



QianBase 异步复制操作指南 1.6.6

2020 年 12 月

版权

© Copyright 2019-2020 贵州易鲸捷信息技术有限公司

公告

本文档包含的信息如有更改，恕不另行通知。

保留所有权利。除非版权法允许，否则在未经易鲸捷预先书面许可的情况下，严禁改编或翻译本手册的内容。易鲸捷对于本文中所包含的技术或编辑错误、遗漏概不负责。

易鲸捷产品和服务附带的正式担保声明中规定的担保是该产品和服务享有的唯一担保。本文中的任何信息均不构成额外的保修条款。

声明

Microsoft® 和 Windows® 是美国微软公司的注册商标。Java® 和 MySQL® 是 Oracle 及其子公司的注册商标。Bosun 是 Stack Exchange 的商标。Apache®、Hadoop®、HBase®、Hive®、openTSDB®、Sqoop® 和 Trafodion® 是 Apache 软件基金会的商标。Esgyn, EsgynDB 和 QianBase 是易鲸捷的商标。

目录

目录.....	I
前言.....	IV
本文简介	IV
目标读者	IV
修订历史	IV
批评与建议	V
相关文档	VI
1 异步复制	1
1.1 异步复制介绍.....	1
1.2 数据库安装后初始化异步 xdc.....	1
1.3 启停 binlog 功能	6
1.3.1 打开数据库集群写 binlog 功能	6
1.3.2 关闭数据库集群写 binlog 功能	11
1.4 基本操作.....	12
1.4.1 atrxdc 命令帮助.....	12
1.4.2 atrxdc_start/stop.....	14
1.4.3 atrxdc -checkconf	14
1.4.4 atrxdc -init	16
1.4.5 atrxdc -enable/disable.....	16
1.4.6 atrxdc -sl.....	18
1.4.7 atrxdc -cl.....	18
1.4.8 atrxdc -set	20
1.4.9 atrxdc -get.....	20
1.4.10 atrxdc -stat	21
1.4.11 checkpoint	22
1.5 hbase binlog 清理策略	22

1.5.1	手工清理 binlog	22
1.5.2	自动清理 binlog	23
1.6	异常问题处理.....	24
1.6.1	备集群启动 atrxdc 服务, 没有任何输出, 也没有启动.....	24
1.6.2	备集群 atrxdc 服务异常, 导致无法同步数据;	24
1.6.3	备集群 atrxdc 服务异常无法同步数据, atrxdc 服务重启无效...	25
1.6.4	普通表增加异步复制属性.....	25
1.6.5	备集群清理 binlog 后, 启动 atrxdc 服务, 没有继续回放 binlog 记录; 25	
1.6.6	当 atrxdc replayer 日志内持续输出 map size is XX 的处理.....	25
1.6.7	如果主集群数据库处理业务正常, hdfs binlogcache 内频繁写入新记录, 说明 hbase binlog 压力过大, 导致 binlog 访问频繁超时.....	26
1.7	主备集群不一致故障处理思路.....	26
1.7.1	当主备集群表数据一致, binlog 数据不一致	26
1.7.2	当主备集群表数据不一致, binlog 数据一致	27
1.7.3	当主备集群表数据不一致, binlog 数据不一致	27
1.8	特殊说明.....	27
1.8.1	版本升级.....	27
1.8.2	由于初始化 binlog 操作(atrxdc -init), 会重置 ttl 配置, 故在执行完 atrxdc -init 操作后, 一定要重新配置 ttl(atrxdc -setttl);	28
1.8.3	由于清理 binlog 操作(atrxdc -cl), 会重置 ttl 配置, 故在执行完 atrxdc -cl 后, 一定要重新配置 ttl(atrxdc -setttl).....	28
1.8.4	在最大性能模式下, 为了提高性能及稳定性, 建议酌情提高 buffersize 设置, 默认为 1000 字节, 一般建议调整为 4M, 但在发生如断电/断网等异常故障时, 会导致丢失更多的缓存数据(丢失当前正在传输的那组 buffersize), 建议生产环境酌情配置	28

2	Binlog Reader 介绍	28
2.1	基本原理.....	28
2.2	基本功能.....	29
2.2.1	支持查询的操作:	29
2.2.2	g 支持的数据类型:	29
2.2.3	暂不支持的内容.....	29
2.3	使用说明:	29
2.3.1	参数说明.....	29
2.4	特殊说明.....	32
2.5	异常问题处理:	33
2.6	升级方式:	33

前言

本文简介

本章概述易鲸捷 QianBase 异步复制基础内容。

目标读者

本手册的目标读者为访问易鲸捷数据库的管理员。

修订历史

版本号	日期	说明
1.0	2020 年 10 月 31 日	首次撰写该手册
1.1	2020 年 11 月 3 日	修改入参 <code>commit_switch</code> 开关默认值说明及其他描述 新增备集群回放 <code>checkpoint</code> 功能，故做完 <code>atrxdc -init</code> 后需要重置 <code>checkpoint</code>
1.1.1	2020 年 12 月 10 日	<code>Binlog_reader</code> 新增入口参数和返回列 0 <code>Atrxdc</code> 补充部分操作命令
1.1.2	2020 年 12 月 24 日	异步复制新增新功能及故障处理
1.6.5	2020 年 12 月 25 日	版本更新
1.6.6	2020 年 12 月 29 日	版本更新

批评与建议

我们支持您对本指南做出的任何批评与建议，并尽力提供符合您需求的文档。

若您发现任何错误、或有任何改进建议，请发邮件至 support@esgyn.cn。

相关文档

文档名称	说明
QianBase 安装部署指南	本文介绍安装 QianBase，包括安装前准备、安装 Hadoop 发行版、故障排除、配置、启用安全功能、提高安全性和卸载 QianBase 等。
易鲸捷 Designer 用户指南	本文介绍易鲸捷图形化数据库管理工具
易鲸捷迁移工具用户指南	本文介绍如何安装和使用易鲸捷迁移工具。
ODB 用户指南	本文介绍了如何使用 odb（一种基于 ODBC 的多线程命令行工具）在易鲸捷数据库上执行各种操作。
易鲸捷加载转换指南	本文介绍如何将数据加载转换到易鲸捷数据库。
QianBase 技术白皮书	本文介绍 QianBase 技术架构，组件介绍，技术特点等。
QianBase 数据库规划文档	本文介绍节点数量规划、数据目录和安装部署目录规划、集群角色分配规划等。
QianBase 管理员手册	本文介绍 QianBase 的日常运维常用系统命令、常用检查 SQL，用户权限配置，连接设置等内容。
QianBase 常见问题提排查与解决	本文介绍如何排查和解决 QianBase 的常见问题。
QianBase 灾难恢复手册	本文介绍 QianBase 灾难恢复设计原理，方案建议以及使用手册。
QianBase 备份恢复手册	本文介绍 QianBase 备份恢复设计原理，方案建议以及使用手册。

