



# OceanBase 0.4.2

## 操作指南

文档版本：Beta 02

发布日期：2014.04.01

支付宝（中国）网络技术有限公司 OceanBase 团队

# 前言

## 概述

本文档主要介绍OceanBase 0.4.2的旁路导、使用DataX导入导出数据、主备集群切换、ChunkServer更换磁盘和rs\_admin使用等操作。

## 读者对象

本文档主要适用于：

- 安装工程师。
- 数据库管理工程师。

## 通用约定

在本文档中可能出现下列各式，它们所代表的含义如下。

格式	说明
警告	表示可能导致设备损坏、数据丢失或不可预知的结果。
注意	表示可能导致设备性能降低、服务不可用。
小窍门	可以帮助您解决某个问题或节省您的时间。
说明	表示正文的附加信息，是对正文的强调和补充。
宋体	表示正文。
<b>粗体</b>	表示命令行中的关键字（命令中保持不变、必须照输的部分）或者正文中强调的内容。
斜体	用于变量输入。
{ a   b   ... }	表示从两个或多个选项中选取一个。
[ ]	表示用“[ ]”括起来的部分在命令配置时是可选的。

## 修订记录

修改记录累积了每次文档更新的说明。最新版本的文档包含以前所有文档版本。

版本和发布日期	说明
Beta 02（2014-04-01）	第一次发布Beta版本，适用于OceanBase 0.4.2。

## 联系我们

如果您有任何疑问或是想了解 OceanBase 的最新开源动态消息，请联系我们：

支付宝（中国）网络技术有限公司·OceanBase 团队

地址：杭州市万塘路 18 号黄龙时代广场 B 座；邮编：310099

北京市朝阳区东三环中路 1 号环球金融中心西塔 14 层；邮编：100020

邮箱：[alipay-oceanbase-support@list.alibaba-inc.com](mailto:alipay-oceanbase-support@list.alibaba-inc.com)

新浪微博：<http://weibo.com/u/2356115944>

技术交流群（阿里旺旺）：853923637

# 目 录

---

1 旁路导入.....	- 1 -
1.1 基本信息 .....	- 1 -
1.2 导入前准备 .....	- 3 -
1.2.1 获取 OceanBase 服务器信息 .....	- 3 -
1.2.2 服务器规划 .....	- 4 -
1.2.3 检查导入环境 .....	- 5 -
1.3 部署 Hadoop 集群 .....	- 5 -
1.3.1 安装前准备 .....	- 5 -
1.3.2 安装 Hadoop .....	- 7 -
1.3.3 配置 Hadoop .....	- 9 -
1.4 旁路导入配置 .....	- 10 -
1.4.1 配置并启动 ImportServer .....	- 10 -
1.4.2 配置并启动 ProxyServer .....	- 13 -
1.4.3 配置 OceanBase .....	- 14 -
1.5 导入数据 .....	- 15 -
1.6 其他操作 .....	- 16 -
2 DataX 导入/导出数据.....	- 21 -
2.1 基本信息 .....	- 21 -
2.2 部署 DataX .....	- 22 -
2.2.1 安装前检查 .....	- 22 -
2.2.2 安装 JDK 补丁包.....	- 22 -
2.2.3 创建安装用户 .....	- 23 -
2.2.4 安装 DataX 软件 .....	- 23 -
2.3 OceanBase 导入/导出.....	- 24 -
3 主备集群切换.....	- 29 -
4 ChunkServer 更换磁盘.....	- 32 -
5 rs_admin 使用 .....	- 34 -

# 1 旁路导入

旁路导入 OceanBase 旁路导入利用 Hadoop 生成 ChunkServer 的数据文件进行直接装载，可以加快 OceanBase 数据批量写入的速度。

## 1.1 基本信息

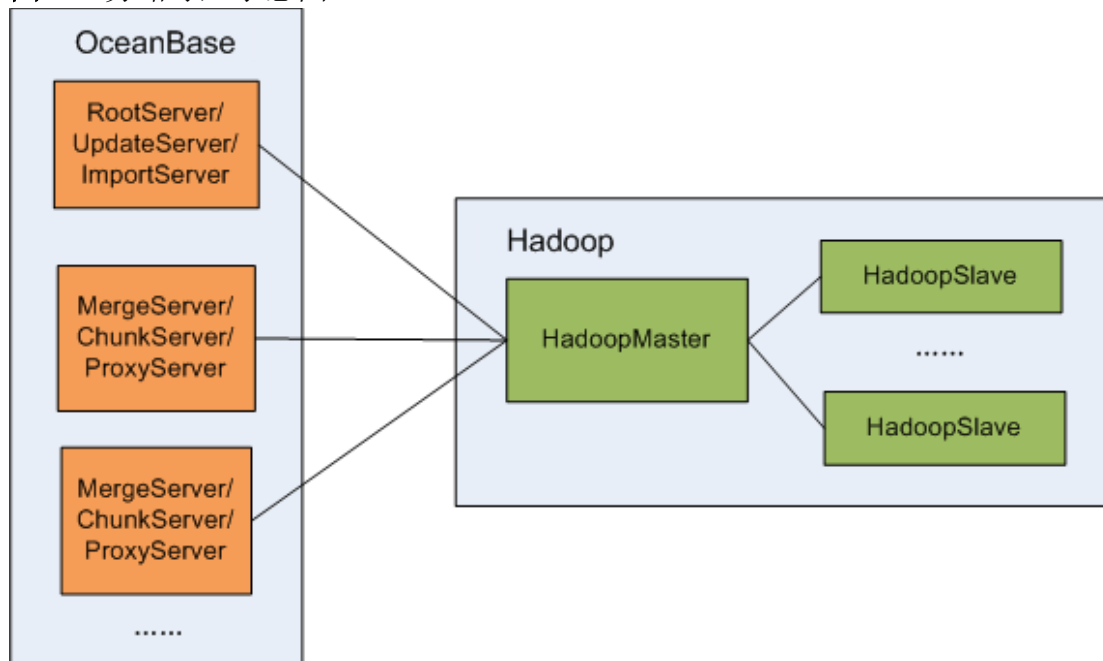
本小节介绍了在进行 OceanBase 旁路导入前，您需要了解的基本信息，包括旁路导入流程、支持格式和注意事项。

### \* 旁路导入示意图和流程

主要使用的场景如下，其结构示意图如[图 1-1](#)所示：

- 上百 G 的海量数据导入。
- 导入前的数据不保留。

图 1-1 旁路导入示意图



旁路导入各服务说明如下：

- **ImportServer**  
启动在 RootServer 上，用于发送生成和加载数据的指令。

- **ProxyServer**  
代理服务，启动在 **ChunkServer** 上，用于隔离 **OceanBase** 和 **Hadoop**，起到中转作用。
- **Hadoop**  
包括 **HadoopMaster**、**HadoopSlave** 和 **HadoopClient**。其中 **HadoopMaster** 和 **HadoopSlave** 组成 **Hadoop** 集群；**HadoopClient** 需要在 **ImportServer** 和 **ProxyServer** 服务器上部署。

旁路导入基本流程如下：

1. **ImportServer** 调用 **mapreduce** 在 **Hadoop** 上生成将指定表的 **Tablet**。
2. **ImportServer** 发送指令给 **RootServer** 发起加载任务。
3. **RootServer** 通过 **ProxyServer** 从 **Hadoop** 上获取 **range**。
4. **RootServer** 发送命令给 **ChunkServer**，使 **ChunkServer** 通过 **ProxyServer** 获取对应 **range** 的 **Tablet**。
5. 全部 **range** 的 **Tablet** 加载完毕后，**RootServer** 修改表的 **table id**，使新数据生效，并删除旧数据。

#### \* 支持格式

旁路导入支持的数据类型为 **int**，**varchar**，**timestamp** 三种。其中时间格式如下：

- **YYYY-MM-DD hh:mm:ss**  
例如：2014-01-01 01:02:03，导入后为“2014-01-01 01:02:03”。
- **YYYY-MM-DD**  
例如：2014-01-01，导入后为“2014-01-01 00:00:00”。
- **YYYYMMDD hh:mm:ss**  
例如：20140101 01:02:03，导入后为“2014-01-01 01:02:03”。
- **YYYYMMDDhhmmss**  
例如：20140101010203，导入后为“2014-01-01 01:02:03”。
- **YYYYMMDD**  
例如：20140101，导入后为“2014-01-01 00:00:00”。

#### \* 注意事项

在进行旁路导入时，需要注意以下几点：

- 进行旁路导入的表中的原数据将会被全部替换，该表的 **table id** 也会被自动改变。
- 列名不能出现“-”和“,”。
- 希望业务人员在提供数据的时候能够保证数据的随机性，这样生成的 **sstable** 会比较均匀。如果是排序过的数据，请联系 **DBA** 调整采样参数。

- 导入的数据必须是明文的，不能是二进制的。**varchar** 类型在 **java** 客户端实际上可以插入二进制的数据，但是旁路导入的时候不能接受二进制的数据。因为旁路导入依赖分隔符来划分不同的列，如果使用二进制的数据会导致读数据失败。
- 导入的数据建议使用没有压缩的，否则压缩率过高会导致单个 **map** 处理的数据过大，造成整体速度较慢的问题。
- 如果用随机采样建议输入的单个文件大小为 **hdfs** 的 **split** 大小(默认为 256m)。**hive** 可以通过 **hive.exec.reducers.bytes.per.reducer** 设置。
- 如果导入失败，请联系 **OceanBase DBA**，请不要简单的进行重试。
- 如果有 **createtime** 和 **modifytime** 字段，建议在建表的时候放在最前面两列。
- 磁盘空间警报建议设置在 **75%**，需要预留一次导入的空间。
- 每日合并和旁路导入一定要错开，如果旁路导入过程中触发每日合并会导致改表导入失败。同时数据恢复将很复杂。

## 1.2 导入前准备

在进行 **OceanBase** 前，请先获取 **OceanBase** 服务器信息、完成服务器规划和检查导入环境。

### 1.2.1 获取 **OceanBase** 服务器信息

在导入数据前，请先获取 **OceanBase** 的安装信息，假设现有 **OceanBase** 服务器信息如表 1-1 所示。

表 1-1 **OceanBase** 服务器信息

服务器	IP	端口	安装用户/密码	安装目录
RootServer	10.10.10.2	服务端口：2500	admin/Abc@123	/home/admin/oceanbase
UpdateServer		服务端口：2700 合并端口：2701		
Listener		服务端口：2800 MySQL 协议端口：2828		
ChunkServer	10.10.10.4	服务端口：2600	admin/Abc@1	/home/admin/o

服务器	IP	端口	安装用户/密码	安装目录
MergeServer	10.10.10.5	服务端口：2800 MySQL 协议端口：2880	23	ceanbase

