BillConveyor概要设计

* 刘纪庆

## 功能描述

一：功能

1 读取批价输出的用户账单文件，生成类似U文件的新型U文件，里面包含用户、群、二级群、账户的优惠信息。

二：业务处理逻辑

处理流程如下：

1：getData()通过po加载参数表在程序启动的时候加载到内存。

2：getSysparam()获取配置文件。

3：process()处理主流程：获取一批文件，打开一个文件，遍历该文件的每条话单，首先查找m\_userMap<user\_id,map<bill\_id,cdr>>里是否有该用户的记录，如果没有记录，依次获取群、二级群、账户的优惠信息，如果有优惠，分别插入m\_groupMap,m\_secondGroupMap,m\_accountMap中；如果有记录，根据user\_id获取value值，也就是map<bill\_id,cdr>，然后根据bill\_id查找map，如果找到，则累加话单里的value值，如果找不到，则插入 map<bill\_id,cdr>中。整个文件遍历完后，把该文件移入work目录，并加#号前缀备份，然后判断处理时间是否超时（配置文件中配置了超时时间），如果超时了，则不再获取后面的文件，调用输出函数，如果没超时，继续判断m\_userMap的大小是否达到了输出条数（配置文件中配置），如果达到了，则不再获取后面的文件，调用输出函数，如果没达到，继续获取下一个文件。然后重复上面的处理流程。

4：outPutFile()输出文件，依次编译m\_userMap, m\_groupMap,m\_secondGroupMap,m\_accountMap输出文件。首先遍历m\_userMap，根据user\_id调用getChannle()分通道方法，获取该用户的输出通道，并插入m\_userChannel map<user\_id,chaanel>记录用户通道信息，然后遍历m\_groupMap,m\_secondGroupMap,m\_accountMap，这时候不需要再调用getChannel()方法获取通道，而是直接查找m\_userChannel通道，的到输出通道。

5：清理work目录（清理还是备份）。

## 功能实现

### 数据模型设计

#### 表结构(物理设计)

### 流程图

流程图，主要描述了程序主流程：



### 优惠获取sql

群、二级群、账户的优惠可以参考eventor里提取的sql：

二级群优惠：

Select distinct a.USER\_ID From tf\_f\_user a,tf\_f\_feepolicy b,td\_b\_discnt c ,td\_b\_feepolicy\_comp d

WHERE b.id = a.user\_id and b.id\_type='0'

and b.feepolicy\_id = c.discnt\_code

and c.b\_discnt\_code = d.feepolicy\_id

and d.event\_type\_id=28

and a.DUMMY\_TAG = '1' And a.acct\_tag='0'

And (a.remove\_tag='0' Or (a.remove\_tag!='0' And nvl(DESTROY\_DATE,nvl(PRE\_DESTROY\_DATE,to\_date('20500101000000','YYYYMMDDHH24MISS')))>to\_date(:1,'YYYYMMDDHH24MISS')))

AND a.open\_date <=to\_date(:2,'YYYYMMDDHH24MISS') and a.PROVINCE\_CODE=:3

群优惠：

Select distinct b.USER\_ID,b.SERIAL\_NUMBER,b.EPARCHY\_CODE,b.OPEN\_DATE

From tf\_f\_feepolicy a, tf\_f\_user b,td\_b\_discnt c ,td\_b\_feepolicy\_comp d

WHERE a.id = b.user\_id and b.dummy\_tag ='1'

and a.feepolicy\_id = c.discnt\_code

and c.b\_discnt\_code = d.feepolicy\_id

and (d.event\_type\_id in (22,24) or (d.event\_type\_id >= '%s' and d.event\_type\_id <= '%s') )

and d.effect\_role\_code = '-2'

AND b.PREPAY\_TAG <> '2' And b.acct\_tag='0'

and b.PROVINCE\_CODE='%s' and b.EPARCHY\_CODE='%s'

And (b.remove\_tag='0' Or (b.remove\_tag!='0' And nvl(b.DESTROY\_DATE,nvl(b.PRE\_DESTROY\_DATE,to\_date('20500101000000','YYYYMMDDHH24MISS')))>to\_date('%s','YYYYMMDDHH24MISS')))

AND b.open\_date <=to\_date('%s','YYYYMMDDHH24MISS')

AND nvl(a.end\_date,to\_date('20500101000000','YYYYMMDDHH24MISS'))>to\_date('%s','YYYYMMDDHH24MISS')

AND a.start\_date <=to\_date('%s','YYYYMMDDHH24MISS')

union

select distinct b.user\_id, b.serial\_number, b.eparchy\_code, b.open\_date

from tf\_b\_adjustbbill a , tf\_f\_user b

where a.id = b.user\_id and a.id\_type = '0'

and a.start\_cycle\_id <= '%d' and a.end\_cycle\_id >= '%d'

and b.dummy\_tag = '1'

and b.PROVINCE\_CODE = '%s' and b.EPARCHY\_CODE='%s'