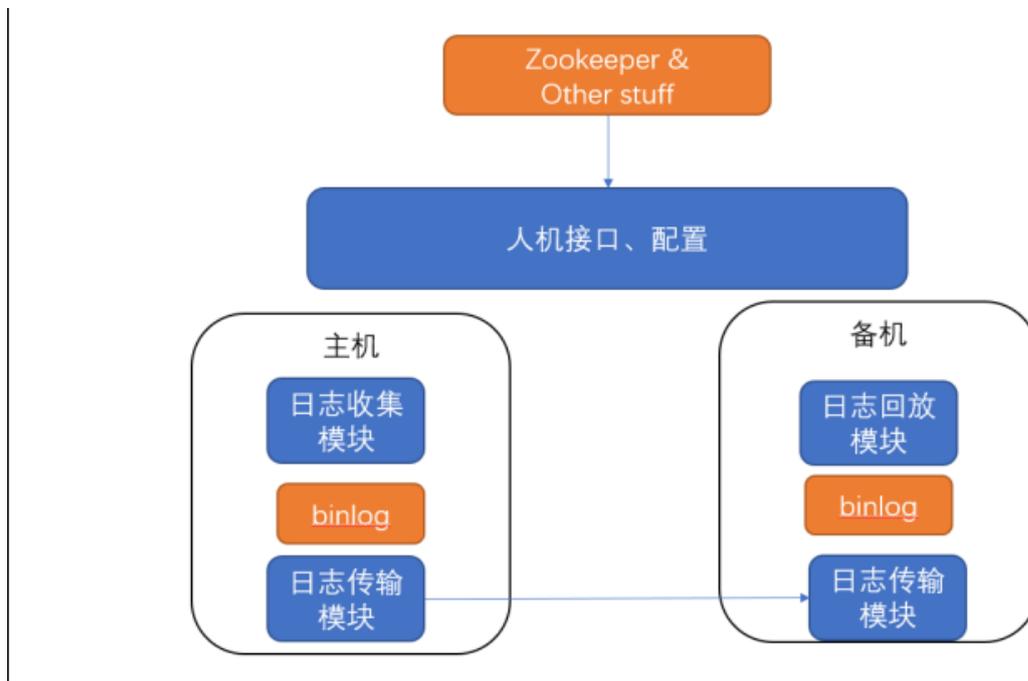
QianBase 数据库扩容指南	本文介绍 QianBase 如何更换节点，增加节点，删除节点等操作。
QianBase 数据库参数调优建议	本文介绍如何进行数据模型优化，CQD 参数优化等。
QianBase 客户端安装手册	本文介绍 QianBase JDBC，ODBC 以及 Trafci 驱动安装。
QianBase JDBC 程序员参考指南	本文介绍 QianBase JDBC 驱动连接设置，开发人员指南。
QianBase ODBC 程序员参考指南	本文介绍 QianBase ODBC 驱动连接设置，开发人员指南。
QianBase SPSQL 存储过程用户手册	本文介绍 QianBase SPSQL 存储过程的使用。
Esgyn DBManager 用户手册	本文介绍图形化数据库监控运维工具 DB Manager 的使用。
QianBase 数据库迁移指南	本文介绍如何将常见关系型数据库（Oracle、MySQL、SQL server 等）迁移至 QianBase。
QianBase SQL 用户手册	本文是 QianBase 的 SQL 使用手册。
QianBase 命令行工具指南	本指南适用于维护和监管 QianBase 数据库的数据库管理员和支持人员。

1 异步复制

本章主要介绍易鲸捷 QianBase 异步复制基础内容

1.1 异步复制介绍

主集群日志收集模块收集用户操作日志，并写入 hbase binlog，备集群通过日志传输模块拉取主集群 binlog 数据，然后在备集群进行回放，并写入备集群 binlog；备集群 atrxdc 服务定时扫描主集群 binlog，发现新数据后开始批量拉取，直到没有新数据结束；



