esProc

Innovated big data computing engine

OGG Incrementally collected data importing into database

Issued by Raqsoft

Content

1 OGG importing data into database Preface

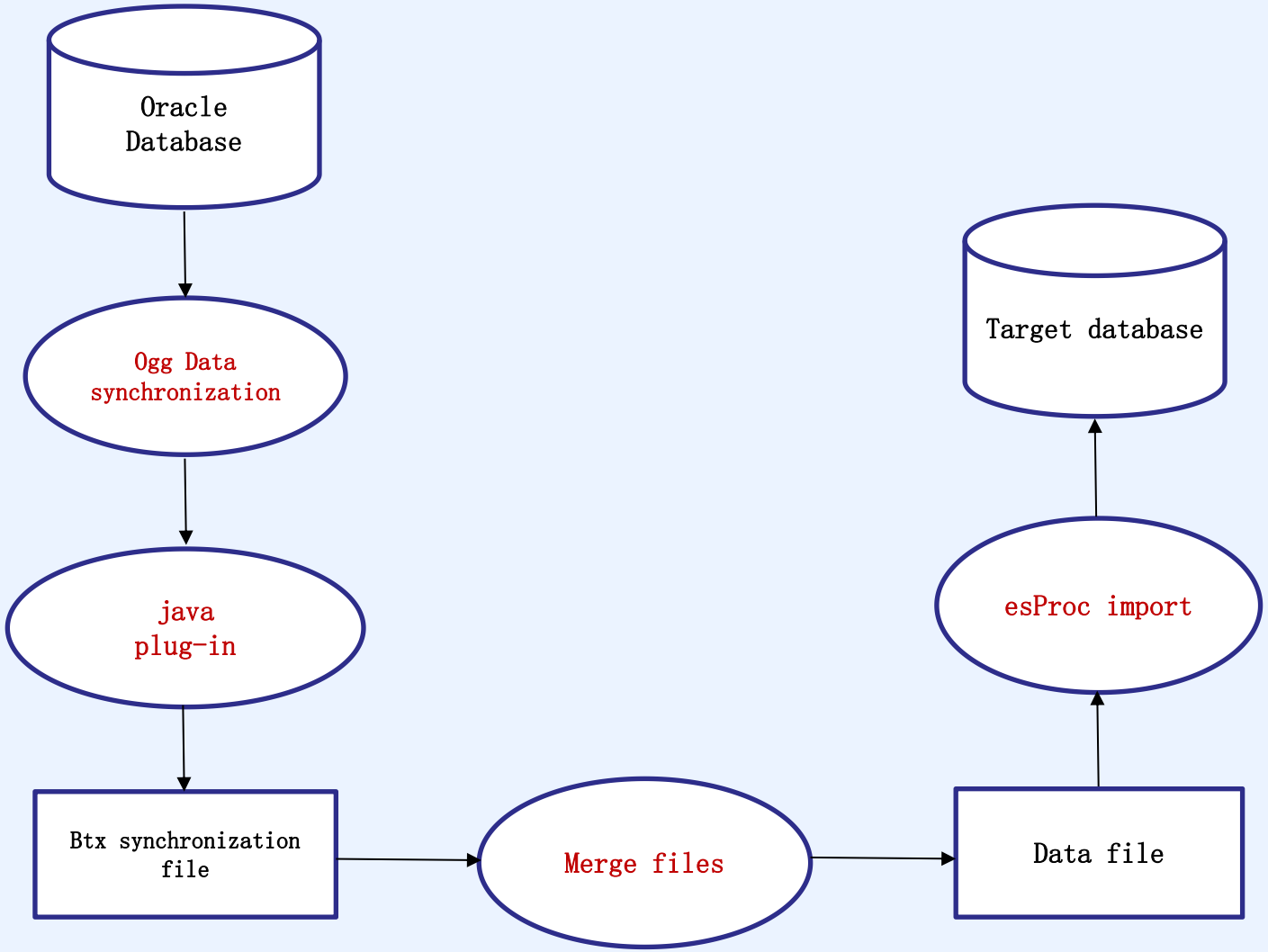
Oracle Golden Gate (Ogg) software is a kind of structured data replication and backup software based on logs. It obtains the incremental changes of data by analyzing the online logs or archive logs of the source database, and then applies these changes to the target database, so as to realize the synchronization of the source database and the target database.