AND b.PREPAY\_TAG <> '2' And b.acct\_tag='0'

And (b.remove\_tag='0'

Or (b.remove\_tag!='0' And nvl(DESTROY\_DATE,nvl(PRE\_DESTROY\_DATE,to\_date('20500101000000','YYYYMMDDHH24MISS')))>to\_date('%s','YYYYMMDDHH24MISS')))

AND b.open\_date <=to\_date('%s','YYYYMMDDHH24MISS')

账户优惠：

select distinct c.id from

(select a.id from tf\_f\_feepolicy a

where a.end\_date >= to\_date(:1,'YYYYMMDDHH24MISS') and a.start\_date <= to\_date(:2,'YYYYMMDDHH24MISS') and a.id\_type ='1' and a.prov\_code=:5

union

select b.id from tf\_b\_adjustbbill b

where to\_date(b.end\_cycle\_id, 'YYYYMM') >= to\_date(:6,'YYYYMMDDHH24MISS') and to\_date(b.start\_cycle\_id, 'YYYYMM') <= to\_date(:7,'YYYYMMDDHH24MISS') and b.id\_type ='1')

c, tf\_f\_account d

where c.id = d.acct\_id and d.PROVINCE\_CODE=:10

说明：BillConveyor启动后不能链接oralce库，所以需要把这些sql转换成程序处理逻辑。

另外注意一下eventor中，CEventCreatorApp::isNeedAccoutUser这个方法中，有一些条件过滤，具体：出帐标志=1或者2的，或者注销标志为x的需要过滤调，网别为WV的过滤掉，集团用户不出账

## 配置文件

类似分拣批价的配置文件形式，分为billconveyor.file\_format.cfg里面定义了输入输出格式，billconveyor.cdr\_attributes\_macro.cfg里面定义宏的数组下标，billconveyor.cfg里面配置了common与通道信息，

参考

<billConveyor>

<cdr\_format>

INCLUDE\_CONFIG =/cbss/billing/user/xingq/BillConveyor/billConveyor.file\_format.cfg

</cdr\_format>

<cdr\_attrib\_define>

INCLUDE\_CONFIG =/cbss/billing/user/xingq/BillConveyor/billConveyor.CDR\_ATTRIBUTES\_MACRO.cfg

