ALIBABA CLOUD

阿里云

数据湖分析

AnalyticDB for MySQL 2.0

文档版本：20201201

阿里云

法律声明

阿里云提醒您在阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读或使用本文档，您的阅读或使用行为将被视为对本声明全部内容的认可。

1. 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档，且仅能用于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息，您应当严格遵守保密义务；未经阿里云事先书面同意，您不得向任何第三方披露本手册内容或提供给任何第三方使用。

2. 未经阿里云事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。

3. 由于产品版本升级、调整或其他原因，本文档内容有可能变更。阿里云保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在阿里云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。

4. 本文档仅作为用户使用阿里云产品及服务的参考性指引，阿里云以产品及服务的“现状”、“有缺陷”和“当前功能”的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的，阿里云不承担任何法律责任。在任何情况下，阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害，包括用户使用或信赖本文档而遭受的利润损失，承担责任（即使阿里云已被告知该等损失的可能性）。

5. 阿里云网站上所有内容，包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计，均由阿里云和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意，任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外，未经阿里云事先书面同意，任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称（包括但不限于单独为或以组合形式包含“阿里云”、“Aliyun”、“万网”等阿里云和/或其关联公司品牌，上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司）。

6. 如若发现本文档存在任何错误，请与阿里云取得直接联系。

通用约定

格式	说明	样例
危险	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	危险 重置操作将丢失用户配置数据。
警告	该类警示信息可能会导致系统重大变更甚至故障，或者导致人身伤害等结果。	警告 重启操作将导致业务中断，恢复业务时间约十分钟。
注意	用于警示信息、补充说明等，是用户必须了解的内容。	注意 权重设置为0，该服务器不会再接受新请求。
说明	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	说明 您也可以通过按Ctrl+A选中全部文件。
>	多级菜单递进。	单击**设置**> **网络**> **设置网络类型**。
粗体	表示按键、菜单、页面名称等UI元素。	在**结果确认**页面，单击**确定**。
`Courier字体`	命令或代码。	执行 `cd /d C:/window` 命令，进入Windows系统文件夹。
斜体	表示参数、变量。	`bae log list --instanceid` *Instance_ID*
[] 或者 [a|b]	表示可选项，至多选择一个。	`ipconfig [-all|-t]`
{} 或者 {a|b}	表示必选项，至多选择一个。	`switch {active|stand}`

目录

1.读写AnalyticDB for MySQL 2.0 05
1.1. 背景信息以及准备工作 05
1.2. 创建OSS Schema和表 07
1.3. 创建AnalyticDB Schema 08
1.4. 执行INSERT FROM SELECT迁移数据 10

1.读写AnalyticDB for MySQL 2.0

1.1. 背景信息以及准备工作

数据湖分析 DLA（Data Lake Analytics）作为云上数据处理的枢纽，通过DLA可以读取存储在对象存储服务OSS（Object Storage Service）中的数据，对其进行清洗后再通过DLA将结果数据写入AnalyticDB for MySQL 2.0，通过AnalyticDB for MySQL 2.0强大的计算引擎，对数据进行毫秒级计算分析。

> **说明** DLA支持的OSS数据格式有LOG、CSV、JSON。

您需要根据DLA和AnalyticDB for MySQL 2.0所属Region，分别在AnalyticDB for MySQL 2.0和DLA中创建相同的VPC，DLA通过AnalyticDB for MySQL 2.0的VPC地址连接AnalyticDB for MySQL 2.0。

华东1（杭州）	cn-hangzhou-g	vpc-bp1g66t4f0onrvbht2et5	vsw-bp1nh5ri8di2q7tkof474
华东2（上海）	cn-shanghai-d	vpc-uf6wxkgst74es59wqareb	vsw-uf6m7k4fcq3pgd0yjfdnm
华北2（北京）	cn-beijing-g	vpc-2zeawsrpzbelyjko7i0ir	vsw-2zea8ct4hy4hwsrcpd52d
华北3（张家口）	cn-zhangjiakou-a	vpc-8vbpi1t7c0devxwfe19sn	vsw-8vbjl32xkft0ewggef6g9
华南1（深圳）	cn-shenzhen-a	vpc-wz9622zx341dy24ozifn3	vsw-wz91ov6gj2i4u2kenpe42
新加坡	ap-southeast-a	vpc-t4n3sczhu5efvwo1gsupf	vsw-t4npcrmzzk64r13e3nhhm
英国（伦敦）	eu-west-1a	vpc-d7ovzdful8490upm8b413	vsw-d7opmgixr2h34r1975s8a
日本（东京）	ap-northeast-1b	vpc-6weiev04rkbhguxqnnh2c	vsw-6wei487pduzx7753blrfr
美国（即佛尼亚）	us-east-1a	vpc-0xiarsc9sd2sen1kmii71	vsw-0xi31nno1mx2rb1sk1u3k
美国（硅谷）	us-west-1a	vpc-rj98sujw7ms8kkpdv9hak	vsw-rj9o2vqz0xjbj9cmjqeiv
中国（香港）	cn-hongkong-b	vpc-j6cotfjiboiskto4htqga	vsw-j6cv4uyb6qybxnpjzph2p

