

**Billing Operation Knowledge**

**ICP Billing**

***Invoicing Manual***

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

**Document History**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Document Title | Version | State | Created by | Date |
| Invoicing Manual Run | 0.1 | Final | Hafizul | 12 January 2015 |
| Invoicing Manual |  | Combine with TroubleShoot | Hafizul | 10 December 2015 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Table of Contents**

1 Introduction 4

1.1 Reference document 4

2 Overview 5

3 Manual Run 6

3.1 Bill\_Stream\_Batch 6

3.2 Updating DB 6

3.3 Pin\_inv\_acct 7

4 Pin\_bill\_day\_large2 8

4.1 Wrapper script 8

5 Invoice Regen 9

5.1 Creating Flist 9

6 TroubleShoot 10

6.1 Common issues and recovery steps 10

6.1.1 Connection and DB Management 10

6.1.2 Server Condition 12

6.1.3 Flist Format 12

6.1.4 DB Locking 12

6.1.5 Billed Accounts 14

6.1.6 Misc issue 15

# Introduction

This document is to describe how to run ICP BRM Invoicing job manually using Flist and preparation step before triggering invoicing.

## 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

Invoicing is the next activity in every Billing Cycle (BP) after billing job completed. Invoicing is needed to export bill in xml form in the next step of ICP Billing and Invoicing. This process involving preparation steps before running invoicing and invoicing process itself.

# Manual Run

## Bill\_Stream\_Batch

Bill stream batch is a job where all billed account for current BP is assign into its bill streams.

These bill streams were assigned according to logic set in the configuration based on total due, types of accounts etc.

To run bill stream batch manually login into Node 1 (10.41.66.10) and follow the instructions.

Path : /app/brm/base/source/apps/tm\_hsbb\_bill\_stream\_batch/

Script Name : tm\_hsbb\_bill\_stream\_batch

Command : nohup tm\_hsbb\_bill\_stream\_batch &

### This part is included in the wrapper script

## Updating DB

Before running invoicing run this query to update ‘/item/sponsor’ item to make item got picked up for invoicing by running the following query in NBRMPRD

update item\_t i

set i.POID\_TYPE='/item/misc'

where i.POID\_ID0 in (

select i.poid\_id0

from item\_t i,bill\_t b

where i.POID\_TYPE='/item/sponsor'

and b.poid\_id0 = i.BILL\_OBJ\_ID0

and b.END\_T = <bill\_date>

and b.INVOICE\_OBJ\_ID0 = 0)

and i.POID\_TYPE='/item/sponsor';

## Pin\_inv\_acct

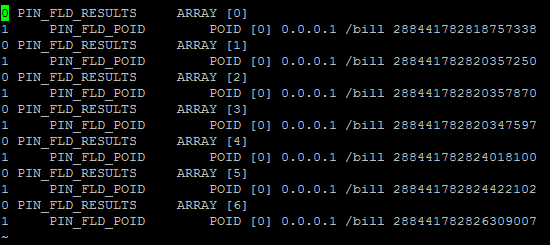
This job will kick start the invoicing for the billed account for the current BP. Invoicing manual will need flist to run as input file so we can know which bill need to be invoice. Manual invoicing job need to be run in the following path in Node1.

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

Command : nohup pin\_inv\_accts –detail <flist\_name> -verbose &

Example : nohup pin\_inv\_accts –detail invoice1 –verbose &

Structure of a Flist



# Pin\_bill\_day\_large2

## Wrapper script

## A wrapper script has been created to run bill\_stream\_batch and pin\_inv\_accts together which located in node 1. If this script triggered, there no need to run bill\_stream\_batch and pin\_inv\_accts manually.

Path : /app/brm/base/bin

Script : pin\_bill\_day\_large2

Command : nohup pin\_bill\_day\_large2&

# Invoice Regen

When running regen invoicing, the process will only rerun the invoicing for bill that already being invoiced without creating change in existing invoice poid.

## Creating Flist

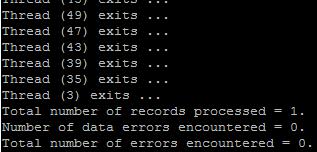
Create Flist the same format in manual invoicing by using bill\_poid of specific bill that wanted to be regen and put in /app/brm/base/apps/pin\_inv.

Flist example



Command : pin\_inv\_accts –detail <flist\_name> -verbose -regen

Example : pin\_inv\_accts –detail inv –verbose -regen



Example of success regen an invoice.

# TroubleShoot

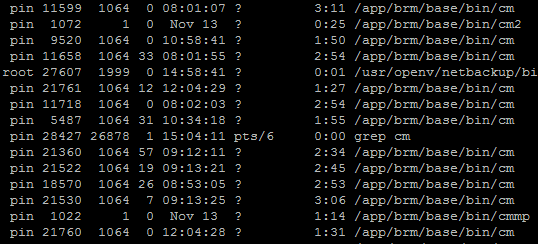
Every billing run need to be monitored the throughput until the process is complete.