Ogg can flexibly move data between similar and heterogeneous systems (including different versions of Oracle database, different hardware platforms) and between Oracle database and non Oracle database (including MSSQL, IBM DB2, mysql, mongodb, etc.), and can copy all or part of data according to the needs of the target database.

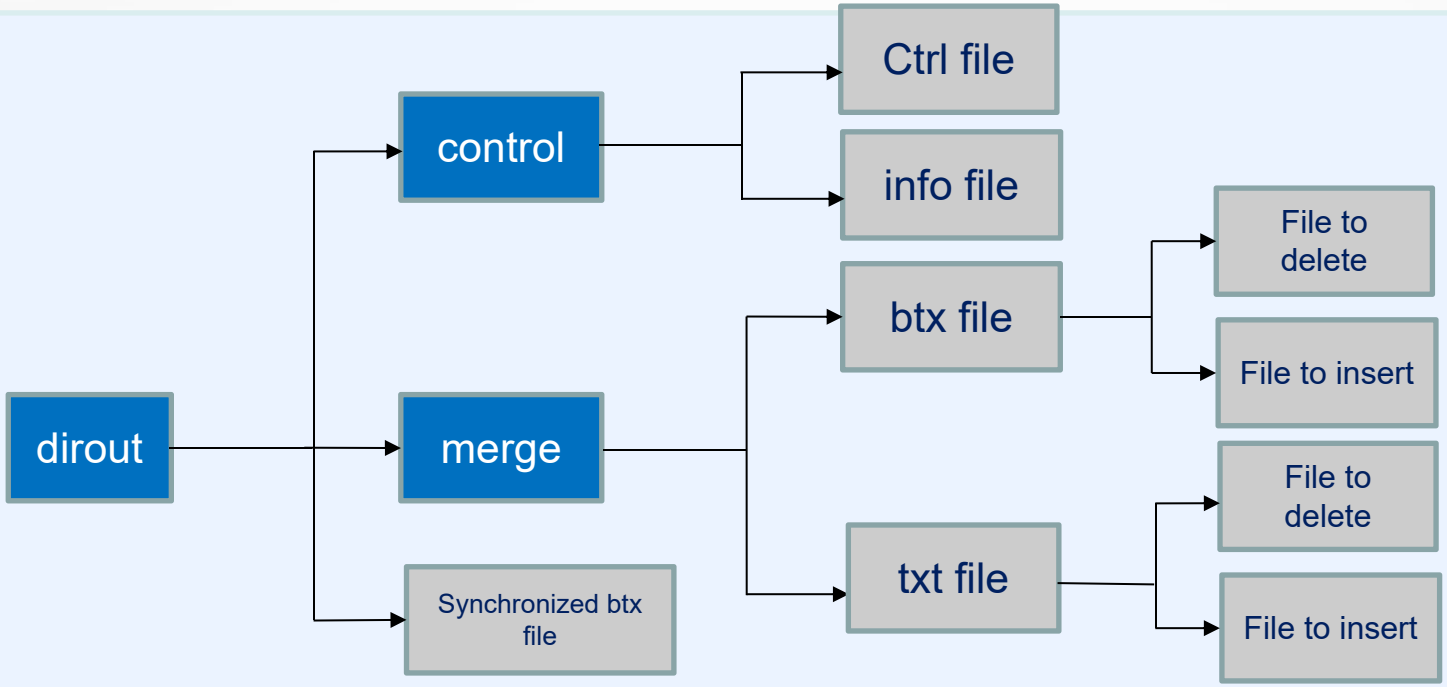
In the face of the target database supported or not supported by Ogg, the configuration settings of Ogg are different. However, by importing different databases with SPL language, the unified interface can be realized, and it is convenient to switch between different target databases.

In this solution, through Java plug-in, the data of Ogg synchronization is summarized and merged into delete event data and insert event data. Combined with esProc SPL language, many relational and non relational databases are supported, so as to simplify the operation of importing data into database and support Oracle to synchronize data to different databases.