德国（法兰克福）	eu-central-1b	vpc-gw8mxe263blkfaurn81ej	vsw-gw8j7409kagfk3rdjwdak
印度（孟买）	ap-south-1a	vpc-a2d9k64h6zy80akzvnhyf	vsw-a2d6wrycvxb71rszsnpc8
澳大利亚（悉尼）	ap-southeast-2b	vpc-p0wsvf29xy47hbsw9440g	vsw-p0w5t4q4pmu8pjj3z5s81

准备工作

1. 开通DLA、AnalyticDB for MySQL 2.0以及OSS服务，且DLA、AnalyticDB for MySQL 2.0、OSS所属Region相同。
2. 在DLA中创建服务访问点，即创建VPC专有网络。详情，请参见创建服务访问点。
3. 在AnalyticDB for MySQL 2.0中创建VPC，DLA将通过AnalyticDB for MySQL 2.0的VPC地址连接AnalyticDB for MySQL 2.0，详情请参见。
4. 如果您的AnalyticDB所属VPC与DLA相同，则跳过本步骤，继续下面的步骤。否则，可通过MySQL命令行工具或者客户端DBeaver等连接AnalyticDB for MySQL 2.0，执行以下SQL修改AnalyticDB for MySQL 2.0的VPC，使DLA能够连接AnalyticDB for MySQL 2.0。

```
ALTER DATABASE ads_database set zone_id='cn-shanghai-d' vpc_id='vpc-uf6wxkgst74es59wqareb' vs
witch_id='vsw-uf6m7k4fcq3pgd0yjfdnm';
```

5. 在AnalyticDB for MySQL 2.0中创建实时表，用于存储OSS数据。

   本示例中的表为shipping和order_table表，表结构信息如下所示。

```
CREATE TABLE ads_database.shipping (
id bigint NOT NULL COMMENT '',
origin_state varchar NOT NULL COMMENT '',
origin_zip varchar NOT NULL COMMENT '',
destination_state varchar NOT NULL COMMENT '',
destination_zip varchar NOT NULL COMMENT '',
package_weight int NOT NULL COMMENT '',
PRIMARY KEY (id)
)
PARTITION BY HASH KEY (id) PARTITION NUM 23
TABLEGROUP ads
OPTIONS (UPDATETYPE='realtime')
COMMENT ''
```

```
CREATE TABLE ads_database.order_table (
 customer_id bigint NOT NULL COMMENT '',
 order_id varchar NOT NULL COMMENT '',
 order_time date NOT NULL COMMENT '',
 order_amount double NOT NULL COMMENT '',
 order_type varchar NOT NULL COMMENT '',
 address varchar NOT NULL COMMENT '',
 city varchar NOT NULL COMMENT '',
 order_season bigint COMMENT '',
 PRIMARY KEY (customer_id)
)
PARTITION BY HASH KEY (customer_id) PARTITION NUM 32
TABLEGROUP ads
OPTIONS (UPDATETYPE='realtime')
COMMENT ''
```