1.2 数据库安装后初始化异步 xdc

1. 数据库安装后，默认状态为未开启 atrxdc 功能，默认 atrxdc 集群角色为 primary；
2. 数据库安装后，需要通过 atrxdc -init 命令，atrxdc -init 命令会初始化 atrxdc 配置，并重构 hbase binlog 表，如下所示：

```
[trafodion@host-10-10-14-101 ~]$ atrxdc -init
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in
[jar:file:/opt/cloudera/parcels/CDH-5.16.2-
```

```
1.cdh5.16.2.p0.8/jars/slf4j-log4j12-
1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in
[jar:file:/opt/trafodion/QianBase_server-
1.6.3/export/lib/orc-tools-1.5.0-
uber.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See
http://www.slf4j.org/codes.html#multiple\_bindings for an
explanation.
SLF4J: Actual binding is of type
[org.slf4j.impl.Log4jLoggerFactory]
atrxdc configuration initialized.
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in
[jar:file:/opt/cloudera/parcels/CDH-5.16.2-
1.cdh5.16.2.p0.8/jars/slf4j-log4j12-
1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in
[jar:file:/opt/trafodion/QianBase_server-
1.6.3/export/lib/orc-tools-1.5.0-
uber.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See
http://www.slf4j.org/codes.html#multiple\_bindings for an
explanation.
SLF4J: Actual binding is of type
[org.slf4j.impl.Log4jLoggerFactory]
EsgynDB Advanced Conversational Interface 1.6.3
Copyright (c) 2015-2020 Esgyn Corporation
```

```
>>initialize trafodion, create xdc metadata;
```

```
--- SQL operation complete.
```

```
>>set parserflags 131072;
```

```
--- SQL operation complete.
```

```
>>create table if not exists
```

```
TRAFODION."_XDC_MD_".XDC_DDL_DUP like
```

```
TRAFODION."_XDC_MD_".XDC_DDL;
```

```
--- SQL operation complete.
```

```
>>exit;
```

```
End of MXCI Session
```

```
reset salt map success, current salt map is:
```

```
host-10-10-14-103: 2
```

```
host-10-10-14-102: 1
```

```
host-10-10-14-104: 3
```

```
host-10-10-14-101: 0
```

```
**WARNING** this command will rebuild binlog,
```

```
continue?[y/n]: y
```

```
Java HotSpot(TM) 64-Bit Server VM warning: Using
```

```
incremental CMS is deprecated and will likely be removed
```

```
in a future release
```

```
20/10/26 10:17:30 INFO Configuration.deprecation:
```

```
hadoop.native.lib is deprecated. Instead, use
```

```
io.native.lib.available
```

```
HBase Shell; enter 'help<RETURN>' for list of supported
commands.
```

```
Type "exit<RETURN>" to leave the HBase Shell
```

```
Version 1.2.0-cdh5.16.2,
```

```
r8e5f7788542324d7c55313bac2627d190b58d4cf, Fri Mar 20
```

```
10:39:33      2020
```

```
disable 'TRAF_RSRVD_5:TRAFODION._DTM_.TRAFODION_BINLOG'
```

```
0 row(s) in 3.6860 seconds
```

```
drop 'TRAF_RSRVD_5:TRAFODION._DTM_.TRAFODION_BINLOG'
```

```
0 row(s) in 3.3030 seconds
```

```
create 'TRAF_RSRVD_5:TRAFODION._DTM_.TRAFODION_BINLOG',
```

```
'mf', 'mt_', {NUMREGIONS => 4, SPLITALGO =>
```

```
'UniformSplit'}
```

```
0 row(s) in 3.1730 seconds
```

```
Hbase::Table -
```

```
TRAF_RSRVD_5:TRAFODION._DTM_.TRAFODION_BINLOG
```

```
--- operation success.
```

3. 如果是主备集群，还需要使用 `atrxdc -set role primary|secondary`，分别配置主备角色，然后重启数据库

配置 `atrxdc` 集群角色为主：

```
[trafodion@host-10-10-14-105 ~]$ atrxdc -set role primary
```

```
***WARN***: operation success, restart EsgynDB to take
affect.
```

```
[trafodion@host-10-10-14-105 ~]$
```

配置atrxdc集群角色为备:

```
[trafodion@host-10-10-14-105 ~]$ atrxdc -set role
secondary
***WARN***: operation success, restart EsgynDB to take
affect.
```

```
[trafodion@host-10-10-14-105 ~]$
```

4. 使用 `atrxdc -checkconf`, 查看 `atrxdc` 状态, 确认配置生效; 如下 option name:

`ATRConfigInitialized` 表示配置生效, `atrxdc` 初始化成功;

```
[trafodion@host-10-10-14-105 ~]$ atrxdc -checkconf
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in
[jar:file:/opt/cloudera/parcels/CDH-5.16.2-
1.cdh5.16.2.p0.8/jars/slf4j-log4j12-
1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in
[jar:file:/opt/trafodion/QianBase_server-
1.6.3/export/lib/orc-tools-1.5.0-
uber.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See
http://www.slf4j.org/codes.html#multiple\_bindings for an
explanation.
SLF4J: Actual binding is of type
[org.slf4j.impl.Log4jLoggerFactory]
option name: clusterRole, zk path:
/trafodion/multi_dc/atr/role, default values: 0, current
value: 0
option name: syncMode, zk path:
/trafodion/multi_dc/atr/syncmode, default values: 2,
```

```
current value: 2
option name: ATRConfigInitialized, zk path:
/trafodion/multi_dc/atr/binlog/initialized, default
values: f, current value: t
option name: binlogPartialNum, zk path:
/trafodion/multi_dc/atr/binlog/partial, default values:
null, current value: 4
option name: binlogMutatorWriterBufferSize, zk path:
/trafodion/multi_dc/atr/binlog/mutatorbuf, default
values: 1000, current value: 1000
option name: binlogConnectionString, zk path:
/trafodion/multi_dc/atr/binlog/binlogconn, default
values: , current value:

[trafodion@host-10-10-14-105 ~]$
```

1.3 启停 binlog 功能

1.3.1 打开数据库集群写 binlog 功能

数据库安装好后, binlog 功能默认是关闭的, 打开 binlog 功能只需要在安装好数据库后, 做如下操作便可以打开 binlog 功能。

1. 检查当前数据库 binlog 是否打开 (由黄色字体表明默认没有打开)

```
[root@gy27 ~]# su - trafodion
Last login: Mon Oct 19 17:43:55 CST 2020 on pts/0
[trafodion@gy27 ~]$ atrxdc -checkconf
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in
[jar:file:/opt/cloudera/parcels/CDH-5.16.2-
```

```
1.cdh5.16.2.p0.8/jars/slf4j-log4j12-
1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in
[jar:file:/opt/trafodion/QianBase_server-
1.6.3D20201016/export/lib/orc-tools-1.5.0-
uber.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See
http://www.slf4j.org/codes.html#multiple_bindings for an
explanation.
SLF4J: Actual binding is of type
[org.slf4j.impl.Log4jLoggerFactory]
ATRXDC configuration not initialized
```

2. trafodion 用户下执行 `atrxdc -init` 便可以打开写 binlog 功能（注意黄色处要输入 Y）

```
[trafodion@gy27 ~]$ atrxdc -init
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in
[jar:file:/opt/cloudera/parcels/CDH-5.16.2-
1.cdh5.16.2.p0.8/jars/slf4j-log4j12-
1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in
[jar:file:/opt/trafodion/QianBase_server-
1.6.3D20201016/export/lib/orc-tools-1.5.0-
uber.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple\_bindings for an
explanation.
SLF4J: Actual binding is of type
[org.slf4j.impl.Log4jLoggerFactory]
```

atrxdc configuration initialized.

.....

... ..

****WARNING** this command will rebuild binlog,**

continue?[y/n]: y

Java HotSpot(TM) 64-Bit Server VM warning: Using incremental CMS is deprecated and will likely be removed in a future release

20/10/19 18:53:07 INFO Configuration.deprecation:

hadoop.native.lib is deprecated. Instead, use io.native.lib.available

HBase Shell; enter 'help<RETURN>' for list of supported commands.