2 Data collection flow chart



3 Directory and data file



Dirout, control and merge are directories and others are files.
Control directory: Store info file and Ctrl index file.
merge directory: Store the merged BTX data file and TXT data file.
Synchronized BTX file: Record data synchronized from Oracel and need to be consolidated before entering database
Merged file: BTX and txt files in the merge directory will generate delete operation files and insert operation files after merging, which will be used as data to import into database.

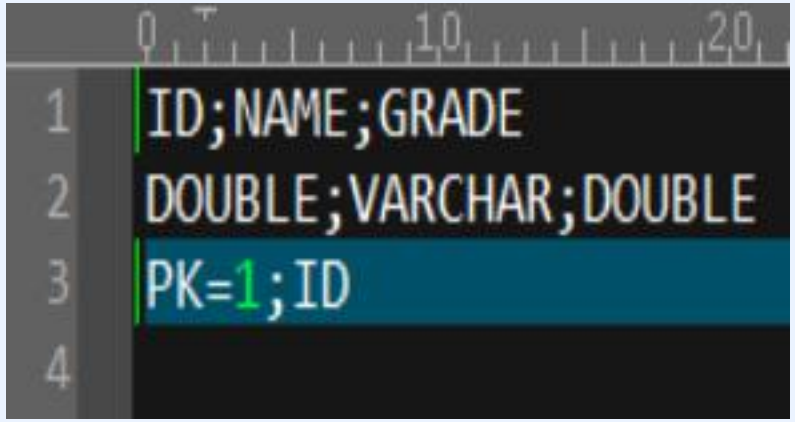
File name naming rules

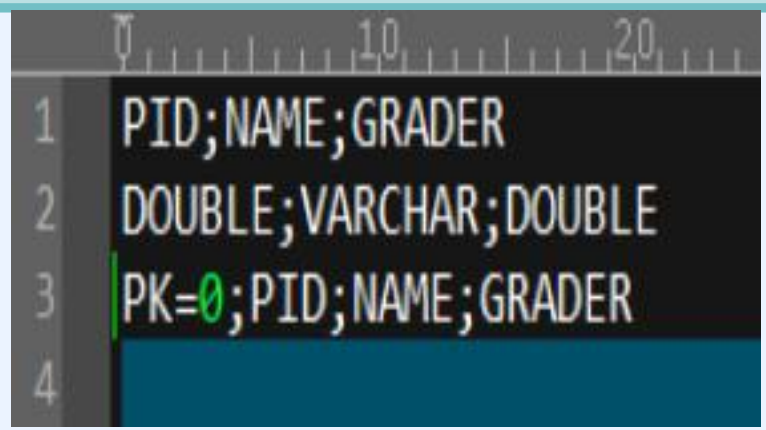
Type	Naming rule	Sample
Synchronized btx file	tablespace_tablename_datetime_sn.btx	OGG_DEMO_2019-06-26_14-10-55_00002.btx
Merged file	PUMP_tablespace_tablename_op_datetime.btx	PUMP_OGG_KULL_I_2019-08-30_19-02-58.btx
Info file	PK_tablespace_tablename_info.txt	PK_OGG_TT_info.txt
Index file	tablespace_tablename_date.ctrl	OGG_TT_2019-08-29_data.ctrl

File structure

Info file

A file that records table structure information, including table name, field name, field type, and primary key.





The content consists of three lines: field name, field type and primary key field respectively.