### 1.2.2 服务器规划

在进行 OceanBase 旁路导入前，我们需要启动 OceanBase 的 ImportServer 和 ProxyServer 服务，并安装 Hadoop。

ImportServer 启动在 RootServer 服务器中，ProxyServer 启动在 ChunkServer 服务器中，其规划如表 1-2 所示。

*说明：*多台 ChunkServer 时，可以启动多个 ProxyServer，也可以在 ChunkServer 服务器中选择一台启动。

表 1-2 ImportServer/ProxyServer 规划

服务器	IP	端口	安装用户/密码	安装目录
ImportServer	10.10.10.2	2900	admin/Abc@123	/home/admin/oceanbase
ProxyServer	10.10.10.4	2650	admin/Abc@123	/home/admin/oceanbase

Hadoop 安装包括 HadoopMaster、HadoopSlave 和 HadoopClient。Hadoop 的详细介绍请参见 Hadoop 官方网站“<http://hadoop.apache.org/>”。本文档仅介绍其与 OceanBase 旁路导入相关的安装、配置。

Hadoop 安装规划如表 1-3 所示。

表 1-3 Hadoop 安装规划

服务器	IP/主机名	端口	安装用户/密码	安装目录
HadoopMaster	10.10.10.41/OB.Hadoop.Master	HDFS 端口：9000 jobtracker 端口：9001	hadoop/Abc@123	/home/hadoop/hadoop-0.19.1
HadoopSlave	10.10.10.42/OB.Hadoop.Slave01 10.10.10.43/OB.Hadoop.Slave02	-	hadoop/Abc@123	/home/hadoop/hadoop-0.19.1



服务器	IP/主机名	端口	安装用户/密码	安装目录
HadoopClient <i>说明: ImportServer 和 ProxyServer 所在的服务器中任均需部署。</i>	10.10.10.2 10.10.10.4	-	admin/Abc@123	/home/admin/hadoop-0.19.1

### 1.2.3 检查导入环境

检查旁路导入环境的操作步骤如下：

1. 以 **admin** 用户登录 RootServer 服务器。
2. 执行以下命令，检查 Python 版本是否大于等于“2.6”。  
**python -V**
3. 执行以下命令，检查 JDK 版本是否大于等于“1.6”。  
**java -version**

## 1.3 部署 Hadoop 集群

主要介绍如何安装部署 Hadoop 集群。本文档仅介绍其与 OceanBase 旁路导入相关的安装、配置。Hadoop 的详细介绍请参见 Hadoop 官方网站“<http://hadoop.apache.org/>”。

### 1.3.1 安装前准备

安装 Hadoop 前请先关闭防火墙、修改主机名、创建安装用户、打通 SSH 权限以及下载安装包。

#### \* 关闭防火墙

关闭防火墙操作步骤如下：

1. 以 **root** 登录 HadoopMaster 服务器（10.10.10.41）。
2. 执行以下命令，关闭防火墙。  
**service iptables stop**
3. 执行以下命令，查看防火墙是否关闭。  
**service iptables status**
4. 参考以上步骤，分别在 HadoopClient 服务器和两台 HadoopSlave 服务器上关闭防火墙。

#### \* 修改主机名

修改主机名操作步骤如下：

1. 以 **root** 用户登录 HadoopMaster 服务器（10.10.10.41）。
2. 用 **vi** 编辑器，将 Hadoop 各服务器的 IP 和主机名加入“/etc/hosts”文件。

```
10.10.10.41 OB.Hadoop.Master
10.10.10.42 OB.Hadoop.Slave01
10.10.10.43 OB.Hadoop.Slave02
```

3. 参考以上步骤，分别在两台 HadoopSlave 服务器上修改“/etc/hosts”文件。

### \* 创建安装用户

创建安装用户操作步骤如下：

1. 以 **root** 用户登录 HadoopMaster 服务器（10.10.10.41）。
2. 执行如下命令，创建 HadoopMaster 的安装用户。  
**useradd -d /home/hadoop -s /bin/bash -m hadoop**
3. 执行如下命令，为用户“hadoopmaster”设置密码。  
**passwd hadoop**
4. 您需要根据系统的提示输入两次密码“Abc@123”。
5. 参考以上步骤，分别在两台 HadoopSlave 服务器上安装用户和密码“hadoop/Abc@123”。

### \* 打通 SSH 权限

在 HadoopMaster 和 HadoopSlave 之间打通 SSH 无密码登录。HadoopMaster 可以使用 SSH 无密码登录 HadoopSlave，HadoopSlave 也可以使用 SSH 无密码登录到 HadoopMaster。

**注意：**HadoopMaster 需要打通自身到自身的 SSH 无密码登录。

打通 SSH 权限的操作步骤如下：

1. 打通 HadoopMaster 到 HadoopSlave 的 SSH 无密码登录。
  - a. 以 **hadoop** 用户登录 HadoopMaster 服务器（10.10.10.41）。
  - b. 执行以下命令，并按“Enter”键，直至生成公钥。  
**ssh-keygen -t rsa**
  - c. 执行以下命令，并根据提示输入登录密码，配置免登录。  
**ssh-copy-id hadoop@10.10.10.41**  
**ssh-copy-id hadoop@10.10.10.42**  
**ssh-copy-id hadoop@10.10.10.43**
2. 参考以下步骤，依次打通两台 HadoopSlave 到 HadoopMaster 的 SSH 无密码登录。
  - a. 依次以 hadoop 用户登录两台 HadoopSlave 服务器（10.10.10.42/10.10.10.43）。
  - b. 执行以下命令，并按“Enter”键，直至生成公钥。  
**ssh-keygen -t rsa**

- c. 执行以下命令，并根据提示输入登录密码，配置免登录。
- ```
ssh-copy-id hadoop@10.10.10.41
```

#### \* 下载安装包

OceanBase 旁路导入采用 Hadoop 0.19.1 版本，获取安装包的操作步骤如下：

1. 以 **hadoop** 用户登录 HadoopMaster 服务器（10.10.10.41）。
2. 执行以下命令，获取 Hadoop 安装包。  
**wget https://archive.apache.org/dist/hadoop/core/hadoop-0.19.1/hadoop-0.19.1.tar.gz**

### 1.3.2 安装 Hadoop

安装 Hadoop 集群操作步骤如下：

1. 以 **hadoop** 用户登录 HadoopMaster 服务器（10.10.10.41）。
2. 执行以下命令，解压缩安装包。  
**tar xvf hadoop-0.19.1.tar.gz**
3. 使用 **vi** 编辑器，修改“/home/hadoop/hadoop-0.19.1/conf/hadoop-site.xml”文件。

```
<configuration>
  <property>
    <name>fs.default.name</name>
    <value>hdfs://10.10.10.41:9000</value>
    <description>HDFS 的 URI，文件系统://namenode 标识:端口号。
    OB.Hadoop.Master 为 HadoopMaster 的机器名</description>
  </property>
  <property>
    <name>mapred.job.tracker</name>
    <value>10.10.10.41:9001</value>
    <description>jobtracker 的标识:端口号，不是 URI。</description>
  </property>
  <property>
    <name>dfs.replication</name>
    <value>1</value>
    <description>副本个数，不配置默认是 3,应小于 HadoopSlave 的机器数量。
  </description>
  </property>
</configuration>
```

4. 使用 **vi** 编辑器，修改“/home/hadoop/hadoop-0.19.1/conf/hadoop-env.sh”文件。

```
.....
# The java implementation to use. Required.
export JAVA_HOME=/usr/lib/java
.....
```

```
# The maximum amount of heap to use, in MB. Default is 1000.
export HADOOP_HEAPSIZE=1000
.....
# Command specific options appended to HADOOP_OPTS when specified
export HADOOP_NAMENODE_OPTS="-Dcom.sun.management.jmxremote
$HADOOP_NAMENODE_OPTS"
export HADOOP_SECONDARYNAMENODE_OPTS="-Dcom.sun.management.jmxremote
$HADOOP_SECONDARYNAMENODE_OPTS"
export HADOOP_DATANODE_OPTS="-Dcom.sun.management.jmxremote
$HADOOP_DATANODE_OPTS"
export HADOOP_BALANCER_OPTS="-Dcom.sun.management.jmxremote
$HADOOP_BALANCER_OPTS"
export HADOOP_JOBTRACKER_OPTS="-Dcom.sun.management.jmxremote
$HADOOP_JOBTRACKER_OPTS"
.....
```

5. 使用 **vi** 编辑器，修改“/home/hadoop/hadoop-0.19.1/conf/masters”文件。

```
10.10.10.41
```

6. 使用 **vi** 编辑器，修改“/home/hadoop/hadoop-0.19.1/conf/slave”文件。

```
10.10.10.42
```

```
10.10.10.43
```

7. 依次执行以下命令，将 **hadoop** 文件夹分别拷贝到规划的 HadoopMaster 和 HadoopClient 服务器中。

*说明：*如果您需要部署多个 HadoopSlave，只需在步骤 9 中添加所有 HadoopSlave 节点的 IP，然后参考本步骤，将 **hadoop** 目录复制到相应的服务器中。

```
scp -r /home/hadoop/hadoop-0.19.1
hadoop@10.10.10.42:/home/hadoop
scp -r /home/hadoop/hadoop-0.19.1
hadoo@10.10.10.43:/home/hadoop
scp -r /home/hadoop/hadoop-0.19.1
admin@10.10.10.2:/home/admin
scp -r /home/hadoop/hadoop-0.19.1
admin@10.10.10.4:/home/admin
```

8. 执行以下命令，格式化 HDFS 文件系统。

```
/home/hadoop/hadoop-0.19.1/bin/hadoop namenode -format
```

9. 执行以下命令，启动 Hadoop 服务。

```
/home/hadoop/hadoop-0.19.1/bin/start-all.sh
```

*说明：*执行 **/home/hadoop/hadoop-0.19.1/bin/stop-all.sh** 命令，可以停止 Hadoop 服务。

验证 Hadoop 集群是否安装成功的操作步骤如下：

1. 以 **hadoop** 用户登录 HadoopMaster 的服务器（10.10.10.41）。
2. 执行 **jps** 命令，系统显示如下：

```
[hadoopmaster@OceanBase041 ~]$ jps
15188 SecondaryNameNode
15000 NameNode
4409 Jps
15283 JobTracker
```

3. 以 **hadoop** 用户分别登录两台 HadoopSlave 服务器（10.10.10.42/10.10.10.43）。
4. 执行 **jps** 命令，系统显示如下：

```
[hadoopslave@OceanBase042 ~]$ jps
9819 TaskTracker
11061 Jps
9693 DataNode
```

