**OceanBase 0.4.2 CS旁路导入重构**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **文档**  **版本** | **修订章节** | **修订原因** | **修订日期** | **修订人** |
| 0.1 | 全部 | 新建 | 2013-08-01 | 永乐 |
| 0.2 | 全部 | 按照主备集群的逻辑修改 | 2013-11-26 | 永乐 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

目录

[目录 2](#_Toc25962)

[重构的意义与限制 4](#_Toc3021)

[cs旁路导入的流程： 5](#_Toc16417)

[主备集群的一些说明 6](#_Toc23287)

[导入任务状态的说明 7](#_Toc2307)

[方案的容灾 7](#_Toc17302)

# 重构的意义与限制

此次重构最重要的是把将ob改造为接收rs\_admin import <tablename> <table\_id> <uri>

命令后就能将指定的表从指定的路径导入进来。

uri格式为有如下两种

* oceanbase://rs\_ip:rs\_port

对应集群复制功能，可以指定复制指定的table

* proxy://data\_source\_name://path/to/sstable/dir

旁路导入后的导入接口，改造后对旁路导入的数据路径有一定的限制。

Partition file必须作为sstable\_dir里面的文件，文件名为\_partition\_file

table相关的信息保存在sstable\_dir里面的文件，文件名为\_table\_info

Sstable都必须在sstable\_dir里面，文件名为table\_id-seq\_no. （seq\_no为10位，0补齐）

hadoop://group/oceanbase/mrsstable\_data/appname/task\_id/table\_name/sstables

\_partition\_file

\_table\_info

1001-0000000000

1001-0000000001

1001-0000000002

1001-0000000003

1001-0000000004

1001-0000000005

…

Mapreduce生成的文件名实际上是我们的jar中指定的，因此只要用户使用我们的jar包来生成sstable，那么这个限制对用户来说就是透明的。

\_table\_info内容包括：

table\_name=table\_name

table\_id=1001

rowkey\_desc=1,4,5,6-5 // 1,4,5,6为类型，分别为int，data time， precise date time， varchar，每个rowkey字段用逗号分割。字符串有长度限制，用-分割。

sstable\_version=

delim=1

# cs旁路导入的流程：

## 用户手工或者自动配置旁路导入配置文件

## 用户在云梯上准备导入文件

## 用户发送导入命令到importserver.

python2.6 importcli.py -t OVERWRITE /path/to/yunti/table1 tablename1 /path/to/yunti/table2 tablename2

## importserver在云梯上生成sstable

## importserver发命令到主主rs

rs\_admin import <tablename> <table\_id> <uri>

uri格式为proxy://hadoop://path/to/sstable/dir

## 主主rs通知各个主rs发起改表的旁路导入

## 所有的主rs分别通过file\_proxy从云梯获取range\_table

ObRootRpcStub::fetch\_range\_table (ObString& table\_name, ObString& uri)

packet\_code=OB\_FETCH\_RANGE\_TABLE

uri=data\_source\_name://path/to/sstable/dir

## 所有的主Rs把rangetable插入到roottable，并对每个table维护一个导入的状态记录数据结构。

## 所有的主rs负责本集群的导入调度，发送拷贝云梯或者oceanbase数据的命令到cs （有N个请求）

ObRootRpcStub::migrate\_tablet(DataSourceDesc& desc);

## cs通过proxy从云梯获取sstable 数据（有N个请求）

fetch\_data(DataSourceDesc& desc)

## cs汇报fetch data结果，走migrateover接口

从云梯拿数据时，如果rs收到下面的错误则会直接认为挂了设置失败标记：OB\_DATA\_SOURCE\_NOT\_EXIST，

OB\_DATA\_SOURCE\_TABLE\_NOT\_EXIST，

OB\_DATA\_SOURCE\_RANGE\_NOT\_EXIST，

OB\_DATA\_SOURCE\_DATA\_NOT\_EXIST，

OB\_SSTABLE\_VERSION\_UNEQUAL

其他错误则会继续重试。

## 主主rs当自己失败或者拷贝完成或者每60s会检测备主rs上导入表的状态。

## 主主rs在12步检测成功则会切换tableid，通知各个主rs更新本集群所有server的schema，并删掉旧的tablet。

## 如果主rs所有旁路导入的表已经完成，则退出旁路导入状态，允许每日合并。

# 主备集群的一些说明

* 所有的rs\_admin命令都是发给主主rs的，导入命令和kill命令有主主rs转发给所有的主rs（包括它自己）。
* 主主rs每个60秒会检查一次各个主rs的状态，如果有返回failed的则发起导入失败的命令。如果检测的rpc失败，则只打印warn日志。ObRootBalancer::stop\_failed\_load\_table\_task
* 主主rs在自己成功后会立刻发起一次检查，如果有failed的则发起导入失败的命令，如果有DOING状态的主rs，则以后每个30s检查一次(配置项load\_data\_check\_status\_interval)。

ObRootBalancer::handle\_load\_table\_done.

这个函数里面有个地方会比较诡异，如果主主rs检测各个主rs的时候rpc调用失败，load\_table\_history这个表会显示FAILED。30秒后如果检测成功了又会变成DONE。 这里会造成importserver认为导入挂了，返回给客户让客户也觉得挂了，实际查看的时候却是导入成功的。但是如果不这样做，如果主主rs和备注rs通信挂了貌似就不会有警报了。

另外这里面切tableid挂掉和set\_status DONE的时候挂掉需要人工处理。

* 主主rs在自己失败后会立刻发起导入失败的命令，同步到各个主rs。

# 导入任务状态的说明

INIT： 初始化的状态，只是用来表示这个结构体是空的，不是导入任务的状态。

NEW： 新加的导入任务的状态为NEW。

PREPARE： 当主主rs成功发送导入命令给所有的主rs后，所有主rs上面的导入命令会 被设置为PREPARE，表示主rs可以开始从proxy拉去rangelist了。

DOING: 当主rs成功拉去rangelist并且插入roottable以后，该任务状态变成DOING，表 示这个集群可以开始拉这个表的tablet了。注意：当主rs检测到某个表的tablet 全部拷贝完成时，改表的状态还是保持DOING，只有主主rs才有权把主rs的 状态切成DONE。

DONE： 导入任务成功的时候的状态，只存在在最后完成切换状态时主主rs发给各个主 rs的rpc中。各个rs收到DONE的状态后会开始清理内部旧的tablet。

FAILED： 失败的状态。

KILLED：只用在主主rs向各个主rs发送的切换状态的rpc中。

COPIED： 值用在主rs响应主主rs的查询状态的rpc中，表示该表对应的tablet已经成 功拷贝并加载。

# 方案的容灾

1. 在rs\_admin发起旁路导入的时候，如果所有主rs的状态没有进入到PREPARE，那么这个导入就失败了，需要手工发起kill命令。
2. 在一个表旁路导入加载完成的瞬间，切换tableid失败或者设置各个主rs的状态失败则会导致导入失败，需要人工发起kill命令。
3. 在导入过程中切换主备rsvip或者主备集群可以容错。
4. 在发起导入的时候每个rs从云梯那rangelist会有3\*3=9次的重试集群，如果有一个rs拉去rangelist连续九次失败，那么整个导入就失败了。
5. 从云梯拉tablet超时或者失败可以容错。