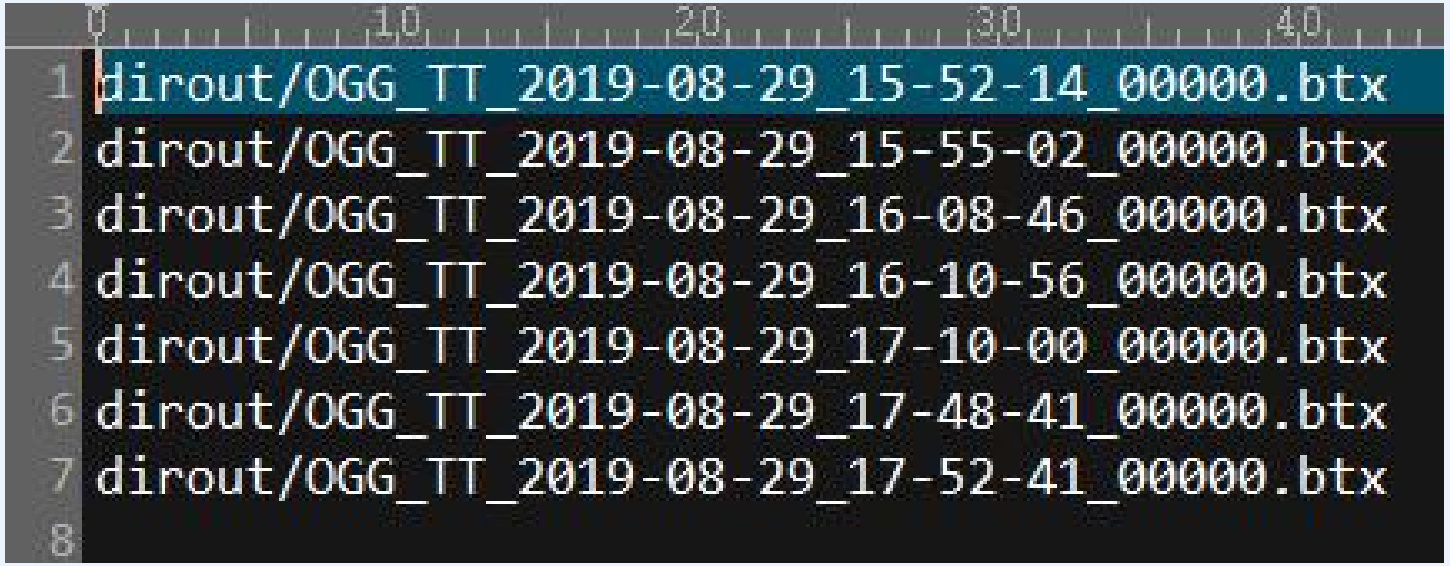
PK = 1 indicates that there is a primary key and the corresponding field name is recorded afterwards;

PK = 0 indicates that there is no primary key and all fields in the table are recorded afterwards.

File structure

Index file

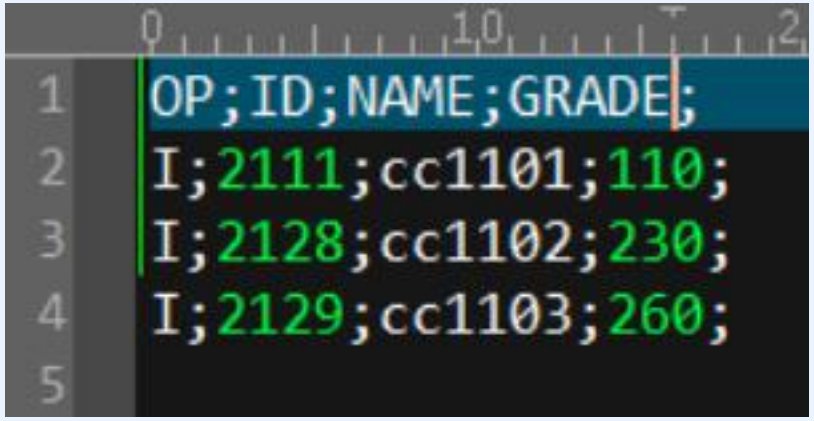
Record the BTX file names saved on a certain day of a table. When SQL DML operation is executed and data is submitted by commit, the corresponding BTX data file will be generated. The index file will record these file names in chronological order for merging BTX files.