### 1.3.3 配置 Hadoop

Hadoop 安装完成后需要在 Hadoop 上创建 OceanBase 旁路导入所需的文件夹，并上传相关的 lib 文件。

1. 以 **hadoop** 用户登录 HadoopMaster 服务器（10.10.10.41）。
2. 依次执行以下命令，创建导入数据所需的文件夹。各文件夹说明如下：
  - **input**: 用于存放导入的数据文件。
  - **output**: 用于存放生成的 sstable 文件。“obtest”为 OceanBase 的 APP 名称。
  - **config**: 用于存放导入数据转换过程中生成的配置文件。
  - **mrllib**: 用于存放导入 OceanBase 数据时所需的“native library”。

```
/home/hadoop/hadoop-0.19.1/bin/hadoop --config
/home/hadoop/hadoop-0.19.1/conf/ fs -mkdir
/home/hadoop/hadoop-0.19.1/oceanbase/input
/home/hadoop/hadoop-0.19.1/bin/hadoop --config
/home/hadoop/hadoop-0.19.1/conf/ fs -mkdir
/home/hadoop/hadoop-0.19.1/oceanbase/output/obtest
/home/hadoop/hadoop-0.19.1/bin/hadoop --config
/home/hadoop/hadoop-0.19.1/conf/ fs -mkdir
/home/hadoop/hadoop-0.19.1/oceanbase/config
/home/hadoop/hadoop-0.19.1/bin/hadoop --config
/home/hadoop/hadoop-0.19.1/conf/ fs -mkdir
/home/hadoop/hadoop-0.19.1/oceanbase/mrllib
```

3. 执行以下命令，给“oceanbase”及其子目录赋予“777”权限。

```
/home/hadoop/hadoop-0.19.1/bin/hadoop --config
/home/hadoop/hadoop-0.19.1/conf/ fs -chmod -R 777
/home/hadoop/hadoop-0.19.1/oceanbase
```

4. 以 **admin** 用户登录 RootServer 服务器（10.10.10.2）。
5. 执行以下命令，将“native library”文件传入 HadoopMaster 节点的“/home/hadoop/hadoop-0.19.1/oceanbase/mrllib”目录中。

*说明：“native library 文件”在 OceanBase 安装目录下，“mrsstable\_lib\_5u”表示适用于 Hadoop 的环境是 redhat5（centos5）或者兼容的 linux 环境，“mrsstable\_lib\_6u”表示适用于 Hadoop 的环境是 redhat6（centos6）或者兼容的 linux 环境。*

```
/home/admin/hadoop-0.19.1/bin/hadoop --config
/home/admin/hadoop-0.19.1/conf/ fs -put
```

**/home/admin/oceanbase/mrsstable\_lib\_6u/\***  
**/home/hadoop/hadoop-0.19.1/oceanbase/mrllib**

## 1.4 旁路导入配置

旁路导入配置包括配置并启动 ImportServer、配置并启动 ProxyServer 和配置 OceanBase。

### 1.4.1 配置并启动 ImportServer

ImportServer 与 RootServer 启动在同一台服务器中，ImportServer 的配置、启动过程如下：

1. 以 **admin** 用户登录 RootServer 服务器（10.10.10.2）。
2. 执行以下文件，拷贝配置文件。  
**cp /home/admin/oceanbase/etc/configuration.xml.template**  
**/home/admin/oceanbase/etc/configuration.xml**
3. 使用 **vi** 编辑器，修改“/home/admin/oceanbase/etc/configuration.xml”文件。

```
<configuration>
  <property>
    <name>mrsstable.native.lib.path</name>
    <value>hdfs://10.10.10.41:9000/home/hadoop/hadoop-
0.19.1/oceanabse/mrllib</value>
    <description>“native library”在 HDFS 上的路径。</description>
  </property>

  <property>
    <name>mrsstable.sstable.version</name>
    <value>2</value>
    <description>“native library”的 sstable 版本。</description>
  </property>

  <property>
    <name>mapred.reduce.tasks</name>
    <value>10</value>
    <description>Reduce 的任务数，同时也是生成的 sstable 的个数。建议业务人员和
    DBA 讨论决定该参数值。
    该参数值可以按照表数据总量除以 256M（sstable 大小）来估算，可以适当调大些。最小值为 2。
    如果有个别表数据量特别大或者特别小，请联系 dba 手动调整对应表的
    reduce 个数。</description>
  </property>

  <property>
    <name>mrsstable.presort.sampler</name>
    <value>random</value>
    <description>采样类型，根据数据特征配置，建议咨询 OceanBase 的 DBA。
  </property>
```

random, 适合于原始数据的每个文件都是随机的。大部分的业务数据都采用这个采样模式。

interval, 适合于每个数据文件都是有序的, 但全局是无序的。

split, 获取 mrsstable.max.sample.split 个输入文件的前

mrsstable.sample.number 个作为采样。</description>

</property>

<property>

<name>mrsstable.max.sample.split</name>

<value>2</value>

<description>最大的采样分裂数。

“split”和“random”采样时, 默认值为“reduceNum / 5”。

“interval”采样时, 默认值为“max(reduceNum \* 0.05, 1)”。</description>

</property>

<property>

<name>mrsstable.sample.number</name>

<value>10000</value>

<description>总采样数。

“split”和“random”采样时, 默认值为“reduceNum \* 10”建议值“10000-50000”。

“interval”采样时, 无须配置。</description>

</property>

<property>

<name>mrsstable.sample.choose.percent</name>

<value>0.001</value>

<description>采样选取百分比。“interval”采样时, it means the frequency with which records will be emitted.

“split”和“random”采样时, 无须配置。</description>

</property>

<property>

<name>mrsstable.input.format</name>

<value>text</value>

<description>导入数据的文件格式。</description>

</property>

<property>

<name>mrsstable.skip.invalid.row</name>

<value>0</value>

<description>1: 有错误的行的时候还能继续执行, 单行的错误需要查看mapreduce 的日志才能发现。

0: 如果有错误的行, 则导致整个旁路导入过程失败。</description>

</property>

.....

4. 执行以下文件, 拷贝配置文件。

```
cp /home/admin/oceanbase/etc/importserver.conf.template  
/home/admin/oceanbase/etc/importserver.conf
```

5. 使用 **vi** 编辑器，修改“/home/admin/oceanbase/etc/importserver.conf”文件。

```
[public]
# OceanBase 的 APP 名称。
app_name = obtest
# 导入状态监控的账号和密码，与“1.4.3 配置 OceanBase”中创建的相同。
ob_user = load_data_monitor
ob_password = import_password
# Listener 的 MySQL 协议端口。
ob_port = 2828
load_table_timeout = 3600

# 30s per retry, start_load_table_retry for each table
start_load_table_retry = 60

#OceanBase 安装目录。
base_dir = /home/admin/oceanbase/

# hadoop env
# default:
#   hadoop_bin_dir = $hadoop_home/bin
# 在 ImportServer 上，HadoopClient 的 Java Home 路径。
java_home = /usr/lib/java
# 在 ImportServer 上，HadoopClient 的安装路径。
hadoop_home = /home/admin/hadoop-0.19.1
# 在 ImportServer 上，HadoopClient 的配置文件目录。
hadoop_conf_dir = /home/admin/hadoop-0.19.1/conf
# Hadoop 上生成的数据保留天数。
keep_hadoop_data_days = 3

# hdfs path
# hdfs_name is the name of hdfs server
# hdfs_data_dir is the path on hdfs to store work data
# HDFS 的 URI，与在“/home/hadoopmaster/hadoop/conf/hadoop-site.xml”中配置的
相同。
hdfs_name = hdfs://10.10.10.41:9000
# 存放生成的 sstable 文件的目录。在“1.3.3 配置 Hadoop”中创建。
hdfs_data_dir = /home/hadoop/hadoop-0.19.1/oceanbase/output/
# 存放导入数据转换过程中生成的配置文件的目录。在“1.3.3 配置 Hadoop”中创建。
hdfs_conf_dir = /home/hadoop/hadoop-0.19.1/oceanbase/config/

# dispatch_time_range format is: start_time,hours
# 这个用于延迟加载的选项，前一个时间点是可以开始加载的时间点，后者是可
以加载点的持续时间。
# 一旦开始加载操作，就算过了这个时间点也不会自动停止。
#dispatch_time_range=01:00,8

[import_server]
# ImportServer 的服务端口。
port = 2900
log_level = debug
# default pid_file = $log_dir/import_server.pid
```



```
# default log_file = $log_dir/import_server.log
# keep_task_state_count = 128

## list the master rs of each cluster

[ob_instances]
## obi_count 为集群数和集群中 RootServer 的 VIP 和 Port。
obi_count=1
obi0_rs_vip=10.10.10.2
obi0_rs_port=2500
obi1_rs_vip=
obi1_rs_port=
```

6. 执行以下命令，启动 ImportServer。

```
python2.6 /home/admin/oceanbase/bin/importserver.py -f  
/home/admin/oceanbase/etc/importserver.conf
```

### 1.4.2 配置并启动 ProxyServer

ProxyServer 在每台 ChunkServer 上均需要启动，配置并启动 ProxyServer 过程如下：

1. 以 **admin** 用户登录 ProxyServer 服务器（10.10.10.4）。
2. 执行以下文件，拷贝配置文件。  
**cp /home/admin/oceanbase/etc/proxyserver.conf.template**  
**/home/admin/oceanbase/etc/proxyserver.conf**
3. 使用 **vi** 编辑器，修改“/home/admin/oceanbase/etc/proxyserver.conf”文件。

```
[public]
task_queue_size=10000
task_thread_count=8
io_thread_count=8
# second
network_timeout=30
# millisecond
task_left_time=300
# 在 ProxyServer 上，HadoopClient 的 java home。
java_home=/usr/lib/java
[hadoop]
# 在 ProxyServer 上，HadoopClient 的配置文件目录。
hdfs_conf_dir=/home/admin/hadoop-0.19.1/conf/
# 在 ProxyServer 上，HadoopClient 的安装路径。
hadoop_home=/home/admin/hadoop-0.19.1/
# mem use to cache hadoop meta (bytes)
mem_limit=10000000
# buffer use to read file from yunti in one work thread, so total memory is buf_size *
task_thread_count (bytes)
buf_size=64000000
```

4. 执行以下命令，启动 ProxyServer。

```
/home/admin/oceanbase/bin/proxyserver -f  
/home/admin/oceanbase/etc/proxyserver.conf -p2650
```

### 1.4.3 配置 OceanBase

执行旁路导入前，需要在 OceanBase 创建旁路导入相关的表，以及修改相应参数。

1. 执行以下命令，连接 OceanBase。  
**mysql -h 10.10.10.2 -P2828 -uadmin -padmin**
2. 执行以下 SQL 语句，进行 OceanBase 数据导入相关配置。

- a. 创建旁路导入需要的表。

```
CREATE TABLE data_source_proxy(  
cluster_id int,  
ip varchar(32),  
port int,  
supported_data_source_name varchar(256),  
PRIMARY KEY(cluster_id, ip, port));
```

```
CREATE TABLE load_data_history(  
start_time timestamp,  
end_time timestamp,  
table_name varchar(256),  
table_id varchar(32),  
status varchar(256),  
uri varchar(1024),  
PRIMARY KEY(table_name, table_id, start_time));
```