Type "exit<RETURN>" to leave the HBase Shell

Version 1.2.0-cdh5.16.2,

r8e5f7788542324d7c55313bac2627d190b58d4cf, Fri Mar 20

10:39:33 2020

disable 'TRAF_RSRVD_5:TRAFODION._DTM_.TRAFODION_BINLOG'

0 row(s) in 3.4740 seconds

drop 'TRAF_RSRVD_5:TRAFODION._DTM_.TRAFODION_BINLOG'

0 row(s) in 1.1770 seconds

create 'TRAF_RSRVD_5:TRAFODION._DTM_.TRAFODION_BINLOG',

'mf', 'mt_', {NUMREGIONS => 4, SPLITALGO =>

'UniformSplit'}

0 row(s) in 3.1600 seconds

Hbase::Table -

TRAF_RSRVD_5:TRAFODION._DTM_.TRAFODION_BINLOG

--- operation success.

3. 再次检查 binlog 是否打开 (如下表明打开)

```
[trafodion@gy27 ~]$ atrxdc -checkconf
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in
[jar:file:/opt/cloudera/parcels/CDH-5.16.2-
1.cdh5.16.2.p0.8/jars/slf4j-log4j12-
1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in
[jar:file:/opt/trafodion/QianBase_server-
1.6.3D20201016/export/lib/orc-tools-1.5.0-
uber.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple\_bindings for an
explanation.
SLF4J: Actual binding is of type
[org.slf4j.impl.Log4jLoggerFactory]
option name: clusterRole, zk path:
/trafodion/multi_dc/atr/role, default values: 0, current
value: 0
option name: syncMode, zk path:
/trafodion/multi_dc/atr/syncmode, default values: 2,
current value: 2
option name: ATRConfigInitialized, zk path:
/trafodion/multi_dc/atr/binlog/initialized, default
values: f, current value: t
```


for max performance, 2 for max reliability

bwsz - binlog mutator

write buffer size, this parameter used in max performance mode

bcstr - binlog cluster

connection string, value syntax: quorum[:port]

rcp - binlog replayer

checkpoint, value syntax: salt,wid,cid

skipwal - binlog skip WAL

option, value syntax: {t | true | f | false}

-get { attribute } - get attribute related to atrxdc

supported attribute:

bp - binlog partition

number

mode - atr sync mode

role - cluster role

bwsz - binlog mutator

write buffer size, this parameter used in max performance mode

bcstr - binlog cluster

connection string

rcp - binlog replayer

checkpoint

skipwal - binlog skip WAL

option

-sl - scan binlog from hbase shell

-stat - print local binlog statistics

-init - initialize atrxdc

configuration

-replayseq - replay sequence

-resetbl - rebuild binlog

-resetrpc - reset atrxdc replay

checkpoint

```

    -resetsm                - reset binlog salt and host
map
    -h | -help | help      - show this usage
<component> could be:
    manager
    worker
    replayer
<args> could be:
    hostname    - the hostname process will be started
    partitionnum - the partition number worker will replay
and copy,
                this option is for worker process
example: atrxdc start manager
        atrxdc start worker host1 2
        atrxdc start replayer host1 [quorumstr]

```

1.4.2 atrxdc_start/stop

用于在备集群启停 atrxdc 拉取回放服务

1.4.3 atrxdc -checkconf

用于查看 atrxdc 相关状态信息:

如集群角色/同步模式/atrxdc 功能开关状态/binlog 分区数/binlog 写 buffer 尺寸等;

- 异步 xdc 功能关闭状态

```

[trafodion@host-10-10-14-105 ~]$ atrxdc -checkconf
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in
[jar:file:/opt/cloudera/parcels/CDH-5.16.2-
1.cdh5.16.2.p0.8/jars/slf4j-log4j12-
1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in
[jar:file:/opt/trafodion/QianBase_server-
1.6.3/export/lib/orc-tools-1.5.0-

```

```
uber.jar!/org/slf4j/impl/StaticLoggerBinder.class]
```

SLF4J: See

http://www.slf4j.org/codes.html#multiple_bindings for an explanation.

SLF4J: Actual binding is of type

```
[org.slf4j.impl.Log4jLoggerFactory]
```

ATRXDC configuration not initialized

- 异步 xdc 功能开启状态

```
[trafodion@host-10-10-14-105 ~]$ atrxdc -checkconf
```

SLF4J: Class path contains multiple SLF4J bindings.

SLF4J: Found binding in

```
[jar:file:/opt/cloudera/parcels/CDH-5.16.2-
```

```
1.cdh5.16.2.p0.8/jars/slf4j-log4j12-
```

```
1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]
```

SLF4J: Found binding in

```
[jar:file:/opt/trafodion/QianBase_server-
```

```
1.6.3/export/lib/orc-tools-1.5.0-
```

```
uber.jar!/org/slf4j/impl/StaticLoggerBinder.class]
```

SLF4J: See

http://www.slf4j.org/codes.html#multiple_bindings for an explanation.

SLF4J: Actual binding is of type

```
[org.slf4j.impl.Log4jLoggerFactory]
```

option name: clusterRole, zk path:

/trafodion/multi_dc/atr/role, default values: 0, current value: 0

option name: syncMode, zk path:

/trafodion/multi_dc/atr/syncmode, default values: 2,

current value: 2

option name: ATRConfigInitialized, zk path:

/trafodion/multi_dc/atr/binlog/initialized, default

values: f, current value: t

option name: binlogPartialNum, zk path:

/trafodion/multi_dc/atr/binlog/partial, default values:

null, current value: 4

option name: binlogMutatorWriterBufferSize, zk path:

/trafodion/multi_dc/atr/binlog/mutatorbuf, default

values: 1000, current value: 1000

option name: binlogConnectionString, zk path:

/trafodion/multi_dc/atr/binlog/binlogconn, default

values: , current value:

1.4.4 atrxdc -init

用于初始化 atrxdc 相关配置，并重构 hbase binlog 表，会导致 hbase binlog 内所有记录均被清除；

一般用于数据库安装好后，进行初始化操作；

在执行完 atrxdc -init 后，必须执行 atrxdc -resetrcp，用于重置 checkpoint，否则会由于 binlog 内已不存在原来的数据，导致 replayer 进程重新回放上一次的 checkpoint 失败；