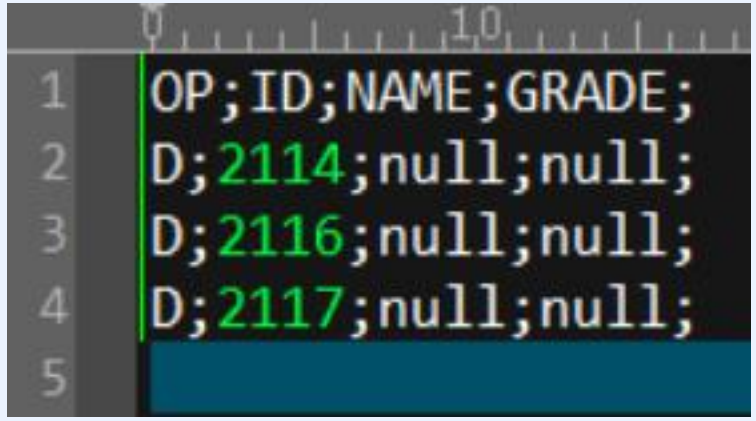


File structure

Data file

The txt files and BTX files stored under merge are in text format and binary format respectively. Their data structures are the same, in the form of **event type + field + data.**





OP represents the event type, with a value of I or D, indicating the insert event and delete event respectively.

The first line is information data. The op event type is followed by the field name. The non first line is data record.

In the delete file, if the database table has a primary key, the data corresponding to the non primary key field is empty.

4 OGG integrated Java plug-in

The Java plug-in developed on the basis of Ogg Adapter Java mainly implements the following interface and generates oggplug.jar file.

```
package com.raqsoft.lib.ogg;

public class SplHandler extends AbstractHandler {
  public void init(DsConfiguration conf, DsMetaData metaData);
    public Status transactionCommit(DsEvent e, DsTransaction tx);
    public Status operationAdded(DsEvent e, DsTransaction tx, DsOperation
dsOperation);
    public void destroy();
}
```

4 OGG integrated Java plug-in

javaue.properties configuration file

```
gg.handlerlist=raq
gg.handler.raq.type=com.raqsoft.lib.ogg.SplHandler
goldengate.userexit.timestamp=utc
goldengate.userexit.nochkpt=true
goldengate.userexit.writers=javawriter

goldengate.log.logname=cuserexit
goldengate.log.level=INFO
goldengate.log.tofile=true

javawriter.stats.display=TRUE
javawriter.stats.full=TRUE

javawriter.bootoptions=-Djava.class.path=.;dirprm;ggjava/resources/classes;ggjava/resou
rces/lib;ggjava/ggjava.jar;dirprm/fastjson-1.2.2.jar;dirprm/oggplug.jar;dirprm/icu4j_3_
4_5.jar;dirprm/dm.jar -Dlog4j.configuration=log4j.properties -Ddebug.trc=true
```

gg.handler.raq.type points to SplHandler library, and add oggplug.jar to the dependency package.

4 OGG integrated Java plug-in

Javaue.rpm configuration file

```
Extract JAVAUE
-- the source-def's must match the trail data
SourceDefs dirdef/ogg.def

-- windows:
CUserExit ggjava_ue.dll CUSEREXIT PassThru IncludeUpdateBefores
-- unix/linux:
--CUserExit libggjava_ue.so CUSEREXIT PassThru IncludeUpdateBefores

GetUpdateBefores
Table OGG.*;
```

Set the connection data source SourceDefs, dependency library, etc.

5 Data synchronization operation

The following describes data synchronization through specific table data operations.

A. Database table structure:

The structure of the table kull without primary key is as follows:

Field	Type	NULL	Note
PID	NUMBER	N	Record ID
NAME	VARCHAR2(20)	Y	Name
GRADE	INTEGER	Y	Grade

The structure of table test with primary key is as follows:

Field	Type	NULL	Note
PNO	INTEGER	N	Batch number
SNO	INTEGER	N	Serial number, constitute the primary key with PNO
NAME	VARCHAR2(20)	Y	Name
VAL	VARCHAR2(256)	Y	Parameter description

5 Data synchronization operation

Kull table data before data operation:

	PID	NAME	GRADE
1	3126	mm1000	1000
2	3127	cc1107	3700

TEST table data before data operation:

	PNO	SNO	NAME	VAL
1	200	200	Am202	Bm202
2	300	100	Am301	Bm301
3	300	200	Am100	Bm100