- b. 打开旁路导入的开关并设置旁路导入线程数。

```
ALTER SYSTEM SET enable_load_data='true'  
SERVER_TYPE=rootserver;  
ALTER SYSTEM SET bypass_sstable_loader_thread_num=5  
SERVER_TYPE=chunkserver;
```

- c. 创建导入状态监控的账号和密码。

```
CREATE USER 'load_data_monitor' IDENTIFIED BY  
'import_password';  
GRANT SELECT ON load_data_history TO  
'load_data_monitor';
```

- d. 插入 ProxyServer 的节点信息。多台 ProxyServer 时，根据以下命令，插入所有 ProxyServer 的信息。

*说明：“1”为集群 ID，“10.10.10.4”为 ProxyServer 的 IP，“2650”为 ProxyServer 的端口，“hadoop”为 ProxyServer 的标识，请勿修改。*

```
INSERT INTO data_source_proxy VALUES(1, '10.10.10.4', 2650, 'hadoop');
```

- e. 根据实际机器节点修改导入的相关的配置。“tablet\_replicas\_num”为副本数，必须小于等于 ChunkServer 服务器数，其余参数含义请参考《OceanBase 0.4.2 参考指南》中的“4 配置项参考”。

```
ALTER SYSTEM SET balance_max_concurrent_migrate_num
```

```

='1' SERVER_TYPE=rootserver;
ALTER SYSTEM SET balance_max_migrate_in_per_cs ='20'
SERVER_TYPE=rootserver;
ALTER SYSTEM SET balance_max_migrate_out_per_cs ='20'
SERVER_TYPE= rootserver;
ALTER SYSTEM SET load_data_max_timeout_per_range
='5m' SERVER_TYPE= rootserver;
ALTER SYSTEM SET tablet_replicas_num='1'
SERVER_TYPE=rootserver;
ALTER SYSTEM SET datasource_timeout ='30'
SERVER_TYPE=chunkserver;
ALTER SYSTEM SET max_migrate_task_count ='2'
SERVER_TYPE=chunkserver;
ALTER SYSTEM SET merge_thread_per_disk ='1'
SERVER_TYPE=chunkserver;
ALTER SYSTEM SET migrate_band_limit_per_second='30M'
SERVER_TYPE=chunkserver;

```

## 1.5 导入数据

在导入数据前，请确认表在 OceanBase 中已经存在。如果表不存在，则先在 OceanBase 中创建。

假设表 test，其信息如下所示：

```
mysql> desc test;
```

field	type	nullable	key	default	extra
c1	int		1	NULL	
c2	varchar(3)		0	NULL	
c3	timestamp		0	NULL	

3 rows in set (0.00 sec)

现需要向表 test 导入以下 5 组数据：

- (1, a, 2014-03-04 18:00:00)
- (2, b, 2014-03-04)
- (3, c, 20140304 18:00:00)
- (4, d, 20140304180000)
- (5, e, 20140304)

1. 以 **admin** 用户登录任意一台 HadoopClient 服务器（10.10.10.2）。
2. 执行以下命令，新建旁路导入表的配置。“importcli.py”操作和参数说明请参见“6 其他操作”。

```
python2.6 /home/admin/oceanbase/bin/importcli.py -s 10.10.10.2 -
p 2900 -t CREATE_BYPASS_CONFIG test delim=44
raw_data_field_count=3 column_infos=c1-0,c2-1,c3-2
```

3. 用 **vi** 编辑器，创建导入文件“/home/admin/test.text”。

```
1,a,2014-03-04 18:00:00
2,b,2014-03-04
3,c,20140304 18:00:00
4,d,20140304180000
5,e,20140304
```

4. 执行以下命令，将“/home/admin/test.text”传入 Hadoop 的“input”文件夹中。

```
/home/admin/hadoop-0.19.1/bin/hadoop --config  
/home/admin/hadoop-0.19.1/conf/ fs -put /home/admin/test.text  
/home/hadoop/hadoop-0.19.1/oceanbase/input
```

5. 执行以下命令，将“test.text”中的数据导入 OceanBase 的 test 表中。

“importcli.py”操作和参数说明请参见“6 其他操作”。

```
python2.6 /home/admin/oceanbase/bin/importcli.py -s 10.10.10.2 -  
p 2900 -t OVERWRITE /home/hadoop/hadoop-  
0.19.1/oceanbase/input/test.text test
```

导入完成后，可连接 OceanBase 查看导入的数据，验证导入是否成功。

1. 执行以下命令，连接 OceanBase。

```
mysql -h 10.10.10.2 -P2828 -uadmin -padmin
```

2. 执行以下 SQL，查询插入的数据。

```
SELECT * FROM test;
```

## 1.6 其他操作

“importcli.py”的操作包括“新建旁路导入表的配置”、“打印旁路导入表的配置”、“删除旁路导入表的配置”、“发起旁路导入”、“查看某状态下的任务”和“结束 OVERWRITE 的任务”等，详细说明如下：

- 新建旁路导入表的配置，参数说明如表 1-4 所示。  

```
python2.6 ./importcli.py -s import_server_ip -p import_server_port -t  
CREATE_BYPASS_CONFIG table_name [delim=1]  
[raw_data_field_count=N] [column_infos=columnA-0,columnB-  
1,columnC-2,...]
```

表 1-4 参数说明

参数	说明
<i>import_server_ip</i>	ImportServer 的 IP。
<i>import_server_port</i>	ImportServer 的端口。
CREATE_BYPASS_CONFIG	新建旁路导入表的配置。
delim	导入数据的分隔符，用 ASCII 码表示。

参数	说明
raw_data_field_count	用于检测导入的行的完整性，默认配置为 OceanBase 中表的列数。
column_infos	<p>OceanBase 中表的列名和导入文件中字段的对应关系，格式为“列名-字段号”，每个对应关系用逗号隔开。</p> <ul style="list-style-type: none"> <li>• 字段号从 0 开始标号。例如：“columnA-0”表示导入数据的第 0 个字段对应 OceanBase 中表的 columnA 列。</li> <li>• “columnA--”表示该列不导入，直接填充 NULL。</li> <li>• 如果不配置，则自动从 0 开始按照列的顺序自动填充。</li> <li>• OceanBase 中“createtime”和“modifytime”字段固定为前两个字段。如果有这两种类型的字段，建议建表的时候写在最前面，不然旁路导入的时候“column_infos”必须手动指定。</li> </ul>

- 打印旁路导入表的配置，参数说明如[表 1-5](#)所示。  

```
python2.6 ./importcli.py -s import_server_ip -p import_server_port -t PRINT_BYPASS_CONFIG table_name
```

表 1-5 参数说明

参数	说明
import_server_ip	ImportServer 的 IP。
import_server_port	ImportServer 的端口。
PRINT_BYPASS_CONFIG	打印旁路导入表的配置。
table_name	需要打印旁路导入配置的表。

- 删除旁路导入表的配置，参数说明如[表 1-6](#)所示。  

```
python2.6 ./importcli.py -s import_server_ip -p import_server_port -t DELETE_BYPASS_CONFIG table_name
```

表 1-6 参数说明

参数	说明
<i>import_server_ip</i>	ImportServer 的 IP。
<i>import_server_port</i>	ImportServer 的端口。
DELETE_BYPASS_CONFIG	删除旁路导入表的配置。
<i>table_name</i>	需要删除旁路导入配置的表。

- 发起旁路导入，参数说明如表 1-7 所示。  

```
python2.6 ./importcli.py -s import_server_ip -p import_server_port -t
OVERWRITE /path/on/hadoop/filename01 table_name01
/path/on/hadoop/filename02 table_name02 ...
```

表 1-7 参数说明

参数	说明
<i>import_server_ip</i>	ImportServer 的 IP。
<i>import_server_port</i>	ImportServer 的端口。
OVERWRITE	发起旁路导入。同一个命令中，Hadoop 的任务全部跑完以后才会发命令给 OceanBase，让 OceanBase 开始拉数据。 建议一每个命令导入一个 table。
<i>table_name01</i>	“ <i>table_name01</i> ”表示需要导入数据的表。
<i>/path/on/hadoop/filename01</i>	“ <i>/path/on/hadoop/filename01</i> ”表示 <i>table_name01</i> 的导入文件在 Hadoop 上的路径。

- 查看某状态下的任务，参数说明如表 1-8 所示。  

```
python2.6 bin/importcli.py -s import_server_ip -p import_server_port -t
STATE task_state
```

表 1-8 参数说明

参数	说明
<i>import_server_ip</i>	ImportServer 的 IP。
<i>import_server_port</i>	ImportServer 的端口。
STATE	<p>查看对应状态的任务。</p> <p>任务状态数最大为 32 个。如果超过了这个限制，那么每次新增任务的时候就会清除 DONE 和 KILLED 状态的 task 的命令直到总 task 数目降到 32 个。</p>
<i>task_state</i>	<p>任务的状态，包括以下几种：</p> <ul style="list-style-type: none"> <li>• ALL：有状态的任务。</li> <li>• DOING：正在导入的任务。</li> <li>• DONE：已经成功完成的任务。</li> <li>• KILLED：被杀死的任务。</li> <li>• NEED_KILL：收到杀死命令但是还没退出的任务。</li> <li>• <i>task_id</i>：也可直接输入某个任务的 ID，进行查看。</li> </ul> <p><b>说明：</b>任务的 ID 可以在“importcli.py”开始运行任务的第一行输出中找到，也可以通过 STATE 命令查看 DOING 状态的任务来查询。</p>

- 结束 OVERWRITE 的任务，参数说明如表 1-9 所示。  
python2.6 bin/importcli.py -s *import\_server\_ip* -p *import\_server\_port* -t KILL *task\_id*

表 1-9 参数说明

参数	说明
<i>import_server_ip</i>	ImportServer 的 IP。
<i>import_server_port</i>	ImportServer 的端口。

参数	说明
KILL	<p>当旁路导入错过在“/home/admin/oceanbase/etc/importserver.conf”文件中设置的时间窗口“dispatch_time_range”时，可能需要把该次旁路导入终止掉，这时候就需要 KILL 命令。</p> <p>这个操作只是把任务状态改成“NEED_KILL”，需要等待任务检测到自己状态被设置成 NEED_KILL 才会退出来，因此该操作有一定的延时和局限性。旁路导入中 <b>mapreduce</b> 过程不会被这个操作停止，拷贝和分发 <b>sstable</b> 的过程可以被终止。如果一个 <b>task</b> 在 <b>mapreduce</b> 阶段接收到 KILL 命令，只有在 <b>mapreduce</b> 结束进入拷贝和分发 <b>sstable</b> 的时候才会退出。如果想要立刻退出，可以用 <b>Hadoop</b> 命令直接发 <b>kill</b> 命令给对应的 <b>mapreduce</b> 任务。</p>
<i>task_id</i>	运行的任务 ID。