## Common issues and recovery steps

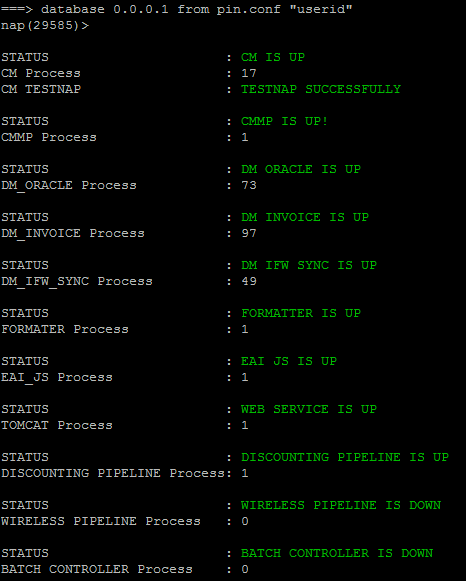
### Connection and DB Management

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. Additional from billing, invoicing need dm\_invoice to up as well.

Command : ps –ef|grep cm , ps –ef|grep dm



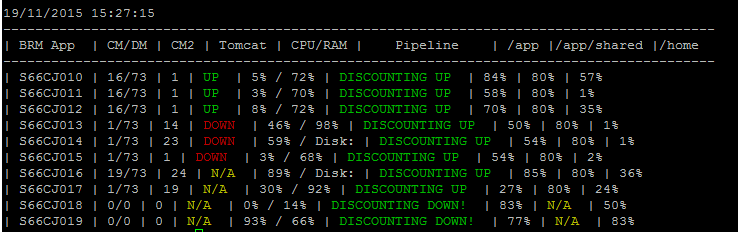
If the tasks is UP and running, the grep will show these result. If nothing appear means that the task is down. Please contact TA team to bring up the task.



Type check command to check the server condition.

### Server Condition

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

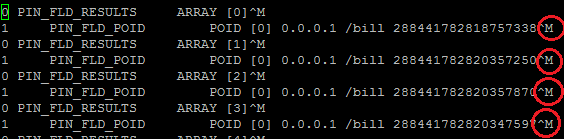


Make sure the disk space and memory of the server is sufficient enough for billing to run

Command check\_all can be used to check server condition

### Flist Format

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



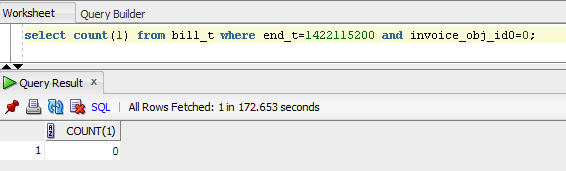
Make sure upload into server as text so that special character (^M) won’t appear.

### 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 in production DB PIN schema.

SQL Query :

Select count(1) from bill\_t where end\_t \_t= <bill\_date> and invoice\_obj\_id0=0;



If the throughput average ~1000 then the throughput is good but if it reach below 500, contact DBA in charge 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'

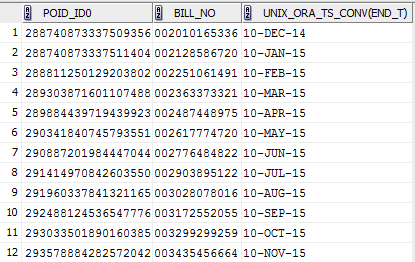
and last\_call\_et>60

order by sid,serial#,q.piece;

### Billed Accounts

For invoicing, every account need to be billed first so it can be invoice. Every billed account can be check in bill\_t table in PROD DB PIN schema.

Command : select poid\_id0,bill\_no,unix\_ora\_ts\_conv(end\_t) from bill\_t where account\_obj\_id0=<account\_poid>;



The account is billed when bill\_no and bill\_date (end\_t) is not NULL.

### 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\_inv

File name : pin\_mta.pinlog

CM : /app/brm/var/cm

CM2 : /app/brm/var/cm2

File name : cm.log

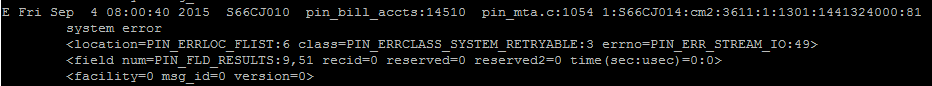
DM\_oracle log usually can be check if there got data error.

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. For invoicing kindly check both **PROD DB and INVOICE DB**

**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.

**POID TYPE**

In item\_t table, check for ‘/item/sponsor’ poid type for that particular account that want to be billed. If exists, update the type to ‘/item/sponsor’. This is applicable for manula bill only as the updating part is already embed in wrapper script for batch run.