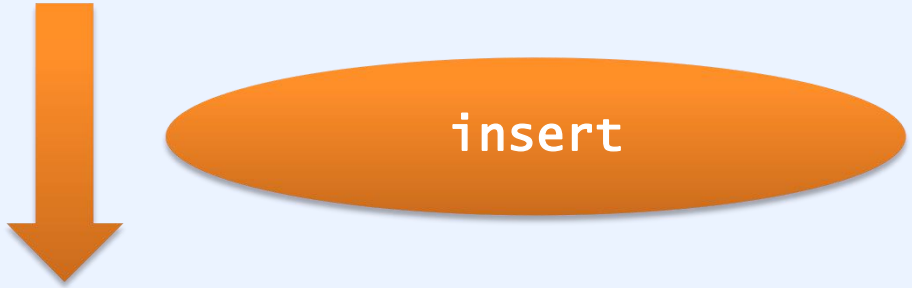
A. Single table data operation: delete

	PID	NAME	GRADE
1	3126	mm1000	1000
2	3127	cc1107	3700


```
savefile :dirout/OGG_KULL_2019-08-30_13-14-43_00015.btx
OP       PID      NAME     GRADER
D        3126     mm1000   1000
D        3127     cc1107   3700
transactionCommit xxxxxxxxxxxxxxxxxxx
```

A. Single table data operation: insert

	PID	NAME	GRADE
1	3126	cc1106	3600
2	3127	cc1107	3700




```
savefile :dirout/OGG_KULL_2019-08-30_13-14-43_00018.btx
OP       PID      NAME     GRADER
I        3126     cc1106   3600
I        3127     cc1107   3700
transactionCommit xxxxxxxxxxxxxxxxxx
```

A. Single table data operation: update

	PID	NAME	GRADE
Before updating	3126	cc1106	3600
After updating	3126	mm1000	1000


```
savefile :dirout/OGG_KULL_2019-08-30_13-14-43_00021.btx
OP      PID     NAME    GRADER
D       3126    cc1106  3600
I       3126    mm1000  1000
transactionCommit xxxxxxxxxxxxxxxxxx
```

A. Single table data DML operation

Data synchronization description

※ DML operation is performed on the data. After each commit submission, Ogg generates the corresponding synchronous BTX file according to the database table, and the file saves the changed data (data of adding, deleting and modifying events).

※ OP = D indicates deletion, Op = I indicates insertion, and update operation is decomposed into deletion and insertion operation.

B. Examples of simultaneous operation of multiple tables

The SQL statements to execute delete, insert and update:

```
delete from KULL;
insert into KULL values(3126, 'cc1106', 3600);
insert into KULL values(3127, 'cc1107', 3700);
update KULL set GRADER=1000,name='mm1000' where PID = 3126;

delete from test;
insert into test values(200, 200, 'Am202', 'Bm202');
insert into test values(300, 100, 'Am301', 'Bm301');
insert into test values(300, 200, 'Am302', 'Bm302');
update test set name='Am100',val='Bm100' where pno=300 and sno=200;
commit;
```

B. Examples of simultaneous operation of multiple tables

Execute delete, insert, update sql statement:

Oracle data

Table kull

	PID	NAME	GRADER
1	3127	cc1107	3700
2	3126	cc1106	3600

Table test

	PNO	SNO	NAME	VAL
1	200	200	Am202	Bm202
2	300	100	Am301	Bm301
3	300	200	Am100	Bm100

Synchronous btx data

savefile :dirout/OGG_KULL_2019-08-30_11-33-40_00012.btx

OP	PID	NAME	GRADER
D	3126	mm1000	1000
D	3127	cc1107	3700
I	3126	cc1106	3600
I	3127	cc1107	3700
D	3126	cc1106	3600
I	3126	mm1000	1000

savefile :dirout/OGG_TEST_2019-08-30_11-33-40_00014.btx

OP	PNO	SNO	NAME	VAL
D	200	200	null	null
D	300	100	null	null
D	300	200	null	null
I	200	200	Am202	Bm202
I	300	100	Am301	Bm301
I	300	200	Am302	Bm302
D	300	200	null	null
I	300	200	Am100	Bm100

B. Examples of simultaneous operation of multiple tables

Data synchronization description

A. The data change of table KULL is similar to that of the previous, and the data change of table test is slightly different when it deletes.
B. Because table test has a primary key, the corresponding value of the non primary key field is empty when deleting.
C. The data records in the SQL statement are recorded into the corresponding synchronous BTX file according to the table name.
D. the sequence of records shall be consistent with the operation sequence.