## 2 DataX 导入/导出数据

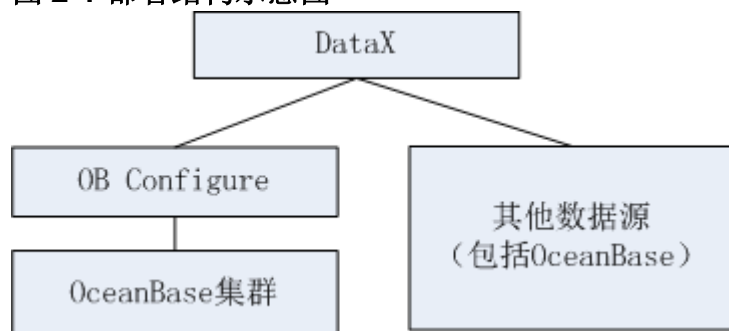
DataX 是阿里巴巴集团内被广泛使用的离线数据同步工具/平台，实现包括 OceanBase、MySQL、Oracle、HDFS、Hive、HBase、OTS、ODPS 等各种异构数据源之间高效的数据同步功能。

本文档仅描述 DataX 用于 OceanBase 的导入/导出相关的信息，如果您需要了解其他 DataX 的相关知识，请参见[“http://gitlab.alibaba-inc.com/atcloud/datax/wikis/home”](http://gitlab.alibaba-inc.com/atcloud/datax/wikis/home)。

### 2.1 基本信息

使用 DataX 对 OceanBase 数据进行导入/导出是指从一个数据源的数据导入到 OceanBase 中或者从 OceanBase 中导出数据到另一个数据源。其部署结构示意图如图 2-1 所示。

图 2-1 部署结构示意图



使用 DataX 导入/导出数据的结构介绍如下：

- **DataX**  
阿里巴巴集团内被广泛使用的离线数据同步工具。
- **OB Configure**  
OceanBase 配置中心，在使用 DataX 导入/导出数据时，提供 OceanBase 的 URL。
- **OceanBase 集群**  
阿里巴巴集团自主研发的海量数据库系统。
- **其他数据源**  
包括 OceanBase、MySQL、Oracle、HDFS、Hive、HBase、OTS、ODPS 等。OceanBase 数据可以通过 DataX 导入上述数据源中，上述数据源的数据也可以通过 DataX 导入 OceanBase。

使用 DataX 导入/导出数据的注意事项如下：

- 由于 OceanBase 在进行数据导入/导出过程中，使用 SQL 方式一次性将数据装载到 UpdateServer 的内存，因此，导入/导出的数据量不能够太大。
- 由于数据导入/导出会采用 SQL 接口往 OceanBase 写数据，故会占用 UpdateServer 的内存资源，因此，建议在业务请求量较低时进行操作（例如：凌晨），否则会影响正常的业务请求。
- 对所有的日期时间格式，统一转化为“yyyy-MM-dd HH:mm:ss”格式，并截断毫秒，只保留到秒级别。

## 2.2 部署 DataX

安装 DataX 主要包括安装前检查、创建安装用户和安装 DataX 软件，假设 DataX 的安装服务器的 IP 为“10.10.10.51”。

### 2.2.1 安装前检查

在安装 DataX 前需要检查安装的服务器是否满足以下内容：

- 处理器为“X86\_64”。
- Python 版本 $\geq$ 2.4。
- Java 版本 $\geq$ 1.6。

安装 DataX 前需要检查的操作步骤如下：

1. 以 **root** 用户登录 DataX 服务器（10.10.10.51）。
2. 执行以下命令，查看处理器型号。

**file /bin/ls**

```
[root@OceanBase051 /root]$ # file /bin/ls
/bin/ls: ELF 64-bit LSB executable, x86-64, version 1 (SYSV), dynamically linked (uses
shared libs), for GNU/Linux 2.6.18, stripped
```

3. 执行以下命令，查看 Python 版本。

**python --version**

```
[root@OceanBase051 /root]$ python --version
Python 2.6.6
```

4. 执行以下命令，查看 Java 版本。

**java -version**

```
[root@OceanBase051 /root]$ java -version
java version "1.6.0_22"
OpenJDK Runtime Environment (IcedTea6 1.10.4) (rhel-1.41.1.10.4.el6-x86_64)
OpenJDK 64-Bit Server VM (build 20.0-b11, mixed mode)
```

### 2.2.2 安装 JDK 补丁包

在 OceanBase 导入/导出时，需要安装 JDK 补丁包，否则会报“java.security.NoSuchAlgorithmException: Cannot find any provider supporting Blowfish”的错误。

1. 根据“2.1 安装前检查”中查看的 Java 版本，下载相应的 JDK 版本补丁。
  - JDK6 补丁：  
<http://www.oracle.com/technetwork/java/javase/downloads/jce-6-download-429243.html>
  - JDK7 补丁：  
<http://www.oracle.com/technetwork/java/javase/downloads/jce-7-download-432124.html>
2. 将 JDK 补丁包解压到目录“\$JAVA\_HOME/jre/lib/security/”下。其中“\$JAVA\_HOME”为 Java 的安装目录。

### 2.2.3 创建安装用户

DataX 的安装目录必须在“/home/taobao”下，因此需要创建 DataX 的安装用户/密码为“taobao/Abc@123”。

1. 以 **root** 用户登录 DataX 服务器（10.10.10.51）。
2. 执行如下命令，创建 HadoopMaster 的安装用户。  
**useradd -d /home/taobao -s /bin/bash -m taobao**
3. 执行如下命令，为用户“taobao”设置密码。  
**passwd taobao**
4. 您需要根据系统的提示输入两次密码“Abc@123”。
5. 为“taobao”赋予“sudo”权限。
  - a. 执行以下命令，添加“/etc/sudoers”文件的写权限。  
**chmod u+w /etc/sudoers**
  - b. 使用 **vi** 编辑器，在“/etc/sudoers”文件中“root ALL= (ALL) ALL”后添加语句，如黑体部分所示。

```
root ALL= (ALL) ALL
taobao ALL=(ALL) ALL
```
  - c. 执行以下命令，删除“/etc/sudoers”文件的写权限。  
**chmod u-w /etc/sudoers**

### 2.2.4 安装 DataX 软件

本小节主要介绍 DataX 的下载、安装和验证是否正确安装的方法。

下载、安装、验证 DataX 的操作步骤如下：

1. 以 **taobao** 用户登录 DataX 服务器（10.10.10.51）。
2. 用 **vi** 编辑器，在环境变量文件“/home/taobao/.bashrc”中，增加 JAVA HOME 的路径。请根据实际情况填写。

```
export JAVA_HOME=/usr/lib/java
```

3. 执行以下命令，使环境变量生效。  
**source /home/taobao/.bashrc**
4. 执行以下命令，下载 DataX 的 RPM 包。  
*说明：暂时还未提供下载链接，如有需要，请直接联系我们获取。*
5. 执行以下命令，安装 DataX。  
**sudo rpm -ivh t\_dp\_dw\_datax\_core\_all-201403051409-736.noarch.rpm**
6. 安装完成后，执行以下命令，如果系统最后显示信息为“completed successfully”，则说明 DataX 安装成功。  
**/home/taobao/datax/bin/datax.py**  
**/home/taobao/datax/jobs/example.xml**

## 2.3 OceanBase 导入/导出

主要介绍使用 DataX 对 OceanBase 数据进行导入和导出的详细过程。

### \* 任务说明

为便于介绍 OceanBase 导入/导出的配置文件，因此，我们选取将一个 OceanBase（起始数据源）的数据导入到另一个 OceanBase（目的数据源）为例。

假设起始数据源中有表 test\_01 信息如下：

```
mysql> desc test_01;
+-----+-----+-----+-----+-----+-----+
| field | type      | nullable | key | default | extra |
+-----+-----+-----+-----+-----+-----+
| c1    | int       |          | 1   | NULL    |       |
| c2    | varchar(3) |          | 0   | NULL    |       |
| c3    | timestamp |          | 0   | NULL    |       |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> select * from test_01;
+-----+-----+-----+
| c1 | c2 | c3                |
+-----+-----+-----+
| 1 | a  | 2014-03-04 18:00:00 |
| 2 | b  | 2014-03-04 00:00:00.123456 |
+-----+-----+-----+
2 rows in set (0.01 sec)
```

### \* 前提条件

使用 DataX 对 OceanBase 导入/导出，需要依赖 OB Configure。因此在进行导入/导出前，请先部署 OB Configure，详细请参见《OceanBase 0.4.2 配置中心

安装指南》。如果您已经部署 OB Configure，请联系 DBA 获取您的作为起始数据源和目的数据源的 URL。

假设现在已获取起始数据源和目的数据源的 URL 如下：

- 起始数据源  
`http://10.10.10.31:8080/ob-config/config.co?dataId=OceanBase01`
- 目的数据源  
`http://10.10.10.31:8080/ob-config/config.co?dataId=OceanBase02`

#### \* 操作步骤

使用 DataX 对 OceanBase 导入/导出的操作步骤如下：

1. 在目的数据源中，执行以下 SQL 语句，创建与需要导入表（test\_01）相同格式的表（test\_02）。  
**CREATE TABLE test\_02(  
c1 int PRIMARY KEY,  
c2 varchar(3),  
c3 timestamp);**
2. 以 **taobao** 用户登录 DataX 服务器（10.10.10.51）。
3. 生成用于导入导出作业的 XML 文件。
  - a. 执行以下命令，进入生产 XML 文件模式。  
**/home/taobao/datax/bin/datax.py -g**
  - b. 根据提示，输入您需要作为起始数据源的具体数字“10”，并按 Enter 键。

```
DataX V 2.0, Build 20140305. From Taobao
Copyright (C) 2010-2014, Alibaba Group. All Rights Reserved.

Data Source :
0   hdfs
1   hbase-0.94
2   hbaselog
3   http
4   stream
5   hive
6   odps
7   sqlserver
8   oracle
9   mysql
10  oceanbasejdbc
11  hbase
Please choose [0-11]: 10
```

- c. 根据提示，输入您需要作为目的数据源具体的数字“0”，并按 Enter 键。

```

Data Destination :
0   oceanbase-0.4
1   stream
2   hbasebulk-cdh4
3   mysql
4   hdfs-2.0
5   oraclejdbc
6   hbase-0.94
7   tairbulk
8   mysqljdbc
9   hbasebulk-cdh3
10  hdfs
11  hbase
12  otsbulk
13  tair
14  oracle
15  ots
16  oceanbase
17  odps
18  hive
Please choose [0-18]: 0
Generate /home/taobao/datax/jobs/oceanbasejdbcreader_to_oceanbasewriter-
0.4_1394703186554.xml

```

4. 使用 **vi** 编辑器，修改“/home/taobao/datax/jobs/oceanbasejdbcreader\_to\_oceanbasewriter-0.4\_1394703186554.xml”文件。参数说明如表 2-1 所示。

