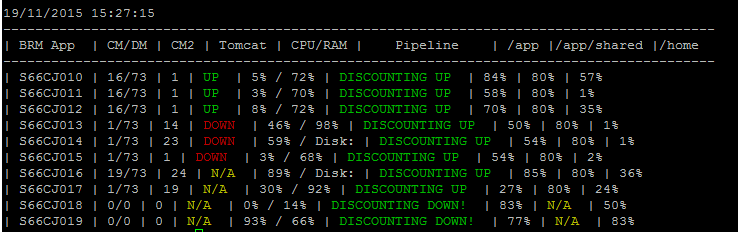
If the tasks is UP and running, the grep will show these result. If nothing appear means that the task is down. Please contact TA team to bring up the task.



Type check command to check the server condition.

### Server Condition

Server condition can cause billing to fail. To check server by running this command.

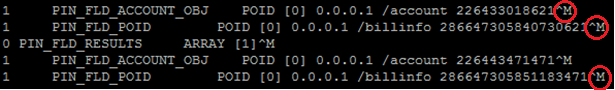


Make sure the disk space and memory of the server is sufficient enough for billing to run

Command check\_all can be used to check server condition

### Flist Format

First of all, before running billing manually make sure that the flist uploaded into server are in the correct format.



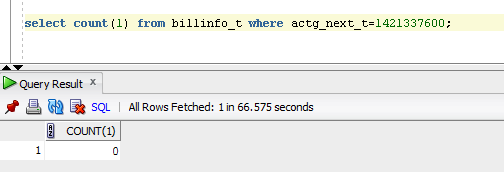
Make sure upload into server as text so that special character (^M) won’t appear.

### DB Locking

While bill is running, make sure to check throughput every hour or every 30 minutes (per convenient. Throughput of billing can be check using the following query.

SQL Query :

Select count(1) from billinfo\_t where actg\_next\_t= <bill\_date>;



If the throughput average ~1000 then the throughput is good but if it reach below 500 contact DBA to check for DB locking. The billing will complete once the query return 0.

Use this SQL query to check for DB locking

SELECT a.sid||decode(request,0,'(holder)','(waiter)'||':blocked by:'||blocking\_session) sess\_id ,lmode,request,a.type,c.object\_name,decode(row\_wait\_obj#,-1,'Holder of Lock !!!',

dbms\_rowid.rowid\_create(1,row\_wait\_obj#,row\_wait\_file#,row\_wait\_block#,row\_wait\_row#)) row\_id,

nvl(SQL\_FULLTEXT,'Holder of Lock !!!')

FROM V$LOCK A, V$LOCKED\_OBJECT B, ALL\_OBJECTS C, V$SESSION D, V$SQL E

WHERE (id1, id2, a.type) in (select id1, id2, type from v$lock where request>0)

AND a.sid = b.session\_id

AND b.object\_id = c.object\_id

AND d.sid = a.sid

AND d.sql\_hash\_value = e.hash\_value(+);

select /\*+ opt\_param('\_optimizer\_cartesian\_enabled','false') \*/

s.username,s.sid,s.serial#,s.last\_call\_et/60 mins\_running,q.sql\_text, s.program from v$session s

join v$sqltext\_with\_newlines q

on s.sql\_address = q.address

where status='ACTIVE'

and type <>'BACKGROUND'

and last\_call\_et>60

order by sid,serial#,q.piece;

### Misc issue

Other issue might occur as data problem, connectivity or memory in billing. All these issue will show in log. There are several log can be check for bill\_run activity. Always check for last update time of the log file. All logs located in Node1.

MTA pinlog : /app/brm/var/pin\_billd

File name : pin\_mta.pinlog

CM : /app/brm/var/cm

CM2 : /app/brm/var/cm2

File name : cm.log

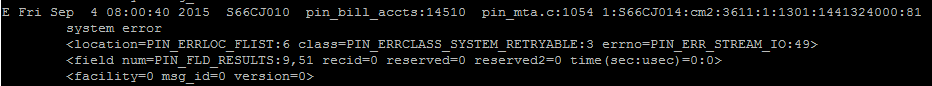
DM\_oracle log usually can be check if there got data error.

DM\_Oracle log : /app/brm/var/dm\_oracle

File name : dm\_oracle.pinlog

**Common Error Message and Solution**

**PIN\_ERR\_STREAM\_IO**



Pin error stream IO is cause by the connection to DB having problem. If this occur, contact TA and DBA team respectively to get updates on connection to DB or problems from DB side.

**PIN\_ERR\_NO\_MEM**

Pin error no memory occurred when there not enough memory for the server where bill is run. This error can be cater by running billing in different node where there are no other job running or not so many job using it.

**PIN\_ERR\_STORAGE**

Pin error storage is caused by not enough memory in table/or DB space. Contact DBA team to solve this issue.

**PIN\_ERR\_TIMEOUT**

Pin error timeout happens when billing process running for a long time and pass the timeout threshold set in the setting. This is usually happened when running billing manually. Run the account in node 7 to avoid this error as node 7 is custom set to cater the timeout issue.

**SUSPENDED ACCOUNT**

For manual billed account, if there are no error shown in log but the account didn’t get picked up for billing, check in table billinfo\_t where billing\_status for the account is suspended. If the account is suspended (1) change it to normal (0).

For suspended account under investigation, kindly get update from Apps team regularly to confirm which account is suspended and will not get billed.