</cdr\_attrib\_define>

<input\_rule>

#以下子标签无含义

<0>

format = INPUT\_FORMAT

init\_values = ATTRI\_BIZ\_TYPE:3;ATTRI\_SOURCE\_TYPE:50

</0>

</input\_rule>

<output\_rule>

#以下子标签无含义

<0>

format = OUTPUT\_FORMAT

init\_values = ATTRI\_BIZ\_TYPE:3;ATTRI\_SOURCE\_TYPE:50

</0>

</output\_rule>

<common>

#优先获取通道配置信息，通道中没有则获取COMMON中配置信息

#dmdb内存库

dmdb\_serv = dmdb

dmdb\_user = app

dmdb\_pass = si-tech

dmdb\_host = 169

dmdb\_port = 110

#资料库配置

act\_user = UOP\_ACT2

act\_pass = CBSS\_UOP\_ACT2

act\_serv = 41NGACT

act\_dbcode = ACTDB1

#参数库配置

param\_user = UOP\_PARAM

param\_pass = CBSS\_UOP\_PARAM

param\_serv = 41NGACT

param\_dbcode = ACTDB41

</common>

<1>

#dmdb内存库

dmdb\_serv = dmdb

dmdb\_user = app

dmdb\_pass = si-tech

dmdb\_host = 169

dmdb\_port = 110

#资料库配置

act\_user = UOP\_ACT2

act\_pass = CBSS\_UOP\_ACT2

act\_serv = 41NGACT

act\_dbcode = ACTDB1

#参数库配置

param\_user = UOP\_PARAM

param\_pass = CBSS\_UOP\_PARAM

param\_serv = 41NGACT

param\_dbcode = ACTDB41

input\_path = /cbss11/billdata1/user/xingq/billConveyor/input

work\_path = /cbss11/billdata1/user/xingq/billConveyor/work

output\_path = /cbss11/billdata1/user/xingq/billConveyor/output

err\_path = /cbss11/billdata1/user/xingq/billConveyor/err

is\_bak = Y

bak\_path = /cbss11/billdata1/user/xingq/billConveyor/bak

province\_code = 11

time\_out = 10

cdr\_max\_mem = 1000

</1>

</billConveyor>

### 输入格式

输入文件名：GPP\_liujq.U.1712.20171207000016

GPP\_liujq：批价文件名

U：用户账单

1712：账单账期

20171207：文件生产时间

000016：文件序列号

<INPUT\_FORMAT>

USER\_ID = ATTRI\_USER\_ID 01

BILL\_TYPE = ATTRI\_BILL\_TYPE 02

BILL\_CYCLE = ATTRI\_BILL\_CYCLE 03

TPOBJID = ATTRI\_TPOBJID 04

BEGIN\_TIME = ATTRI\_BEGIN\_TIME 05

END\_TIME = ATTRI\_END\_TIME 06

BILL\_ID = ATTRI\_BILL\_ID 07

VALUE = ATTRI\_VALUE 08

DISCNTVALUE = ATTRI\_DISCNTVALUE 09

ADJUSTBEFORE = ATTRI\_ADJUSTBEFORE 10

TYPE = ATTRI\_TYPE 11

GROUP\_USERID = ATTRI\_GROUP\_USERID 12

MAX\_VALUE = ATTRI\_MAX\_VALUE 13

MINCYCLE = ATTRI\_MINCYCLE 14

MINVALUE = ATTRI\_MINVALUE 15

LEFTVALUE = ATTRI\_LEFTVALUE 16

RESVALUE = ATTRI\_RESVALUE 17

</INPUT\_FORMAT>

|  |  |
| --- | --- |
| 输入字段 | 含义 |
| USER | 用户id |
| BILL\_TYPE | 账单累计类型0-用户账单，1-sum1,2-sum2,3-账户账单，7-sum4 |
| BILL\_CYCLE | 账期 |
| TPOBJID | 政策实例：只有帐单类型为1或2时有效,其它填0 |
| BEGIN\_TIME | 起始时间：只有帐单类型为2时有效,其它填0 |
| END\_TIME | 结束时间：只有帐单类型为2时有效,其它填0 |
| BILL\_ID | 账单明细账目 |
| VALUE | 账单值 |
| DISCNTVALUE | 账单优惠后实际值 |
| ADJUSTBEFORE | 用户调账费用，只有账单为0时有效，其他填0 |
| TYPE | 账单类型：只有帐单类型为2时有效,其它填0，0-集团，1-用户 |
| GROUP\_USERID | 群id，只有账单类型为2时有效，其他填0 |
| MAX\_VALUE | 套餐最大值，只有账单类型为1或2时有效，其他填0 |
| MINCYCLE | 可用的最小账期，只有账单类型为1或2时有效，其他填0 |
| MINVALUE | 可用的最小账期结余流量，只有账单类型为1或2时有效，其他填0 |
| LEFTVALUE | 其他账期结余流量，只有账单类型为1或2时有效，其他填0 |
| RESVALUE | 预留字段 |

### 输出格式

**URN文件命名规则：**

URNCC\_YYYYMMMMDDHHMISSXXXX\_ZZ

CC：U文件输出的通道号，根据用户的user\_id后四位找到相应的通道号 （用哪个规则表？）

YYYYMM：U文件文件归属的帐期

MMDDHHMISS：文件产生的时间（不包含YY年）

XXXX：流水号

ZZ:billconveyor进程通道号

注：URN文件是billconveyor进程产生的文件。

注：source\_type可以写死，用户用90，群91，二级群92，账户93 ，MONTH\_END\_TAG写死为1。

<OUTPUT\_FORMAT>

SOURCE\_TYPE = ATTRI\_SOURCE\_TYPE 01

BIZ\_TYPE = ATTRI\_BIZ\_TYPE 02

MONTH\_END\_TAG = ATTRI\_MONTH\_END\_TAG 03

USER\_ID = ATTRI\_USER\_ID 04

MSISDN = ATTRI\_MSISDN 05

START\_DATE = ATTRI\_START\_DATE 06

START\_TIME = ATTRI\_START\_TIME 07

HOME\_AREA\_CODE = ATTRI\_HOME\_AREA\_CODE 08

OPEN\_DATETIME = ATTRI\_OPEN\_DATETIME 09

ACCT\_DAY = ACCT\_DAY 10

BILL\_CYCLE = ATTRI\_BILL\_CYCLE 11

TPOBJID = ATTRI\_TPOBJID 12

BEGIN\_TIME = ATTRI\_BEGIN\_TIME 13

END\_TIME = ATTRI\_END\_TIME 14

BILL\_ID = ATTRI\_BILL\_ID 15

VALUE = ATTRI\_VALUE 16

DISCNTVALUE = ATTRI\_DISCNTVALUE 17

ADJUSTBEFORE = ATTRI\_ADJUSTBEFORE 18

TYPE = ATTRI\_TYPE 19

GROUP\_USERID = ATTRI\_GROUP\_USERID 20

MAX\_VALUE = ATTRI\_MAX\_VALUE 21

MINCYCLE = ATTRI\_MINCYCLE 22

MINVALUE = ATTRI\_MINVALUE 23

LEFTVALUE = ATTRI\_LEFTVALUE 24

RESVALUE = ATTRI\_RESVALUE 25

</OUTPUT\_FORMAT>