**说明：**如果您的起始数据源或者目的数据源为其他类型，则其配置说明请参见[“http://gitlab.taobao.ali.com/atcloud/datax/wikis/home”](http://gitlab.taobao.ali.com/atcloud/datax/wikis/home)。

```

<?xml version="1.0" encoding="UTF-8"?>

<jobs>
  <job id="oceanbasejdbcreader_to_oceanbasewriter-0.4_job">
    <reader>
      <plugin>oceanbasejdbcreader</plugin>
      <param key="config-url" value="http://10.10.10.31:8080/ob-config/config.co?dataId=OceanBase01"/>
      <param key="table" value="test_01"/>
      <param key="column" value="c1,c2,c3"/>
      <!--
      <param key="where" value="?" />
      <param key="sql" value="?" />
      -->
      <param key="concurrency" value="1"/>
    </reader>
    <writer>
      <plugin>oceanbasewriter-0.4</plugin>
      <param key="config-url" value="http://10.10.10.31:8080/ob-config/config.co?dataId=OceanBase02"/>
      <param key="table" value="test_02"/>
      <param key="column" value="c1,c2,c3"/>
      <param key="error-limit" value="1"/>
      <param key="batch-size" value="512"/>
    </writer>
  </job>
</jobs>

```

```

    <param key="concurrency" value="1"/>
  </writer>
</job>
</jobs>

```

表 2-1 参数说明

数据源	参数	说明
起始数据源	config-url	必填项。表示 OceanBase 连接串，请联系 OceanBase DBA 获取。
	table	必填项。表示需要导出的表的表名。
	column	必填项。表示需要导出的表中的列名集合。以英文状态的逗号进行分隔。
	where	非必填项。表示需要导出的表中 WHERE 过滤条件。 建议不要填写。
	sql	非必填项。当 WHERE 过滤条件不足以描述所筛选的条件时，用户可以通过该配置型来自定义筛选 SQL。当用户配置了这一项之后，DataX 系统就会忽略“table”、“column”和“where”这些配置型，直接使用这个配置项的内容对数据进行筛选。  例如：需要进行多表 join 后同步数据，可使用“select a,b from table_a join table_b on table_a.id = table_b.id;”。
	concurrency	非必填项。表示导出并发量。
目的数据源	config-url	必填项。表示 OceanBase 连接串，请联系 OceanBase DB 获取。
	table	必填项。表示需要导入的表的表名。
	column	必填项。表示需要导入的表中的列名集合。以英文状态的逗号进行分隔。

数据源	参数	说明
	error-limit	<p>非必填项。表示导入过程中允许数据错误的行数。</p> <p>0: 就立即终止导入。</p> <p>1: 表示容错 1 行，即 1 行错误还能继续执行。</p> <p>2: 表示容错 2 行，即 2 行以内的错误还能继续执行。</p> <p>依次类推。</p>
	batch-size	<p>非必填项。表示 writer 的 buffer 大小，每个线程累计行数到达这个值的时候批量插入。</p>
	concurrency	<p>非必填项。表示导入并发量。</p>

5. 执行以下命令，进行 DataX 导入/导出。  
**`/home/taobao/datax/bin/datax.py`**  
**`/home/taobao/datax/jobs/oceanbasejdbcreader_to_oceanbasewriter-0.4_1394703186554.xml`**
6. 导入完成后，连接目的数据源，查看 test\_02 是否与起始数据源中 test\_01 的数据相同。如果相同，则说明导入成功。



# 3 主备集群切换

主要介绍 OceanBase 主备集群手动切换主备的操作方法。

## \* 基本信息

假设现有主备集群信息如[表 3-1](#)所示。

表 3-1 集群信息

集群	各 Server 安装目录	集群 ID	RootServer IP	RootServer Port
主集群	/home/admin/oceanbase	1	10.10.10.2	2500
备集群	/home/admin/oceanbase	2	10.10.10.12	2500

说明：如果 RootServer 为主备时，此处的 IP 为 VIP。

## \* 前提条件

在进行主备切换前，请确认以下内容：

- OceanBase 主备集群已经正确安装，并正常运行。
- 已完成数据备份。
- 切换时间为业务低峰期。

## \* 预期结果

执行主备集群切换后，“10.10.10.2”所在的集群变为备集群，“10.10.10.12”所在的集群变为主集群。

## \* 操作步骤

OceanBase 主备集群切换操作步骤如下：

1. 将主集群切换为备集群。
  - a. 在主集群 RootServer 服务器中执行以下命令，查看是否为主集群。  
**/home/admin/oceanbase/bin/rs\_admin -r 10.10.10.2 -p 2500  
get\_obi\_role**

```
[admin@OceanBase002 ~]$ /home/admin/oceanbase/bin/rs_admin -r 10.10.10.2
-p 2500 get_obi_role
server[10.10.10.2:2500], timeout=10000000
get_obi_role...
obi_role=MASTER
```

- b. 执行以下命令，将原主集群切换为备集群。

**/home/admin/oceanbase/bin/rs\_admin -r 10.10.10.2 -p 2500  
set\_obi\_role -o OBI\_SLAVE**

```
[admin@OceanBase002 ~]$ /home/admin/oceanbase/bin/rs_admin -r 10.10.10.2
-p 2500 set_obi_role -o OBI_SLAVE
server[10.10.10.2:2500], timeout=10000000
set_obi_role...role=1
Okay
```

- c. 执行以下命令，查看是否切换成功。

**/home/admin/oceanbase/bin/rs\_admin -r 10.10.10.2 -p 2500  
get\_obi\_role**

```
[admin@OceanBase002 ~]$ /home/admin/oceanbase/bin/rs_admin -r 10.10.10.2
-p 2500 get_obi_role
server[10.10.10.2:2500], timeout=10000000
get_obi_role...
obi_role=SLAVE
```

2. 将备集群切换为主集群。

- a. 在备集群 RootServer 服务器中执行以下命令，查看是否为备集群。

**/home/admin/oceanbase/bin/rs\_admin -r 10.10.10.12 -p 2500  
get\_obi\_role**

```
[admin@OceanBase012 ~]$ /home/admin/oceanbase/bin/rs_admin -r
10.10.10.12 -p 2500 get_obi_role
server[10.10.10.12:2500], timeout=10000000
get_obi_role...
obi_role=SLAVE
```

- b. 执行以下命令，将原备集群切换为主集群。

**/home/admin/oceanbase/bin/rs\_admin -r 10.10.10.12 -p 2500  
set\_obi\_role -o OBI\_MASTER**

```
[admin@OceanBase012 ~]$ /home/admin/oceanbase/bin/rs_admin -r
10.10.10.12 -p 2500 set_obi_role -o OBI_MASTER
server[10.10.10.12:2500], timeout=10000000
set_obi_role...role=1
Okay
```

- c. 执行以下命令，查看是否切换成功。

**/home/admin/oceanbase/bin/rs\_admin -r 10.10.10.12 -p 2500  
get\_obi\_role**

```
[admin@OceanBase012 ~]$ /home/admin/oceanbase/bin/rs_admin -r
10.10.10.12 -p 2500 get_obi_role
server[10.10.10.12:2500], timeout=10000000
get_obi_role...
obi_role=MASTER
```

3. 依次执行以下命令，修改集群中记录的主集群 RootServer 的 IP 和 Port。

```
/home/admin/oceanbase/bin/rs_admin -r 10.10.10.2 -p 2500  
set_config -o  
master_root_server_ip=10.10.10.12,master_root_server_port=2500  
/home/admin/oceanbase/bin/rs_admin -r 10.10.10.12 -p 2500  
set_config -o  
master_root_server_ip=10.10.10.12,master_root_server_port=2500
```

4. 连接新主集群中的任意一台 MergeServer，并修改内部表相关信息。

- a. 执行以下 SQL 语句，设置主集群信息。

```
ALTER SYSTEM SET master_root_server_ip='10.10.10.12'  
server_type=rootserver, master_root_server_port=2500  
server_type=rootserver;
```

- b. 执行以下 SQL 语句，查看主集群信息。

```
SHOW PARAMETERS LIKE 'master_root_server%';
```

name	server_ip	server_port	server_type	cluster_id	value
info					
master_root_server_ip	10.10.10.2	2500	rootserver	1	10.10.10.12   master OceanBase instance vip address
master_root_server_port	10.10.10.2	2500	rootserver	2	2500   master OceanBase instance listen port
master_root_server_ip	10.10.10.12	2500	rootserver	1	10.10.10.12   master OceanBase instance vip address
master_root_server_port	10.10.10.12	2500	rootserver	2	2500   master OceanBase instance listen port

- c. 执行以下 SQL 语句，查看集群 ID 和集群角色。“cluster\_role”的值为“1”表示主集群，“2”表示备集群。

```
SELECT cluster_vip, cluster_id, cluster_role FROM __all_cluster;
```

cluster_vip	cluster_id	cluster_role
10.10.10.2	1	2
10.10.10.12	2	1

如果查询结果与预期不同，请执行以下语句，修改集群角色。

```
update __all_cluster set cluster_role=1 where cluster_id=2;  
update __all_cluster set cluster_role=2 where cluster_id=1;
```

# 4 ChunkServer 更换磁盘

当 ChunkServer 中挂载的磁盘出现故障而需要更换磁盘或者需要在 ChunkServer 中增加磁盘时，需要进行 ChunkServer 更换磁盘操作。

假设 IP 为“10.10.10.4”，服务端口为“2600”的 ChunkServer 服务器上，“/data”下挂载了 8 块磁盘，其中挂载的磁盘“/data/4”需要更换。其操作步骤如下：

1. 登录需要换盘的 ChunkServer 服务器。

2. 执行以下命令，将需要更换的磁盘进行手动下盘。

```
~/oceanbase/bin/cs_admin -s 10.10.10.4 -p 2600 -i "uninstall_disk 4"
```

*说明：*指令将坏盘上的所有的 Tablet 从 RootTable 和 TabletImage 上删除，并在内存中将该盘标示为“ERROR”，同时删除软连接，使该磁盘不可以被读写。

```
[admin@OceanBase004 ~]$ ~/oceanbase/bin/cs_admin -s 10.10.10.4 -p 2600 -i
"uninstall_disk 4"
...
uninstall_disk:4 Okay!
...
```

3. 执行以下命令，查看磁盘信息。

```
~/oceanbase/bin/cs_admin -s 10.10.10.4 -p 2600 -i "show_disk"
```

```
[admin@OceanBase004 ~]$ ~/oceanbase/bin/cs_admin -s 10.10.10.4 -p 2600 -i
"show_disk"
...
disk:01 status:OK
disk:02 status:OK
disk:03 status:OK
disk:04 status:error
disk:05 status:OK
disk:06 status:OK
disk:07 status:OK
disk:08 status:OK
...
```