注意事项

- AnalyticDB for MySQL 2.0为主键覆盖逻辑，若整个INSERT FROM SELECT的ETL任务失败，需要整体重试。
- 在AnalyticDB for MySQL 2.0中查询数据时，存在一定的延迟，延迟时长取决于AnalyticDB for MySQL 2.0中的ECU规格。
- 建议将ETL任务尽量切成小的单位分批执行。例如，OSS中的数据为200GB，在业务允许的情况下，200GB的数据切分为100个文件夹，每个文件夹存储2GB数据。在DLA中创建100张表，100张表分别做ETL，单个ETL任务失败，只重试单个ETL任务。
- OSS数据写入AnalyticDB for MySQL 2.0后，需要修改AnalyticDB for MySQL 2.0的VPC地址，否则除了DLA可以连接AnalyticDB for MySQL 2.0以外，其他阿里云服务将无法通过AnalyticDB for MySQL 2.0的当前VPC地址访问AnalyticDB for MySQL 2.0。

```
ALTER DATABASE ads_database set zone_id='cn-shanghai-d' vpc_id='vpc-uf6wxkgst*****' vswitch_id='vsw-uf6m7k4fcq3pgd0y****';
```

1.2. 创建OSS Schema和表

本示例中dla_table_1、dla_table_2分别与AnalyticDB中的shipping、order_table表信息一致。

1. 登录DLA控制台。
2. 单击左侧导航栏的**SQL访问点**，然后**登录DMS**创建OSS Schema。

```
CREATE SCHEMA oss_data_schema with DBPROPERTIES(
 catalog='oss',
 location = 'oss://oss_bucket_name/table/'
 );
```

   - `catalog='oss'`：指定创建的是OSS Schema。

- location：数据文件所在的OSS Bucket的目录，需以 / 结尾表示目录。

  后续建表的LOCATION所指向的数据文件，必须在这个OSS目录下。本示例中table就是OSS中存放文件的Object。

创建OSS表

```
CREATE EXTERNAL TABLE IF NOT EXISTS dla_table_1 (
 id bigint NOT NULL COMMENT '',
 origin_state varchar NOT NULL COMMENT '',
 origin_zip varchar NOT NULL COMMENT '',
 destination_state varchar NOT NULL COMMENT '',
 destination_zip varchar NOT NULL COMMENT '',
 package_weight int NOT NULL COMMENT ''
)
ROW FORMAT DELIMITED FIELDS TERMINATED BY '|'
STORED AS TEXTFILE
LOCATION 'oss://oss_bucket_name/table/';
```

```
CREATE EXTERNAL TABLE IF NOT EXISTS dla_table_2 (
  customer_id bigint NOT NULL COMMENT '',
  order_id varchar NOT NULL COMMENT '',
  order_time date NOT NULL COMMENT '',
  order_amount double NOT NULL COMMENT '',
  order_type varchar NOT NULL COMMENT '',
  address varchar NOT NULL COMMENT '',
  city varchar NOT NULL COMMENT '',
  order_season bigint COMMENT '',
  PRIMARY KEY (customer_id)
)
ROW FORMAT DELIMITED FIELDS TERMINATED BY '|'
STORED AS TEXTFILE
LOCATION 'oss://oss_bucket_name/table/';
```

1.3. 创建AnalyticDB Schema

本文主要介绍在DLA中如何创建AnalyticDB Schema。

创建Schema

> **注意**
>
> AnalyticDB for MySQL 2.0所属VPC与DLA所属VPC必须相同。

1. 登录DLA控制台。
2. 单击左侧导航栏的**SQL访问点管理**，然后单击**登录DMS**创建AnalyticDB for MySQL 2.0 Schema。

```
CREATE SCHEMA ads_database_schema WITH DBPROPERTIES
(
 CATALOG = 'ads',
 LOCATION = 'jdbc:mysql://ads-database-********-vpc.cn-shanghai-1.ads.aliyuncs.com:10001/ads_database',
 USER='AnalyticDB AccessKey ID',
 PASSWORD='AnalyticDB AccessKey Secret'
);
```

   - `CATALOG = 'ads'`：指定创建的是AnalyticDB for MySQL 2.0 Schema。
   - LOCATION：AnalyticDB for MySQL 2.0的连接信息，格式为`连接地址:端口号/AnalyticDB for MySQL 2.0集群名称`，通过AnalyticDB 控制台获取。

创建表

Schema创建成功后，接下来就可以在DLA中创建表，表分为以下两种情况：

- 表的表名、字段名或者字段类型与AnalyticDB for MySQL 2.0表信息不一致。
- 表的表名、字段名或者字段类型与AnalyticDB for MySQL 2.0表信息完全一致。

