1 长交易处理

GoldenGate在提交前将交易缓存在内存和硬盘中，一旦进程停止则会丢弃这些信息，重启时根据检查点重新读取

GoldenGate提供了一些命令来查看和处理这些长交易

查看长交易的方法

Ggsci> send extract <进程名> , showtrans [thread n] [count n]

其中，<进程名>为所要察看的进程名，如exta/dpea/repa等；

Thread n是可选的，表示只查看其中一个节点上的未提交交易，只用于RAC；

Count n也是可选的，表示只显示n条记录。例如,查看extsz进程中节点1上最长的10个交易，可以通过下列命令：

Ggsci> send extract exta [, showtrans thread 1  count 10]

强制提交或者回滚指定事务

Ggsci> SEND EXTRACT <进程名>, SKIPTRANS <5.17.27634> [THREAD <2>]//跳过交易

Ggsci>SEND EXTRACT <进程名>, FORCETRANS <5.17.27634> [THREAD <1>] //强制提交

说明：使用这些命令只会让GoldenGate进程跳过或者认为该交易已经提交，但并不改变数据库中的交易，他们依旧存在于数据库中。因此，强烈建议使用数据库中提交或者回滚交易而不是使用GoldenGate处理。

可在extract进程中配置长交易告警

extract exta

……

warnlongtrans 12h, checkintervals 10m

exttrail /backup/goldengate/dirdat/ra

….

以上表示GoldenGate会每隔10分钟检查一下长交易，如果有超过12个小时的长交易，GoldenGate会在根目录下的ggserr.log里面加入一条告警信息。可以通过察看ggserr.log或者在ggsci中执行view ggsevt命令查看这些告警信息。以上配置可以有助于及时发现长交易并予以处理。

注意： ggserr.log保存了自mgr启动以来所有的命令历史记录，以及进程启动和终止的信息。可以通过查看它检查命令是否正确，也可通过监控它有没有ERROR字样监控复制是否正常运行。

2 配置自动删除过期队列文件

以goldengate用户登录系统，进入安装目录执行./ggsci；

执行edit param mgr编辑管理进程参数，加入或修改以下行

purgeoldextracts <my\_ogg\_path>/dirdat/\*, usecheckpoints, minkeepdays 2

其中，第一个参数为队列位置，\*可匹配备份中心所有队列文件；第二个参数表示是根据检查点计算时间；第三个参数表示最小保留多少天，后面的数字为天数。例如，如果希望只保留队列/ggs/dirdat/rt文件10天，可以配置如下：

purgeoldextracts <my\_ogg\_path>/dirdat/rt, usecheckpoints, minkeepdays 10

重启mgr进程

GGSCI > stop mgr

输入y确认停止

GGSCI > start mgr

注：临时停止mgr进程并不影响数据复制进程。

3 配置自动定时重启extract和replicat进程

以golengate用户登录系统，进入安装目录执行ggsci；

执行edit param mgr编辑管理进程参数，加入以下行

AUTORESTART ER \*, RETRIES 3, WAITMINUTES 5, RESETMINUTES 60

以上参数表示每5分钟尝试重新启动所有进程，共尝试三次。以后每60分钟清零，再按照每5分钟尝试一次共试3次。

停止和重启mgr进程，使修改后的参数文件生效

GGSCI > stop mgr

GGSCI > start mgr

4 配置密码加密

可在参数文件中使用默认密钥，而不是明文指出密码字符！

GGSCI中提供了密码加密的命令 ENCRYPT PASSWORD，该命令可以使用缺省密钥或者自定义密钥.（使用自定义密钥请参考GoldenGate的相关文档）.

使用缺省密钥进行密码加密的步骤如下：

首先在ggsci中使用ENCRYPT PASSWORD生成对应密码的加密字符串：

GGSCI (db) 3> ENCRYPT PASSWORD oggs

Expecting either 'ENCRYPTKEY' or an encryption algorithm

以上显示需要一个 ENCRYPTKEY和加密算法

使用默认密钥

GGSCI (db) 4> ENCRYPT PASSWORD oggs,ENCRYPTKEY DEFAULT

Using default key...

Encrypted password:  AACAAAAAAAAAAAEAYDTFAJAIXFMALHRF

Algorithm used:  BLOWFISH

其次，在ext/rep参数文件中配置密钥：

USERID goldengate, PASSWORD AACAAAAAAAAAAAEAYDTFAJAIXFMALHRF, ENCRYPTKEY DEFAULT

配置完毕即可启动进程验证是否成功。

5 修改复制表结构

当数据库需要复制的表结构有所改变，如增加列，改变某些列的属性等表结构改变后，可以按照下列步骤执行：

-各进程使用lag exta验证无较大延迟（秒一级即可）；

-各进程执行info exta,showch并验证ext进程重启所需日志存在；

-按顺序停止源和目标端进程；

-源和目标端执行修改表结构;

-如表有主键（或唯一索引）且本次修改未修改主键，则直接重启进程；

-但如果表无主键或者本次修改了主键（或唯一索引）则需继续执行下列步骤重建附加日志；

ggsci> dblogin userid goldengate, password XXXXXX

ggsci> delete trandata schema.mytable

ggsci> add  trandata schema.mytable

-重新启动源端和目标端的抓取和复制进程。

注意：表结构变化后必须要重启ext/rep进程！否则各进程将按照旧的结构进行复制，会导致数据无法正确复制。

6 增加复制表（新表，无基础数据）

1)    对于新增表，首先要在源端对其加入supplemental log。

GGSCI > dblogin userid oggs, password oggs

GGSCI > info trandata <schema>.<table name>

如果不是enable则需要手动加入：

GGSCI > add trandata <schema>.<table name>

2)    在源端系统上ggsci中执行stop extXX停止源端的抓取进程（注意检查所需最老归档日志是否存在）；

3)    在目标端系统上ggsci中执行stop repXX停止目标端的复制进程；

4)    在源端修改ext进程参数，添加新表:

Ggsci> edit param extXX

在文件最后加入一行“table myschema.newtable;” 注意写全schema和表的名称。

在目标端修改rep进程所包含的列表文件:

GGSCI>edit param repXX

在assumetargetdefs下面加入一行：

Handlecollisions  //注：如果是空表的话不用加此冲突处理参数

It is NOT recommended to have HANDLECOLLISIONS active all of the time, because you may introduce data inconsistencies to the target.？？？

结尾加入一行“map \*\*\*.XXX, target \*\*\*.XXX;”注意写全schema和表的名称。

在源端执行exp将该表数据导出；（可选步骤，仅用于需要执行初始化的表；如表为新表，可以直接在备份端执行建表sql，然后执行重起即可）

在目标端执行imp将该表数据导入；（可选步骤，仅用于需要执行初始化的表）

在目标端系统上启动复制进程 repXX

GGSCI > start  repXX

在源端系统上启动源端的抓取进程extXX和传输进程dpeXX

GGSCI > start  er \*

在目标端系统上去掉冲突处理（如果不需要去掉冲突处理则无需做此步）

GGSCI > lag repXX

直到AT EOF，停止repXX进程:

ggsci>stop repXX

修改repXX.prm:

ggsci>edit param repXX.prm

在handlecollions前面加—将其注释掉;

重新启动repXX:

ggsci>start repXX

即可进入正常复制状态。

添加删除复制表后必须要重启ext/rep进程！--修改参数需要重启

6 增加复制表（旧表，有基础数据）

1）.检查新表是否有补充日志

ggsci

dblogin userid goldengate

info trandata test2.dxmy

add trandata test2.dxmy

2）.在源头及目标端检查是否有该表的捕捉传输应用的限制

3）.停止源端的捕获及传输进程，参数文件中添加新表的同步，启动该进程。

4）.在目标端停止应用进程，找到csn

stop repfullo

info repfullo

GGSCI (single2) 6> info repfullo

REPLICAT   REPFULLO  Last Started 2017-06-01 19:32   Status STOPPED

Checkpoint Lag       00:00:27 (updated 00:00:07 ago)

Log Read Checkpoint  File /ggs/dirdat/tt000016

                     2017-06-01 20:12:25.594875  RBA 3357111

logdump

open  /ggs/dirdat/tt000016

ghdr on

detail on

detail data

ggstoken on

usertoken on

ggstokens detail

pos    3357111

n

GGS tokens:

TokenID x52 'R' ORAROWID         Info x00  Length   20

4141 4156 7374 4141 4541 4141 4151 6641 4655 0001 | AAAVstAAEAAAAQfAFU..

TokenID x4c 'L' LOGCSN           Info x00  Length    7

3232 3131 3430 37                                 | 2211407

TokenID x36 '6' TRANID           Info x00  Length    8

342e 3230 2e39 3431                               | 4.20.941

找到csn：2211407

5）.    按照csn-1从源端导出数据

flashback\_scn:2211407-1

6）.    将导出文件传输到目的端

7）.    目标端导入该表(导入前确认是否存在相应的表空间)

8）.    修改目标端进程参数

edit param repfulln

map test2.dxmy, target test2.dxmy;

9）.    收集目标端新表的统计信息

exec dbms\_stats.gather\_table\_stats(OWNNAME => 'TEST2', TABNAME => 'DXMY', CASCADE => TRUE);

.

10）.    启动目标端进程

start repfulln

7 删除复制表

要移除某个复制表，则执行以下步骤

直接源端的进程参数文件中注释掉包含该表的table行

目标端同样在rep进程里面注释掉包含该表的map行

源端使用info exta,showch查看检查点并验证归档日志是否存在，然后重启ext进程

目标端直接重启rep进程

添加删除复制表后必须要重启ext/rep进程！--修改参数需要重启

8 拆分replicat进程（将一部分表的同步放到新的应用进程里）

stop replicat

GGSCI (db) stop repfull

找到停止的位置

GGSCI (db) 5> info repfull,showch

REPLICAT   REPFULL   Last Started 2018-04-20 15:08   Status RUNNING

Checkpoint Lag       00:00:00 (updated 00:00:02 ago)

Log Read Checkpoint  File /u04/ogg/dirdat/tg000003

                     2018-04-20 16:41:04.873647  RBA 5047

Current Checkpoint Detail:

Read Checkpoint #1

  GGS Log Trail

  Startup Checkpoint (starting position in the data source):

    Sequence #: 3

    RBA: 2063

    Timestamp: 2018-04-20 15:08:37.000000

    Extract Trail: /u04/ogg/dirdat/tg

  Current Checkpoint (position of last record read in the data source):

    Sequence #: 3

    RBA: 5047

    Timestamp: 2018-04-20 16:41:04.873647

    Extract Trail: /u04/ogg/dirdat/tg

添加新的replicat进程，参数与原replicat一致，修改映射的表信息

启动新的replicat进程(从停止位置)

alter replicat repfull,extseqno 3,extrba 5047

9 合并replicat进程

停止dp进程，不产生新的trail file，有了一个统一的应用进程位置

拆分逆操作

启动