4. 修复磁盘后，将新磁盘挂载到“/data/4”。

*注意：*挂载的磁盘“/data/4”中不能有 SSTable 目录，否则会导致上盘失败。

5. 执行以下命令，进行手动上盘。执行成功后，“/disk/4”将被软连接到“~/oceanbase/data/4”上。

```
~/oceanbase/bin/cs_admin -s 10.10.10.4 -p 2600 -i "install_disk /data/4"
```

```
[admin@OceanBase004 ~]$ ~/oceanbase/bin/cs_admin -s 10.10.10.4 -p 2600 -i
"install_disk /data/4"
...
install_disk:/data/4 mount to disk_no:4 Okay!
...
```

6. 执行以下命令，查看磁盘信息。

**`~/oceanbase/bin/cs_admin -s 10.10.10.4 -p 2600 -i "show_disk"`**

```
[admin@OceanBase004 ~]$ ~/oceanbase/bin/cs_admin -s 10.10.10.4 -p 2600 -i  
"show_disk"
```

```
...
```

```
disk:01 status:OK
```

```
disk:02 status:OK
```

```
disk:03 status:OK
```

```
disk:04 status:OK
```

```
disk:05 status:OK
```

```
disk:06 status:OK
```

```
disk:07 status:OK
```

```
disk:08 status:OK
```

```
...
```

# 5 rs\_admin 使用

---

rs\_admin 是 RootServer 的辅助运维工具，提供各种查询和设置的功能。

*小窍门：* 您可以执行“`~/oceanbase/bin/rs_admin -help`”命令查询 rs\_admin 的命令和参数。

## \* boot\_recover

格式：

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port boot_strap
```

含义：

当 RootServer 端的“first\_tablet\_meta”丢失时，需要通过该命令来恢复。

## \* boot\_strap

格式：

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port boot_strap
```

含义：

集群首次启动时初始化操作。包括新建内部表，初始化 Schema/权限信息等。

## \* change\_log\_level

格式：

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port  
change_log_level -o DEBUG
```

含义：

调整日志输出级别，设置成 **DEBUG** 可以查看更多运行细节，一般用于查问题是使用。默认级别为 **INFO**。

举例：

```
[admin@OceanBase002 ~]$ ~/oceanbase/bin/rs_admin -r 10.10.10.2 -p 2500 change_log_level -  
o DEBUG  
server[10.10.10.2:2500], timeout=20000000  
do_change_log_level, level=5...  
Okay
```

## \* change\_table\_id

格式：

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port  
change_table_id -o table_id=table_id
```

含义：

旁路导入时使用。修改 table 的 table\_id。新的 table\_id 也是有系统自动生成的。比如修改 3001 号表的 table\_id，新的 table\_id 则为 max\_table\_id。

举例：

```
[admin@OceanBase002 ~]$ ~/oceanbase/bin/rs_admin -r 10.10.10.2 -p 2500 change_table_id -o  
table_id=3001  
server[10.10.10.2:2500], timeout=20000000  
change_table_id, table_id=3001...  
Okay
```

### \* **change\_ups\_master**

格式：

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port  
change_ups_master -o ups_ip=UpdateServer_ip -o  
ups_port=UpdateServer_port
```

含义：

修改集群内的主 UpdateServer 为指定的 UpdateServer。当指定的 UPS 状态为 notsync 时，需要加上 force 参数。

举例：

```
[admin@OceanBase002 ~]$ ~/oceanbase/bin/rs_admin -r 10.10.10.2 -p 2500  
change_ups_master -o ups_ip=10.10.10.2 -o ups_port=2700  
server[10.10.10.2:2500], timeout=20000000  
change_ups_master, ups_ip=10.10.10.2 ups_port=2700 force=0...  
Okay
```

### \* **check\_root\_table**

格式：

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port  
check_root_table -o cs_ip=ChunkServer_ip -o cs_port=ChunkServer_port
```

含义：

当需要下线 ChunkServer 时，可以通过该命令来检查，如果直接下线该 ChunkServer 是否会造成数据丢失。

举例：

```
[admin@OceanBase002 ~]$ ~/oceanbase/bin/rs_admin -r 10.10.10.2 -p 2500 check_root_table -  
o cs_ip=10.10.10.4 -o cs_port=2600  
server[10.10.10.2:2500], timeout=20000000  
check root_table integrity...expect chunkserver addr:cs_addr=10.10.10.4:2600...  
Okay check result: roottable is integrity, expect cs=10.10.10.4:2600
```

### \* **check\_schema**

格式:

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port  
check_schema
```

含义:

用户在用 sql 语句更新 schema 以后, 可能会存在 schema 信息错误, 比如修改压缩算法时, 压缩算法的名字写错了, 写成 lze, 这种错误在执行 sql 的时候是不会报错的, 只有在每日合并的时候才会报错。所以用户需要主动调用 check\_schema 的函数, 来检查系统的 schema 是否合理。

举例:

```
[admin@OceanBase002 ~]$ ~/oceanbase/bin/rs_admin -r 10.10.10.2 -p 2500 check_schema  
server[10.10.10.2:2500], timeout=20000000  
do_rs_admin, cmd=21...  
Okay
```

### \* **check\_tablet\_version**

格式:

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port  
check_tablet_version -o tablet_version=tablet_version
```

含义:

检查所有的在线 Tablet 是否都已经合并到该版本。

举例:

```
[admin@OceanBase002 ~]$ ~/oceanbase/bin/rs_admin -r 10.10.10.2 -p 2500  
check_tablet_version -o tablet_version=2  
server[10.10.10.2:2500], timeout=20000000  
Okay, check_version=2, already have some tablet not reach this version
```

### \* **clean\_error\_msg**

格式:

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port  
clean_error_msg
```

含义:

RootServer 在每日合并过程中如果出现错误时, RootServer 会定期会报警, 提醒 DBA 来处理, 当问题处理完以后, 需要将这个报警任务给删除掉。

### \* **clean\_root\_table**

格式:



```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port  
clean_root_table
```

含义：

重新生成一份新的 **roottable**，在以下情况中使用：

- **RootTable** 中存在错误的 **range** 或者 **index**。
- **RootTable** 中 **Tablet** 的数目太多，需要清除掉失效的 **Tablet**。

### **\* cs\_create\_table**

格式：

```
~/oceanbase/bin/rs_admin -r ChunkServer_ip -p ChunkServer_port  
cs_create_table -o table_id=table_id -o table_version=table_version
```

含义：

命令发给 **ChunkServer**，强制在 **ChunkServer** 上创建空 **tablet**。一般只有 **create table** 失败的情况下，才会使用该手段，作为一种运维补救方法。

举例：

```
[admin@OceanBase002 ~]$ ~/oceanbase/bin/rs_admin -r 10.10.10.4 -p 2600 cs_create_table -o  
table_id=501 -o table_version=1  
server[10.10.10.4:2600], timeout=20000000  
force cs create empty table cs_ip=10.10.10.4 cs_port=2600 version=1 table_id=501...  
[SUCC] create empty table:cs[10.10.10.4:2600], table[501], version[1], ret[0]
```

### **\* do\_check\_point**

格式：

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port  
do_check_point
```

含义：

执行“**check\_point**”，防止 **RootServer** 重启时，需要回放过多的日志。

### **\* dump\_cs\_tablet\_info**

格式：

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port  
dump_cs_tablet_info -o cs_ip=ChunkServer_ip -o cs_port=ChunkServer_port
```

含义：

统计 **ChunkServer** 上 **Tablet** 的个数。

举例：

```
[admin@OceanBase002 ~]$ ~/oceanbase/bin/rs_admin -r 10.10.10.2 -p 2500  
dump_cs_tablet_info -o cs_ip=10.10.10.4 -o cs_port=2600  
server[10.10.10.2:2500], timeout=20000000
```

```
dump chunkserver tablet info, cs_ip=10.10.10.4, cs_port=2600...
Okay
tablet_num=1
```

### **\* dump\_migrate\_info**

格式:

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port
dump_migrate_info
```

含义:

统计 ChunkServer 的迁移信息。需要在“~/oceanbase/bin/rs\_admin.log”中查看。

举例:

```
[admin@OceanBase002 ~]$ ~/oceanbase/bin/rs_admin -r 10.10.10.2 -p 2500
dump_migrate_info
server[10.10.10.2:2500], timeout=20000000
do_rs_admin, cmd=9...
Okay
```

### **\* dump\_root\_table**

格式:

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port
dump_root_table
```

含义:

按序打印出 RooTTable 的内容。需要在“~/oceanbase/bin/rs\_admin.log”中查看。

### **\* dump\_server\_info**

格式:

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port
dump_server_info
```

含义:

打印 ChunkServer 的信息。需要在“~/oceanbase/bin/rs\_admin.log”中查看。

举例:

```
[admin@OceanBase002 ~]$ ~/oceanbase/bin/rs_admin -r 10.10.10.2 -p 2500
dump_migrate_info
server[10.10.10.2:2500], timeout=20000000
do_rs_admin, cmd=9...
Okay
```

### **\* dump\_unusual\_tablets**

格式:

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port  
dump_unusual_tablets
```

含义：

打印副本或者版本不满足要求的 **tablet** 信息。需要在“~/oceanbase/bin/rs\_admin.log”中查看。

举例：

```
[admin@OceanBase002 ~]$ ~/oceanbase/bin/rs_admin -r 10.10.10.2 -p 2500  
dump_unusual_tablets  
server[10.10.10.2:2500], timeout=20000000  
do_rs_admin, cmd=8...  
Okay
```

### **\* enable\_balance|disable\_balance**

格式：

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port  
enable_balance|disable_balance
```

含义：

开启 **balance** 负载均衡。

举例：

```
[admin@OceanBase002 ~]$ ~/oceanbase/bin/rs_admin -r 10.10.10.2 -p 2500 enable_balance  
server[10.10.10.2:2500], timeout=20000000  
do_rs_admin, cmd=11...  
Okay
```

### **\* enable\_rereplication|disable\_rereplication**

格式：

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port  
enable_rereplication|disable_rereplication
```

含义：

开启或者关闭复制过程。

举例：

```
[admin@OceanBase002 ~]$ ~/oceanbase/bin/rs_admin -r 10.10.10.2 -p 2500  
enable_rereplication  
server[10.10.10.2:2500], timeout=20000000  
do_rs_admin, cmd=13...  
Okay
```

### **\* force\_create\_table**

格式：

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port  
force_create_table -o table_id=table_id
```

含义：

强制新建空 **tablet**。该命令只能在备集群上执行成功，这是为了防止 **trigger\_create\_table** 消息丢失而增加的运维手段。

举例：

在主机群上强制建立 3001 号表的空 **tablet**。

### **\* force\_drop\_table**

格式：

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port  
force_drop_table -o table_id=table_id
```

含义：

强制 **drop** 某张表。该命令只能在备集群上执行成功，这是为了防止 **trigger\_drop\_table** 丢失而增加的运维手段。

举例：

在主集群上删除 3001 号表。

### **\* get\_config**

格式：

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port get_config
```