DLA中的表名、列名与AnalyticDB for MySQL 2.0表信息一致（建议一致）

以下示例在DLA中创建shipping表。

```
CREATE EXTERNAL TABLE shipping (
 id bigint NOT NULL COMMENT '',
 origin_state varchar NOT NULL COMMENT '',
 origin_zip varchar NOT NULL COMMENT '',
 destination_state varchar NOT NULL COMMENT '',
 destination_zip varchar NOT NULL COMMENT '',
 package_weight int NOT NULL COMMENT '',
 PRIMARY KEY (id)
)
```

DLA中的表名、列名与AnalyticDB for MySQL 2.0表信息不一致

以下示例在DLA中创建与order_table对应的表order_table1。其中，order_table1中部分字段名与order_table不同，order1_id数据类型与order_id不同。

```
CREATE EXTERNAL TABLE order_table1 (
 customer1_id bigint NOT NULL COMMENT '',
 order1_id bigint NOT NULL COMMENT '',
 order1_time date NOT NULL COMMENT '',
 order1_amount double NOT NULL COMMENT '',
 order1_type varchar NOT NULL COMMENT '',
 address1 varchar NOT NULL COMMENT '',
 city1 varchar NOT NULL COMMENT '',
 order1_season bigint COMMENT '',
 PRIMARY KEY (customer1_id)
)
tblproperties (
  table_mapping = 'ads_database.order_table',
  column_mapping = 'customer1_id,customer_id; order1_id,order_id; order1_time:order_time,
  order1_amount:order_amount, order1_type:order_type, address1:address,
  city1:city,order1_season:order_season'
);
```

- tblproperties：DLA中的表与源表的映射关系
- table_mapping：表名映射
- column_mapping：字段（列）映射

下一步

执行INSERT FROM SELECT将OSS数据写入AnalyticDB for MySQL 2.0

1.4. 执行INSERT FROM SELECT迁移数据

您可以通过MySQL客户端、MySQL命令行工具或者DMS for Data Lake Analytics连接DLA，然后执行INSERT INTO SELECT将OSS中的数据写入AnalyticDB for MySQL 2.0。

注意事项

- INSERT INTO SELECT通常为长时运行任务，建议您通过**异步执行**方式。
- 通过MySQL命令行写入数据时，需要在命令行中指定 `-c` 参数，用来识别MySQL语句前的Hint。

  ```
  mysql -hxxx -Pxxx -uxxx -pxxx db_name -c
  ```

  - h：DLA的**连接串信息**，通过DLA控制台获取。如果您的MySQL客户端或者MySQL命令行工具安装在与DLA同地域的ECS上，且ECS所属VPC与DLA所属VPC相同，则使用DLA **VPC专有网络**连接串。否则，使用DLA**经典网络**连接串。
  - P：DLA的连接**端口**，通过DLA控制台获取。
  - u：DLA用户名，详情，请参见账号类型。
  - db_name：DLA中的Schema名。

- 如果在INSERT INTO子句和SELECT子句中没有指定列信息，请确保源表和目标表的列顺序一致、列类型对应匹配。
- 如果在INSERT INTO子句和SELECT子句中指定列信息，请确保列的顺序符合业务需要、列类型对应匹配。

写入数据

异步执行以下SQL，将OSS table目录下dla_table_1文件中的数据插入AnalyticDB for MySQL 2.0的ads_database数据库中的shipping表。

```
-- OSS到AnalyticDB for MySQL 2.0的全量数据插入
/*+run-async=true*/
INSERT INTO ads_database_schema.shipping
SELECT * FROM oss_data_schema.dla_table_1;
```

异步执行以下SQL，将OSS table目录下dla_table_1文件中 `order_amount > 2` 的数据插入AnalyticDB for MySQL 2.0中ads_database数据库中的order_table表。

```
-- OSS到AnalyticDB for MySQL 2.0的数据插入，包含对OSS数据的筛选逻辑
/*+run-async=true*/
INSERT INTO ads_database_schema.order_table1 (customer1_id, order1_id, order1_time, order1_amount,order1_type,address1,city1,order1_season)
SELECT customer_id, order_id, order_time, order_amount,order_type,address,city,order_season
FROM oss_data_schema.dla_table_2
WHERE order_amount > 2
LIMIT 10000;
```

查询数据

异步执行INSERT INTO SELECT将返回一个task id，在AnalyticDB for MySQL 2.0中通过task id轮询任务执行情况，如果status为**SUCCESS**，则数据写入成功。

```
SHOW query_task WHERE id = 'q201812241524sh385b7d4c0049869';
```