1.4.5 atrxdc -enable/disable

- 用于开启或关闭 atrxdc 功能；

当 atrxdc 功能开启，我们所有 ddl/dml 操作记录，均会写入 hbase binlog；

当 atrxdc 功能关闭，我们所有的 ddl/dml 操作记录，除建表元数据外，其他均不会写入 hbase binlog；

- 关闭 atrxdc 写 binlog 功能

```
[trafodion@host-10-10-14-105 ~]$ atrxdc -disable
```

```
SLF4J: Class path contains multiple SLF4J bindings.
```

```
SLF4J: Found binding in
[jar:file:/opt/cloudera/parcels/CDH-5.16.2-
1.cdh5.16.2.p0.8/jars/slf4j-log4j12-
1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in
[jar:file:/opt/trafodion/QianBase_server-
1.6.3/export/lib/orc-tools-1.5.0-
uber.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See
http://www.slf4j.org/codes.html#multiple\_bindings for an
explanation.
```

```
SLF4J: Actual binding is of type
[org.slf4j.impl.Log4jLoggerFactory]
```

● 开启 atrxdc 写 binlog 功能

```
[trafodion@host-10-10-14-105 ~]$ atrxdc -enable
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in
[jar:file:/opt/cloudera/parcels/CDH-5.16.2-
1.cdh5.16.2.p0.8/jars/slf4j-log4j12-
1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in
[jar:file:/opt/trafodion/QianBase_server-
1.6.3/export/lib/orc-tools-1.5.0-
uber.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See
http://www.slf4j.org/codes.html#multiple\_bindings for an
explanation.
```

```
SLF4J: Actual binding is of type
[org.slf4j.impl.Log4jLoggerFactory]
```

```
[trafodion@host-10-10-14-105 ~]$
```

1.4.6 atrxdc -sl

用于查看 hbase binlog 内除表元数据之外的所有 ddl/dml 记录

```
[trafodion@host-10-10-14-105 ~]$ atrxdc -sl
Java HotSpot(TM) 64-Bit Server VM warning: Using
incremental CMS is deprecated and will likely be removed
in a future release
20/10/26 11:49:39 INFO Configuration.deprecation:
hadoop.native.lib is deprecated. Instead, use
io.native.lib.available
HBase Shell; enter 'help<RETURN>' for list of supported
commands.
Type "exit<RETURN>" to leave the HBase Shell
Version 1.2.0-cdh5.16.2,
r8e5f7788542324d7c55313bac2627d190b58d4cf, Fri Mar 20
10:39:33      2020

scan 'TRAF_RSRVD_5:TRAFODION._DTM_.TRAFODION_BINLOG',
{COLUMNS => 'mf:mq'}

ROW                                COLUMN+CELL
0 row(s) in 0.2670 seconds
```

1.4.7 atrxdc -cl

用于清理 hbase binlog 内除表元数据之外的所有 ddl/dml 记录

```
[trafodion@host-10-10-14-105 ~]$ atrxdc -cl
**WARNING** this command will clear binlog, atrxdc
service depend on this log, continue?[y/n]: y
Java HotSpot(TM) 64-Bit Server VM warning: Using
incremental CMS is deprecated and will likely be removed
in a future release
```

```
20/10/26 11:31:24 INFO Configuration.deprecation:
hadoop.native.lib is deprecated. Instead, use
io.native.lib.available
HBase Shell; enter 'help<RETURN>' for list of supported
commands.
Type "exit<RETURN>" to leave the HBase Shell
Version 1.2.0-cdh5.16.2,
r8e5f7788542324d7c55313bac2627d190b58d4cf, Fri Mar 20
10:39:33      2020
```

```
disable 'TRAF_RSRVD_5:TRAFODION._DTM_.TRAFODION_BINLOG'
0 row(s) in 3.6690 seconds
```

```
alter 'TRAF_RSRVD_5:TRAFODION._DTM_.TRAFODION_BINLOG',
'delete' => 'mf'
Updating all regions with the new schema...
4/4 regions updated.
Done.
0 row(s) in 2.4690 seconds
```

```
alter 'TRAF_RSRVD_5:TRAFODION._DTM_.TRAFODION_BINLOG',
{NAME=> 'mf'}
Updating all regions with the new schema...
4/4 regions updated.
Done.
0 row(s) in 2.2060 seconds
```

```
enable 'TRAF_RSRVD_5:TRAFODION._DTM_.TRAFODION_BINLOG'
```

0 row(s) in 3.2740 seconds

SLF4J: Class path contains multiple SLF4J bindings.

SLF4J: Found binding in

[jar:file:/opt/cloudera/parcels/CDH-5.16.2-

1.cdh5.16.2.p0.8/jars/slf4j-log4j12-

1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]

SLF4J: Found binding in

[jar:file:/opt/trafodion/QianBase_server-

1.6.3/export/lib/orc-tools-1.5.0-

uber.jar!/org/slf4j/impl/StaticLoggerBinder.class]

SLF4J: See

http://www.slf4j.org/codes.html#multiple_bindings for an explanation.

SLF4J: Actual binding is of type

[org.slf4j.impl.Log4jLoggerFactory]

--- Operation success.

1.4.8 atrxdc -set

用于配置 atrxdc 相关配置,

- 配置角色, atrxdc -set role primary|secondary
- 配置同步模式为最大可靠/最大性能, atrxdc -set mode 1|2
- 配置最大性能模式下每次将缓存数据刷入 binlog 的 buffersize(单位为字节), 该参数只在最大性能模式下生效, atrxdc -set bwsz 4194304
- 配置 binlog 记录是否写 wal 日志, 默认为 f, 即写 wal 日志, atrxdc -set skipwal t|f

1.4.9 atrxdc -get

查看 atrxdc 相关信息

- 查看当前 binlog 分区数, atrxdc -get bp

- 查看当前同步模式, `atrxdc -get mode`
- 查看当前集群角色, `atrxdc -get role`
- 查看备集群当前回放检查点, `atrxdc -get rcp`
- 查看每次将缓存数据刷入 binlog 的 buffersize, 该参数只在最大性能模式下生效; `atrxdc -get bwsz`
- 查看 binlog 记录是否写 wal 日志, `atrxdc -get skipwal`