含义：

打印配置项到屏幕上。

举例：

```
[admin@OceanBase002 ~]$ ~/oceanbase/bin/rs_admin -r 10.10.10.2 -p 2500 get_config  
server[10.10.10.2:2500], timeout=20000000  
tablet_replicas_num=3  
retry_times=3  
read_master_master_ups_percent=50  
first_meta_filename=first_tablet_meta  
.....
```

### **\* get\_obi\_config**

暂时未使用。

### **\* get\_obi\_role**

格式：

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port get_obi_role
```

含义：

获取集群的主备角色。

举例：

```
[admin@OceanBase002 ~]$ ~/oceanbase/bin/rs_admin -r 10.10.10.2 -p 2500 get_obi_role
server[10.10.10.2:2500], timeout=20000000
get_obi_role...
obi_role=MASTER
```

### **\* get\_row\_checksum**

暂时未使用。

### **\* import**

格式：

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port import
table_name table_id uri
```

含义：

旁路导入命令。增加旁路导入任务。

### **\* import\_tablets**

格式：

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port
import_tablets -o table_id=table_id
```

含义：

旁路导入命令。有旁路导入的脚本来调用，RootServer 收到请求后，会要求 ChunkServer 加载某个 table 的 Tablet。

### **\* kill\_import**

格式：

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port kill_import
table_name table_id
```

含义：

旁路导入命令，删除一个旁路导入任务。

### **\* log\_move\_to\_debug**

格式：

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port
log_move_to_debug
```

含义:

将 RootServer 的日志设置成 Debug 级别。

### **\* log\_move\_to\_error**

格式:

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port  
log_move_to_error
```

含义:

将 RootServer 的日志设置成 Error 级别。

### **\* print\_root\_table**

格式:

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port  
print_root_table -o table_id=table_id
```

含义:

按 table\_id 打印 roottable 的内容

举例:

```
[admin@OceanBase002 ~]$ ~/oceanbase/bin/rs_admin -r 10.10.10.2 -p 2500 print_root_table -o  
table_id=2  
server[10.10.10.2:2500], timeout=20000000  
do_print_root_table, table_id=2...  
  
extend:min| 11e|  
extend:max|occupy_size 0|record_count 0|1_port 15834|1_ms_port 15844|1_ipv4  
136767498|1_tablet_version 1|2_port 15834|2_ms_port 15844|2_ipv4  
119990282|2_tablet_version 1|
```

### **\* print\_schema**

格式:

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port print_schema  
-o schema_location=schema_location
```

含义:

生成 schema 的可读格式到指定的文件中。

举例:

```
[admin@OceanBase002 ~]$ ~/oceanbase/bin/rs_admin -r 10.10.10.2 -p 2500 print_schema -o  
schema_location=/tmp/a.ini  
server[10.10.10.2:2500], timeout=20000000  
do_print_schema...  
write schema to file. file_name=/tmp/a.ini
```

### **\* refresh\_schema**

格式:

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port  
refresh_schema
```

含义:

更新 schema。在用户用 sql 语句的形式修改三张核心表的内容时，需要主动调用一次 refresh\_schema。RootServer 收到命令后，会从内部表中读出最新的 schema 并缓存起来。

举例:

```
[admin@OceanBase002 ~]$ ~/oceanbase/bin/rs_admin -r 10.10.10.2 -p 2500 refresh_schema  
server[10.10.10.2:2500], timeout=20000000  
do_rs_admin, cmd=18...  
Okay
```

### **\* reload\_config**

格式:

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port reload_config
```

含义:

修改配置文件后，需要重新加载配置文件。

### **\* restart\_cs**

格式:

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port restart_cs -o  
server_list=<ip1+ip2+...+ipn>[,cancel]
```

含义:

重启 ChunkServer。命令发出后，ChunkServer 会直接退出。

### **\* set\_config**

格式:

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port set_config -o  
configuration_parameter=value
```

含义:

临时设置配置项的值，只修改了 RootServer 内存中值，如果 RootServer 重启的话，则修改会丢失。

举例:

```
[admin@OceanBase002 ~]$ ~/oceanbase/bin/rs_admin -r 10.10.10.2 -p 2500 set_config -o
network_timeout=5
server[10.10.10.2:2500], timeout=20000000
config_str: network_timeout=5
OKay
```

### \* **set\_master\_ups\_config**

格式:

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port
set_master_ups_config -o
master_master_ups_read_percentage=master_master_ups_read_percentage
-o slave_master_ups_read_percentage=slave_master_ups_read_percentage
```

含义:

设置集群内主 UpdateServer 的流量分配。如果本集群为主集群的话，则集群内主 UPS 的流量按照 `master_master_ups_read_percentage` 的值来定，如果是备集群的话，则集群内主 UPS 的流量按照 `read_salve_master_ups_percentage` 来算，其他 UPS 则平分剩余的流量。

举例:

```
[admin@OceanBase002 ~]$ ~/oceanbase/bin/rs_admin -r 10.10.10.2 -p 2500
set_master_ups_config -o master_master_ups_read_percentage=30 -o
slave_master_ups_read_percentage=40
server[10.10.10.2:2500], timeout=20000000
set_master_ups_config,
master_master_ups_read_percentage=30,read_salve_master_ups_percentage=40...
OKay
```

### \* **set\_obi\_config**

暂时未使用。

### \* **set\_obi\_role**

格式:

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port set_obi_role -
o OBI_MASTER
```

含义:

设置集群的主备角色，可以设置为“OBI\_MASTER”或者“OBI\_SLAVE”。

举例:

```
[admin@OceanBase002 ~]$ ~/oceanbase/bin/rs_admin -r 10.10.10.2 -p 2500 set_obi_role -o
OBI_MASTER
server[10.10.10.2:2500], timeout=20000000
set_obi_role...role=0
OKay
```



## \* **set\_ups\_config**

格式:

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port  
set_ups_config -o ups_ip=UpdateServer_ip -o ups_port=UpdateServer_port -  
o configuration_parameter=value -o configuration_parameter=value ...
```

含义:

设置单个 UPS 的流量分配。比如设置集群 server 访问 10.232.38.7: 5814 这台 UPS 的流量, 正常读请求的流量设置为 30%, 每日合并过程的流量设置成 40%

举例:

```
[admin@OceanBase002 ~]$ ~/oceanbase/bin/rs_admin -r 10.10.10.2 -p 2500 set_ups_config -o  
ups_ip=10.10.10.2 -o ups_port=2700 -o ms_read_percentage=40 -o cs_read_percentage=30  
server[10.10.10.2:2500], timeout=20000000  
set_ups_config, ups_ip=10.10.10.2 ups_port=2700 ms_read_percentage=40  
cs_read_percentage=30...  
Okay
```

## \* **shutdown**

格式:

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port shutdown -o  
server_list=<cs_ip1+cs_ip2+...+cs_ipn>[,cancel]
```

含义:

下线 ChunkServer 的功能。RootServer 会先将该 ChunkServer 上的 tablet 全部迁走, 迁完以后 RootServer 会通知该 ChunkServer 可以下线了。

举例:

```
[admin@OceanBase002 ~]$ ~/oceanbase/bin/rs_admin -r 10.10.10.2 -p 2500 shutdown -o  
server_list=10.10.10.4  
server[10.10.10.2:2500], timeout=20000000  
do_shutdown_servers, server_list=10.10.10.4 cancel=0...  
Okay
```

## \* **split\_tablet**

暂时未使用。

## \* **stat**

格式:

```
~/oceanbase/bin/rs_admin -r RootServer_ip -p RootServer_port -t  
request_timeout_us command -o suboptions
```

含义:

查询 RootServer 的状态值。该命令需要带参数表达式，如[表 5-1](#)所示。

**表 5-1 参数表达式**

参数	含义
all_server	打印所有 Server 的状态信息。
client_conf	暂时未使用。
copy_count	统计复制的次数。
cs	打印所有 ChunkServer 的 IP 地址和端口。
cs_num	打印 ChunkServer 的个数。
data_size	暂时未使用。
fail_get_count	打印 get 请求失败的次数。
fail_scan_count	打印 scan 请求失败的次数。
frozen_time	打印执行版本冻结的次数和最近一次冻结的时间。
frozen_version	打印 UpdateServer 汇报的最新版本。
get_obi_role_count	统计执行“get_obi_role”的次数。
local_time	打印 RootServer 服务器的当前时间。
log_file_id	打印当前记录的 RootServer 的 CommitLog 日志文件名。
log_id	打印当前 RootServer 的 CommitLog 日志序列号。
mem	打印 RootServer 服务器的内存使用情况。
merge	打印每日合并的状态。
migrate_count	统计迁移的次数。
ms	打印所有 MergeServer 的 IP 地址和端口，包括 LMS。
ms_num	打印 MergeServer 的个数。

参数	含义
ops_get	打印 get 请求的次数。
ops_scan	打印 scan 请求的次数。
pid	打印 RootServer 的进程号。
prog_version	打印 RootServer 的版本号。
reserve	保留项，暂时未使用。
row_count	暂时未使用。
rs_slave_num	打印备 RootServer 的个数。
rs_slave	打印备 RootServer 的信息。如果 RootServer 为单机时，显示为空。
rs_status	打印 RootServer 的状态。
schema_version	暂时未使用。
shutdown_cs	下线 ChunkServer。RootServer 会将该 ChunkServer 上的 tablet 全部迁移到其他 ChunkServer，迁移完成后，日志通知用户迁移已经完成，可以直接将该 ChunkServer 下线掉。
sstable_dist	按表统计 SSTable 在每个 ChunkServer 上的分布情况。
start_time	打印 RootServer 的启动时间。
table_count	打印 table 的个数。
table_num	暂时未使用。
tablet_count	打印 Tablet 的个数。
tablet_num	打印 RootTable 的大小，并不表示 Tablet 的个数。如果需要查看 Tablet 个数请使用“dump_root_table”命令，然后查看日志。

参数	含义
unusual_tablets_num	打印不正常的 Tablet 的个数。 <i>说明：</i> 不正常是指“副本不够或者没有合并到最新版本”。该命令一般用于检查集群 <i>Tablet</i> 状态。
ups	打印主 UpdateServer 的租约剩余时间、主备 UpdateServer 的 IP 地址、端口、状态等信息。

举例：

```
[admin@OceanBase002 ~]$ ~/oceanbase/bin/rs_admin -r 10.10.10.2 -p 2500 stat -o common
server[10.10.10.2:2500], timeout=20000000
do_rs_stat, key=1...
start_time: Fri Mar 28 16:04:00 2014
local_time: Fri Mar 28 17:10:31 2014
prog_version: OceanBase 0.4.2.15(1676.el6)
pid: 11642
obi_role: MASTER
```

#### \* **switch\_schema**

暂时未使用。