Since once the commit transaction is submitted in the SQL operation, a synchronous record file will be generated. There may be many synchronous BTX files, especially in the case of frequent operation of multiple tables. Therefore, it is necessary to merge the files afterwards to prepare for importing into database conveniently.

6 Background program merge data

※ Merging data refers to: For the synchronized BTX file data, in a certain period of time, multiple file data are consolidated according to different table names, and processed according to the sequence of data operations to reduce the problem of too many data files.

※ After merging, each table will generate delete event data file, insert event data file (if the corresponding event exists), and the processed synchronization BTX file will be deleted.

※ Merging includes both merging of files and of data records.

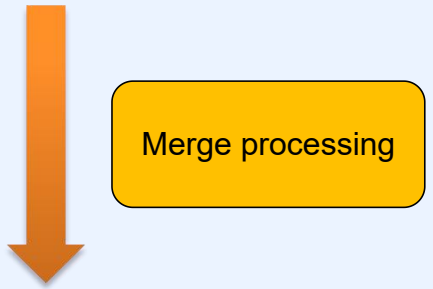
※ Merge.bat, the merge program, runs in the background. By default of the parameters, it automatically performs the merge once an hour. The generated files are stored in the directory of dirout/merge.

※ Start the background program:
>merge.bat ./

Take the merge of kull table without primary key as an example

A. File merge

dirout/OGG_KULL_2019-08-30_13-14-43_00015.btx
dirout/OGG_KULL_2019-08-30_13-14-43_00018.btx
dirout/OGG_KULL_2019-08-30_13-14-43_00021.btx



Merge processing

dirout/merge/PUMP_OGG_KULL_D_2019-08-30_14-38-33.btx
dirout/merge/PUMP_OGG_KULL_I_2019-08-30_14-38-33.btx

Take the merge of kull table without primary key as an example

B. Data merge

Synchronization btx data

OP	PID	NAME	GRADER
D	3126	mm1000	1000
D	3127	cc1107	3700
I	3126	cc1106	3600
I	3127	cc1107	3700
D	3126	cc1106	3600
I	3126	mm1000	1000

Merge processing

Data after merging

序号	OP	PID	NAME	GRADER
1	D	3126	cc1106	3600
2	D	3127	cc1107	3700
3	D	3126	mm1000	1000

序号	OP	PID	NAME	GRADER
1	I	3127	cc1107	3700
2	I	3126	mm1000	1000

If the same record is inserted first and then deleted, its insertion data will not be recorded. Such as inserting record (3126, 'cc1106', 3600)

Take the merge of test table with primary key as an example

A. File merge

dirout/OGG_TEST_2019-08-30_11-33-40_00014.btx

Merge processing

dirout/merge/PUMP_OGG_TEST_D_2019-08-30_15-07-54.btx
dirout/merge/PUMP_OGG_TEST_I_2019-08-30_15-07-54.btx

Take the merge of test table with primary key as an example

B. Data merge

Synchronization btx data

OP	PNO	SNO	NAME	VAL
D	200	200	null	null
D	300	100	null	null
D	300	200	null	null
I	200	200	Am202	Bm202
I	300	100	Am301	Bm301
I	300	200	Am302	Bm302
D	300	200	null	null
I	300	200	Am100	Bm100

Merge processing

Data after merging

序号	OP	PNO	SNO	NAME	VAL
1	D	300	200	(null)	(null)
2	D	300	100	(null)	(null)
3	D	200	200	(null)	(null)

序号	OP	PNO	SNO	NAME	VAL
1	I	200	200	Am202	Bm202
2	I	300	100	Am301	Bm301
3	I	300	200	Am100	Bm100

1. Because the table has primary key, only PNO and SNO fields are required for delete event, and other field values can be ignored.