1.4.10 atrxdc -stat

用于查看当前集群的 binlog 信息

```
[trafodion@host-10-10-14-105 ~]$ atrxdc -stat
binlog zk quorum: host-10-10-14-108,host-10-10-14-106,host-10-10-14-107
```

```
=====
=====
=
| S | latest WID | latest CID | version |
TN | write time | latest mutate table |
=====
=====
=
| 0 | 0 | 0 | 0 |
01/01 08:00:00.000 |
| 1 | 0 | 0 | 0 |
01/01 08:00:00.000 |
| 2 | 0 | 0 | 0 |
01/01 08:00:00.000 |
| 3 | 0 | 0 | 0 |
```

01/01 08:00:00.000 |

```
=====
=====
=
```

1.4.11 checkpoint

checkpoint 即备集群 atrxdc 记录的当前回放位置;

- 查看 checkpoint;

atrxdc -get rcp

- 重置 checkpoint, 则启动 atrxdc 回放服务后, 会从 binlog 起始位置开始回放;

```
[trafodion@host-10-10-14-101 ~]$ atrxdc -resetrpc
```

1.5 hbase binlog 清理策略

1.5.1 手工清理 binlog

- 异步复制支持针对 hbase binlog 内记录的清理, 支持清理
- 全部 binlog 记录, 也支持清理某段时间之前的记录;
- 清理时间段单位可以按天/时/分

```
[trafodion@host-10-10-14-105 ~]$ atrxdc -cl 1 day ago
**WARNING** this command will clear binlog, atrxdc
service depend on this log, continue?[y/n]: y
**WARNING** the binlog before 10/25/2020 11:33:52 AM will
be delete, continue?[y/n]: y
--- Operation success.
```

```
[trafodion@host-10-10-14-105 ~]$ atrxdc -cl 5 hour ago
**WARNING** this command will clear binlog, atrxdc
```

```
service depend on this log, continue?[y/n]: y
**WARNING** the binlog before 10/26/2020 06:34:47 AM will
be delete, continue?[y/n]: y
--- Operation success.
[trafodion@host-10-10-14-105 ~]$
```

```
[trafodion@host-10-10-14-105 ~]$ atrxdc -cl 10 minute ago
**WARNING** this command will clear binlog, atrxdc
service depend on this log, continue?[y/n]: y
**WARNING** the binlog before 10/26/2020 11:25:16 AM will
be delete, continue?[y/n]: y
--- Operation success.
[trafodion@host-10-10-14-105 ~]$
```

- 支持清理指定具体时间之前的数据

```
[trafodion@host-10-10-14-105 ~]$ atrxdc -cl "2020/10/26
11:30:20"
**WARNING** this command will clear binlog, atrxdc
service depend on this log, continue?[y/n]: y
**WARNING** the binlog before 10/26/2020 11:30:20 AM will
be delete, continue?[y/n]: y
--- Operation success.
```

1.5.2 自动清理 binlog

- 设置 ttl 间隔，hbase 会自动清理该 ttl 之前的 binlog 记录（单位：秒），默认配置是 3 天；
- 配置 ttl 示例

```
atrxdc -setttl 259200
```

- 查看 binlog 表 ttl 配置, NAME => 'mf'行, 红字部分即 ttl 配置

```
hbase(main):005:0> desc
'TRAF_RSRVD_5:TRAFODION._DTM_.TRAFODION_BINLOG'
Table TRAF_RSRVD_5:TRAFODION._DTM_.TRAFODION_BINLOG is
ENABLED
TRAF_RSRVD_5:TRAFODION._DTM_.TRAFODION_BINLOG
COLUMN FAMILIES DESCRIPTION
{NAME => 'mf', BLOOMFILTER => 'ROW', VERSIONS => '1',
IN_MEMORY => 'false', KEEP_DELETED_CELLS => 'FALSE',
DATA_BLOCK_ENCODING => 'NONE', TTL => '259200 SECONDS (3
DAY
S)', COMPRESSION => 'NONE', MIN_VERSIONS => '0',
BLOCKCACHE => 'true', BLOCKSIZE => '65536',
REPLICATION_SCOPE => '0'}
{NAME => 'mt_', BLOOMFILTER => 'ROW', VERSIONS => '1000',
IN_MEMORY => 'false', KEEP_DELETED_CELLS => 'FALSE',
DATA_BLOCK_ENCODING => 'NONE', TTL => 'FOREVER', COMPRES
SION => 'NONE', MIN_VERSIONS => '0', BLOCKCACHE =>
'true', BLOCKSIZE => '65536', REPLICATION_SCOPE => '0'}
2 row(s) in 0.1150 seconds
```

1.6 异常问题处理

1.6.1 备集群启动 atrxdc 服务, 没有任何输出, 也没有启动

解决办法: 检查主备集群 xdc 配置是否正确, 一般是由于 xdc 配置错误或没有配置导致;

1.6.2 备集群 atrxdc 服务异常, 导致无法同步数据;

解决办法:

- 一般 atrxdc manager/worker 进程日志会有相关报错信息。
- 尝试重启备集群 atrxdc 服务，一般 atrxdc 服务会继续拉取数据。

1.6.3 备集群 atrxdc 服务异常无法同步数据， atrxdc 服务重启无效

解决办法：

- 可以查看 atrxdc manager/worker 进程日志
- 分别清理主备集群 binlog
- 通过 br 恢复数据，主集群 backup/export-备集群 import/restore

1.6.4 普通表增加异步复制属性

解决办法：普通表如果要修改成异步表，需要在备集群也创建相同普通表，否则无法同步，会导致 atrxdc 服务 block。

1.6.5 备集群清理 binlog 后，启动 atrxdc 服务，没有继续回放 binlog 记录；

解决办法：一般是由于清理 binlog 后，没有重置检查点导致，重置检查点(atrxdc -resetrep)即可；

1.6.6 当 atrxdc replayer 日志内持续输出 map size is XX 的处理

一般出现该日志，说明备集群回放时发现当前事务不完整，导致备集群回放 block；
问题原因：

- 可能事务比较大，包含多条记录，备集群只拉取了一部分，备集群回放比较快，只读取了一部分 binlog 记录；
- 可能主备集群网络故障，导致只拉取了一部分；检查网络，确认网络正常后，如果过段时间不再输出该信息，说明后续备集群拉取完整了，不用管；
- 如果一段时间内一直持续输出，说明可能确实是事务丢了一部分记录，导致事务不完整；

- 通过 `atrxdc -stat`，对比主备集群 binlog 记录，如果主备集群 binlog 记录不同，说明备集群拉取不完整

解决办法：

备集群执行 `atrxdc -cl`，然后备集群通过 `atrxdc_start` 重启 `atrxdc` 服务，重新拉取回放；

- 通过 `atrxdc -stat`，对比主备集群 binlog 记录，如果主备集群 binlog 记录相同，说明主集群写入 binlog 不完整

解决办法：

这个比较麻烦，需要停止服务，然后主备集群分别做 `atrxdc -cl`，清理 binlog，然后通过备份恢复策略，来保证主备集群数据一致，然后备集群启动 `atrxdc` 服务，然后主集群开始对外提供服务；

1.6.7 如果主集群数据库处理业务正常，hdfs binlogcache 内频繁写入新记录，说明 hbase binlog 压力过大，导致 binlog 访问频繁超时

解决办法：如果主集群 hbase binlog 压力过大，导致 binlog 频繁超时，如果 hdfs binlogcache 频繁写入文件，则可以尝试在 hbase 内增加 `BINLOG_PUT_RPC_TIMEOUT` 参数配置，建议值 `BINLOG_PUT_RPC_TIMEOUT=10000`，单位：ms，实际生产环境可酌情调整
查看 hdfs binlogcache 文件：

```
hdfs dfs -ls /user/trafodion/binlogcache
```

1.7 主备集群不一致故障处理思路

1.7.1 当主备集群表数据一致，binlog 数据不一致

解决办法：

1. 查看 atrxdc 拉取和回放日志，确认备集群拉取回放是否结束；
2. 可能备集群 binlog 不完整，清理备集群 binlog 记录，并重启 atrxdc 服务重新拉取回放；
3. 可能主集群 binlog 不完整，这就很麻烦了，需要停主集群服务，清理主备集群 binlog，并重启备集群 atrxdc 服务；

1.7.2 当主备集群表数据不一致，binlog 数据一致

解决办法：

1. 查看 atrxdc 拉取和回放日志，确认备集群拉取回放是否结束
2. 可能主集群 binlog 不完整，需要停主集群服务，清理主备集群 binlog，并重启备集群 atrxdc 服务；

1.7.3 当主备集群表数据不一致，binlog 数据不一致

解决办法：

1. 查看 atrxdc 拉取和回放日志，确认备集群拉取回放是否结束
2. 可能备集群 binlog 不完整，清理备集群 binlog 记录，并重启 atrxdc 服务重新拉取回放；
3. 可能主集群 binlog 不完整，这就很麻烦了，需要停主集群服务，清理主备集群 binlog，并重启备集群 atrxdc 服务；

1.8 特殊说明

1.8.1 版本升级

数据库版本升级后，atrxdc 相关配置会保留，如果不涉及 atrxdc 更新，可以直接

工作；如果涉及 atrxdc 更新，且影响较大，建议在保证主备集群数据一致，且主集群没有业务处理后，做 atrxdc -init 初始化 binlog，并重置 checkpoint；

1.8.2 由于初始化 binlog 操作(atrxdc -init)，会重置 ttl 配置，故在执行完 atrxdc -init 操作后，一定要重新配置 ttl(atrxdc -setttl)；

1.8.3 由于清理 binlog 操作(atrxdc -cl)，会重置 ttl 配置，故在执行完 atrxdc -cl 后，一定要重新配置 ttl(atrxdc -setttl)

1.8.4 在最大性能模式下，为了提高性能及稳定性，建议酌情提高 buffersize 设置，默认为 1000 字节，一般建议调整为 4M，但在发生如断电/断网等异常故障时，会导致丢失更多的缓存数据(丢失当前正在传输的那组 buffersize)，建议生产环境酌情配置

2 Binlog Reader 介绍

本章主要讲解 binlog_reader 相关内容

2.1 基本原理

binlog_reader 读取一条日志，是一个 protobuf，反序列化之后，可以获取该日志对应的 RegionName，进而提取出表的名字。将表的主要元数据信息：列的类型，主键的列表写到 binlog 表的 meta CF 里面去，key 是表名。Binlog_reader 接口通过读取表的元数据信息，用于后续针对 ddl/dml 操作记录的查询。

我们通过 binlog_reader 接口，针对 binlog 内每张表支持串行查询，即 1 个连接可以查询多张表，但不能多个连接并行查询 1 张表，主要是考虑数据时序问题，并行查询结果无法保证时序。目前处理机制通过 binlog_reader 查询时，会返回所有表的 ddl 记录 (truncate 也属于 ddl 事件范围内)，以及目标表的 dml 记录；

2.2 基本功能

2.2.1 支持查询的操作:

Binlog_reader 支持查询表级 dml 操作, 目前 binlog_reader 查询 ddl 支持常用的操作, 针对表/索引的 create/alter/drop 等操作的查询;另外支持了基于 sqlci/trafci 的 truncate 操作的查询, 不支持 truncate hbase 表的操作。

2.2.2 g 支持的数据类型:

目前支持的数据类型包括: INT, LARGEINT, NUMERIC, CHAR, VARCHAR, DATE, TIMESTAMP, 最新版本增加了 FLOAT 和 DECIMAL。

2.2.3 暂不支持的内容

目前暂不支持 sequence、savepoint、lob、function 等;

2.3 使用说明:

2.3.1 参数说明

易鲸捷数据库的 TMUDF 请参考易鲸捷数据库手册。用户可以使用标准的 select 语句读取 TMUDF 的输出。

接口参数示例:

```
select * from udf("_LIBMGR_".BINLOG_READER(Last-commit-id, Last-timestamp, tablelist, timeout,column_delimiter,commit_switch));
```

方法:

```
"_LIBMGR_".BINLOG_READER
```

2.3.1.1 入口参数

LAST-COMMIT-ID

类型: LARGEINT, 对应 java 的 long 类型

含义: 记录上一次数据同步复制完成的最大的 commit ID

LAST-TIMESTAMP

类型: LARGEINT, 对应 java 的 long 类型

含义: 记录上一次数据同步返回的最大的时间戳。这个值可以加快 binlog 断点

续读的起始点。

TABLE-LIST

类型： 字符串

含义:表示本 TMUDF 需要读取的 binlog 对应的表。不在这个列表中的表的 binlog 将不会返回。

应至少包含 schema 和 table 两部分，支持*星号为通配符；

如果有多张表，可以用”,”分号分割表名。

IDLE-TIMEOUT

类型： int

含义： TMUDF 的空闲超时时间。正常情况下，系统不断有数据变化，因此一次调用 TMUDF 将不停地返回。但如果系统在 idle timeout 时间内无增量数据，本次 TMUDF 将结束。

目前设置每 100ms scan 一次 hbase binlog，我们设置 idle timeout 入参，表示等待的 scan hbase 次数，如果当等待 scan binlog 这么多次都没有新数据，则本次查询返回，表示查询结束；

因为 TMUDF 本身有缓存，在 idle timeout 之前，缓存的数据无法返回，因此这个时间不应该设置很长；

COLUMN_DELIMITER

含义： 列分隔符，指返回 values 内的 sql 语句，不同列之间的分隔符，目前支持所有字符；

如果忽略该参数，则默认逗号分隔；

用于将返回结果 values 内的值，按照指定的分隔符作为不同 column 之间的分隔进行拆分；

COMMIT_SWITCH

输入参数为 0/1，用来开启或关闭是否返回 commit 记录；

默认值为 0（关闭 commit），即不会返回 commit 记录；

该参数用来标识通过 binlog_reader 接口查询，是否返回 commit 记录；

TRX_COUNT

用于确定本次查询返回的最大事务数量；

当事务过大或者包含针对多表操作等，可能会出现当次查询事务不完整的情况；

2.3.1.2 返回值

TABLE_NAME

表名，去掉 HBase Namespace。去掉 Catalog 的部分。为 Schema.Table 格式。比如”SEABASE.T1”

注：commit 记录不会输出 table_name

VALUES

VALUES 有两种可能的内容

如果 Operation 是 INSERT, 则 VALUES 就是所有列的值, 以逗号隔开的标准 SQL 兼容语法表示。

如果 Operation 是 DELETE, 则 VALUES 是所有主键列的值, 以逗号隔开。

EVENT

目前支持的 event 包括：DDL/COMMIT/INSERT/UPDATE/UPSERT。

其中所有 ddl 操作都会由 ddl 事物描述，不做详细区分，truncate 操作也作为 ddl 事件处理；

后续版本会添加一个 sequence 对应的事件，用于同步 sequence

COMMIT_ID

事务提交 ID，代表了操作的先后顺序。

同步软件应记录下本次已经同步的最大 commit_ID，并以该值作为下一次调用 BINLOG_READER 的入口参数。这样就可以顺序地读取和回放日志，而避免重复，也避免遗漏。

LAST_TS

事务提交的时间。

这个值提供给同步软件，同步软件记录下上一次读取的最大的 timestamp，并将该值传入 BINLOG_READER。这个不是必须的，但是这有助于提高 binlog_reader

定位起始日志的性能，否则，binlog_reader 将从头开始找日志起点。

LINE_NUM

指每条数据返回的行数，因为 values 设置最大长度为 2048 字节，超过后，会拆分成多行返回，指每条数据返回的行数，因为 values 设置最大长度为 2048 字节，超过后，会拆分成多行返回，line_num 列就是记录该行是该条操作记录的第几行；如果只有 1 行，则 line_num 为 0；如果包含多行，则 line_num 从 1 开始计数；

OPTIME

操作时间戳

TX_ID

事务 id

SEQ_NUM

用来描述一个事务内多个 ddl/dml 操作的先后次序，是针对一个事务内相同表的所有操作按顺序排序的，同一事务内不同表的 seq_num 独立排序；

seq_num 是在 binlog_reader 内记录的，不是在 binlog 内，所以当在一个事务内包含多表操作时，seq_num 也是按照表内排序的，不同表之间 seq_num 无关；

主要用于英方中间件读取到数据后，确定回放的顺序，比如一个事务执行到一半，中间件或对端数据库挂了，那么需要从中断的地方开始重新回放，需要明确上一次回放到哪个事务的哪条 sql 失败的，根据 seq_num 来确定的，这样从上一个失败的 sql 开始继续回放；

TX_TOTAL_NUM

指一个事务提交涉及的目标表 region 数，当一个事务在多个 region 提交，或者包含针对多张表的操作时，会拆分成多条记录；

该值可以用来验证事务的完整性；

2.4 特殊说明

- 针对索引表 update 动作特殊处理，在一个事务内，针对多表做 update，如果表上有主键约束，通过 binlog_reader 查询时，update 返回的是 delete+insert；

如果没有主键约束，则返回 update；

- 通过 binlog_reader 查询，会返回目标表的 dml 记录，以及所有异步表的 ddl 记录；commit 记录是否返回由入参开关决定；
- binlog 内每个事物，通过 binlog_reader 解析得到 1 条或多条 dml/ddl 记录+1 条 commit 记录；
- 当使用 binlog_reader 接口查询，为了保证事务回放顺序，建议查询条件使用 commit_id/line_num/seq_num 排序

2.5 异常问题处理：

- binlog 内目标表只有少量数据，但通过 binlog_reader 查询，一直返回相同记录；

解决办法：一般是由于 hbase binlog 内缺少该表的元数据记录导致，需要 drop/create 目标表，用来在 binlog 内写入元数据。

- 由于通过 binlog_reader 接口查询数据时，访问的是 binlog 内数据，如果 binlog 内没有目标表的元数据，则无法查询到该表的表名和列信息，故在数据库 atrxdc 运行后，后期只能通过 atrxdc -cl 来清理 binlog，这样能保留元数据。

2.6 升级方式：

- 可以通过更新 lib_mgmt.jar 包跟新 binlog_reader

目前 binlog_reader 已经与 sql 底层分离，以后升级 binlog_reader，只要替换 \$STRAF_HOME/export/lib/lib_mgmt.jar 包，然后升级 lib_mgmt.jar 包即可；

```
10:42:15 SQL>initialize trafodion,upgrade library management;
```

```
--- SQL operation complete.
```

```
10:52:00 SQL>
```

- 通过升级数据库版本来更新