2. For delete event records, if there are duplicate records, they shall be deduplicated first and then stored. The same is true for duplicate data in the insert event records.

Merge program introduction

```
/*************************************************************
 * Usage: merge.bat path key:value,....
 * key: model, datetime, interval, filetype
 * 1. model:使用模式，分为自动处理auto，手动处理manual、缺省为自动auto;
 * 2. datetime: 手动处理合并文件的启始时间，时间格式为yyyy-MM-dd HH:00:00
 * 3. interval: 自动处理的间隔时间，缺省为60分钟(单位为分钟);
 * 4. filetype: 输出文件格式，分为txt、btx文件格式，缺省为btx
 * Auto Example:
 * merge.bat ./ "model:auto,datetime:2019-07-05 15:00:00,interval:5,filetype:txt"
*************************************************************/
```

model: Automatic or manual processing. If it is manual, the interval parameter is invalid, and it will execute the operation immediately; if it is automatic, the DataTime parameter is invalid, and it will execute the operation regularly.

datetime: Custom merge time, 1 hour period starting from the time on the hour. This parameter can be used in manual mode if data has not been merged for some time in the past.

interval: Interval time, in minutes. In auto merge mode, merge is performed every given interval.

filetype: Output file type, support binary format BTX file, text format TXT file.

7 Using SPL to import data into database

To import the BTX file data into the database, the basic process is to carry out the deletion first and then the insertion after data loading.
The esProc SPL script is as follows:

	A	B
1	=connect("mysql")	
2	=A1.query("select * from kull").keys(PID)	
3	=file("D:/app/orcl/product/ggs/dirout/merge/PUMP_OGG_KULL_D_2019-08-30_14-38-33.btx").import@b()	
4	=file("D:/app/orcl/product/ggs/dirout/merge/PUMP_OGG_KULL_I_2019-08-30_14-38-33.btx").import@b()	
5	=A4.run(PID=int(PID))	
6	="delete from kull where "	
7	for A3	="PID="+A7.PID+" and " + "NAME='"+A7.NAME+"' and " + "GRADER="+A7.GRADER
8		>A1.execute(A6+B7)
9	=mysql.update@i(A4:A2,kull, PID, NAME, GENDER; PID)	
10	=A1.query("select * from kull")	
11	>A1.close()	

7 Using SPL to import data into database

Result after execution:

序号	PID	NAME	GRADER
1	2127	cc1107	2700
2	2129	cc1109	2900
3	3127	cc1107	3700
4	3126	mm1000	1000

The first two are the original data of the database, and the second two are synchronous data.
The SPL script for the table test to be imported into database is similar, which is skipped here.
The merged BTX file data can be imported into different types of databases according to the actual needs.

Processing method of BTX or txt files that have been processed:
After confirming that the data in storage is correct, the BTX or txt files can be deleted or moved to other locations for backup. During processing, attention shall be paid to avoid repeated importing of data.

7 Using SPL to import data into database

Instructions for importing the same table into database:

1. Each BTX file is relatively independent. It records the data changes in a period of time.

2. Each time SPL makes the same processing to import file into database, passing the file name to be processed as a parameter can reuse the SPL script.

3. The execution order of importing into database operation is to delete first and insert later. Deleting event data has an impact on the existing data in the database, as well as the inserted data. If you do not delete first, you may have duplicate primary key problems.

4. For multiple BTX file importing into database operation, in terms of time sequence, the file generated first is executed first, which is consistent with the DML data operation sequence.

When processing the import of file into database, for the data file in the format of TXT, it can also be realized with other tools in combination with the info file table information.
Automatic importing into database processing, which can be executed by adding the importing into database processing program to the timer. When the dirout/merging.lck file is not detected, it indicates that the merge program is not in the process of merging (or has finished processing), and the importing into database operation can be performed at this time.

8 Summary

After the configuration of OGG, start the processes at both ends and the background merge program. After the DML operation, OGG automatically collects the data and generates the synchronous BTX files. After the background program merges the BTX files, SPL is used for importing into database processing. After understanding the principle and operation process of OGG data collection, what users need to do is how to import the combined data into the database.

The simplified flow chart is as follows: