数据湖探索

最佳实践

文档版本	01
发布日期	2024-12-27

华为技术有限公司

商标声明


和其他华为商标均为华为技术有限公司的商标。
本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

安全声明

漏洞处理流程

华为公司对产品漏洞管理的规定以“漏洞处理流程”为准，该流程的详细内容请参见如下网址：
https://www.huawei.com/cn/psirt/vul-response-process
如企业客户须获取漏洞信息，请参见如下网址：
https://securitybulletin.huawei.com/enterprise/cn/security-advisory

目 录

1 最佳实践内容概览

表 1-1 DLI 最佳实践

方案	说明
使用DLI分析车联网场景驾驶行为数据	使用DLI进行车联网场景驾驶行为数据分析。
使用DLI将CSV数据转换为Parquet数据	使用DLI将CSV数据转换为Parquet数据的方法。
使用DLI分析电商BI报表	以某商城真实的用户、商品、评论数据（脱敏后）为基础，介绍使用DLI进行电商BI报表分析的方法。
使用DLI分析账单消费数据	以DLI实际消费数据为样例，介绍使用DLI进行账单分析和成本优化的措施。
使用DLI分析电商实时业务数据	使用DLI Flink完成电商业务实时数据的分析处理。
配置DBeaver连接DLI进行数据查询和分析	介绍DBeaver连接DLI并提交SQL查询的操作步骤。
配置DBT连接DLI进行数据调度和分析	介绍使用DBT提交DLI作业的操作步骤。
配置YongHong BI连接DLI进行数据查询和分析	介绍YongHong BI连接DLI的操作步骤。
配置PowerBI通过Kyuubi连接DLI进行数据查询和分析	介绍PowerBI基于Kyuubi连接DLI，以访问和分析DLI中的数据的操作步骤。
配置Fine BI通过Kyuubi连接DLI进行数据查询和分析	介绍Fine BI基于Kyuubi连接DLI的操作步骤。
配置SuperSet通过Kyuubi连接DLI进行数据查询和分析	介绍Superset基于Kyuubi连接DLI的操作步骤。
配置Tableau通过Kyuubi连接DLI进行数据查询和分析	介绍Tableau基于Kyuubi连接DLI的操作步骤。

方案	说明
配置Beeline通过Kyuubi连接DLI进行数据查询和分析	介绍Beeline基于Kyuubi连接DLI的操作步骤。

2 使用 DLI 分析车联网场景驾驶行为数据

应用场景

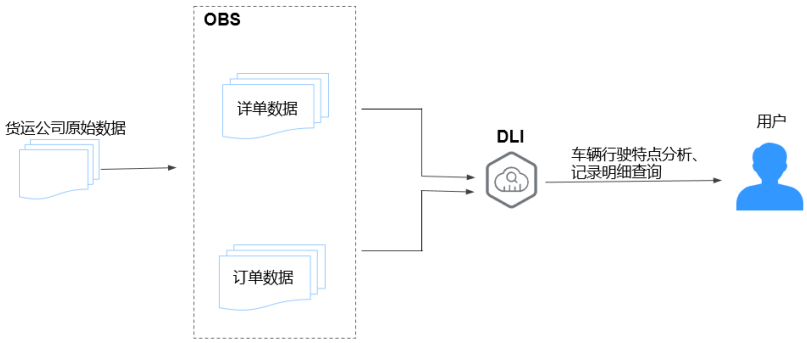
在车联网领域，云计算与大数据为企业提供了强大的分析挖掘能力，可以帮助企业和车队管理者更加科学、便捷地进行车辆数据管理与分析。

方案架构

根据已有的某货运公司车辆定时上报的详单数据和货运订单数据，DLI可以完成对该货运公司车辆行驶特点分析、记录明细的查询。

详细的数据说明请参考**数据说明**。

图 2-1 方案简介



流程指导

使用DLI进行驾驶行为数据分析的操作过程主要包括以下步骤：

步骤1：上传数据。将数据上传到对象存储服务OBS，为后面使用DLI完成数据分析做准备。

步骤2：分析数据。使用DLI对待分析的数据进行查询。

示例代码

具体样例数据及详细SQL语句可以通过**数据包**进行下载。

方案优势

- 数据免搬迁：DLI支持与多种数据源的对接，直接通过SQL建表就可以完成数据源的映射。
- 简单易用：直接使用标准SQL编写指标分析逻辑，无需关注背后复杂的分布式计算平台。
- 按需计费：日志分析按时效性要求按周期进行调度，每次调度之间存在大量空闲期。DLI按需计费只在使用期间收费，有效节约队列成本。

资源和成本规划

表 2-1 资源和成本规划

资源	资源说明	成本说明
OBS	需要创建一个OBS桶将数据上传到对象存储服务OBS，为后面使用DLI完成数据分析做准备。	OBS的使用涉及以下几项费用： • 存储费用：静态网站文件存储在OBS中产生的**存储费用**。 • 请求费用：用户访问OBS中存储的静态网站文件时产生的**请求费用**。 • 流量费用：用户使用自定义域名通过公网访问OBS时产生的**流量费用**。 实际产生的费用与存储的文件大小、用户访问所产生的请求次数和流量大小有关，请根据自己的业务进行预估。
DLI	在创建SQL作业前需购买队列，使用DLI的队列资源时，按照队列CU时进行计费。	如购买按需计费的队列，在使用队列资源时，按照队列CU时进行计费。 以小时为单位进行结算。不足一小时按一小时计费，小时数按整点计算。队列CU时按需计费的计算费用=单价*CU数*小时数。

数据说明

- 详单数据

  车辆上报的详单数据，包括定时上报的位置记录和异常的驾驶行为触发的告警事件数据。

  表 2-2 详单数据

字段名称	字段类型	字段说明
driverID	string	驾驶员ID
carNumber	string	车牌号
latitude	double	纬度
longitude	double	经度

字段名称	字段类型	字段说明
speed	int	速度
direction	int	方向
siteName	string	地点
time	timestamp	记录上报时间
isRapidlySpeedup	int	急加速标识，“1”表示急加速，“0”表示非急加速
isRapidlySlowdown	int	急减速
isNeutralSlide	int	空挡滑行
isNeutralSlideFinished	int	空挡滑行结束
neutralSlideTime	bigint	空挡滑行时长
isOverspeed	int	超速
isOverspeedFinished	int	超速结束
overspeedTime	bigint	超速时长
isFatigueDriving	int	疲劳驾驶
isHthrottleStop	int	停车轰油门
isOilLeak	int	用油异常

- 订单数据
  订单数据记录了货运订单相关的信息。

  表 2-3 订单数据

字段名称	字段类型	字段说明
orderNumber	string	订单号
driverID	string	驾驶员ID
carNumber	string	车牌号
customerID	string	客户ID
sourceCity	string	出发城市
targetCity	string	到达城市
expectArriveTime	timestamp	期望送达时间
time	timestamp	记录产生时间

字段名称	字段类型	字段说明
action	string	事件类型，包括创建订单、开始发货、货物送达、订单签收等事件

步骤 1：上传数据

将数据上传到对象存储服务OBS，为后面使用DLI完成数据分析做准备。

1. 下载OBS Browser+。下载地址请参考《**对象存储服务工具指南**》。
2. 安装OBS Browser+。安装步骤请参考《**对象存储服务工具指南**》。
3. 登录OBS Browser+。OBS Browser+支持AK方式登录，以及授权码登录两种登录方式。登录步骤请参考《**对象存储服务工具指南**》。
4. 通过OBS Browser+上传数据。

   在OBS Browser+页面单击“创建桶”，按照要求选择“区域”和填写“桶名”（例如：dli-demo），其他参数保持默认或根据需要选择，创建桶成功后，返回桶列表，单击桶dli-demo。OBS Browser+提供强大的拖拽上传功能，您可以将本地的一个或多个文件或者文件夹拖拽到对象存储的对象列表或者并行文件系统的对象列表中；同时您也可以将文件或文件夹拖拽到指定的目录上，这样可以上传到指定的目录中。

   单击**Best_Practice_01.zip**获取本示例的测试数据，将“Best_Practice_01.zip”压缩包解压。后续操作说明如下：

   - 详单数据：将解压后Data目录下的“detail-records”文件夹上传到OBS桶根目录下。
   - 订单数据：将解压后Data目录下的“order-records”文件夹上传到OBS桶根目录下。

步骤 2：分析数据

使用DLI对分析的数据进行查询。

1. 创建数据库、表。
   a. 在Console页面上方菜单栏中单击“产品”，单击“大数据”分类中的“数据湖探索 DLI”。
   b. 在DLI控制台总览页面左侧，单击“SQL编辑器”，进入SQL作业编辑器页面。
   c. 在SQL作业编辑器左侧，选择“数据库”页签，单击  创建demo数据库，请参见**图2-2**。

图 2-2 创建数据库

说明

“default”为内置数据库，不能使用该数据库名。

d. 选择demo数据库，在编辑框中输入以下SQL语句：

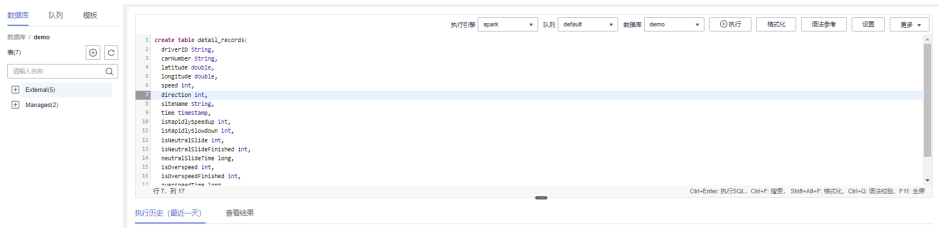
```
create table detail_records(
  driverID String,
  carNumber String,
  latitude double,
  longitude double,
  speed int,
  direction int,
  siteName String,
  time timestamp,
  isRapidlySpeedup int,
  isRapidlySlowdown int,
  isNeutralSlide int,
  isNeutralSlideFinished int,
  neutralSlideTime long,
  isOverspeed int,
  isOverspeedFinished int,
  overspeedTime long,
  isFatigueDriving int,
  isHthrottleStop int,
  isOilLeak int) USING CSV OPTIONS (PATH 'obs://dli-demo/detail-records/');
```

说明

使用该案例时，需将上述SQL语句中的文件路径修改为实际存放详单数据的OBS路径。

e. 单击“执行”，创建详单表detail_records，请参见图2-3。

图 2-3 创建详单表



f. 执行以下SQL语句，在demo数据库下创建告警事件表event_records，步骤同1.d和1.e。

```
create table event_records(
  driverID String,
  carNumber String,
  latitude double,
  longitude double,
  speed int,
  direction int,
  siteName String,
  time timestamp,
  isRapidlySpeedup int,
  isRapidlySlowdown int,
  isNeutralSlide int,
  isNeutralSlideFinished int,
  neutralSlideTime long,
  isOverspeed int,
  isOverspeedFinished int,
  overspeedTime long,
  isFatigueDriving int,
  isHthrottleStop int,
  isOilLeak int)
```

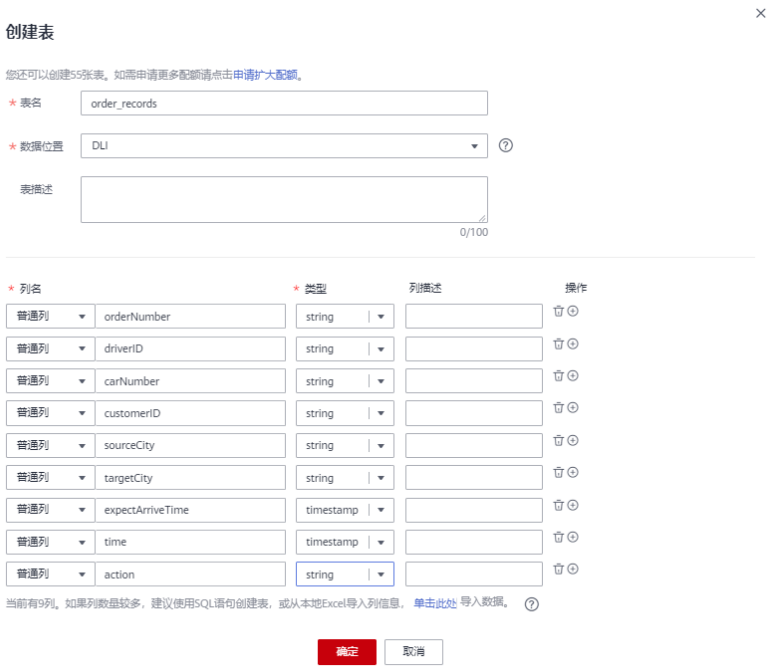
g. 执行以下SQL语句，将告警事件数据从详单中抽取出来插入到event_records表中。

```
insert into table event_records
(select *
from detail_records
where isRapidlySpeedup > 0
OR isRapidlySlowdown > 0
OR isNeutralSlide > 0
OR isNeutralSlideFinished > 0
OR isOverspeed > 0
OR isOverspeedFinished > 0
OR isFatigueDriving > 0
OR isHthrottleStop > 0
OR isOilLeak > 0)
```

h. 使用另一种方式创建订单表order_records。

在SQL作业编辑器左侧，选择“数据库”页签，单击数据库“demo”，单击表菜单右边的加号，创建表，数据位置选择DLI，请参见图2-4。字段类型请参见订单数据。

图 2-4 创建订单表

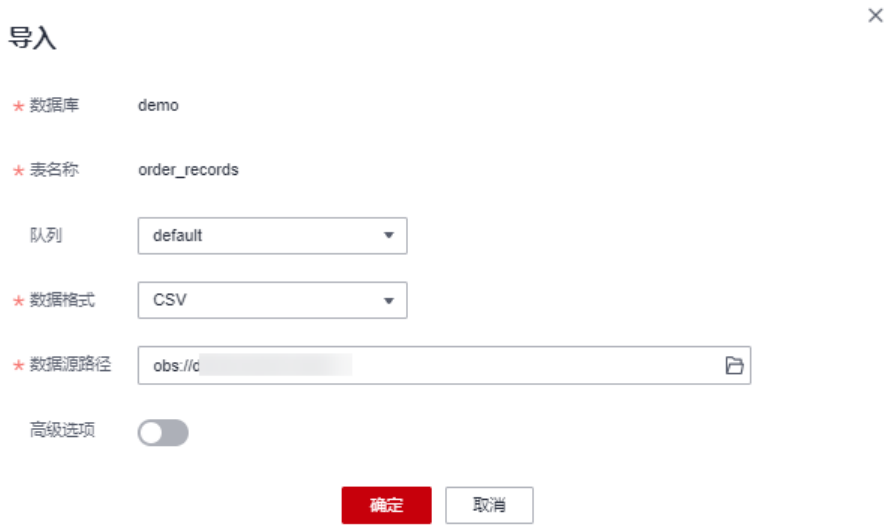


i. 将OBS数据导入到order_records表，单击“数据管理 > 库表管理”，单击demo数据库，进入“表管理”页面，单击order_records表对应“操作”列中的“更多”>“导入”，数据格式选择“CSV”，数据源路径为“obs://dli-demo/order-records/”，参数配置完成后单击“确定”。请参见图2-5。

说明

导入数据时，默认时间戳格式为“yyyy-MM-dd HH:mm:ss”，如果采用其他日期格式，可打开“高级选项”手动输入（本示例该选项不做修改）。

图 2-5 导入表数据



2. 执行查询

a. 执行以下SQL语句，对所有司机在某段时间的异常告警事件进行统计。

说明

常用查询语句可以在SQL编辑器中，选择“更多 > 设为模板”设置为模板。设为模板后，后续可以在模板管理页面找到对应模板进行SQL查询和修改。

具体操作为：选择“作业模板 > SQL模板 > 自定义模板”，在对应模板的操作列，单击“执行”会跳转到SQL语句编辑器，修改查询条件可以很方便地查找对应的数据。

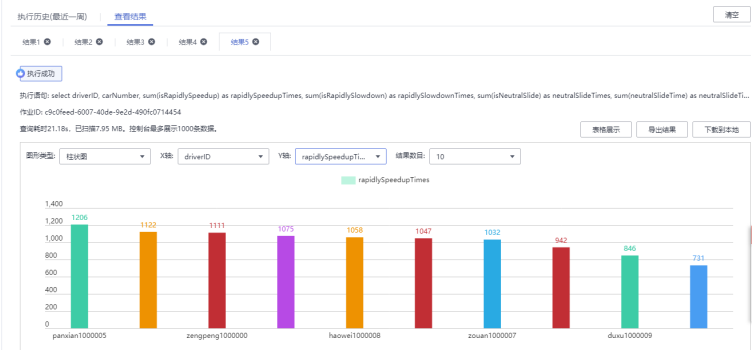
```
select
  driverID,
  carNumber,
  sum(isRapidlySpeedup) as rapidlySpeedupTimes,
  sum(isRapidlySlowdown) as rapidlySlowdownTimes,
  sum(isNeutralSlide) as neutralSlideTimes,
  sum(neutralSlideTime) as neutralSlideTimeTotal,
  sum(isOverspeed) as overspeedTimes,
  sum(overspeedTime) as overspeedTimeTotal,
  sum(isFatigueDriving) as fatigueDrivingTimes,
  sum(isHthrottleStop) as hthrottleStopTimes,
  sum(isOilLeak) as oilLeakTimes
from
  event_records
where
  time >= "2017-01-01 00:00:00"
  and time <= "2017-02-01 00:00:00"
group by
  driverID,
  carNumber
order by
  rapidlySpeedupTimes desc,
  rapidlySlowdownTimes desc,
  neutralSlideTimes desc,
  neutralSlideTimeTotal desc,
  overspeedTimes desc,
  overspeedTimeTotal desc,
  fatigueDrivingTimes desc,
  hthrottleStopTimes desc,
  oilLeakTimes desc
```

在查询结果中，单击“结果图形化”：

- “图形类型”选择“柱状图”
- “X轴”选择“driverID”
- “Y轴”选择“rapidlySpeedupTimes”
- “结果数目”选择“10”

展示结果如下：

图 2-6 急加速



b. 执行以下SQL语句，查询某个司机在某个时间段的详细记录。

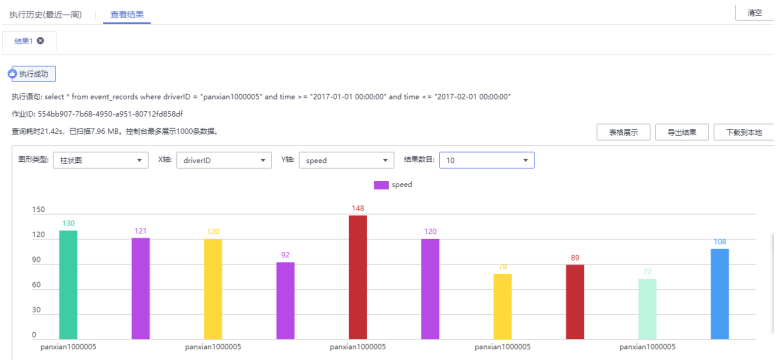
```
select
  *
from
  event_records
where
  driverID = "panxian1000005"
  and time >= "2017-01-01 00:00:00"
  and time <= "2017-02-01 00:00:00"
```

在查询结果中，单击 “结果图形化”：

- “图形类型”选择“柱状图”
- “X轴”选择“driverID”
- “Y轴”选择“speed”
- “结果数目”选择“10”

展示结果如下：

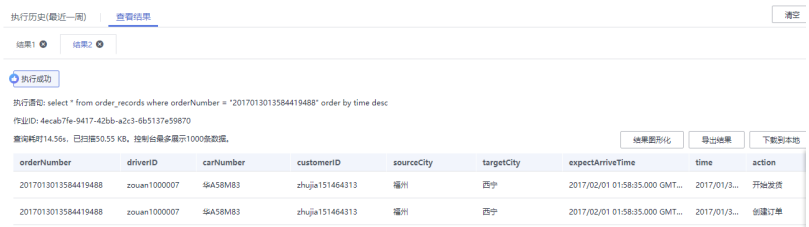
图 2-7 超速记录



c. 执行以下SQL语句，查询订单信息。

```
select
  *
from
  order_records
where
  orderNumber = "2017013013584419488"
order by
  time desc
```

图 2-8 订单信息



d. 执行以下SQL语句，根据司机和发车时间信息查询司机的详细行驶特点。

```
select
  driverID,
  carNumber,
  latitude,
  longitude,
```



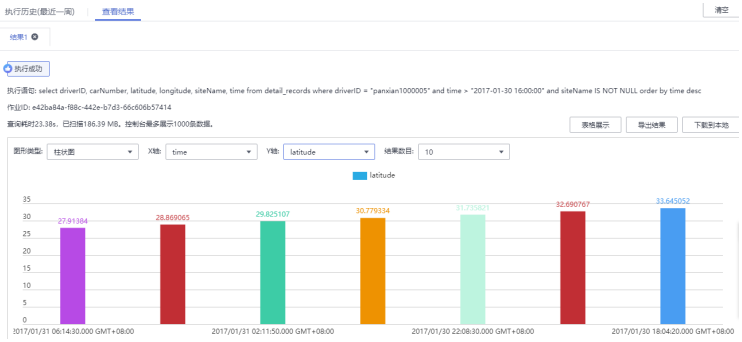
```
  siteName,
  time
from
  detail_records
where
  driverID = "panxian1000005"
  and time > "2017-01-30 16:00:00"
  and siteName IS NOT NULL
order by
  time desc
```

在查询结果中，单击 “结果图形化”：

- “图形类型”选择“柱状图”
- “X轴”选择“time”
- “Y轴”选择“latitude”
- “结果数目”选择“10”

展示结果如下：

图 2-9 行驶信息



3 使用 DLI 将 CSV 数据转换为 Parquet 数据

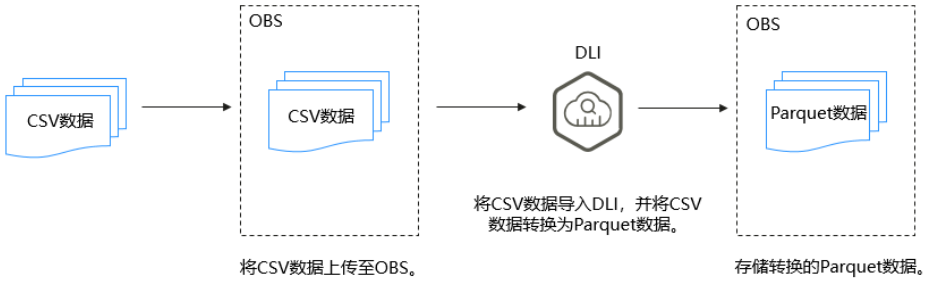
应用场景

Parquet是面向分析型业务的列式存储格式，这种格式可以加快查询速度，查询Parquet格式数据时，只检查所需要的列并对它们的值执行计算，也就是说，只读取一个数据文件或表的一小部分数据。Parquet还支持灵活的压缩选项，因此可以显著减少磁盘上的存储。使用DLI可轻松将CSV格式数据转换为Parquet格式数据。

方案架构

将CSV格式的数据上传到对象存储服务OBS，使用DLI将CSV数据转换为Parquet数据，并将转换后的Parquet数据存储到OBS中。

图 3-1 方案简介



流程指导

使用DLI将CSV数据转换为Parquet数据主要包括以下步骤：

步骤1：创建并上传数据。将数据上传到对象存储服务OBS。

步骤2：使用DLI将CSV数据转换为Parquet数据。将CSV数据导入DLI，并将CSV数据转换为Parquet数据。

方案优势

- **提升查询性能**

如果您在HDFS上拥有基于文本的数据文件或者表，而且正在使用Spark SQL对数据执行查询操作，那么推荐将文本数据文件转换为Parquet数据文件，转换需要时间，但查询性能的提升在某些情况下可能达到约30倍或更高。

- **节省存储空间**

  Parquet还支持灵活的压缩选项，因此可以显著减少磁盘上的存储。存储的节省可高达约75%。

资源和成本规划

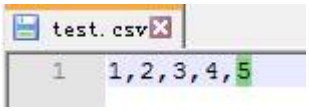
表 3-1 资源和成本规划

资源	资源说明	成本说明
OBS	需要创建一个OBS桶将数据上传到对象存储服务OBS，为后面使用DLI完成数据分析做准备。	OBS的使用涉及以下几项费用： • 存储费用：静态网站文件存储在OBS中产生的**存储费用**。 • 请求费用：用户访问OBS中存储的静态网站文件时产生的**请求费用**。 • 流量费用：用户使用自定义域名通过公网访问OBS时产生的**流量费用**。 实际产生的费用与存储的文件大小、用户访问所产生的请求次数和流量大小有关，请根据自己的业务进行预估。
DLI	在创建SQL作业前需购买队列，使用DLI的队列资源时，按照队列CU时进行计费。	如购买按需计费的队列，在使用队列资源时，按照队列CU时进行计费。 以小时为单位进行结算。不足一小时按一小时计费，小时数按整点计算。队列CU时按需计费的计算费用=单价*CU数*小时数。

步骤 1：创建并上传数据

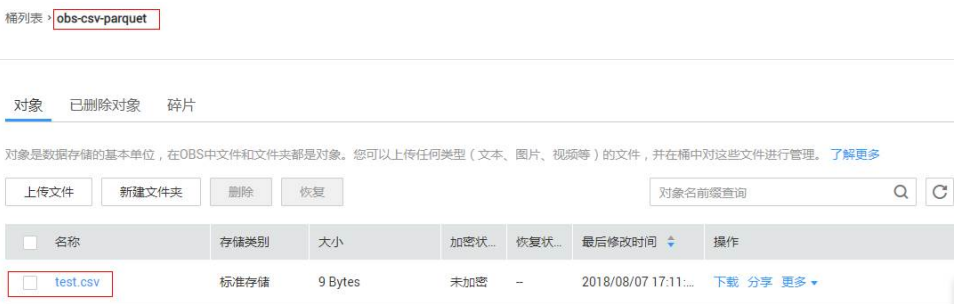
1. 创建CSV数据，例如，如**图3-2**所示test.csv：

   图 3-2 创建 test.csv 文件

2. 在OBS上建桶obs-csv-parquet，并将test.csv文件上传至OBS，如**图3-3**所示：

   图 3-3 上传 CSV 数据至 OBS

3. 在OBS上创建一个新的桶obs-parquet-data用于存储转换的Parquet数据。

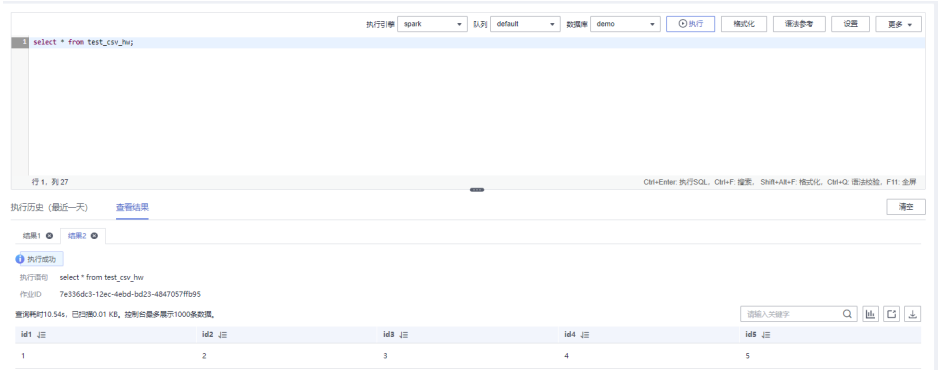
步骤 2：使用 DLI 将 CSV 数据转换为 Parquet 数据

1. 在DLI控制台总览页面左侧，单击“SQL编辑器”，进入SQL作业编辑器页面。
2. 在SQL作业编辑器左侧，选择“数据库”页签，单击创建名字为demo的数据库。
3. 在DLI的SQL编辑窗口，执行引擎选择“spark”，队列选择“default”，数据库选择为“demo”。输入以下建表语句，创建OBS表test_csv_hw并导入test.csv数据。

```
create table test_csv_hw(id1 int, id2 int, id3 int, id4 int, id5 int)
 using csv
 options(
 path 'obs://obs-csv-parquet/test.csv'
 )
```

4. 在DLI的SQL编辑窗口，执行以下语句可以查询表test_csv_hw中的数据。

图 3-4 查询表 test_csv_hw



5. 在DLI的SQL编辑窗口中创建OBS表test_parquet_hw。

```
create table `test_parquet_hw` (`id1` INT, `id2` INT, `id3` INT, `id4` INT, `id5` INT)
using parquet
options (
path 'obs://obs-parquet-data/'
)
```

说明

不需要指明具体的文件，因为在将数据从CSV格式转换为Parquet格式之前，不存在任何Parquet文件。

6. 在DLI的SQL编辑窗口中将CSV数据转换为Parquet数据并存储在OBS中。

```
insert into test_parquet_hw select * from test_csv_hw
```

7. 检查结果，如图3-5所示，系统自动创建了一个文件用于保存结果。

图 3-5 保存 Parquet 数据

4 使用 DLI 分析电商 BI 报表

应用场景

某电商商城在保持高速发展的同时，沉淀了数亿的忠实用户，积累了海量的真实数据。如何利用BI工具从历史数据中找出商机，是大数据应用在精准营销中的关键问题，也是所有电商平台在做智能化升级时所需要的核心技术。

本案例以某商城真实的用户、商品、评论数据（脱敏后）为基础，利用数据湖探索来分析用户和商品的各种数据特征，可为营销决策、广告推荐、信用评级、品牌监控、用户行为预测提供高质量的信息。

流程指导

使用DLI进行电商数据分析的操作过程主要包括以下步骤：

步骤1：上传数据。将数据上传到对象存储服务OBS，为后面使用DLI完成数据分析做准备。

步骤2：分析数据。使用DLI对待分析的数据进行查询。

具体样例数据及详细SQL语句可以通过**数据包**进行下载。

数据说明

为保护用户的隐私和数据安全，所有数据均已进行了采样和脱敏。

- 用户数据

  表 4-1 用户数据

字段名称	字段类型	字段说明	取值范围
user_id	int	用户ID	脱敏
age	int	年龄段	-1表示未知
gender	int	性别	• 0表示男 • 1表示女 • 2表示保密

字段名称	字段类型	字段说明	取值范围
rank	Int	用户等级	有顺序的级别枚举，越高级别数字越大
register_time	string	用户注册日期	单位：天

- 商品数据

  表 4-2 商品数据

字段名称	字段类型	字段说明	取值范围
product_id	int	商品编号	脱敏
a1	int	属性1	枚举，-1表示未知
a2	int	属性2	枚举，-1表示未知
a3	int	属性3	枚举，-1表示未知
category	int	品类ID	脱敏
brand	int	品牌ID	脱敏

- 评价数据

  表 4-3 评价数据

字段名称	字段类型	字段说明	取值范围
deadline	string	截止时间	单位：天
product_id	int	商品编号	脱敏
comment_num	int	累计评论数分段	• 0表示无评论 • 1表示有1条评论 • 2表示有2-10条评论 • 3表示有11-50条评论 • 4表示大于50条评论
has_bad_comment	int	是否有差评	0表示无，1表示有
bad_comment_rate	float	差评率	差评数占总评论数的比重

- 行为数据

表 4-4 行为数据

字段名称	字段类型	字段说明	取值范围
user_id	int	用户编号	脱敏
product_id	int	商品编号	脱敏
time	string	行为时间	-
model_id	string	模块编号	脱敏
type	string	• 浏览（指浏览商品详情页） • 加入购物车 • 购物车删除 • 下单 • 关注 • 点击	-

步骤 1：上传数据

将数据上传到对象存储服务OBS，为后面使用DLI完成数据分析做准备。

1. 下载OBS Browser+。下载地址请参考《**对象存储服务工具指南**》。
2. 安装OBS Browser+。安装步骤请参考《**对象存储服务工具指南**》。
3. 登录OBS Browser+。OBS Browser+支持AK方式登录，以及授权码登录两种登录方式。登录步骤请参考《**对象存储服务工具指南**》。
4. 通过OBS Browser+上传数据。

   在OBS Browser+页面单击“创建桶”，按照要求选择“区域”和填写“桶名”（例如：DLI-demo），创建桶成功后，返回桶列表，单击桶DLI-demo。OBS Browser+提供强大的拖拽上传功能，您可以将本地的一个或多个文件或者文件夹拖拽到对象存储的对象列表或者并行文件系统的对象列表中；同时您也可以将文件或文件夹拖拽到指定的目录上，这样可以上传到指定的目录中。

   单击**Best_Practice_04.zip**获取本示例的测试数据，解压“Best_Practice_04.zip”压缩包，解压后将data文件夹上传到OBS桶根目录下。测试数据目录说明如下：

   - user表数据：data/JData_User
   - product表数据：data/JData_Product
   - comment表数据：data/JData_Product/JData_Comment
   - action表数据：data/JData_Action

步骤 2：分析数据

1. 创建数据库、表

   a. 在portal页面上方菜单栏中单击“产品”，单击“大数据”分类中的“数据湖探索 DLI”。

   b. 创建demo数据库，在DLI控制台总览页面，选择“作业管理 > SQL作业”，单击“创建作业”，进入SQL作业编辑器。

c. 在SQL作业编辑器左侧，选择“数据库”页签，单击创建demo数据库，请参见图4-1。

图 4-1 创建数据库

说明

“default”为内置数据库，不能创建名为“default”的数据库。

d. 选择demo数据库，在编辑框中输入以下SQL语句：

```
create table user(
  user_id int,
  age int,
  gender int,
  rank int,
  register_time string
) USING csv OPTIONS (path "obs://DLI-demo/data/JData_User")
```

说明

上述SQL语句中的文件路径为实际存放数据的OBS路径。

e. 单击“执行”，创建用户信息表user。

f. 用相同的方法创建商品表，评价表，行为表。

- 商品表

```
create table product(
  product_id int,
  a1 int,
  a2 int,
  a3 int,
  category int,
  brand int
) USING csv OPTIONS (path "obs://DLI-demo/data/JData_Product")
```


- 评价表

```
create table comment(
  deadline string,
  product_id int,
  comment_num int,
  has_bad_comment int,
  bad_comment_rate float
) USING csv OPTIONS (path "obs://DLI-demo/data/JData_Comment")
```

- 行为表

```
create table action(
  user_id int,
  product_id int,
  time string,
  model_id string,
  type string
) USING csv OPTIONS (path "obs://DLI-demo/data/JData_Action");
```

2. 执行查询

   常用查询语句可以设置为模板，下次查询的时候在模板管理页面可以查看，具体操作可参见《数据湖探索用户指南》中的**《模板管理》**。

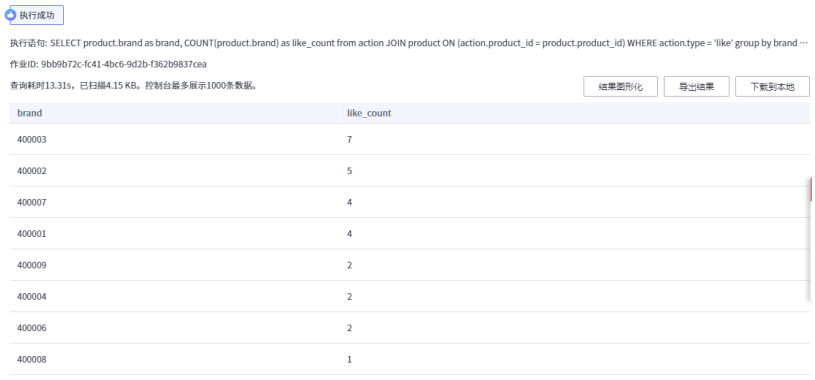
   - 分析出10大用户点赞数最多的产品

     i. 执行以下SQL语句，可以分析出10大用户点赞数最多的产品。

```
SELECT
  product.brand as brand,
  COUNT(product.brand) as like_count
from
  action
  JOIN product ON (action.product_id = product.product_id)
WHERE
  action.type = 'like'
group by
  brand
ORDER BY like_count desc
limit
  10
```

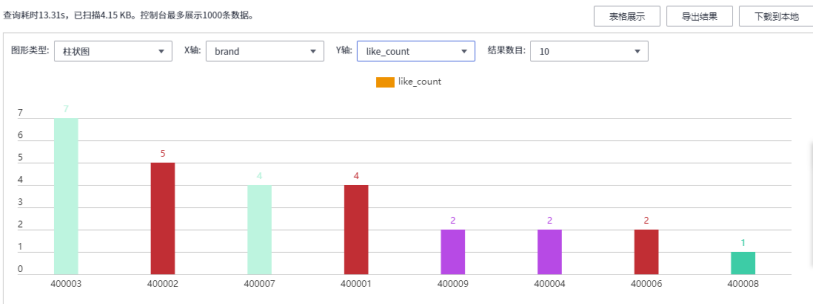
     ii. 单击“执行”，运行结果如**图4-2**所示：

图 4-2 查询结果



     iii. 单击  “结果图形化”，对结果进行图形展示：

图 4-3 结果图形化



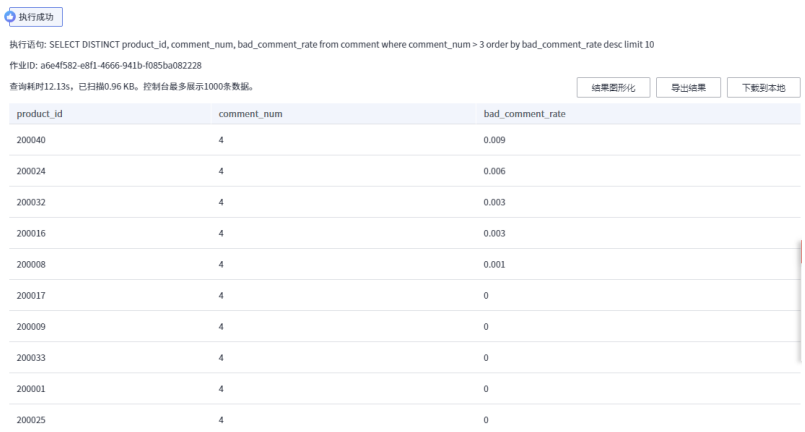
- 分析出10大评级最差的商品

  i. 执行以下SQL语句，可以分析出10大评级最差的商品。

```
SELECT
  DISTINCT product_id,
  comment_num,
  bad_comment_rate
from
  comment
where
  comment_num > 3
order by
  bad_comment_rate desc
limit
  10
```

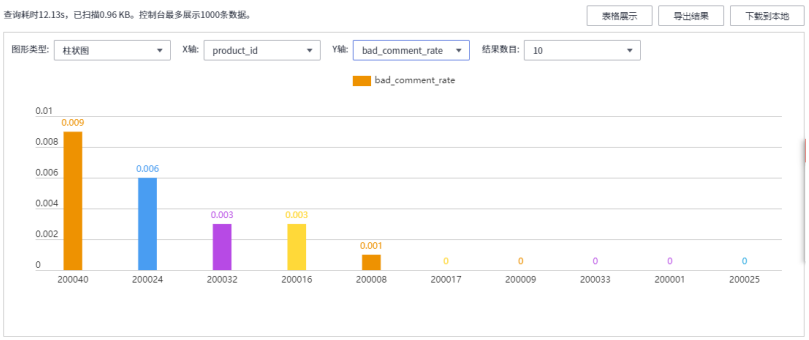
  ii. 单击“执行”，运行结果如图4-4所示：

图 4-4 查询结果



  iii. 单击  “结果图形化”，对结果进行图形展示：

图 4-5 结果图形化



此外，还可以分析用户的年龄分布、性别比例、商品评价情况、购买情况、浏览情况等。

5 使用 DLI 分析账单消费数据

应用场景

本文主要介绍如何使用华为云DLI上的实际消费数据（文中涉及账户的信息已脱敏），在DLI的大数据分析平台上进行分析，找出费用优化的空间，并给出使用DLI过程中降低成本的一些优化措施。

流程介绍

使用DLI进行账单分析与优化的操作过程主要包括以下步骤：

步骤1：获取消费数据。获取账户的实际消费数据。

步骤2：分析账户消费结构并优化。在DLI上分析账户消费结构，找出开支较大的资源或用户，并给出降底成本的优化措施。

资源和成本规划

表 5-1 资源和成本规划

资源	资源说明	成本说明
DLI	数据湖探索（DLI）作为华为云大数据分析平台，其计费项包括存储费用与计算费用两项，计费类型包括包周期（包年包月），套餐包和按需计费三种。	DLI目前支持三种作业：SQL作业，Flink作业和Spark作业。 SQL作业的计费包括存储计费和计算计费，其中计算计费有包年包月计费和按需计费两种。 ● 包年包月计费根据购买周期进行扣费，推荐使用包年包月模式，价格优惠且在周期内独享计算资源。 ● 按需计费以小时为单位进行扣费。按需计费又分为按CU时计费和按扫描数据量计费，这两种计费方式是互斥的，可根据需要选择其中一种。建议优先选择按CU时计费，可资源独享，且成本核算清晰。同时，按CU时计费还提供套餐包的购买和使用。 – CU时资费=CU数*使用时长*单价。使用时长按自然小时计费，不足一个小时按一个小时计费。 – 扫描数据量资费=执行SQL时产生的扫描数据量*单价。如果计算任务超时或失败，则本次计算不收取费用。 ● Flink作业和Spark作业的计费只有计算计费，具体计费规则与SQL作业相同。 具体计费规则可以参考华为云官网**价格详情**。 。

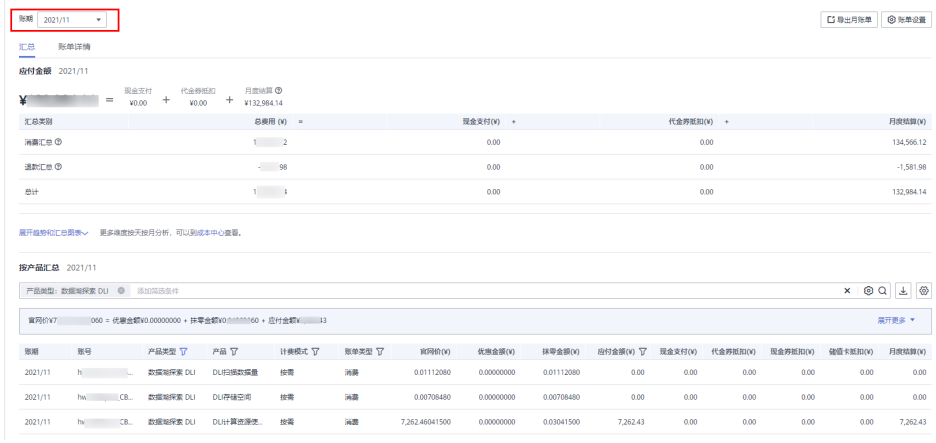
步骤 1：获取消费数据

1. 获取消费明细数据。
   a. 使用华为云账户登录控制台。
   b. 通过“费用中心”>“费用账单”进入费用中心。

图 5-1 费用账单

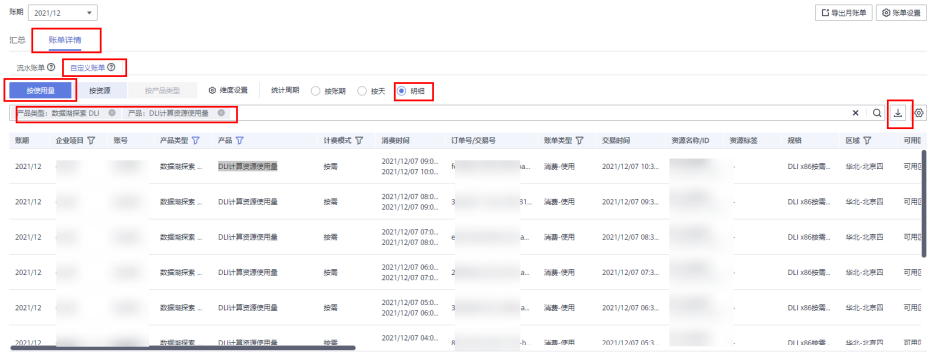
c. 在“费用账单”界面，选择对应的“账期”，在“按产品汇总”下的搜索框下，选择“产品类型 > 数据湖探索DLI”。在消费汇总中可以发现DLI计算资源使用量消费最多。

图 5-2 费用汇总



d. 单击“账单详情 > 自定义账单”，单击“维度设置”，选择“按使用量”。“统计周期”选择“明细”。在显示数据的标题行，“产品类型”选择“数据湖探索 DLI”，“产品”选择“DLI计算资源使用量”，单击“导出账单”。

图 5-3 导出消费数据



e. 左侧导航栏，选择“导出记录”。下载对应的消费明细数据。

步骤 2：分析账户消费结构并优化

1. 在DLI上进行消费明细分析。
   a. 将1下载的消费明细数据上传到已建好的OBS桶中。
   b. 在数据湖探索服务中创建表。
      i. 登录DLI控制台，左侧导航栏单击“SQL编辑器”，执行引擎选择“spark”，选择执行的队列和数据库。本次演示队列和数据库选择“default”。
      ii. 下载的文件中包含时间用量等，按表头意义在DLI上创建表，具体可以参考如下示例，其中amount列为费用。

```
CREATE TABLE `spending` (
 account_period string,
 EnterpriseProject string,
 EnterpriseProjectID string,
 accountID string,
 product_type_code string,
 product_type string,
 product_code string,
 product_name string,
 product_id string,
 mode string,
 time1 string,
 use_start string,
 use_end string,
 orderid string,
 ordertime string,
 resource_type string,
 resource_id string,
 resouce_name string,
 tag string,
 skuid string,
 `c22name` STRING,
 `c23name` STRING,
 `c24name` STRING,
 `c25name` STRING,
 `c26name` STRING,
 `c27name` STRING,
 `c28name` STRING,
 `c29name` STRING,
 size STRING,
 `c31name` STRING,
 `c32name` STRING,
 `c33name` STRING,
 `c34name` STRING,
 `c35name` STRING,
 `amount` STRING,
```



```
  `c37name` STRING,
  `c38name` STRING,
  `c39name` STRING,
  `c40name` STRING,
  `c41name` STRING,
  `c42name` STRING,
  `c43name` STRING,
  `c44name` STRING,
  `c45name` STRING,
  `c46name` STRING,
  `c47name` STRING,
  `c48name` STRING,
  `c49name` STRING,
  `c50name` STRING,
  `c51name` STRING,
  `c52name` STRING,
  `c53name` STRING,
  `c54name` STRING
) USING csv options (
  path 'obs://xxx/Spendings(ByTransaction)_20200501_20200531.csv',
  header true)
```

c. 查询该时间内消费最高的resource_id, resource_name。

通过以下语句，可以发现sql和flink队列使用的费用均为1842元，在总费用3754元中占比为98%。

```
select resource_id, resouce_name, sum(size)
  as usage, sum(amount)
  as sum_amount
  from spending
  group by resource_id, resouce_name
  order by sum_amount desc
```

图 5-4 查询结果

resource_id	resouce_name	usage	sum_amount
d91d4616-b10c-471a-820d-e676e6c5f4b4	sql	5264	1842.3999999999896
8163cc27-89ce-4bac-aa85-38cb753ee425	flink	5264	1842.3999999999896
9bd07369-f8ca-4bfb-b3e7-0e391ef7dd8b	null	48	14.399999999999999
dd3a12ff-c0af-4ad1-bbc1-858bf4d3661c	dlitest	32	11.2
f8265ef5-eb5f-4eff-b8d6-9ca91ed20009	test	16	5.6

d. 使用以下语句具体分析sql和flink这两个资源消费的时间段。

```
select * from spending where resource_id = 'd91d4616-b10c-471a-820d-e676e6c5f4b4' order by ordertime
```

可以发现sql队列从2020-05-14 17:00:00 GMT+08:00开始，每小时产生5.6元费用，持续到2020-05-28 10:00:00 GMT+08:00，说明这个sql队列在这段时间内持续使用。

同样，也可以发现flink队列在2020-05-14 17:00:00 GMT+08:00到2020-05-28 10:00:00 GMT+08:00这段 内持续使用。

2. 优化建议。

通过以上分析，了解到sql和flink这两个队列几乎是在持续使用的，建议通过购买包周期队列来降低使用成本。另外，对于明确需要使用多少CU时的作业，也可以提前购买对应的CU时套餐包，来降低使用成本。

企业中的业务模式较多且经常变化，成本管理员通常并不能全面及时了解花销较大的业务在哪里，哪些是合理的，哪些是不合理的，通过在DLI中对费用明细进行分析，可以及时发现企业花销不合理的地方，及时进行成本管理，进一步降低企业使用华为云的成本。

6 使用 DLI 分析电商实时业务数据

应用场景

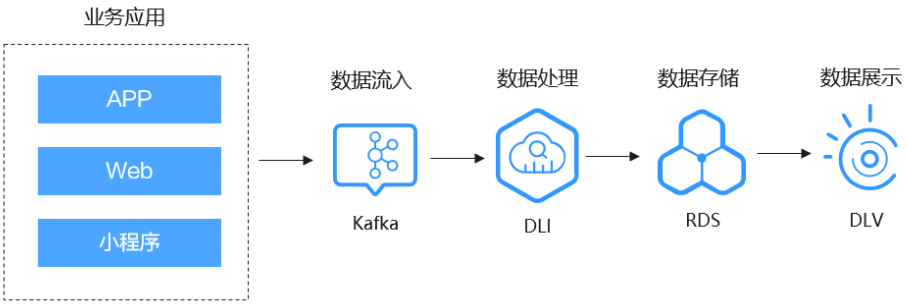
当前线上购物无疑是最火热的购物方式，而电商平台则又可以以多种方式接入，例如通过web方式访问、通过app的方式访问、通过微信小程序的方式访问等等。而电商平台则需要每天统计各平台的实时访问数据量、订单数、访问人数等等指标，从而能在显示大屏上实时展示相关数据，方便及时了解数据变化，有针对性地调整营销策略。而如何高效快捷地统计这些指标呢?

假设平台已经将每个商品的订单信息实时写入Kafka中，这些信息包括订单ID、订单生成的渠道(即web方式、app方式等)、订单时间、订单金额、折扣后实际支付金额、支付时间、用户ID、用户姓名、订单地区ID等信息。而我们需要做的，就是根据当前可以获取到的业务数据，实时统计每种渠道的相关指标，输出存储到数据库中，并进行大屏展示。

方案架构

使用DLI Flink完成电商业务实时数据的分析处理，获取各个渠道的销售汇总数据。

图 6-1 方案简介



流程指导

使用DLI Flink进行电商实时业务数据分析的操作过程主要包括以下步骤：

步骤1：创建资源。在您的账户下创建作业需要的相关资源，涉及VPC、DMS、DLI、RDS。

步骤2：获取DMS连接地址并创建Topic。获取DMS Kafka实例连接地址并创建DMS Topic。

步骤3：创建RDS数据库表。获取RDS实例内网地址，登录RDS实例创建RDS数据库及MySQL表。

步骤4：创建DLI增强型跨源。创建DLI增强型跨源，并测试队列与RDS、DMS实例连通性。

步骤5：创建并提交Flink作业。创建DLI Flink OpenSource SQL作业并运行。

步骤6：查询结果。查询Flink作业结果，使用DLV进行大屏展示。

方案优势

- 跨源分析：数据免搬迁，就可以关联分析存在OBS中的各个渠道的销售汇总数据。
- 纯SQL操作：DLI已对接多个数据源，直接通过SQL建表就可以完成数据源的映射。

资源和成本规划

表 6-1 资源和成本规划

资源	资源说明	成本说明
OBS	需要创建一个OBS桶将数据上传到对象存储服务OBS，为后面使用DLI完成数据分析做准备。	OBS的使用涉及以下几项费用： • 存储费用：静态网站文件存储在OBS中产生的**存储费用**。 • 请求费用：用户访问OBS中存储的静态网站文件时产生的**请求费用**。 • 流量费用：用户使用自定义域名通过公网访问OBS时产生的**流量费用**。 实际产生的费用与存储的文件大小、用户访问所产生的请求次数和流量大小有关，请根据自己的业务进行预估。
DLI	在创建SQL作业前需购买队列，使用DLI的队列资源时，按照队列CU时进行计费。	如购买按需计费的队列，在使用队列资源时，按照队列CU时进行计费。 以小时为单位进行结算。不足一小时按一小时计费，小时数按整点计算。队列CU时按需计费的计算费用=单价*CU数*小时数。
VPC	VPC丰富的功能帮助您灵活管理云上网络，包括创建子网、设置安全组和网络ACL、管理路由表、申请弹性公网IP和带宽等。	VPC本身不收取费用。 但如有互联网访问需求，您需要购买弹性公网IP。弹性公网IP提供“包年/包月”和“按需计费”两种计费模式。 了解**VPC计费说明**。

资源	资源说明	成本说明
DMS Kafka	Kafka提供的消息队列服务，向用户提供计算、存储和带宽资源独占式的Kafka专享实例。	Kafka版支持按需和包周期两种付费模式。Kafka计费项包括Kafka实例和Kafka的磁盘存储空间。 了解**Kafka计费说明**。
RDS MySQL	数据库 RDS for MySQL提供在线云数据库服务。	RDS对您选择的数据库实例、数据库存储和备份存储（可选）收费。 了解**RDS计费说明**。
DLV	DLV适配云上云下多种数据源，提供丰富多样的可视化组件，快速定制数据大屏。	使用DLV服务的费用主要是DLV包年包月套餐的费用，您可以根据实际使用情况，选择合适的版本规格。 了解DLI**产品价格详情**。

数据说明

- 数据源表：电商业务订单详情宽表

字段名	字段类型	说明
order_id	*string*	订单ID
order_channel	*string*	订单生成的渠道(即web方式、app方式等)
order_time	*string*	订单时间
pay_amount	*double*	订单金额
real_pay	*double*	实际支付金额
pay_time	*string*	支付时间
user_id	*string*	用户ID
user_name	*string*	用户姓名
area_id	*string*	订单地区ID

- 结果表：各渠道的销售总额实时统计表。

字段名	字段类型	说明
begin_time	*varchar(32)*	开始统计指标的时间
channel_code	*varchar(32)*	渠道编号
channel_name	*varchar(32)*	渠道名
cur_gmv	*double*	当天GMV

字段名	字段类型	说明
cur_order_user_count	*bigint*	当天付款人数
cur_order_count	*bigint*	当天付款订单数
last_pay_time	*varchar(32)*	最近结算时间
flink_current_time	*varchar(32)*	Flink数据处理时间

步骤 1：创建资源

如表6-2所示，完成VPC、DMS、RDS、DLI、DLV资源的创建。

表 6-2 创建资源

资源类型	说明	操作指导
VPC	VPC为资源提供云上的网络管理服务。 **资源网络规划说明：** ● Kafka与MySQL实例指定的VPC需为同一VPC。 ● Kafka与MySQL实例所属VPC网段不得与创建的DLI队列网段冲突。	创建VPC和子网
DMS Kafka	本例中以DMS Kafka实例作为数据源。	DMS Kafka入门指引
RDS MySQL	本例中以使用RDS提供在线云数据库服务。	购买RDS for MySQL实例
DLI	DLI提供实时业务数据分析。 创建DLI队列时请创建“包年包月”或者“按需-专属资源”模式的通用队列，否则无法创建增强型网络连接。	DLI 创建队列
DLV	DLV实时展现DLI队列处理后的结果数据。	DLV 创建大屏

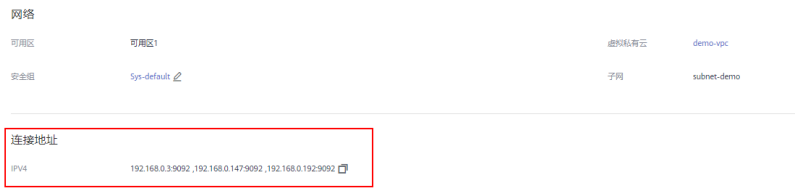
步骤 2：获取 DMS 连接地址并创建 Topic

1. 在控制台单击“服务列表”，选择“分布式消息服务DMS”，单击进入DMS服务控制台页面。在“Kafka专享版”页面找到您所创建的Kafka实例。

图 6-2 Kafka 实例

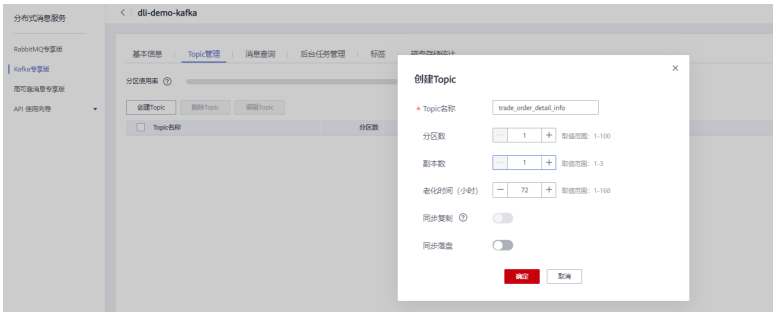
2. 进入实例详情页面。单击“基本信息”，获取“连接地址”。

图 6-3 获取连接地址



3. 单击“Topic管理”，创建一个Topic：trade_order_detail_info。

图 6-4 创建 Topic



Topic配置如下：

- 分区数：1
- 副本数：1
- 老化时间：72h
- 同步落盘：否

步骤 3：创建 RDS 数据库表

1. 在控制台单击“服务列表”，选择“云数据库RDS”，单击进入RDS页面。在“实例管理页面”，找到您已经创建的RDS实例，获取其内网地址。

图 6-5 内网地址

2. 单击所创建RDS实例的“登录”，跳转至“数据管理服务-DAS”。输入相关账户信息，单击“测试连接”。显示连接成功后，单击“登录”，进入“实例登录”页面。

图 6-6 实例登录

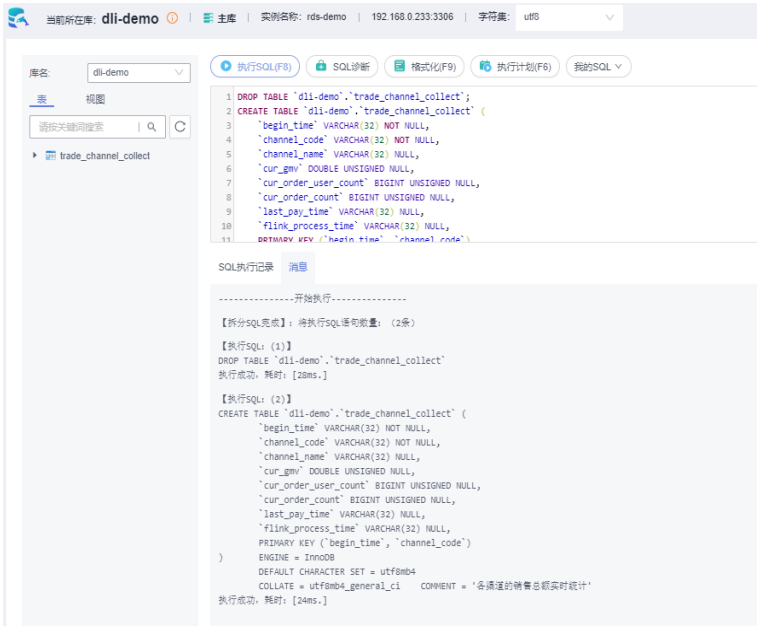
3. 登录RDS实例后，单击“新建数据库”，创建名称为“dli-demo”的数据库。

图 6-7 创建数据库

4. 单击“SQL操作”>“SQL查询”，执行如下SQL创建测试用MySQL表，表相关字段含义在**•数据说明**中有详细介绍。

```
DROP TABLE `dli-demo`.`trade_channel_collect`;
CREATE TABLE `dli-demo`.`trade_channel_collect` (
    `begin_time` VARCHAR(32) NOT NULL,
    `channel_code` VARCHAR(32) NOT NULL,
    `channel_name` VARCHAR(32) NULL,
    `cur_gmv` DOUBLE UNSIGNED NULL,
    `cur_order_user_count` BIGINT UNSIGNED NULL,
    `cur_order_count` BIGINT UNSIGNED NULL,
    `last_pay_time` VARCHAR(32) NULL,
    `flink_current_time` VARCHAR(32) NULL,
    PRIMARY KEY (`begin_time`, `channel_code`)
)   ENGINE = InnoDB
    DEFAULT CHARACTER SET = utf8mb4
    COLLATE = utf8mb4_general_ci
    COMMENT = '各渠道的销售总额实时统计';
```


图 6-8 创建表



步骤 4：创建 DLI 增强型跨源

1. 在控制台单击“服务列表”，选择“数据湖探索”，单击进入DLI服务页面。单击“资源管理 > 队列管理”，查询创建的DLI队列。

图 6-9 队列列表

2. 单击“全局配置 > 服务授权”，选中“VPC Administrator”，单击“更新委托权限”，赋予DLI操作用户VPC资源的权限，用于创建VPC的“对等连接”。

图 6-10 更新委托权限

3. 单击“跨源连接 > 增强型跨源 > 创建”，配置如下连接信息后单击“确定”。
   - 连接名称：增强型跨源名称。
   - 弹性资源池：选择您所创建的通用队列。
   - 虚拟私有云：选择 Kafka 与 MySQL 实例所在的VPC。
   - 子网：选择 Kafka 与 MySQL 实例所在的子网。

图 6-11 创建增强型跨源

增强型跨源创建完成后，在跨源列表中，对应的跨源连接状态会显示为“已激活”。

单击跨源连接的名称，详情页面显示连接状态为“ACTIVE”。

图 6-12 跨源连接状态

图 6-13 详情

4. 测试队列与RDS、DMS实例连通性。

   a. 单击“队列管理”，选择您所使用的队列，单击“操作”列中的“更多”>“测试地址连通性”。

图 6-14 检测地址连通性

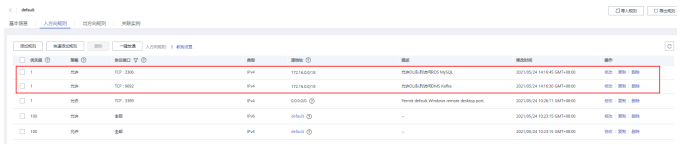
b. 输入DMS Kafka实例连接地址和步RDS MySQL实例内网地址，进行网络连通性测试。

测试结果显示可达，则DLI队列与Kafka、MySQL实例的网络已经联通。

图 6-15 测试结果

如果测试结果不可达，需要修改实例所在VPC的安全组规则，放开9092、3306端口对DLI队列的限制，DLI队列网段信息可以在队列的详情页中获取。

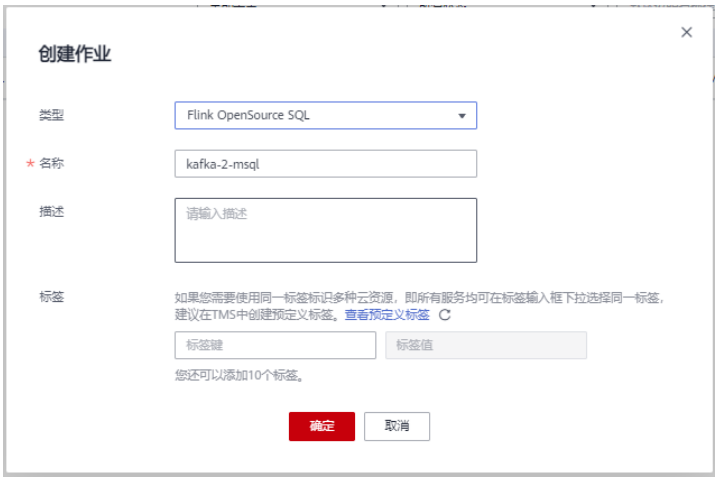
图 6-16 安全组规则



步骤 5：创建并提交 Flink 作业

1. 单击DLI控制台左侧“作业管理”，选择“Flink作业”。单击“创建作业”。
   - 类型：选择作业类型为：Flink OpenSource SQL。
   - 名称：自定义。

图 6-17 创建 Flink 作业



2. 单击“确定”，进入作业编辑作业页面，具体SQL示例如下，部分参数值需要根据RDS和DMS对应的信息进行修改。

```
--********************************************************************--
-- 数据源：trade_order_detail_info (订单详情宽表)
--********************************************************************--
create table trade_order_detail (
  order_id string,      -- 订单ID
  order_channel string,   -- 渠道
  order_time string,     -- 订单创建时间
  pay_amount double,     -- 订单金额
  real_pay double,      -- 实际付费金额
  pay_time string,      -- 付费时间
  user_id string,      -- 用户ID
  user_name string,     -- 用户名
  area_id string       -- 地区ID
) with (
  "connector.type" = "kafka",
  "connector.version" = "0.10",
  "connector.properties.bootstrap.servers" = "xxxx:9092,xxxx:9092,xxxx:9092", -- Kafka连接地址
  "connector.properties.group.id" = "trade_order",   -- Kafka groupID
  "connector.topic" = "trade_order_detail_info",     -- Kafka topic
  "format.type" = "json",
  "connector.startup-mode" = "latest-offset"
);

--********************************************************************--
-- 结果表：trade_channel_collect (各渠道的销售总额实时统计)
--********************************************************************--
create table trade_channel_collect(
  begin_time string,       --统计数据的开始时间
  channel_code string,      -- 渠道编号
  channel_name string,      -- 渠道名
  cur_gmv double,         -- 当天GMV
  cur_order_user_count bigint, -- 当天付款人数
  cur_order_count bigint,    -- 当天付款订单数
  last_pay_time string,     -- 最近结算时间
  flink_current_time string,
  primary key (begin_time, channel_code) not enforced
) with (
  "connector.type" = "jdbc",
  "connector.url" = "jdbc:mysql://xxxx:3306/xxxx",    -- mysql连接地址，jdbc格式
  "connector.table" = "xxxx",             -- mysql表名
  "connector.driver" = "com.mysql.jdbc.Driver",
  'pwd_auth_name'= '***xxxxx***', --DLI侧创建的Password类型的跨源认证名称。使用跨源认证则无需在作业中配置账号和密码。
  "connector.write.flush.max-rows" = "1000",
  "connector.write.flush.interval" = "1s"
);
```



```
--********************************************************************--
-- 临时中间表
--********************************************************************--
create view tmp_order_detail
as
select *
    , case when t.order_channel not in ("webShop", "appShop", "miniAppShop") then "other"
           else t.order_channel end as channel_code --重新定义统计渠道 只有四个枚举值[webShop、
appShop、miniAppShop、other]
    , case when t.order_channel = "webShop" then _UTF16"网页商城"
           when t.order_channel = "appShop" then _UTF16"app商城"
           when t.order_channel = "miniAppShop" then _UTF16"小程序商城"
           else _UTF16"其他" end as channel_name --渠道名称
from (
    select *
        , row_number() over(partition by order_id order by order_time desc ) as rn --去除重复订单数据
        , concat(substr("2021-03-25 12:03:00", 1, 10), " 00:00:00") as begin_time
        , concat(substr("2021-03-25 12:03:00", 1, 10), " 23:59:59") as end_time
    from trade_order_detail
    where pay_time >= concat(substr("2021-03-25 12:03:00", 1, 10), " 00:00:00") --取今天数据，为了方
便运行，这里使用"2021-03-25 12:03:00"替代cast(LOCALTIMESTAMP as string)
    and real_pay is not null
) t
where t.rn = 1;

-- 按渠道统计各个指标
insert into trade_channel_collect
select
      begin_time  --统计数据的开始时间
    , channel_code
    , channel_name
    , cast(COALESCE(sum(real_pay), 0) as double) as cur_gmv --当天GMV
    , count(distinct user_id) as cur_order_user_count --当天付款人数
    , count(1) as cur_order_count --当天付款订单数
    , max(pay_time) as last_pay_time --最近结算时间
    , cast(LOCALTIMESTAMP as string) as flink_current_time --flink任务中的当前时间
from tmp_order_detail
where pay_time >= concat(substr("2021-03-25 12:03:00", 1, 10), " 00:00:00")
group by begin_time, channel_code, channel_name;
```

说明

作业逻辑说明如下：

1. 创建一个Kafka源表，用来从Kafka指定Topic中读取消费数据；
2. 创建一个结果表，用来通过JDBC向MySQL中写入结果数据。
3. 实现相应的处理逻辑，以实现各个指标的统计。

   为了简化最终的处理逻辑，使用创建视图进行数据预处理。

   1. 利用over窗口条件和过滤条件结合以去除重复数据（该方式是利用了top N的方法），同时利用相应的内置函数concat和substr将当天的00:00:00作为统计的开始时间，当天的23:59:59作为统计结束时间，并筛选出支付时间在当天凌晨00:00:00后的订单数据进行统计（**为了方便模拟数据的构造，这里使用"2021-03-25 12:03:00"替代cast(LOCALTIMESTAMP as string)**）。
   2. 根据这些数据的订单渠道利用内置的条件函数设置channel_code和channel_name的值，从而获取了源表中的字段信息，以及begin_time、end_time和channel_code、channel_name的值。
4. 根据需要对相应指标进行统计和筛选，并将结果写入到结果表中。

3. 选择所创建的DLI通用队列提交作业。

图 6-18 Flink Opensource SQL 作业

4. 等待作业状态会变为“运行中”，单击作业名称，可以查看作业详细运行情况。

图 6-19 作业运行状态

5. 使用Kafka客户端向指定topic发送数据，模拟实时数据流。
   具体方法请参考**DMS-连接实例生产消费信息**。

图 6-20 模拟实时数据流

6. 发送命令如下：

```
sh kafka_2.11-2.3.0/bin/kafka-console-producer.sh --broker-list KafKa连接地址 --topic Topic名称
```

示例数据如下：

```
{"order_id":"202103241000000001", "order_channel":"webShop", "order_time":"2021-03-24 10:00:00", "pay_amount":"100.00", "real_pay":"100.00", "pay_time":"2021-03-24 10:02:03", "user_id":"0001", "user_name":"Alice", "area_id":"330106"}
{"order_id":"20210324160606001", "order_channel":"appShop", "order_time":"2021-03-24 16:06:06", "pay_amount":"200.00", "real_pay":"180.00", "pay_time":"2021-03-24 16:10:06", "user_id":"0001", "user_name":"Alice", "area_id":"330106"}
{"order_id":"20210325120202001", "order_channel":"miniAppShop", "order_time":"2021-03-25 12:02:02", "pay_amount":"60.00", "real_pay":"60.00", "pay_time":"2021-03-25 12:03:00", "user_id":"0002", "user_name":"Bob", "area_id":"330110"}
{"order_id":"20210325150505001", "order_channel":"qqShop", "order_time":"2021-03-25 15:05:05", "pay_amount":"500.00", "real_pay":"400.00", "pay_time":"2021-03-25 15:10:00", "user_id":"0003", "user_name":"Cindy", "area_id":"330108"}
{"order_id":"20210325202020001", "order_channel":"webShop", "order_time":"2021-03-24 20:20:20", "pay_amount":"600.00", "real_pay":"480.00", "pay_time":"2021-03-25 00:00:00", "user_id":"0004", "user_name":"Daisy", "area_id":"330102"}
{"order_id":"20210326080808001", "order_channel":"webShop", "order_time":"2021-03-25 08:08:08", "pay_amount":"300.00", "real_pay":"240.00", "pay_time":"2021-03-25 08:10:00", "user_id":"0004", "user_name":"Daisy", "area_id":"330102"}
{"order_id":"20210326131130001", "order_channel":"webShop", "order_time":"2021-03-25 13:13:13", "pay_amount":"100.00", "real_pay":"100.00", "pay_time":"2021-03-25 16:16:16", "user_id":"0004", "user_name":"Daisy", "area_id":"330102"}
{"order_id":"20210327060606001", "order_channel":"appShop", "order_time":"2021-03-25 06:06:06", "pay_amount":"50.50", "real_pay":"50.50", "pay_time":"2021-03-25 06:07:00", "user_id":"0001", "user_name":"Alice", "area_id":"330106"}
{"order_id":"20210327060606002", "order_channel":"webShop", "order_time":"2021-03-25 06:06:06", "pay_amount":"66.60", "real_pay":"66.60", "pay_time":"2021-03-25 06:07:00", "user_id":"0002", "user_name":"Bob", "area_id":"330110"}
{"order_id":"20210327060606003", "order_channel":"miniAppShop", "order_time":"2021-03-25 06:06:06", "pay_amount":"88.80", "real_pay":"88.80", "pay_time":"2021-03-25 06:07:00", "user_id":"0003", "user_name":"Cindy", "area_id":"330108"}
{"order_id":"20210327060606004", "order_channel":"webShop", "order_time":"2021-03-25 06:06:06", "pay_amount":"99.90", "real_pay":"99.90", "pay_time":"2021-03-25 06:07:00", "user_id":"0004", "user_name":"Daisy", "area_id":"330102"}
```

7. 单击DLI控制台左侧“作业管理”>“Flink作业”，单击**3**提交的Flink作业。在作业详情页面，可以看到处理的数据记录数。

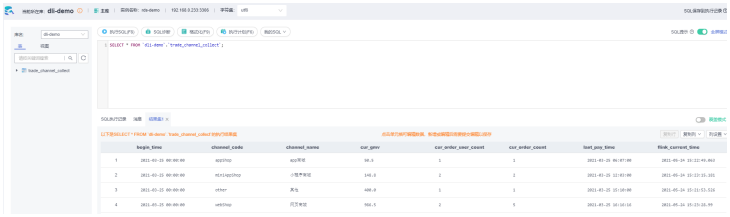
图 6-21 Flink 作业详情

步骤 6：查询结果

1. 参考2，登录MySQL实例，执行如下SQL语句，即可查询到经过Flink作业处理后的结果数据。

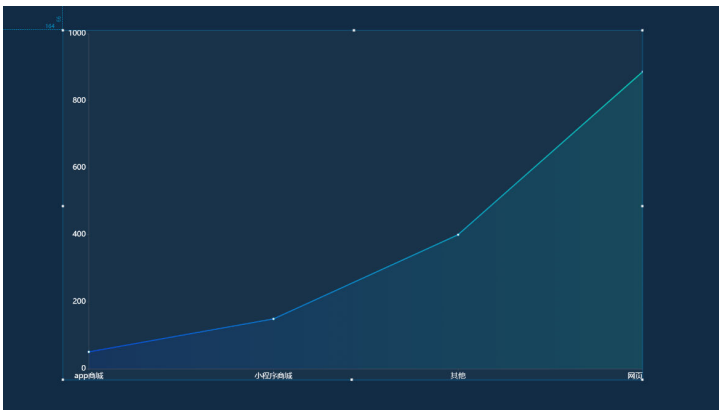
```
SELECT * FROM `dli-demo`.`trade_channel_collect`;
```

图 6-22 查询结果



2. 配置DLV大屏，执行SQL查询RDS MySQL，即可以实现大屏实时展示。具体配置方法可参考**DLV开发大屏**。

图 6-23 大屏展示



7 使用 BI 工具连接 DLI 分析数据

7.1 BI 工具连接 DLI 方案概述

BI工具是数据分析的强大助手，提供数据可视化、报表生成和仪表板创建等功能。

DLI服务通过对数据的融合分析处理，可以为BI工具提供标准的、有效的高质量数据，供给后续的数据统计分析使用。

通过连接到DLI，BI工具可以更加灵活的使用DLI访问和分析数据，帮助企业快速做出基于数据的决策。

DLI为BI工具提供了便捷的连接方法：

- DBeaver、DBT和YongHong BI，可以直接通过DLI提供的驱动连接到DLI。这简化了配置过程，使得用户能够直接利用这些工具的强大功能。
- PowerBI、Fine BI、SuperSet、Tableau和Beeline，它们可以通过Kyuubi建立与DLI的连接。Kyuubi是一个分布式SQL查询引擎，提供了标准的SQL接口，使得BI工具能够通过Kyuubi与DLI进行交互，执行数据查询和分析。

说明

BI工具连接DLI的方案中使用了DLI SDK V2。

- 2024年5月起，新用户可以直接使用DLI SDK V2，无需开通白名单。
- 对于2024年5月之前开通并使用DLI服务的用户，如需使用“DLI SDK V2”功能，必须提交工单申请加入白名单。

7.2 配置 DBeaver 连接 DLI 进行数据查询和分析

DBeaver 是一个免费且开源的数据库管理工具，支持多种数据库，通过DBeaver这款可视化数据库管理工具可以查看数据库结构、执行SQL查询和脚本、浏览和导出数据等。本节操作介绍DBeaver连接DLI服务的操作步骤。

操作前准备

- **工具包**
  - **DLI的JDBC驱动：**

单击**dli-jdbc-x.x.x.jar**获取JDBC驱动，驱动名称：huaweicloud-dli-jdbc-xxx-dependencies.jar。

- **DBeaver客户端安装包：**

  DBeaver官网提供了针对不同操作系统的客户端安装包，单击**下载DBeaver**访问DBeaver官网下载系统对应的DBeaver客户端安装包并完成安装 。推荐使用24.0.3版本的DBeaver。

- **连接信息：**

  表 7-1 连接信息

类别	说明	获取方式
DLI AKSK	AK/SK认证就是使用AK/SK对请求进行签名，从而通过身份认证。	**获取AK/SK**
DLI Endpoint地址	地区与终端节点，即云服务在不同Region有不同的访问域名。	**获取EndPoint**
DLI所在的项目ID	项目编号，用于资源隔离。	**获取项目ID**
DLI区域信息	DLI所属区域信息	**地区和终端节点**

步骤 1：在 DBeaver 新建 DLI JDBC 驱动

1. 在**DBeaver**单击“数据库 > 驱动管理器”，创建新的驱动连接。

   使用驱动类加载DLI的JDBC驱动，请确保使用的jar包为huaweicloud-dli-jdbc-2.1.1-jar-with-dependencies.jar。

   图 7-1 新建驱动连接

2. 打开创建新驱动界面。
3. 在“设置”页输入驱动相关参数说明，单击“确定”创建驱动。

   驱动参数配置说明请参考**表7-2**。

图 7-2 编辑驱动连接

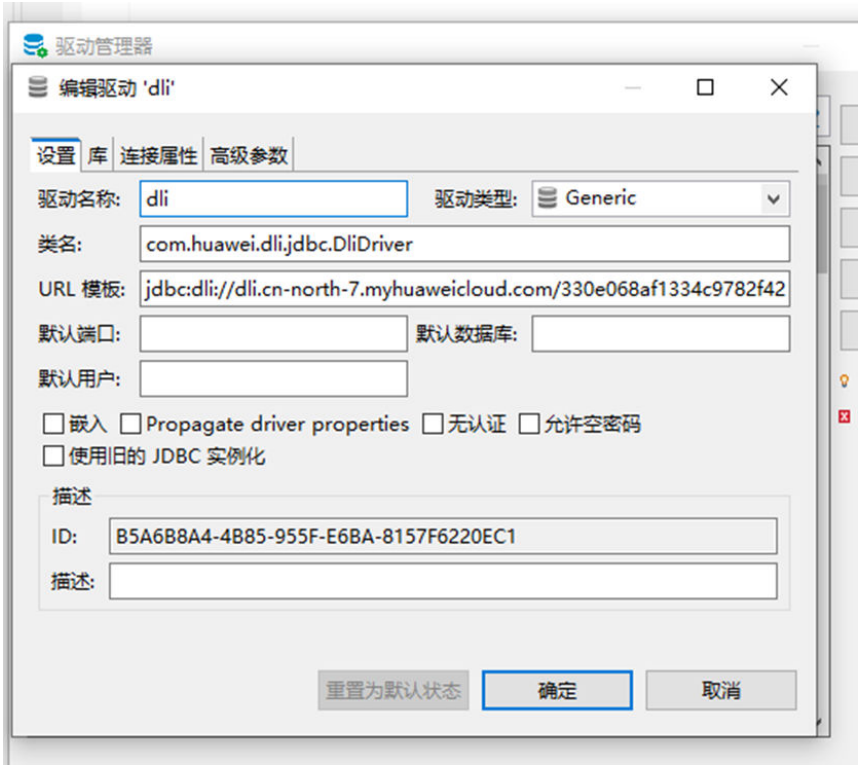


表 7-2 驱动相关参数说明

参数	说明
驱动名称	命名为便于识别的名称，例如GaussDB Driver。
驱动类型	驱动类型选择Generic。
类名	类名
URL模板	DLI JDBC驱动连接的格式： DLI JDBC驱动连接配置示例请参考**•DLIJDBC驱动连接的格式：**和**•DLIJDBC驱动连接配置示例：** 。 jdbc:dli://<endPoint>/projectId?<key1>=<val1>;<key2>=<val2>...
默认端口	需要连接的数据库端口。
Default Database	需要连接的数据库名。
Default User	账号名称。默认root。

- **DLI JDBC驱动连接的格式**
  jdbc:dli://<endPoint>/projectId?<key1>=<val1>;<key2>=<val2>...
  ? 后面接其他配置项，每个配置项以 key=value 的形式列出，配置项之间以；隔开。
- **DLI JDBC驱动连接配置示例**


```
jdbc:dli://dli.ap-southeast-2.myhuaweicloud.com/0b33ea2a7e0010802fe4c009bb05076d?databasename=tpch;queuename=auto;accesskey=XXXX;secretkey=XXXXX;regionname=ap-southeast-2;enginetype=trino;catalog=lfcatgalog
```

详细参数说明请参考**表7-3**和**表7-4**。

表 7-3 驱动连接配置信息参数说明

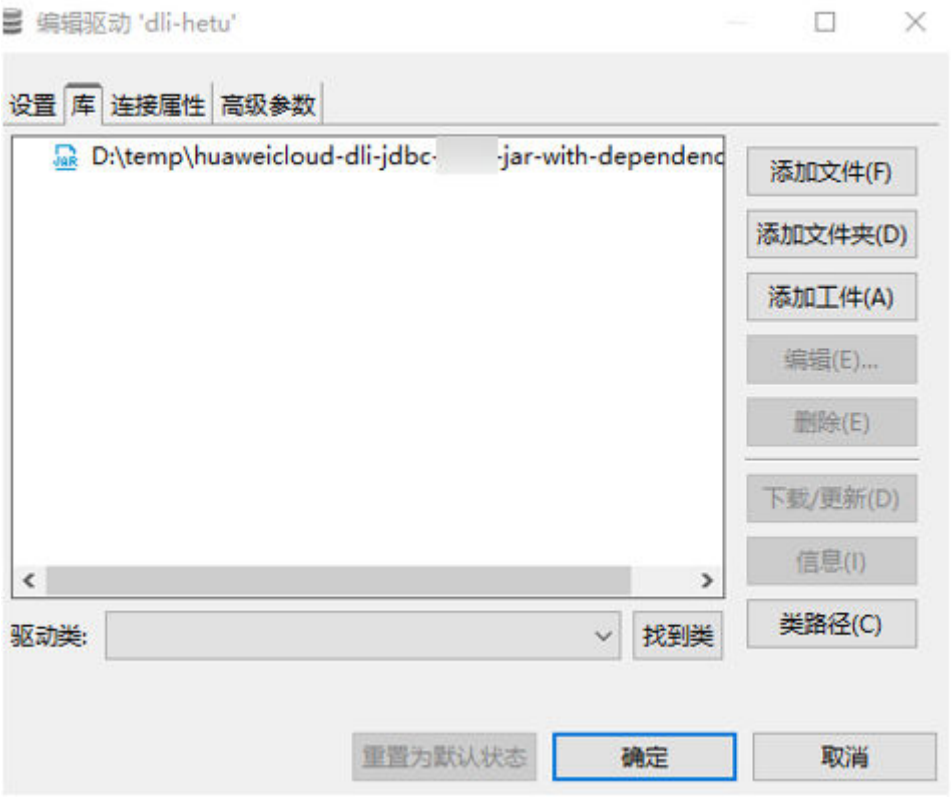
参数	说明	获取方式
endPoint	地区与终端节点，即云服务在不同Region有不同的访问域名。	**获取EndPoint**
projectId	DLI资源所在的项目ID。	**获取项目ID**
<key1>=<val1>	连接中? 后面接其他配置项，每个配置项以 key=value 的形式列出，配置项之间以；隔开。	请参考**表7-4**

表 7-4 key=value 参数说明

参数	说明	是否必选	示例
queuename	DLI服务的队列名称。	是	dli_test
databasename	数据库名称	是	tpch
accesskey和secretkey	AK/SK认证密钥。 如果使用AK/SK认证方式。	是	accesskey=your-access-key secretkey=your-secret-key
regionname	DLI的区域名称。 如果使用AK/SK认证方式时配置。	是	-
charset	JDBC编码方式。 默认为UTF-8。	否	-
enginetype	spark或trino 指spark队列或者hetu队列（默认使用spark）	否	trino
catalog	元数据catalog名称。 使用Lakeformation catgalog时必填，对应的Lakeformation catgalog名称。	否	lfcatalog

4. 在“库”页中，单击添加文件，添加1中的huaweicloud-dli-jdbc-xxx-dependencies.jar。

图 7-3 上传驱动

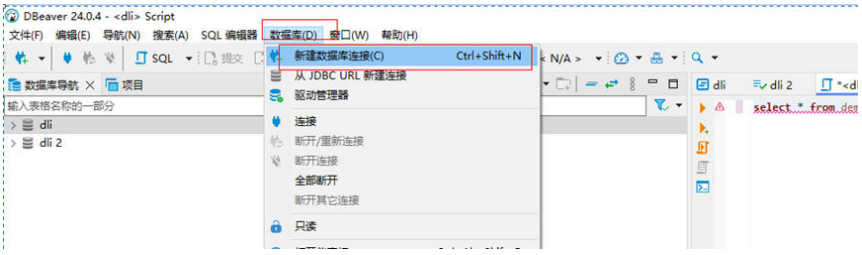


5. 添加后驱动类为空，需要单击“找到类”。识别出来的驱动类，需要与“设置”页的“类名”一致。
6. 单击“确定”，驱动设置完成。

步骤 2：测试连接数据库

1. 在DBeaver客户端单击“数据库 > 新建数据库连接”，选择**步骤1：在DBeaver新建DLI JDBC驱动**中创建的数据驱动。

图 7-4 新建数据库连接



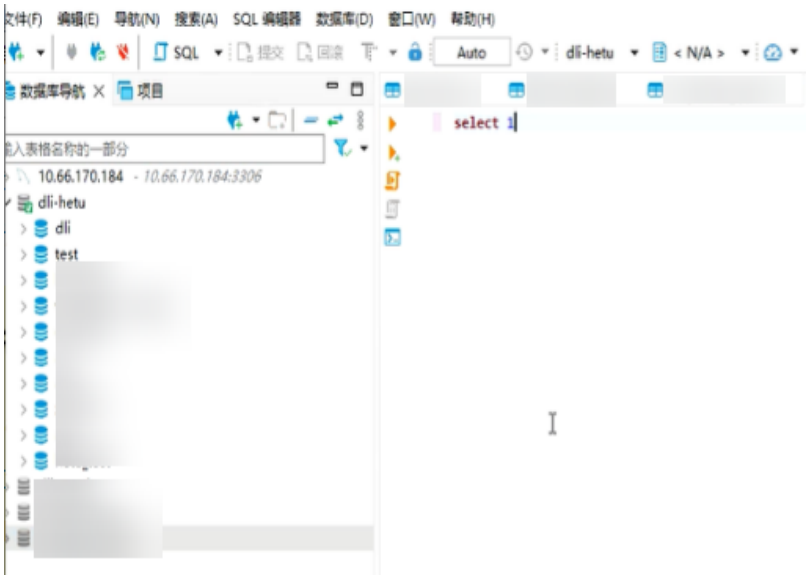
2. 单击“完成”，即可连接到DLI。在“数据库导航”栏可查看到连接的数据库信息。
3. 通过新建的连接即可对DLI执行后续的数据查询相关工作。

步骤 3：在 DBeaver 编写 SQL 查询

在DBeaver建立与DLI的连接后，即可在DBeaver编写SQL查询：

1. 您可以在左侧的数据库导航面板中选择数据库对象，然后在中间的查询编辑器中编写SQL语句。
2. 编写完查询后，可以单击工具栏上的“运行”按钮（通常是一个绿色的播放图标）来执行查询。
3. 查询执行后，结果将显示在查询编辑器下方的数据网格中。

图 7-5 在 DBeaver 编写 SQL 查询



7.3 配置 DBT 连接 DLI 进行数据调度和分析

DBT（Data Build Tool），是一款开源的数据建模和转换工具，运行在Python环境上。DBT连接DLI，用来定义和执行SQL转换，支持从数据集成、转换到分析的整个数据生命周期管理，适用于大规模数据分析项目和复杂的数据分析场景。

本节操作介绍DBT连接DLI的操作步骤。

操作前准备

- **环境要求**

  确保您的系统环境满足以下要求。
  - 操作系统：Windows 或 Linux
  - DBT是一个基于Python的工具，请确保已安装了Python。

    Python 版本：Python 3.8 或更高版本，推荐使用 Python 3.8
- **获取dli-dbt驱动包：**

  单击**dli-jdbc-x.x.x.jar**获取JDBC驱动，驱动名称：huaweicloud-dli-jdbc-xxx-dependencies.jar。
- **连接信息：**

表 7-5 连接信息

类别	说明	获取方式
DLI AKSK	AK/SK认证就是使用AK/SK对请求进行签名，从而通过身份认证。	**获取AK/SK**
DLI Endpoint地址	地区与终端节点，即云服务在不同Region有不同的访问域名。	**获取EndPoint**
DLI所在的项目ID	项目编号，用于资源隔离。	**获取项目ID**
DLI区域信息	DLI所属区域信息	**地区和终端节点**

步骤 1：部署 DBT 环境

1. **安装dbt-core**

   使用pip安装建议版本的dbt-core：

   ```
   pip install dbt-core==1.7.9
   ```

   说明

   pip是Python的包管理工具，通常与Python一起安装。

   如果尚未安装pip，可以通过Python内置的ensurepip模块安装：

   python -m ensurepip

2. **安装dli-sdk-python**

   执行安装命令：

   ```
   python setup.py install
   ```

3. **安装dli-dbt**

   从DLI管理控制台下载dli-dbt驱动。

   执行安装命令：

   ```
   python setup.py install
   ```

   安装完成后，可以通过运行以下命令来验证dbt是否正确安装：

   ```
   dbt --version
   ```

步骤 2：配置 DBT 连接 DLI

配置profiles.yml文件用于保存DBT与DLI的连接信息。

在安装DBT的服务器的主目录下找到 .dbt 目录，创建或编辑 profiles.yml 文件。

例如，在Windows系统中，路径可能是 C:\Users\用户名\.dbt\profiles.yml。

配置文件内容应包含DBT与DLI的连接配置，例如：

```
profiles:
  - name: dbt_dli
    target: dev
    outputs:
      dev:
        type: dli
        region: your-region-name
        project_id: your-project_id
        access_id：your-ak
        secret_key：your-sk
```

```
queue：your-queue-name
database: your-dli-database
schema: your-dli-schema
```

表 7-6 DBT 连接 DLI 参数说明

参数	是否必选	说明	配置样例
type	是	数据源类型，本例配置为dli。	dli
region	是	DLI的区域名称和服务名称。	ap-southeast-2
project_id	是	DLI资源所在的项目ID。 **获取项目ID**	0b33ea2a7e0010802fe4c009bb05076d
access_id和secret_key	是	AK/SK认证密钥。	-
queue	是	DLI服务的队列名称。	dli_test
database	是	数据目录名称。默认使用dli数据目录。 如果使用Lakeformation元数据，填写具体的数据目录名称。	dli
schema	是	提交作业使用的DLI的数据库名称。	tpch

步骤 3：测试使用 DBT 提交作业至 DLI

1. **初始化DBT项目**

   在空目录下执行以下命令以初始化DBT项目：

   ```
   dbt init
   ```

2. **配置dbt_project.yml文件**

   在项目根目录下创建或编辑 dbt_project.yml文件。

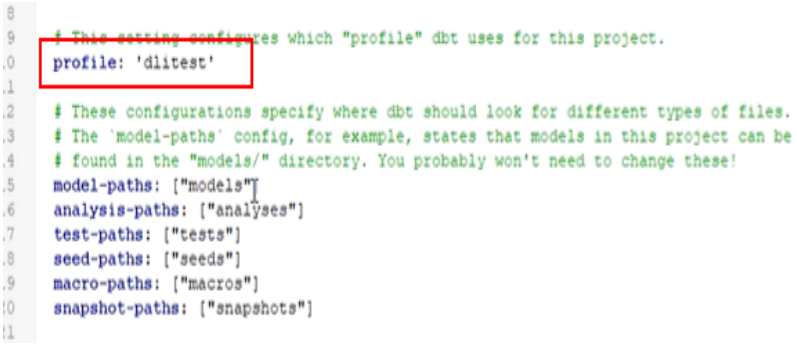
   参考**dbt_project.yml**配置项目。

   确保**步骤2：配置DBT连接DLI**profile文件中已设置该项目的profiles.yml中定义的数据源名称。

   图 7-6 profile 文件

图 7-7 dbt_project.yml 文件中配置的 profile



3. **验证配置**

   执行以下命令测试DBT配置是否正确：

   ```
   dbt debug
   ```

4. **执行项目作业**

   测试通过后执行以下命令来执行您的数据模型。

   ```
   dbt run
   ```

7.4 配置 YongHong BI 连接 DLI 进行数据查询和分析

YongHong BI是一款企业级数据分析工具。支持数据可视化、报表制作、数据分析和决策支持的功能，帮助企业洞察业务数据，提升决策效率。

本节操作介绍YongHong BI连接DLI的操作步骤。

操作前准备

- **环境要求：**
  - 已安装YongHong BI。
- **DLI的JDBC驱动：**

  单击**dli-jdbc-x.x.x.jar**获取JDBC驱动，驱动名称：huaweicloud-dli-jdbc-xxx-dependencies.jar。
- **连接信息：**

表 7-7 连接信息

类别	说明	获取方式
DLI AKSK	AK/SK认证就是使用AK/SK对请求进行签名，从而通过身份认证。	**获取AK/SK**
DLI Endpoint地址	地区与终端节点，即云服务在不同Region有不同的访问域名。	**获取EndPoint**
DLI所在的项目ID	项目编号，用于资源隔离。	**获取项目ID**
DLI区域信息	DLI所属区域信息	**地区和终端节点**

步骤 1：配置 YongHong BI 新建 DLI 数据连接

1. 启动YongHong BI。
2. 在YongHong BI界面的单击“添加数据源”。
3. 在“选择数据源类型”页面中选择数据源类型为“GENERIC”。
4. 添加数据源的相关配置，请参见图7-8。

   驱动：上传下载的DLI JDBC驱动。

   URL：后面填写DLI jdbc的URL，URL的格式见表7-8，属性配置项说明见表7-9。

   指定数据库：

   说明

   - “表结构模式”可填写需访问的数据库名称，如果填写，后续创建数据集时，刷新表，页面上只可见该数据库下的表。如果不填写，后续创建数据集时，刷新表，页面上会显示所有数据库下的表。
   - 其他选项不需要填写，也无需勾选“需要登录”选项。

图 7-8 添加数据源配置

表 7-8 数据库连接参数

参数	描述
URL	URL的格式如下。 ***jdbc:dli://<endPoint>/<projectId>?<key1>=<val1>;<key2>=<val2>…*** **说明** • endpoint指DLI的终端节点，具体请参考**地区和终端节点**。 • projectId指项目编号，从华为云“基本信息>我的凭证”页面获取项目编号。 • “？”后面接其他配置项，每个配置项以“key=value”的形式列出，配置项之间以“;”隔开，详见**表7-9**

表 7-9 属性配置项

属性项（key）	必须配置	默认值（value）	描述
enginetype	否	-	配置执行作业的引擎类型。
queuename	是	-	DLI服务的队列名称。
databasename	否	-	默认访问的数据库，URL中若不填此项，访问数据库的表时需采用db.table方式（如 select * from dbother.tabletest）。
accesskey	authentication mode=aksk时必须配置	-	JDBC认证方式。
secretkey	authentication mode=aksk时必须配置	-	JDBC认证方式。
regionname	authentication mode=aksk时必须配置	-	具体请参考**地区和终端节点**。
servicename	authentication mode=aksk时必须配置	-	由于是对接DLI，所以servicename=dli。
catalog	否	dli	配置执行作业读取的元数据类型。

5. 在“添加数据源配置”页面工具栏中单击“测试连接”，测试通过后，单击“保存”，填写数据源名称，保存该数据源。

   说明

   目前没有根目录保存权限，需保存到已建文件夹目录下。

步骤 2：在 YongHong BI 创建 DLI 的数据集

步骤1 在YongHong BI SaaS生产环境主页，单击左侧导航栏中的“创建数据集”。

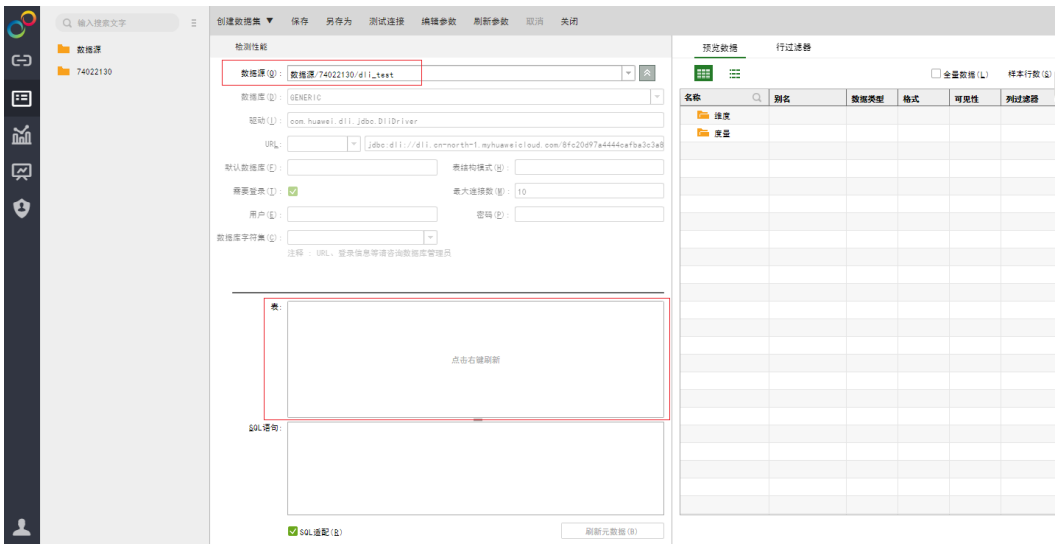
图 7-9 创建数据集

步骤2 在“数据集类型”页面中，选择创建“SQL数据集”，请参见图7-10。

图 7-10 创建 SQL 数据集

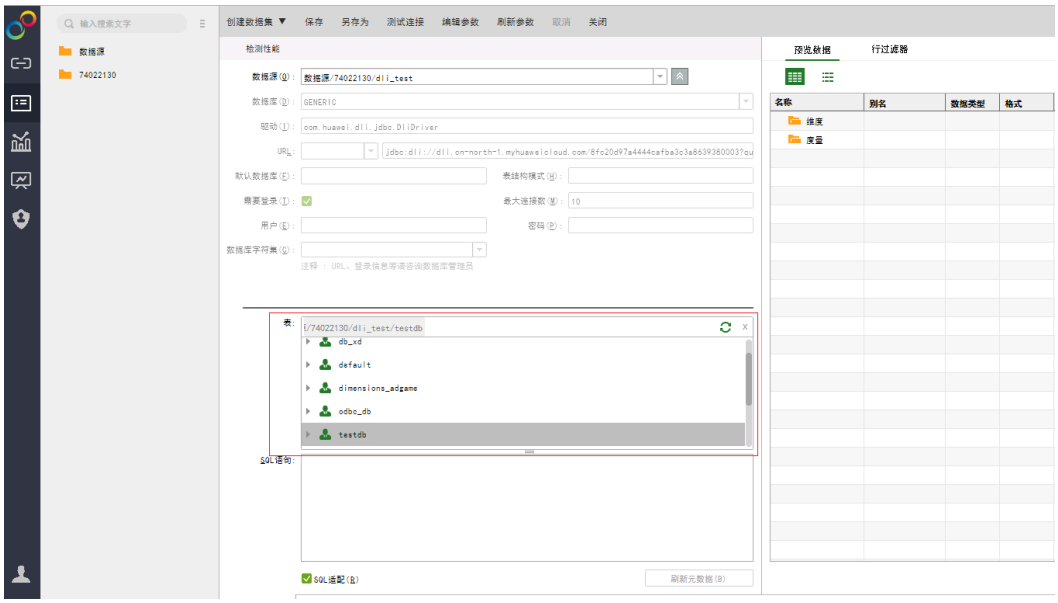
步骤3 在“创建数据集”页面中，左侧“数据源”栏选择已添加的DLI数据源，请参见图7-11。

图 7-11 选择数据源



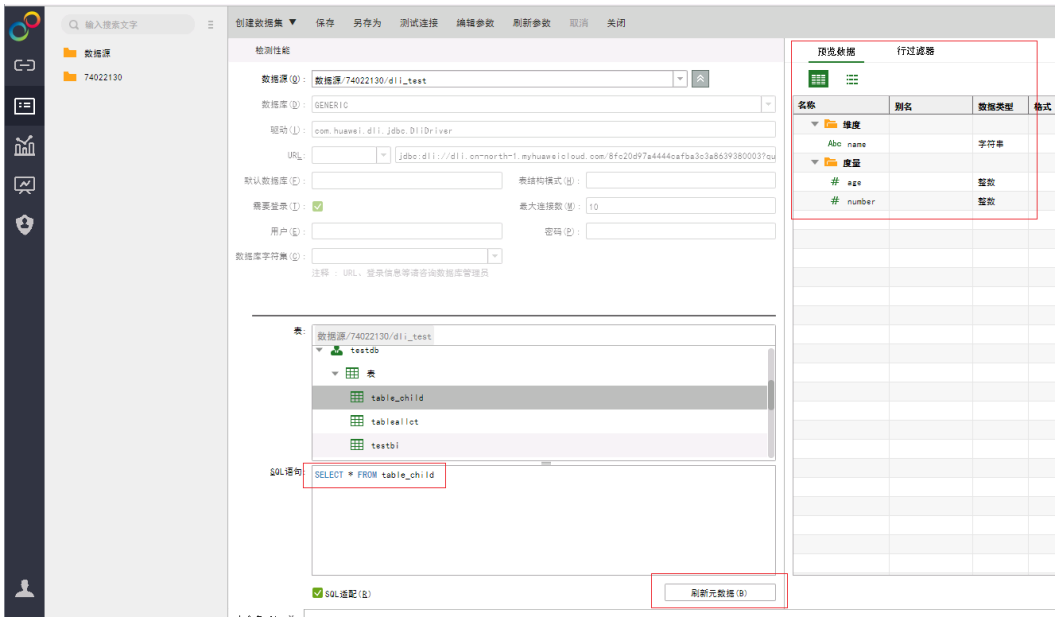
步骤4 左侧“表”栏中单击右键，刷新表，将列出所有数据库及数据库下面的数据表（这是添加数据源时，“表结构模式”没有配置时的情况），请参见图7-12。

图 7-12 刷新数据表



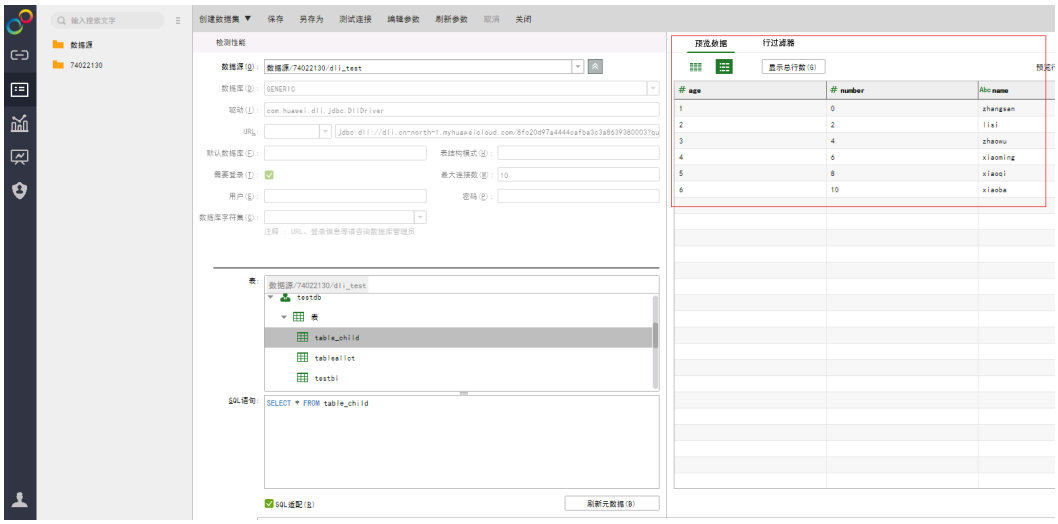
步骤5 在左侧“SQL语句”栏中执行表查询命令”select * from table_name“，单击“刷新元数据”，再单击右侧“预览数据”栏下左侧的“预览元数据”  ，可查询出该表的元数据（包括字段，字段类型等），请参见图7-13。

图 7-13 查询数据表



步骤6 单击右侧“预览数据”栏下右侧的“数据细节”  ，可查询出该表的数据，请参见图7-14。

图 7-14 查询数据表数据



步骤7 在“创建数据集”页面工具栏中单击“保存”，完成创建数据集。

----结束

在YongHong BI连接DLI数据源并创建和数据集后，即可在YongHong BI中按需制作BI图表。

7.5 配置 PowerBI 通过 Kyuubi 连接 DLI 进行数据查询和分析

Power BI提供了数据集成、数据仓库、报告和数据可视化等功能，能够将复杂的数据转换为易于理解和交互的可视化图表和仪表板，从而帮助企业做出基于数据的决策。

Kyuubi是一个分布式SQL查询引擎，它允许用户通过标准的SQL接口来访问和分析数据。

将Power BI与Kyuubi对接，通过Kyuubi访问DLI进行数据查询和分析，简化了数据访问流程，提供了数据的统一管理和分析能力，从而获得更深入的数据洞察。

本节操作介绍PowerBI基于Kyuubi连接DLI，以访问和分析DLI中的数据的操作步骤。

操作流程

图 7-15 操作流程

- **步骤1：安装并配置Kyuubi连接DLI**：安装并配置Kyuubi，确保Kyuubi可以连接到DLI。
- **步骤2：配置ODBC连接Kyuubi**：安装ODBC驱动，配置ODBC驱动连接到Kyuubi服务器。
- **步骤3：配置Power BI使用ODBC连接到Kyuubi**：在BI工具中创建一个新的数据连接，使用ODBC作为数据源，通过ODBC连接Kyuubi。

步骤 1：安装并配置 Kyuubi 连接 DLI

如需使用外网访问Kyuubi请确保弹性云服务器绑定弹性公网IP，并配置安全组入方向开启10009和3309端口。

步骤1 安装JDK。

在安装和使用Kyuubi前，确保您的开发环境已安装JDK。

Java SDK要求使用JDK1.8或更高版本。考虑到后续版本的兼容性，推荐使用1.8版本。

1. 下载JDK。

   从**Oracle官网**下载并安装JDK1.8版本安装包。

   本例使用jdk-8u261-linux-x64.tar.gz。
2. 将jdk上传到linux服务器对应的目录下并执行解压命令，此处上传到/usr/local目录下。
   ```
   sudo tar -xzf jdk-8u261-linux-x64.tar.gz -C /usr/local/
   ```
3. 配置环境变量。

   编辑.bashrc或.profile文件，添加以下行：
   ```
   export JAVA_HOME=/usr/local/jdk-1.8.0_261
   export PATH=$PATH:$JAVA_HOME/bin
   ```
4. 执行以下命令应用环境变量。
   ```
   source ~/.bashrc
   ```
5. 执行命令**java -version**，检查是否安装成功，如下显示版本号信息说明java环境安装成功。
   ```
   java version "1.8.0_261"
   Java(TM) SE Runtime Environment (build 1.8.0_261-b12)
   Java HotSpot(TM) 64-Bit Server VM (build 25.261-b12, mixed mode)
   ```

步骤2 安装Kyuubi

1. 访问**Apache Kyuubi**的下载Kyuubi安装包。了解更多**Kyuubi安装操作**。
2. 解压下载的Kyuubi安装包。
   ```
   tar -xzf kyuubi-{version}-bin.tar.gz
   ```
3. 配置环境变量（可选）：

   将Kyuubi的bin目录添加到PATH环境变量中，确保可以在任何位置调用Kyuubi的脚本。

步骤3 配置Kyuubi连接DLI

1. 在Kyuubi的根目录下添加DLI驱动。

   在“**DLI SDK DOWNLOAD**”页面，单击Kyuubi驱动包链接，下载对应版本的驱动包。

   并将该驱动放在kyuubi根目录/externals/engines/jdbc。

   确保插件用户组和权限与其他Jar保持一致。

2. 执行以下命令修改Kyuubi配置文件。
   cd $KYUUBI_HOME/confvi kyuubi-defaults.conf
   配置项说明请参考**表7-10**。

   表 7-10 kyuubi 配置参数说明

配置项	说明	是否必选	示例
kyuubi.engine.type	JDBC服务类型。这里请指定为dli。	是	jdbc
kyuubi.engine.jdbc.type	引擎类型。请使用dli。	是	dli
kyuubi.engine.jdbc.driver.class	连接JDBC服务使用的驱动类名。请使用com.huawei.dli.jdbc.DliDriver	是	com.huawei.dli.jdbc.DliDriver
kyuubi.engine.jdbc.connection.url	JDBC服务连接的URL。 格式：jdbc:dli://{dliendpoint} /{projectId}	是	jdbc:dli://{dliendpoint} /{projectId}
kyuubi.engine.jdbc.session.initialize.sql	用于指定在建立JDBC会话时执行的初始化SQL语句。	否	select 1 如果在DLI的管理控制台看到select 1，代表初始化成功。
kyuubi.frontend.protocols	用于指定Kyuubi服务支持的前端协议。Kyuubi支持多种前端协议，允许用户通过不同的接口与Kyuubi进行交互。	是	– mysql – thrift_binary

配置项	说明	是否必选	示例
kyuubi.engine.dli.schema.show.name	用于指定当用户执行show schemas或show databases语句时，Kyuubi引擎如何展示数据源接口的模式名称。 – true：表示在展示模式名称时，包含 DLI 的名称作为前缀。 – false：表示在展示模式名称时，不包含 DLI 的名称。 例如如果配置为true，并且有一个DLI名称为 hive，那么在执行show schemas时，输出为hive.default的格式。 如果配置为false，输出为default的格式。	否	– true – false
kyuubi.engine.dli.jdbc.connection.region	DLI的区域名称和服务名称。	是	regionname=ap-southeast-2
kyuubi.engine.dli.jdbc.connection.queue	DLI服务的队列名称。	是	dli_test
kyuubi.engine.dli.jdbc.connection.database	用于指定Kyuubi引擎通过JDBC连接到DLI数据源时默认使用的数据库名称。	是	tpch
kyuubi.engine.dli.jdbc.connection.ak	AK/SK认证密钥。 如果使用AK/SK认证方式。	是	accesskey=your-access-key
kyuubi.engine.dli.jdbc.connection.sk	DLI的区域名称和服务名称。 如果使用AK/SK认证方式时配置。	是	secretkey=your-secret-key
kyuubi.engine.dli.jdbc.connection.project	DLI资源所在的项目ID。	是	0b33ea2a7e0010802fe4c009bb05076d
kyuubi.engine.dli.sql.limit.time.sec	SQL查询的执行时间限制。 默认600s	否	300

配置项	说明	是否必选	示例
kyuubi.engine.dli.result.line.num.limit	SQL查询的返回的最大条数。 默认返回10万条。 配置为-1代表不限制返回的条数。	是	50000
kyuubi.engine.dli.small.file.merge	配置是否开启小文件自动合并。默认为false，代表不开启。 – true：开启 – false：不开启	是	true
kyuubi.engine.dli.bi.type	用于指定BI工具类型。 支持fine/ grafana/ superset/ tableau/ power/dbt/yongHong	是	fine
kyuubi.engine.dli.boolean.type.to.int	定义DLI的Boolean类型数据是以1/0返回，还是true/false返回 当BI工具类型为Grafana时，需要设置为true。 – true：按1/0返回（1：代表true，0：fales）。 – false：按true/false返回。 默认取值false。	否	false
kyuubi.engine.dli.set.conf.transform.to.annotation	支持在SQL中设置set spark参数。 PowerBI、FineBI、SuperSet、DBT需要设置为true。	否	true
kyuubi.engine.dli.set.conf.sql.suffix	支持在SQL中尾端设置set spark参数。 PowerBI、DBT需要设置为true。	否	true
kyuubi.engine.dli.result.cache.enable	是否开启库表数据缓存，开启后自动缓存库表信息。默认为true。 – true：开启 – false：不开启	否	true

配置项	说明	是否必选	示例
kyuubi.engine.dli.cache.limit.line.num	配置缓存的最大条数。 默认缓存10万条。 配置为-1代表不限制缓存的最大条数。	否	1000
kyuubi.engine.dli.cache.time.sec	配置缓存的时间。 默认为1800s。	否	1800
kyuubi.operation.incremental.collect	kyuubi会预加载select结果数据到缓存加快读取数据，数据量较大的场景防止内存OOM建议关闭。	否	false 配置为false代表关闭预加载。
kyuubi.engine.jdbc.memory	jdbc engine进程内存 默认为1g，建议改成5g以上加大jdbc engine进程内存使用	否	5g

3. 快速启动kyuubi。

   进入云服务器的根目录/bin执行以下命令启动kyuubi。

   ```
   cd /bin
   ./kyuubi start  restart
   ```

   连接成功后，可以执行SQL查询来测试Kyuubi与DLI的连接是否正常工作。

步骤4 （可选）配置主机的host文件提高Kyuubi的访问效率

为了提高Kyuubi的访问效率，建议在主机的/etc/hosts 配置Kyuubi主机IP的映射关系。

1. 执行ifconfig查看主机IP地址。

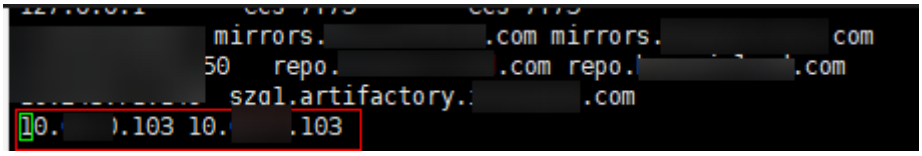
   图 7-16 查看主机 IP 地址

   ```
   [root@ecs-7f75 ~]# ifconfig
   eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
           inet 10.6    .103  netmask 255.255.255.0  broadcast 10.63.0.255
           inet6 fe80::f816:3eff:fefd:3a50  prefixlen 64  scopeid 0x20<link>
           ether fa:16:3e:fd:3a:50  txqueuelen 1000  (Ethernet)
           RX packets 6654431  bytes 2845894229 (2.6 GiB)
           RX errors 0  dropped 0  overruns 0  frame 0
           TX packets 4585886  bytes 1125818425 (1.0 GiB)
           TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

   lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
           inet 127.0.0.1  netmask 255.0.0.0
           inet6 ::1  prefixlen 128  scopeid 0x10<host>
           loop  txqueuelen 1000  (Local Loopback)
           RX packets 1502680  bytes 307935807 (293.6 MiB)
           RX errors 0  dropped 0  overruns 0  frame 0
           TX packets 1502680  bytes 307935807 (293.6 MiB)
           TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
   ```

2. 将该IP配置在/etc/hosts文件中。

图 7-17 在/etc/hosts 文件中配置 IP 地址



----结束

步骤 2：配置 ODBC 连接 Kyuubi

步骤1 安装ODBC驱动

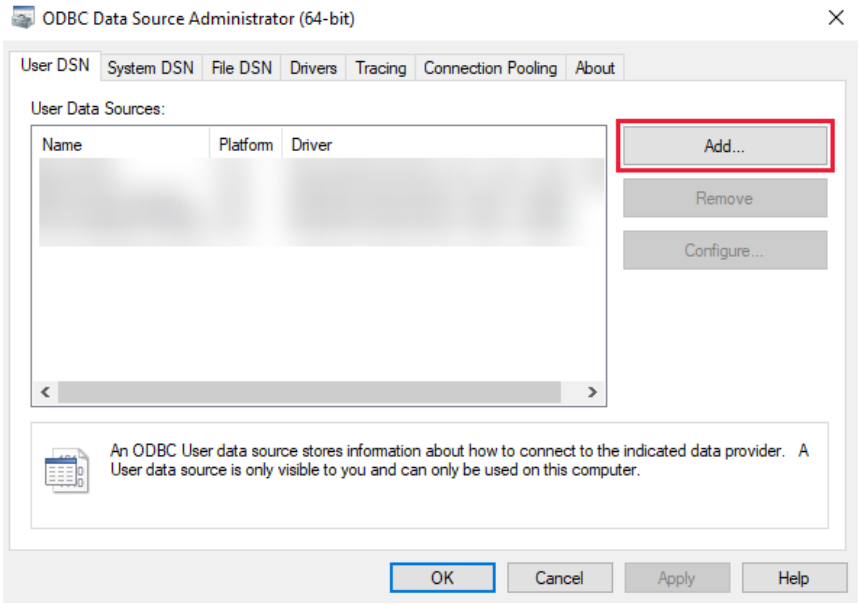
根据数据库类型，需要在本地主机上安装相应的ODBC驱动。本例使用Hive数据库类型。

- **Cloudera Hive ODBC**，推荐使用v2.5.12。
- **Microsoft Hive ODBC**，推荐使用v2.6.12.1012。

步骤2 配置ODBC连接Kyuubi

1. 在Windows系统中，打开“控制面板 > 管理工具 > 数据源 (ODBC)”。
2. 配置新的ODBC数据源。
   a. 在ODBC中单击“User DSN”。
   b. 单击“Add”创建新的数据源。
   c. 选择Hive ODBC Driver，单击“OK”。

图 7-18 ODBC 新建数据源连接

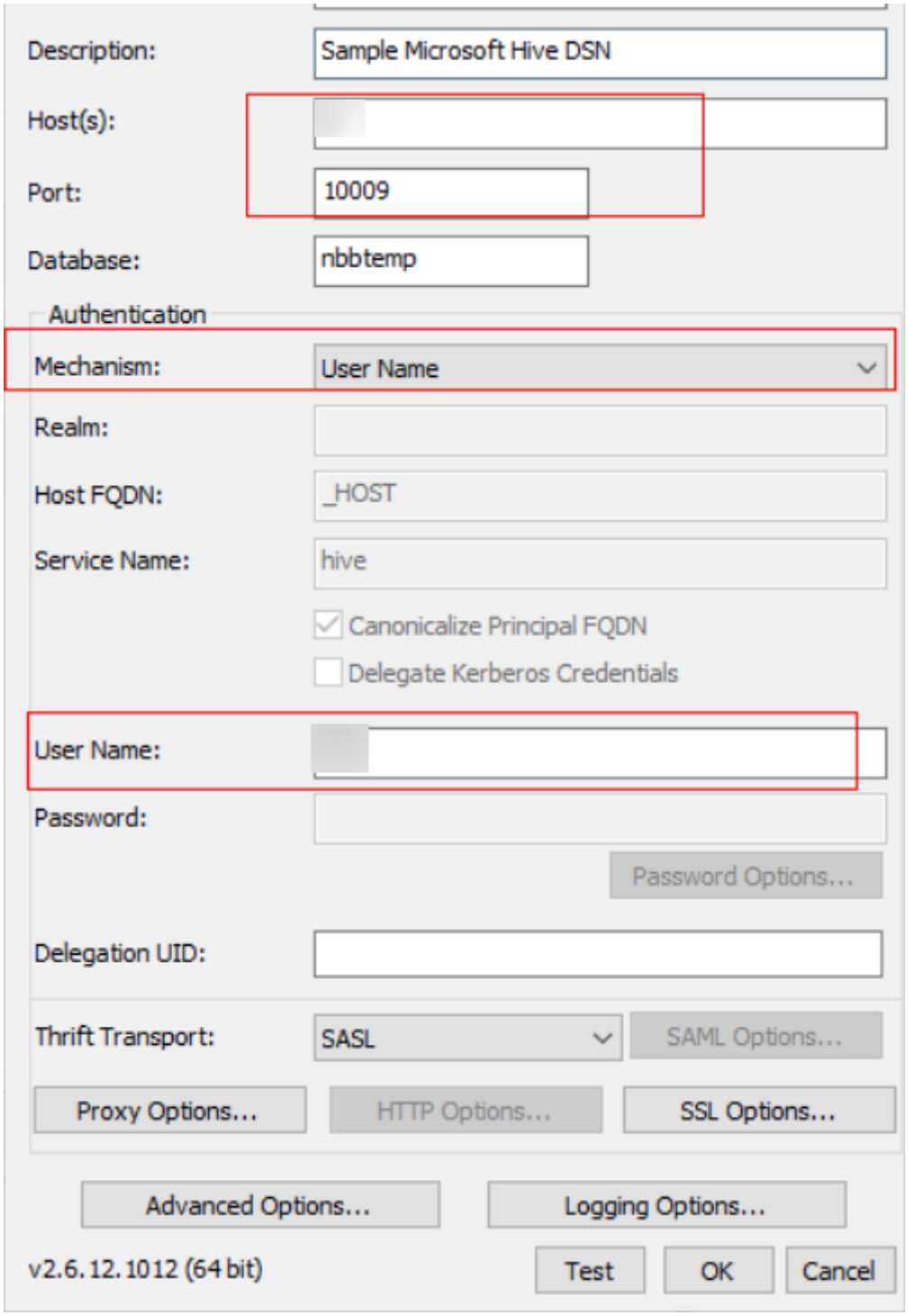


3. 在创建的新数据源配置界面中，输入Kyuubi服务器的相关信息。
   - 数据库名称：本例输入DLI数据库名称。
   - 服务器地址：输入Kyuubi服务器的弹性公网IP地址。
   - 端口号：Kyuubi服务监听的端口，使用Hive Thrift协议，默认端口10009。

- 用户名和密码：按需配置Kyuubi服务器用户名和密码。

按需配置其他高级选项，然后保存配置。

图 7-19 ODBC 配置数据源连接信息



4. 单击“Test”测试数据源连接是否成功，如果连接正常单击“OK”保存连接。

----结束

步骤 3：配置 Power BI 使用 ODBC 连接到 Kyuubi

1. 单击并安装PowerBI。获取**PowerBI安装包**。
2. 打开Power BI Desktop。
3. 单击“主页”选项卡下的“获取数据”按钮。
4. 在“获取数据”窗口中，选择“更多...”以查看其他数据源选项。
5. 从列表中选择“ODBC”作为数据源类型，然后单击“连接”。
6. 在弹出的“ODBC驱动管理器”窗口中，选择**步骤2：配置ODBC连接Kyuubi**配置的ODBC数据源名称，单击“确定”。

   Power BI将使用ODBC连接到Kyuubi，并允许你预览和选择数据库中的表和视图。

   说明

   在预览库表时请选择limit，否则分区表将全表扫描。

常用操作：SQL 作业参数设置

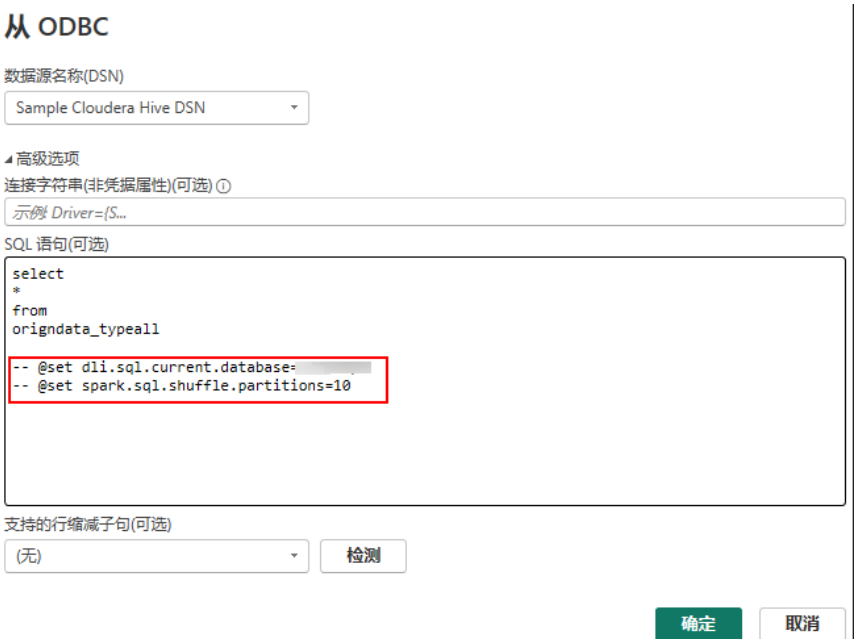
根据安装ODBC驱动类型选择配置方法：

- **使用Cloudera Hive ODBC (v2.5.12)驱动**

  只需在sql语句的末尾添加注解参数。

  ```
  -- @set 参数
  示例：
  -- @set dli.sql.current.database=tpch
  -- @set dli.sql.shuffle.partitions=100
  ```

  图 7-20 ODBC 配置参数示例（Cloudera Hive ODBC）

- **使用Microsoft Hive ODBC (v2.6.12.1012)驱动**

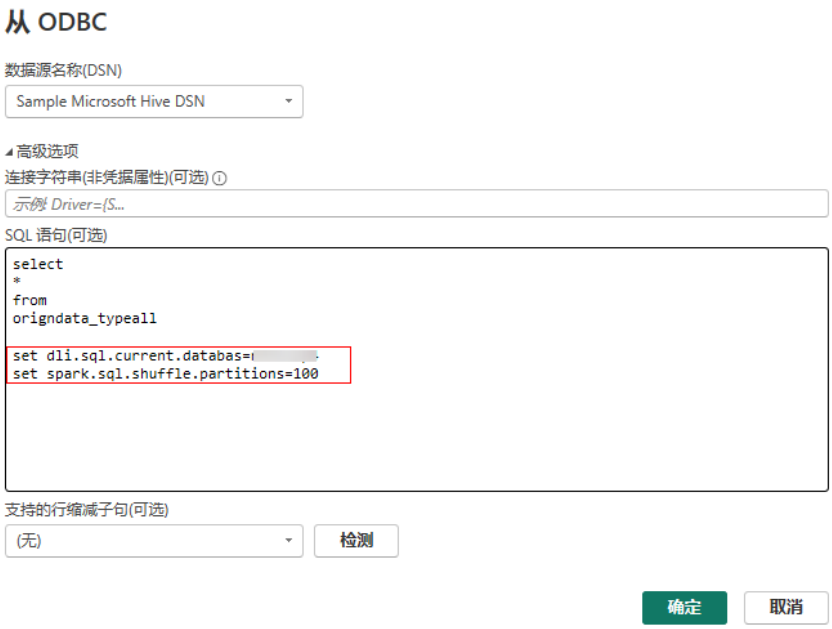
  a. 确保kyuubi的/conf/kyuubi-defaults.conf配置打开如下参数开关。

  ```
  kyuubi.engine.dli.set.conf.transform.to.annotation=true
  kyuubi.engine.dli.set.conf.sql.suffix=true
  ```

b. 在sql语句的末尾添加注解参数。

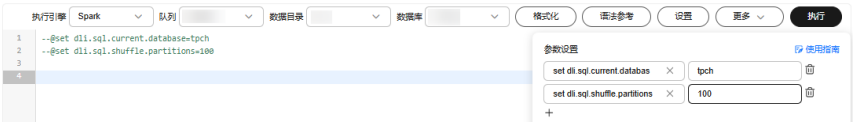
```
set 参数
示例：
set dli.sql.current.database=tpch
set dli.sql.shuffle.partitions=100
```

图 7-21 ODBC 配置参数示例（Microsoft Hive ODBC）



在DLI的SQL编辑器的执行效果：

图 7-22 在 DLI 的 SQL 编辑器查看配置的参数



7.6 配置 Fine BI 通过 Kyuubi 连接 DLI 进行数据查询和分析

Fine BI是一款智能可视化工具，专注于数据分析和可视化。它支持连接多种数据源，能够将复杂的数据转换为直观的图表和仪表板，快速获得数据洞察。

Kyuubi是一个分布式 SQL 查询引擎，它提供了标准的SQL接口，使用户能够方便地访问和分析存储在大数据平台中的数据。

通过将Fine BI与Kyuubi对接，用户可以利用Kyuubi访问DLI进行数据查询和分析。这种集成简化了数据访问流程，提供了数据的统一管理和分析能力，使得用户能够更深入地洞察数据。

本节操作介绍Fine BI基于Kyuubi连接DLI，以访问和分析DLI中的数据的操作步骤。

操作流程

图 7-23 操作流程

- **步骤1：安装并配置Kyuubi连接DLI**：安装并配置Kyuubi，确保Kyuubi可以连接到DLI。
- **步骤2：Fine BI安装数据连接驱动**：配置Fine BI安装数据连接驱动。
- **步骤3：配置Fine BI连接Kyuubi**：在BI工具中创建一个新的数据连接，通过JDBC连接Kyuubi。

步骤 1：安装并配置 Kyuubi 连接 DLI

如需使用外网访问Kyuubi请确保弹性云服务器绑定弹性公网IP，并配置安全组入方向开启10009和3309端口。

步骤1 安装JDK。

在安装和使用Kyuubi前，确保您的开发环境已安装JDK。

Java SDK要求使用JDK1.8或更高版本。考虑到后续版本的兼容性，推荐使用1.8版本。

1. 下载JDK。

   从**Oracle官网**下载并安装JDK1.8版本安装包。

   本例使用jdk-8u261-linux-x64.tar.gz。
2. 将jdk上传到linux服务器对应的目录下并执行解压命令，此处上传到/usr/local目录下。

   ```
   sudo tar -xzf jdk-8u261-linux-x64.tar.gz -C /usr/local/
   ```
3. 配置环境变量。

   编辑.bashrc或.profile文件，添加以下行：

   ```
   export JAVA_HOME=/usr/local/jdk-1.8.0_261
   export PATH=$PATH:$JAVA_HOME/bin
   ```
4. 执行以下命令应用环境变量。

   ```
   source ~/.bashrc
   ```
5. 执行命令**java -version**，检查是否安装成功，如下显示版本号信息说明java环境安装成功。

   ```
   java version "1.8.0_261"
   Java(TM) SE Runtime Environment (build 1.8.0_261-b12)
   Java HotSpot(TM) 64-Bit Server VM (build 25.261-b12, mixed mode)
   ```

步骤2 安装Kyuubi

1. 访问**Apache Kyuubi**的下载Kyuubi安装包。了解更多**Kyuubi安装操作**。
2. 解压下载的Kyuubi安装包。

   ```
   tar -xzf kyuubi-{version}-bin.tar.gz
   ```

3. 配置环境变量（可选）：

   将Kyuubi的bin目录添加到PATH环境变量中，确保可以在任何位置调用Kyuubi的脚本。

步骤3 配置Kyuubi连接DLI

1. 在Kyuubi的根目录下添加DLI驱动。

   在“**DLI SDK DOWNLOAD**”页面，单击Kyuubi驱动包链接，下载对应版本的驱动包。

   并将该驱动放在kyuubi根目录/externals/engines/jdbc。

   确保插件用户组和权限与其他Jar保持一致。

2. 执行以下命令修改Kyuubi配置文件。

   cd $KYUUBI_HOME/confvi kyuubi-defaults.conf

   配置项说明请参考**表7-11**。

表 7-11 kyuubi 配置参数说明

配置项	说明	是否必选	示例
kyuubi.engine.type	JDBC服务类型。这里请指定为dli。	是	jdbc
kyuubi.engine.jdbc.type	引擎类型。请使用dli。	是	dli
kyuubi.engine.jdbc.driver.class	连接JDBC服务使用的驱动类名。请使用com.huawei.dli.jdbc.DliDriver	是	com.huawei.dli.jdbc.DliDriver
kyuubi.engine.jdbc.connection.url	JDBC服务连接的URL。 格式：jdbc:dli://{dliendpoint} /{projectId}	是	jdbc:dli://{dliendpoint} /{projectId}
kyuubi.engine.jdbc.session.initialize.sql	用于指定在建立JDBC会话时执行的初始化SQL语句。	否	select 1 如果在DLI的管理控制台看到select 1，代表初始化成功。
kyuubi.frontend.protocols	用于指定Kyuubi服务支持的前端协议。Kyuubi支持多种前端协议，允许用户通过不同的接口与Kyuubi进行交互。	是	- mysql - thrift_binary

配置项	说明	是否必选	示例
kyuubi.engine.dli.schema.show.name	用于指定当用户执行show schemas或show databases语句时，Kyuubi引擎如何展示数据源接口的模式名称。 – true：表示在展示模式名称时，包含 DLI 的名称作为前缀。 – false：表示在展示模式名称时，不包含 DLI 的名称。 例如如果配置为true，并且有一个DLI名称为 hive，那么在执行show schemas时，输出为hive.default的格式。 如果配置为false，输出为default的格式。	否	– true – false
kyuubi.engine.dli.jdbc.connection.region	DLI的区域名称和服务名称。	是	regionname=ap-southeast-2
kyuubi.engine.dli.jdbc.connection.queue	DLI服务的队列名称。	是	dli_test
kyuubi.engine.dli.jdbc.connection.database	用于指定Kyuubi引擎通过JDBC连接到DLI数据源时默认使用的数据库名称。	是	tpch
kyuubi.engine.dli.jdbc.connection.ak	AK/SK认证密钥。 如果使用AK/SK认证方式。	是	accesskey=your-access-key
kyuubi.engine.dli.jdbc.connection.sk	DLI的区域名称和服务名称。 如果使用AK/SK认证方式时配置。	是	secretkey=your-secret-key
kyuubi.engine.dli.jdbc.connection.project	DLI资源所在的项目ID。	是	0b33ea2a7e0010802fe4c009bb05076d
kyuubi.engine.dli.sql.limit.time.sec	SQL查询的执行时间限制。 默认600s	否	300

配置项	说明	是否必选	示例
kyuubi.engine.dli.result.line.num.limit	SQL查询的返回的最大条数。 默认返回10万条。 配置为-1代表不限制返回的条数。	是	50000
kyuubi.engine.dli.small.file.merge	配置是否开启小文件自动合并。默认为false，代表不开启。 – true：开启 – false：不开启	是	true
kyuubi.engine.dli.bi.type	用于指定BI工具类型。 支持fine/ grafana/ superset/ tableau/ power/dbt/yongHong	是	fine
kyuubi.engine.dli.boolean.type.to.int	定义DLI的Boolean类型数据是以1/0返回，还是true/false返回 当BI工具类型为Grafana时，需要设置为true。 – true：按1/0返回（1：代表true，0：fales）。 – false：按true/false返回。 默认取值false。	否	false
kyuubi.engine.dli.set.conf.transform.to.annotation	支持在SQL中设置set spark参数。 PowerBI、FineBI、SuperSet、DBT需要设置为true。	否	true
kyuubi.engine.dli.set.conf.sql.suffix	支持在SQL中尾端设置set spark参数。 PowerBI、DBT需要设置为true。	否	true
kyuubi.engine.dli.result.cache.enable	是否开启库表数据缓存，开启后自动缓存库表信息。默认为true。 – true：开启 – false：不开启	否	true

配置项	说明	是否必选	示例
kyuubi.engine.dli.cache.limit.line.num	配置缓存的最大条数。 默认缓存10万条。 配置为-1代表不限制缓存的最大条数。	否	1000
kyuubi.engine.dli.cache.time.sec	配置缓存的时间。 默认为1800s。	否	1800
kyuubi.operation.incremental.collect	kyuubi会预加载select结果数据到缓存加快读取数据，数据量较大的场景防止内存OOM建议关闭。	否	false 配置为false代表关闭预加载。
kyuubi.engine.jdbc.memory	jdbc engine进程内存 默认为1g，建议改成5g以上加大jdbc engine进程内存使用	否	5g

3. 快速启动kyuubi。

   进入云服务器的根目录/bin执行以下命令启动kyuubi。

   ```
   cd /bin
   ./kyuubi start  restart
   ```

   连接成功后，可以执行SQL查询来测试Kyuubi与DLI的连接是否正常工作。

步骤4 （可选）配置主机的host文件提高Kyuubi的访问效率

为了提高Kyuubi的访问效率，建议在主机的/etc/hosts 配置Kyuubi主机IP的映射关系。

1. 执行ifconfig查看主机IP地址。

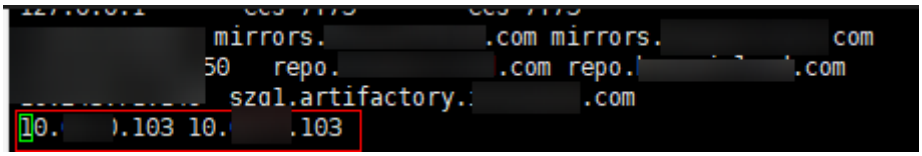
   图 7-24 查看主机 IP 地址

   ```
   [root@ecs-7f75 ~]# ifconfig
   eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
           inet 10.6    .103  netmask 255.255.255.0  broadcast 10.63.0.255
           inet6 fe80::f816:3eff:fefd:3a50  prefixlen 64  scopeid 0x20<link>
           ether fa:16:3e:fd:3a:50  txqueuelen 1000  (Ethernet)
           RX packets 6654481  bytes 2845894229 (2.6 GiB)
           RX errors 0  dropped 0  overruns 0  frame 0
           TX packets 4585886  bytes 1125818425 (1.0 GiB)
           TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

   lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
           inet 127.0.0.1  netmask 255.0.0.0
           inet6 ::1  prefixlen 128  scopeid 0x10<host>
           loop  txqueuelen 1000  (Local Loopback)
           RX packets 1502680  bytes 307935807 (293.6 MiB)
           RX errors 0  dropped 0  overruns 0  frame 0
           TX packets 1502680  bytes 307935807 (293.6 MiB)
           TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
   ```

2. 将该IP配置在/etc/hosts文件中。

图 7-25 在/etc/hosts 文件中配置 IP 地址



----结束

步骤 2：Fine BI 安装数据连接驱动

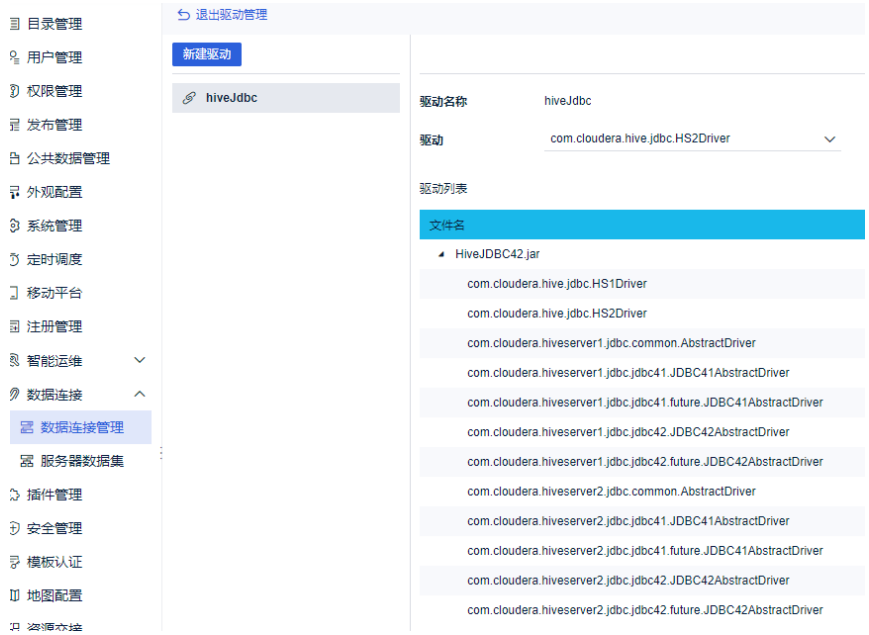
步骤1 下载并安装Fine BI

1. 获取**Fine BI安装包**
2. 找到下载的Fine BI安装程序文件。
3. 双击运行安装程序。
4. 按照安装向导的指示进行操作，包括接受许可协议、选择安装类型（典型安装或自定义安装）、设置安装目录等。

步骤2 配置Fine BI集成JDBC驱动

1. 下载数据驱动。获取**Hive Jdbc驱动包**，推荐使用v2.6.23版本。
2. Fine BI集成驱动插件。
   a. 打开Fine BI。
   b. 单击“数据连接 > 数据连接管理”。
   c. 单击“新建驱动”，在驱动列表中选择**步骤2.1**中的驱动。

图 7-26 Fine BI 安装数据连接驱动



----结束

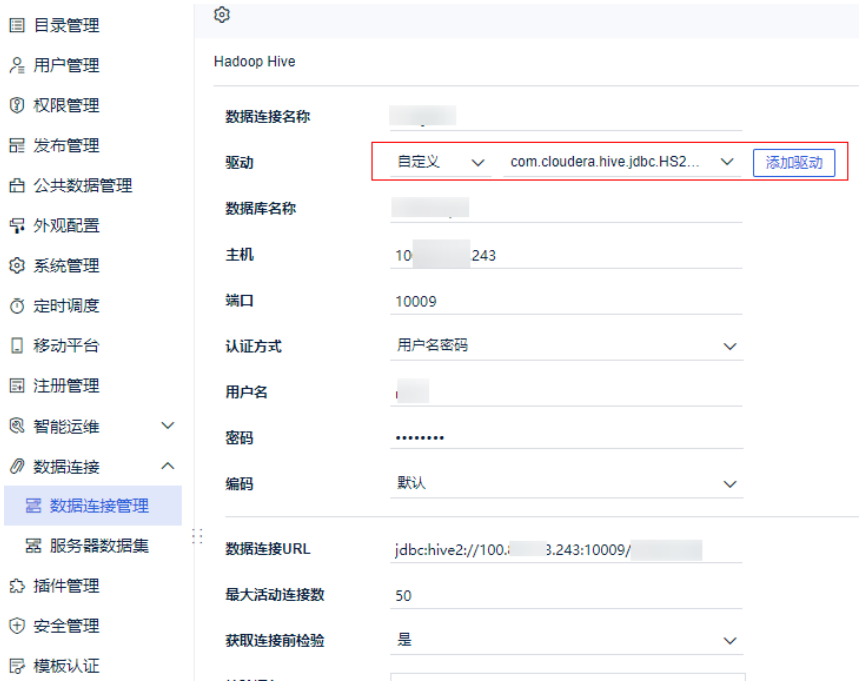
步骤 3：配置 Fine BI 连接 Kyuubi

1. 打开Fine BI。
2. 单击“数据连接 > 数据连接管理”。
3. 单击“添加数据源”创建一个新的数据源。
4. 在数据源的向导中选择数据库类型。本例选择“Hadoop Hive”
5. 配置数据库连接信息：
   - 数据连接名称：用户自定义数据连接名称。
   - 驱动：选择**步骤2.1**中的驱动。
   - 数据库名称：DLI的数据库名称。
   - 主机：安装Kyuubi主机IP地址。
   - 端口：访问Kyuubi主机的端口。端口默认10009
   - 认证方式：本例选择密码认证方式。
   - 用户名：Kyuubi数据库的访问凭证，Kyuubi用户名。
   - 密码：Kyuubi数据库的访问凭证，Kyuubi访问密码。
   - 数据连接URL：

     Hive 0.11.0及之后版本：

     ```
     jdbc:hive2://localhost:10009/databasename
     本例：jdbc:hive2://100.xx.xxx.243:10009/tpch
     ```

图 7-27 配置 Hadoop Hive 数据连接



常用操作：SQL 作业参数设置

1. 确保kyuubi的/conf/kyuubi-defaults.conf配置打开如下参数开关。

   ```
   kyuubi.engine.dli.set.conf.transform.to.annotation=true
   ```

2. 在sql语句的末尾添加注解参数。

   ```
   set 参数
   示例：
   ```

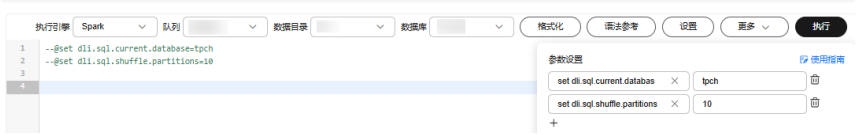


```
set dli.sql.current.database=tpch
set dli.sql.shuffle.partitions=10
```

图 7-28 FineBI 参数配置示例

在DLI的SQL编辑器的执行效果：Set参数会修改为注释提交到DLI侧执行。

图 7-29 在 DLI 的 SQL 编辑器查看配置的参数



7.7 配置 SuperSet 通过 Kyuubi 连接 DLI 进行数据查询和分析

Superset是一个开源的数据探索和可视化平台，支持对数据进行快速、直观的探索，同时支持创建丰富的数据可视化和交互式仪表板。

Kyuubi是一个分布式 SQL 查询引擎，它提供了标准的SQL接口，使用户能够方便地访问和分析存储在大数据平台中的数据。

通过将Superset与Kyuubi对接，用户可以利用Kyuubi访问DLI进行数据查询和分析。这种集成简化了数据访问流程，提供了数据的统一管理和分析能力，使得用户能够更深入地洞察数据。

本节操作介绍Superset基于Kyuubi连接DLI，以访问和分析DLI中的数据的操作步骤。

操作流程

图 7-30 操作流程

- **步骤1：安装并配置Kyuubi连接DLI**：安装并配置Kyuubi，确保Kyuubi可以连接到DLI。
- **步骤2：安装SuperSet并配置数据连接驱动**：配置Superset安装数据连接驱动。
- **步骤3：配置SuperSet连接kyuubi**：在BI工具中创建一个新的数据连接，通过JDBC连接Kyuubi。

步骤 1：安装并配置 Kyuubi 连接 DLI

如需使用外网访问Kyuubi请确保弹性云服务器绑定弹性公网IP，并配置安全组入方向开启10009和3309端口。

步骤1 安装JDK。

在安装和使用Kyuubi前，确保您的开发环境已安装JDK。

Java SDK要求使用JDK1.8或更高版本。考虑到后续版本的兼容性，推荐使用1.8版本。

1. 下载JDK。

   从**Oracle官网**下载并安装JDK1.8版本安装包。

   本例使用jdk-8u261-linux-x64.tar.gz。
2. 将jdk上传到linux服务器对应的目录下并执行解压命令，此处上传到/usr/local目录下。

   ```
   sudo tar -xzf jdk-8u261-linux-x64.tar.gz -C /usr/local/
   ```
3. 配置环境变量。

   编辑.bashrc或.profile文件，添加以下行：

   ```
   export JAVA_HOME=/usr/local/jdk-1.8.0_261
   export PATH=$PATH:$JAVA_HOME/bin
   ```
4. 执行以下命令应用环境变量。

   ```
   source ~/.bashrc
   ```
5. 执行命令**java -version**，检查是否安装成功，如下显示版本号信息说明java环境安装成功。

   ```
   java version "1.8.0_261"
   Java(TM) SE Runtime Environment (build 1.8.0_261-b12)
   Java HotSpot(TM) 64-Bit Server VM (build 25.261-b12, mixed mode)
   ```

步骤2 安装Kyuubi

1. 访问**Apache Kyuubi**的下载Kyuubi安装包。了解更多**Kyuubi安装操作**。
2. 解压下载的Kyuubi安装包。

   ```
   tar -xzf kyuubi-{version}-bin.tar.gz
   ```

3. 配置环境变量（可选）：

   将Kyuubi的bin目录添加到PATH环境变量中，确保可以在任何位置调用Kyuubi的脚本。

步骤3 配置Kyuubi连接DLI

1. 在Kyuubi的根目录下添加DLI驱动。

   在“**DLI SDK DOWNLOAD**”页面，单击Kyuubi驱动包链接，下载对应版本的驱动包。

   并将该驱动放在kyuubi根目录/externals/engines/jdbc。

   确保插件用户组和权限与其他Jar保持一致。

2. 执行以下命令修改Kyuubi配置文件。

   cd $KYUUBI_HOME/confvi kyuubi-defaults.conf

   配置项说明请参考**表7-12**。

表 7-12 kyuubi 配置参数说明

配置项	说明	是否必选	示例
kyuubi.engine.type	JDBC服务类型。这里请指定为dli。	是	jdbc
kyuubi.engine.jdbc.type	引擎类型。请使用dli。	是	dli
kyuubi.engine.jdbc.driver.class	连接JDBC服务使用的驱动类名。请使用com.huawei.dli.jdbc.DliDriver	是	com.huawei.dli.jdbc.DliDriver
kyuubi.engine.jdbc.connection.url	JDBC服务连接的URL。 格式：jdbc:dli://{dliendpoint} /{projectId}	是	jdbc:dli://{dliendpoint} /{projectId}
kyuubi.engine.jdbc.session.initialize.sql	用于指定在建立JDBC会话时执行的初始化SQL语句。	否	select 1 如果在DLI的管理控制台看到select 1，代表初始化成功。
kyuubi.frontend.protocols	用于指定Kyuubi服务支持的前端协议。Kyuubi支持多种前端协议，允许用户通过不同的接口与Kyuubi进行交互。	是	- mysql - thrift_binary

配置项	说明	是否必选	示例
kyuubi.engine.dli.schema.show.name	用于指定当用户执行show schemas或show databases语句时，Kyuubi引擎如何展示数据源接口的模式名称。 – true：表示在展示模式名称时，包含 DLI 的名称作为前缀。 – false：表示在展示模式名称时，不包含 DLI 的名称。 例如如果配置为true，并且有一个DLI名称为 hive，那么在执行show schemas时，输出为hive.default的格式。 如果配置为false，输出为default的格式。	否	– true – false
kyuubi.engine.dli.jdbc.connection.region	DLI的区域名称和服务名称。	是	regionname=ap-southeast-2
kyuubi.engine.dli.jdbc.connection.queue	DLI服务的队列名称。	是	dli_test
kyuubi.engine.dli.jdbc.connection.database	用于指定Kyuubi引擎通过JDBC连接到DLI数据源时默认使用的数据库名称。	是	tpch
kyuubi.engine.dli.jdbc.connection.ak	AK/SK认证密钥。 如果使用AK/SK认证方式。	是	accesskey=your-access-key
kyuubi.engine.dli.jdbc.connection.sk	DLI的区域名称和服务名称。 如果使用AK/SK认证方式时配置。	是	secretkey=your-secret-key
kyuubi.engine.dli.jdbc.connection.project	DLI资源所在的项目ID。	是	0b33ea2a7e0010802fe4c009bb05076d
kyuubi.engine.dli.sql.limit.time.sec	SQL查询的执行时间限制。 默认600s	否	300

配置项	说明	是否必选	示例
kyuubi.engine.dli.result.line.num.limit	SQL查询的返回的最大条数。 默认返回10万条。 配置为-1代表不限制返回的条数。	是	50000
kyuubi.engine.dli.small.file.merge	配置是否开启小文件自动合并。默认为false，代表不开启。 – true：开启 – false：不开启	是	true
kyuubi.engine.dli.bi.type	用于指定BI工具类型。 支持fine/ grafana/ superset/ tableau/ power/dbt/yongHong	是	fine
kyuubi.engine.dli.boolean.type.to.int	定义DLI的Boolean类型数据是以1/0返回，还是true/false返回 当BI工具类型为Grafana时，需要设置为true。 – true：按1/0返回（1：代表true，0：fales）。 – false：按true/false返回。 默认取值false。	否	false
kyuubi.engine.dli.set.conf.transform.to.annotation	支持在SQL中设置set spark参数。 PowerBI、FineBI、SuperSet、DBT需要设置为true。	否	true
kyuubi.engine.dli.set.conf.sql.suffix	支持在SQL中尾端设置set spark参数。 PowerBI、DBT需要设置为true。	否	true
kyuubi.engine.dli.result.cache.enable	是否开启库表数据缓存，开启后自动缓存库表信息。默认为true。 – true：开启 – false：不开启	否	true

配置项	说明	是否必选	示例
kyuubi.engine.dli.cache.limit.line.num	配置缓存的最大条数。 默认缓存10万条。 配置为-1代表不限制缓存的最大条数。	否	1000
kyuubi.engine.dli.cache.time.sec	配置缓存的时间。 默认为1800s。	否	1800
kyuubi.operation.incremental.collect	kyuubi会预加载select结果数据到缓存加快读取数据，数据量较大的场景防止内存OOM建议关闭。	否	false 配置为false代表关闭预加载。
kyuubi.engine.jdbc.memory	jdbc engine进程内存 默认为1g，建议改成5g以上加大jdbc engine进程内存使用	否	5g

3. 快速启动kyuubi。

   进入云服务器的根目录/bin执行以下命令启动kyuubi。

   ```
   cd /bin
   ./kyuubi start  restart
   ```

   连接成功后，可以执行SQL查询来测试Kyuubi与DLI的连接是否正常工作。

步骤4 （可选）配置主机的host文件提高Kyuubi的访问效率

为了提高Kyuubi的访问效率，建议在主机的/etc/hosts 配置Kyuubi主机IP的映射关系。

1. 执行ifconfig查看主机IP地址。

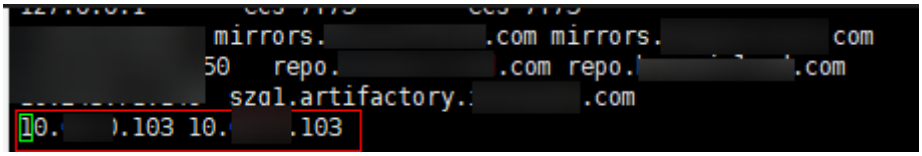
   图 7-31 查看主机 IP 地址

   ```
   [root@ecs-7f75 ~]# ifconfig
   eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
           inet 10.6    .103  netmask 255.255.255.0  broadcast 10.63.0.255
           inet6 fe80::f816:3eff:fefd:3a50  prefixlen 64  scopeid 0x20<link>
           ether fa:16:3e:fd:3a:50  txqueuelen 1000  (Ethernet)
           RX packets 6654431  bytes 2845894229 (2.6 GiB)
           RX errors 0  dropped 0  overruns 0  frame 0
           TX packets 4585886  bytes 1125818425 (1.0 GiB)
           TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

   lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
           inet 127.0.0.1  netmask 255.0.0.0
           inet6 ::1  prefixlen 128  scopeid 0x10<host>
           loop  txqueuelen 1000  (Local Loopback)
           RX packets 1502680  bytes 307935807 (293.6 MiB)
           RX errors 0  dropped 0  overruns 0  frame 0
           TX packets 1502680  bytes 307935807 (293.6 MiB)
           TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
   ```

2. 将该IP配置在/etc/hosts文件中。

图 7-32 在/etc/hosts 文件中配置 IP 地址



----结束

步骤 2：安装 SuperSet 并配置数据连接驱动

步骤1 下载并安装SuperSet。

详细安装操作指导请参考**安装SuperSet**

以**Docker安装Superset**为例：

1. 安装Docker：
   确保当前主机系统上安装了Docker。
2. 拉取Superset Docker镜像：
   ```
   docker pull apache/superset
   ```
3. 启动Superset容器：
   ```
   docker run -p 8088:8088 apache/superset
   ```
   启动Superset容器，并将容器的8088端口映射到宿主机的8088端口。
4. 访问Superset：
   在浏览器中访问 http://localhost:8088，并使用默认的用户名和密码登录（通常为admin/admin）。

步骤2 下载数据驱动。获取**Apach Hive驱动包**，推荐使用pyhive 0.7.0版本。

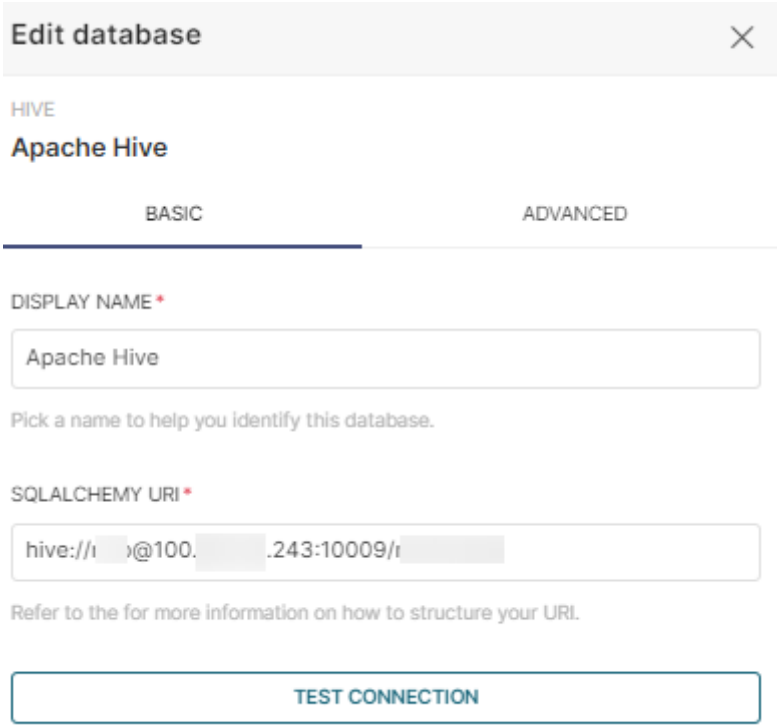
安装操作指导请参考**安装Hive驱动**。

----结束

步骤 3：配置 SuperSet 连接 kyuubi

1. 打开并登录Superset。
2. 单击“Data > Databases”。
3. 单击“Add Database”。
   在弹出的Database窗口中，选择在步骤**步骤2：安装SuperSet并配置数据连接驱动**安装的驱动。
4. 配置数据连接的信息。
   - DISPLAY NAME：自定义数据连接名称。
   - SQL ALCHEMY URI：配置数据连接的URL
     ```
     数据库类型://username:password@host:port/database
     本例：hive://username:password@100.xx.xxx.243:10009/tpch
     ```


图 7-33 SuperSet 配置数据连接

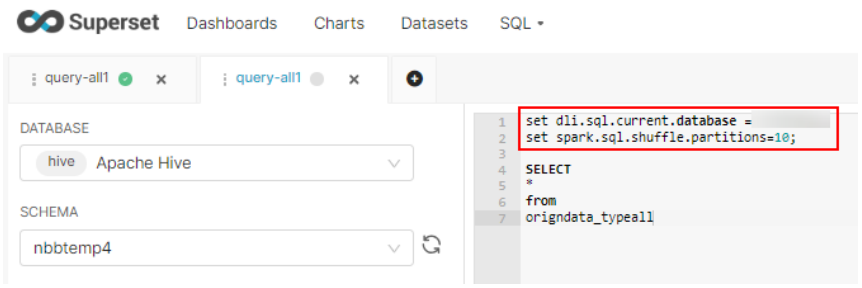


5. 单击“TEST connection”测试数据源连接是否成功，如果连接正常单击“OK”保存连接。

常用操作：SQL 作业参数设置

1. 确保kyuubi的/conf/kyuubi-defaults.conf配置打开如下参数开关。
   ```
   kyuubi.engine.dli.set.conf.transform.to.annotation=true
   ```
2. 在sql语句的末尾添加注解参数。
   ```
   set 参数
   示例:
   set dli.sql.current.database=tpch
   set dli.sql.shuffle.partitions=10
   ```

图 7-34 SuperSet 参数配置示例



在DLI的SQL编辑器的执行效果：Set参数会修改为注释提交到DLI侧执行。

图 7-35 在 DLI 的 SQL 编辑器查看配置的参数

7.8 配置 Tableau 通过 Kyuubi 连接 DLI 进行数据查询和分析

Tableau是一款数据分析和可视化工具，支持通过拖放式界面连接到各种数据源，创建交互式和共享式的数据可视化，从而将数据转化为可操作的见解。

Kyuubi是一个分布式 SQL 查询引擎，它提供了标准的SQL接口，使用户能够方便地访问和分析存储在大数据平台中的数据。

通过将Tableau与Kyuubi对接，用户可以利用Kyuubi访问DLI进行数据查询和分析。这种集成简化了数据访问流程，提供了数据的统一管理和分析能力，使得用户能够更深入地洞察数据。

本节操作介绍Tableau基于Kyuubi连接DLI，以访问和分析DLI中的数据的操作步骤。

操作流程

图 7-36 操作流程

- **步骤1：安装并配置Kyuubi连接DLI**：安装并配置Kyuubi，确保Kyuubi可以连接到DLI。
- **步骤2：配置ODBC连接Kyuubi**：配置Superset安装数据连接驱动。
- **步骤3：配置Tableau使用ODBC连接到Kyuubi**：在BI工具中创建一个新的数据连接，通过JDBC连接Kyuubi。

步骤 1：安装并配置 Kyuubi 连接 DLI

如需使用外网访问Kyuubi请确保弹性云服务器绑定弹性公网IP，并配置安全组入方向开启10009和3309端口。

步骤1 安装JDK。

在安装和使用Kyuubi前，确保您的开发环境已安装JDK。

Java SDK要求使用JDK1.8或更高版本。考虑到后续版本的兼容性，推荐使用1.8版本。

1. 下载JDK。

   从**Oracle官网**下载并安装JDK1.8版本安装包。

   本例使用jdk-8u261-linux-x64.tar.gz。

2. 将jdk上传到linux服务器对应的目录下并执行解压命令，此处上传到/usr/local目录下。

   ```
   sudo tar -xzf jdk-8u261-linux-x64.tar.gz -C /usr/local/
   ```

3. 配置环境变量。

   编辑.bashrc或.profile文件，添加以下行：

   ```
   export JAVA_HOME=/usr/local/jdk-1.8.0_261
   export PATH=$PATH:$JAVA_HOME/bin
   ```

4. 执行以下命令应用环境变量。

   ```
   source ~/.bashrc
   ```

5. 执行命令**java -version**，检查是否安装成功，如下显示版本号信息说明java环境安装成功。

   ```
   java version "1.8.0_261"
   Java(TM) SE Runtime Environment (build 1.8.0_261-b12)
   Java HotSpot(TM) 64-Bit Server VM (build 25.261-b12, mixed mode)
   ```

步骤2 安装Kyuubi

1. 访问**Apache Kyuubi**的下载Kyuubi安装包。了解更多**Kyuubi安装操作**。
2. 解压下载的Kyuubi安装包。

   ```
   tar -xzf kyuubi-{version}-bin.tar.gz
   ```

3. 配置环境变量（可选）：

   将Kyuubi的bin目录添加到PATH环境变量中，确保可以在任何位置调用Kyuubi的脚本。

步骤3 配置Kyuubi连接DLI

1. 在Kyuubi的根目录下添加DLI驱动。

   在“**DLI SDK DOWNLOAD**”页面，单击Kyuubi驱动包链接，下载对应版本的驱动包。

   并将该驱动放在kyuubi根目录/externals/engines/jdbc。

   确保插件用户组和权限与其他Jar保持一致。

2. 执行以下命令修改Kyuubi配置文件。

   cd $KYUUBI_HOME/confvi kyuubi-defaults.conf

   配置项说明请参考**表7-13**。

   表 7-13 kyuubi 配置参数说明

配置项	说明	是否必选	示例
kyuubi.engine.type	JDBC服务类型。这里请指定为dli。	是	jdbc
kyuubi.engine.jdbc.type	引擎类型。请使用dli。	是	dli

配置项	说明	是否必选	示例
kyuubi.engine.jdbc.driver.class	连接JDBC服务使用的驱动类名。请使用com.huawei.dli.jdbc.DliDriver	是	com.huawei.dli.jdbc.DliDriver
kyuubi.engine.jdbc.connection.url	JDBC服务连接的URL。 格式：jdbc:dli://{dliendpoint} /{projectId}	是	jdbc:dli://{dliendpoint} /{projectId}
kyuubi.engine.jdbc.session.initialize.sql	用于指定在建立JDBC会话时执行的初始化SQL语句。	否	select 1 如果在DLI的管理控制台看到select 1，代表初始化成功。
kyuubi.frontend.protocols	用于指定Kyuubi服务支持的前端协议。Kyuubi支持多种前端协议，允许用户通过不同的接口与Kyuubi进行交互。	是	– mysql – thrift_binary
kyuubi.engine.dli.schema.show.name	用于指定当用户执行show schemas或show databases语句时，Kyuubi引擎如何展示数据源接口的模式名称。 – true：表示在展示模式名称时，包含 DLI 的名称作为前缀。 – false：表示在展示模式名称时，不包含 DLI 的名称。 例如如果配置为true，并且有一个DLI名称为 hive，那么在执行show schemas时，输出为hive.default的格式。 如果配置为false，输出为default的格式。	否	– true – false
kyuubi.engine.dli.jdbc.connection.region	DLI的区域名称和服务名称。	是	regionname=ap-southeast-2
kyuubi.engine.dli.jdbc.connection.queue	DLI服务的队列名称。	是	dli_test
kyuubi.engine.dli.jdbc.connection.database	用于指定Kyuubi引擎通过JDBC连接到DLI数据源时默认使用的数据库名称。	是	tpch

配置项	说明	是否必选	示例
kyuubi.engine.dli.jdbc.connection.ak	AK/SK认证密钥。 如果使用AK/SK认证方式。	是	accesskey=your-access-key
kyuubi.engine.dli.jdbc.connection.sk	DLI的区域名称和服务名称。 如果使用AK/SK认证方式时配置。	是	secretkey=your-secret-key
kyuubi.engine.dli.jdbc.connection.project	DLI资源所在的项目ID。	是	0b33ea2a7e0010802fe4c009bb05076d
kyuubi.engine.dli.sql.limit.time.sec	SQL查询的执行时间限制。 默认600s	否	300
kyuubi.engine.dli.result.line.num.limit	SQL查询的返回的最大条数。 默认返回10万条。 配置为-1代表不限制返回的条数。	是	50000
kyuubi.engine.dli.small.file.merge	配置是否开启小文件自动合并。默认为false，代表不开启。 – true：开启 – false：不开启	是	true
kyuubi.engine.dli.bi.type	用于指定BI工具类型。 支持fine/ grafana/ superset/ tableau/ power/dbt/yongHong	是	fine
kyuubi.engine.dli.boolean.type.to.int	定义DLI的Boolean类型数据是以1/0返回，还是true/false返回 当BI工具类型为Grafana时，需要设置为true。 – true：按1/0返回（1：代表true，0：fales）。 – false：按true/false返回。 默认取值false。	否	false

配置项	说明	是否必选	示例
kyuubi.engine.dli.set.conf.transform.to.annotation	支持在SQL中设置set spark参数。 PowerBI、FineBI、SuperSet、DBT需要设置为true。	否	true
kyuubi.engine.dli.set.conf.sql.suffix	支持在SQL中尾端设置set spark参数。 PowerBI、DBT需要设置为true。	否	true
kyuubi.engine.dli.result.cache.enable	是否开启库表数据缓存，开启后自动缓存库表信息。默认为true。 – true：开启 – false：不开启	否	true
kyuubi.engine.dli.cache.limit.line.num	配置缓存的最大条数。 默认缓存10万条。 配置为-1代表不限制缓存的最大条数。	否	1000
kyuubi.engine.dli.cache.time.sec	配置缓存的时间。 默认为1800s。	否	1800
kyuubi.operation.incremental.collect	kyuubi会预加载select结果数据到缓存加快读取数据，数据量较大的场景防止内存OOM建议关闭。	否	false 配置为false代表关闭预加载。
kyuubi.engine.jdbc.memory	jdbc engine进程内存 默认为1g，建议改成5g以上加大jdbc engine进程内存使用	否	5g

3. 快速启动kyuubi。

   进入云服务器的根目录/bin执行以下命令启动kyuubi。

   ```
   cd /bin
   ./kyuubi start  restart
   ```

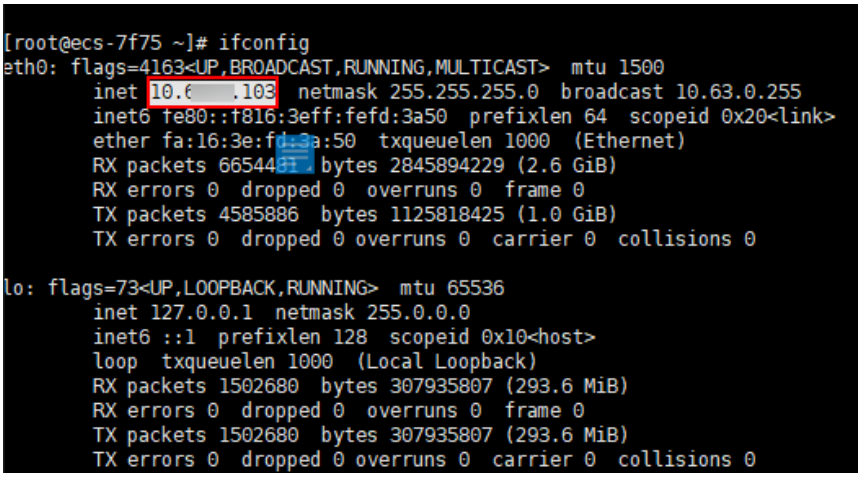
   连接成功后，可以执行SQL查询来测试Kyuubi与DLI的连接是否正常工作。

步骤4 （可选）配置主机的host文件提高Kyuubi的访问效率

为了提高Kyuubi的访问效率，建议在主机的/etc/hosts 配置Kyuubi主机IP的映射关系。

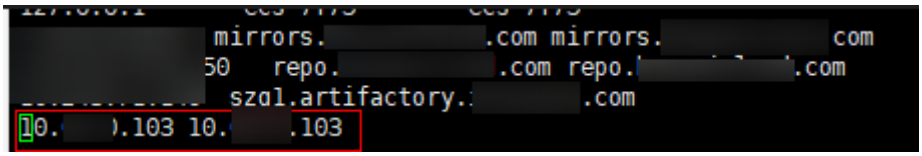
1. 执行ifconfig查看主机IP地址。

图 7-37 查看主机 IP 地址



2. 将该IP配置在/etc/hosts文件中。

图 7-38 在/etc/hosts 文件中配置 IP 地址



----结束

步骤 2：配置 ODBC 连接 Kyuubi

步骤1 安装ODBC驱动

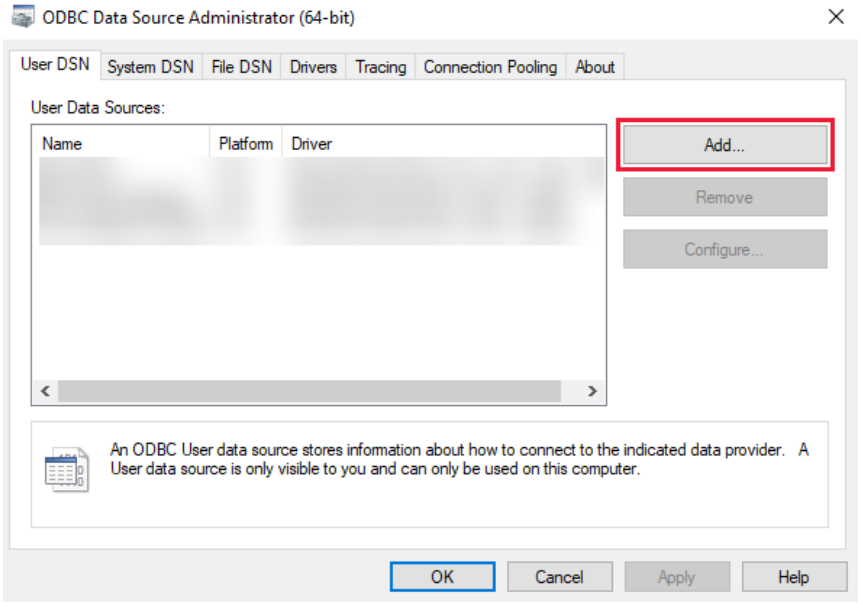
根据数据库类型，需要在本地主机上安装相应的ODBC驱动。本例使用Hive数据库类型。

- Cloudera Hive ODBC，推荐使用v2.5.12。
- Microsoft Hive ODBC，推荐使用v2.6.12.1012。

步骤2 配置ODBC连接Kyuubi

1. 在Windows系统中，打开“控制面板 > 管理工具 > 数据源 (ODBC)”。
2. 配置新的ODBC数据源。
   a. 在ODBC中单击“User DSN”。
   b. 单击“Add”创建新的数据源。
   c. 选择Hive ODBC Driver，单击“OK”。

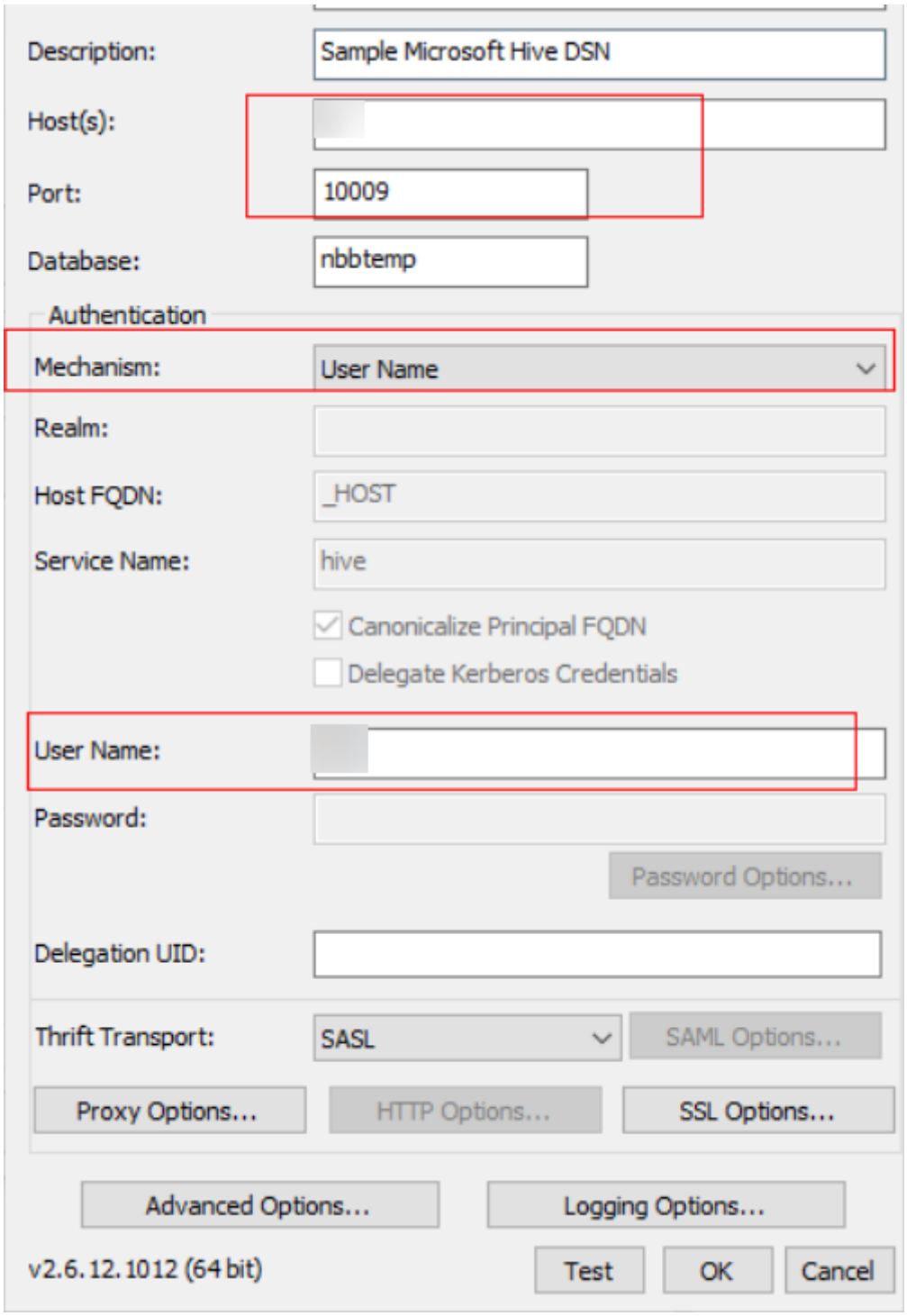
图 7-39 ODBC 新建数据源连接



3. 在创建的新数据源配置界面中，输入Kyuubi服务器的相关信息。
   - 数据库名称：本例输入DLI数据库名称。
   - 服务器地址：输入Kyuubi服务器的弹性公网IP地址。
   - 端口号：Kyuubi服务监听的端口，使用Hive Thirft协议，默认端口10009。
   - 用户名和密码：按需配置Kyuubi服务器用户名和密码。

   按需配置其他高级选项，然后保存配置。

图 7-40 ODBC 配置数据源连接信息



4. 单击“Test”测试数据源连接是否成功，如果连接正常单击“OK”保存连接。

----结束

步骤 3：配置 Tableau 使用 ODBC 连接到 Kyuubi

1. 单击并安装Tableau。获取**Tableau安装包**。
2. 打开Tableau。

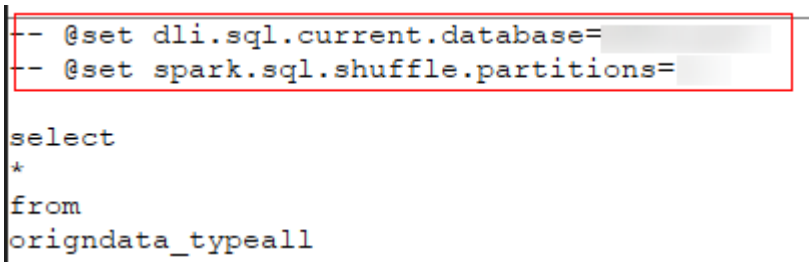
3. 在开始页面的“连接”窗格中，选择你想要连接的数据源类型。本例选择Hive类型的数据连接。
4. 配置数据连接信息。
   - 连接：Hive
   - 服务器：Kyuubi主机的IP地址。
   - 端口：连接Kyuubi的端口，Hive Thrift协议对接，默认端口10009。
   - 身份验证：本例选择用户名的认证方式。
   - 用户名：Kyuubi用户名。
5. 单击“登录”连接Kyuubi。

常用操作：SQL 作业参数设置

在sql语句的末尾添加注解参数。

```
-- @set 参数
示例：
-- @set dli.sql.current.database=tpch
-- @set dli.sql.shuffle.partitions=10
```

图 7-41 Tableau 参数配置示例



在DLI的SQL编辑器的执行效果：Set参数会修改为注释提交到DLI侧执行。

图 7-42 在 DLI 的 SQL 编辑器查看配置的参数

7.9 配置 Beeline 通过 Kyuubi 连接 DLI 进行数据查询和分析

Beeline是数据分析师和数据工程师的重要工具之一，适用于大规模数据处理的场景。Beeline提供了的SQL引擎，使得用户可以使用SQL的语言来执行数据查询、数据分析和管理任务。

Kyuubi是一个分布式 SQL 查询引擎，它提供了标准的SQL接口，使用户能够方便地访问和分析存储在大数据平台中的数据。

通过将Beeline与Kyuubi对接，用户可以利用Kyuubi访问DLI进行数据查询和分析。这种集成简化了数据访问流程，提供了数据的统一管理和分析能力，使得用户能够更深入地洞察数据。

本节操作介绍Beeline基于Kyuubi连接DLI，以访问和分析DLI中的数据的操作步骤。

操作流程

图 7-43 操作流程

- **步骤1：安装并配置Kyuubi连接DLI**：安装并配置Kyuubi，确保Kyuubi可以连接到DLI。
- **步骤2：配置Beeline连接Kyuubi**：在BI工具中创建一个新的数据连接，通过JDBC连接Kyuubi。

步骤 1：安装并配置 Kyuubi 连接 DLI

如需使用外网访问Kyuubi请确保弹性云服务器绑定弹性公网IP，并配置安全组入方向开启10009和3309端口。

步骤1 安装JDK。

在安装和使用Kyuubi前，确保您的开发环境已安装JDK。

Java SDK要求使用JDK1.8或更高版本。考虑到后续版本的兼容性，推荐使用1.8版本。

1. 下载JDK。

   从**Oracle官网**下载并安装JDK1.8版本安装包。

   本例使用jdk-8u261-linux-x64.tar.gz。
2. 将jdk上传到linux服务器对应的目录下并执行解压命令，此处上传到/usr/local目录下。

   ```
   sudo tar -xzf jdk-8u261-linux-x64.tar.gz -C /usr/local/
   ```
3. 配置环境变量。

   编辑.bashrc或.profile文件，添加以下行：

   ```
   export JAVA_HOME=/usr/local/jdk-1.8.0_261
   export PATH=$PATH:$JAVA_HOME/bin
   ```
4. 执行以下命令应用环境变量。

   ```
   source ~/.bashrc
   ```
5. 执行命令**java -version**，检查是否安装成功，如下显示版本号信息说明java环境安装成功。

   ```
   java version "1.8.0_261"
   Java(TM) SE Runtime Environment (build 1.8.0_261-b12)
   Java HotSpot(TM) 64-Bit Server VM (build 25.261-b12, mixed mode)
   ```

步骤2 安装Kyuubi

1. 访问**Apache Kyuubi**的下载Kyuubi安装包。了解更多**Kyuubi安装操作**。
2. 解压下载的Kyuubi安装包。

   ```
   tar -xzf kyuubi-{version}-bin.tar.gz
   ```

3. 配置环境变量（可选）：

   将Kyuubi的bin目录添加到PATH环境变量中，确保可以在任何位置调用Kyuubi的脚本。

步骤3 配置Kyuubi连接DLI

1. 在Kyuubi的根目录下添加DLI驱动。

   在“**DLI SDK DOWNLOAD**”页面，单击Kyuubi驱动包链接，下载对应版本的驱动包。

   并将该驱动放在kyuubi根目录/externals/engines/jdbc。

   确保插件用户组和权限与其他Jar保持一致。

2. 执行以下命令修改Kyuubi配置文件。

   cd $KYUUBI_HOME/confvi kyuubi-defaults.conf

   配置项说明请参考**表7-14**。

表 7-14 kyuubi 配置参数说明

配置项	说明	是否必选	示例
kyuubi.engine.type	JDBC服务类型。这里请指定为dli。	是	jdbc
kyuubi.engine.jdbc.type	引擎类型。请使用dli。	是	dli
kyuubi.engine.jdbc.driver.class	连接JDBC服务使用的驱动类名。请使用com.huawei.dli.jdbc.DliDriver	是	com.huawei.dli.jdbc.DliDriver
kyuubi.engine.jdbc.connection.url	JDBC服务连接的URL。 格式：jdbc:dli://{dliendpoint} /{projectId}	是	jdbc:dli://{dliendpoint} /{projectId}
kyuubi.engine.jdbc.session.initialize.sql	用于指定在建立JDBC会话时执行的初始化SQL语句。	否	select 1 如果在DLI的管理控制台看到select 1，代表初始化成功。
kyuubi.frontend.protocols	用于指定Kyuubi服务支持的前端协议。Kyuubi支持多种前端协议，允许用户通过不同的接口与Kyuubi进行交互。	是	- mysql - thrift_binary

配置项	说明	是否必选	示例
kyuubi.engine.dli.schema.show.name	用于指定当用户执行show schemas或show databases语句时，Kyuubi引擎如何展示数据源接口的模式名称。 – true：表示在展示模式名称时，包含 DLI 的名称作为前缀。 – false：表示在展示模式名称时，不包含 DLI 的名称。 例如如果配置为true，并且有一个DLI名称为 hive，那么在执行show schemas时，输出为hive.default的格式。 如果配置为false，输出为default的格式。	否	– true – false
kyuubi.engine.dli.jdbc.connection.region	DLI的区域名称和服务名称。	是	regionname=ap-southeast-2
kyuubi.engine.dli.jdbc.connection.queue	DLI服务的队列名称。	是	dli_test
kyuubi.engine.dli.jdbc.connection.database	用于指定Kyuubi引擎通过JDBC连接到DLI数据源时默认使用的数据库名称。	是	tpch
kyuubi.engine.dli.jdbc.connection.ak	AK/SK认证密钥。 如果使用AK/SK认证方式。	是	accesskey=your-access-key
kyuubi.engine.dli.jdbc.connection.sk	DLI的区域名称和服务名称。 如果使用AK/SK认证方式时配置。	是	secretkey=your-secret-key
kyuubi.engine.dli.jdbc.connection.project	DLI资源所在的项目ID。	是	0b33ea2a7e0010802fe4c009bb05076d
kyuubi.engine.dli.sql.limit.time.sec	SQL查询的执行时间限制。 默认600s	否	300

配置项	说明	是否必选	示例
kyuubi.engine.dli.result.line.num.limit	SQL查询的返回的最大条数。 默认返回10万条。 配置为-1代表不限制返回的条数。	是	50000
kyuubi.engine.dli.small.file.merge	配置是否开启小文件自动合并。默认为false，代表不开启。 – true：开启 – false：不开启	是	true
kyuubi.engine.dli.bi.type	用于指定BI工具类型。 支持fine/ grafana/ superset/ tableau/ power/dbt/yongHong	是	fine
kyuubi.engine.dli.boolean.type.to.int	定义DLI的Boolean类型数据是以1/0返回，还是true/false返回 当BI工具类型为Grafana时，需要设置为true。 – true：按1/0返回（1：代表true，0：fales）。 – false：按true/false返回。 默认取值false。	否	false
kyuubi.engine.dli.set.conf.transform.to.annotation	支持在SQL中设置set spark参数。 PowerBI、FineBI、SuperSet、DBT需要设置为true。	否	true
kyuubi.engine.dli.set.conf.sql.suffix	支持在SQL中尾端设置set spark参数。 PowerBI、DBT需要设置为true。	否	true
kyuubi.engine.dli.result.cache.enable	是否开启库表数据缓存，开启后自动缓存库表信息。默认为true。 – true：开启 – false：不开启	否	true

配置项	说明	是否必选	示例
kyuubi.engine.dli.cache.limit.line.num	配置缓存的最大条数。 默认缓存10万条。 配置为-1代表不限制缓存的最大条数。	否	1000
kyuubi.engine.dli.cache.time.sec	配置缓存的时间。 默认为1800s。	否	1800
kyuubi.operation.incremental.collect	kyuubi会预加载select结果数据到缓存加快读取数据，数据量较大的场景防止内存OOM建议关闭。	否	false 配置为false代表关闭预加载。
kyuubi.engine.jdbc.memory	jdbc engine进程内存 默认为1g，建议改成5g以上加大jdbc engine进程内存使用	否	5g

3. 快速启动kyuubi。

   进入云服务器的根目录/bin执行以下命令启动kyuubi。

   ```
   cd /bin
   ./kyuubi start  restart
   ```

   连接成功后，可以执行SQL查询来测试Kyuubi与DLI的连接是否正常工作。

步骤4 （可选）配置主机的host文件提高Kyuubi的访问效率

为了提高Kyuubi的访问效率，建议在主机的/etc/hosts 配置Kyuubi主机IP的映射关系。

1. 执行ifconfig查看主机IP地址。

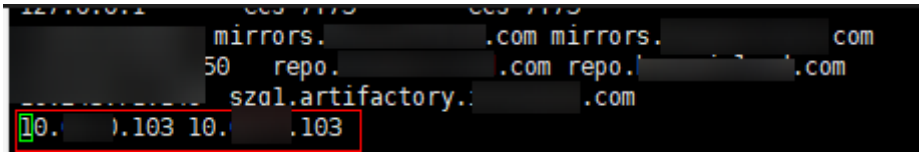
   图 7-44 查看主机 IP 地址

   ```
   [root@ecs-7f75 ~]# ifconfig
   eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
           inet 10.6    .103  netmask 255.255.255.0  broadcast 10.63.0.255
           inet6 fe80::f816:3eff:fefd:3a50  prefixlen 64  scopeid 0x20<link>
           ether fa:16:3e:fd:3a:50  txqueuelen 1000  (Ethernet)
           RX packets 6654431  bytes 2845894229 (2.6 GiB)
           RX errors 0  dropped 0  overruns 0  frame 0
           TX packets 4585886  bytes 1125818425 (1.0 GiB)
           TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

   lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
           inet 127.0.0.1  netmask 255.0.0.0
           inet6 ::1  prefixlen 128  scopeid 0x10<host>
           loop  txqueuelen 1000  (Local Loopback)
           RX packets 1502680  bytes 307935807 (293.6 MiB)
           RX errors 0  dropped 0  overruns 0  frame 0
           TX packets 1502680  bytes 307935807 (293.6 MiB)
           TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
   ```

2. 将该IP配置在/etc/hosts文件中。

图 7-45 在/etc/hosts 文件中配置 IP 地址



----结束

步骤 2：配置 Beeline 连接 Kyuubi

您可以用Kyuubi Beeline工具连接Kyuubi Server。

```
kyuubi-beeline -n user1 -u "jdbc:hive2://<kyuubi-server-host>:<port>/"
```

- <kyuubi-server-host> 是Kyuubi Server的主机名或IP地址。
- <port> 是Kyuubi Server监听的端口，默认是10009。

配置样例：

```
kyuubi-beeline -n user1 -u "jdbc:hive2://kyuubi-server-1:10009/"
```

常用操作：SQL 作业参数设置

1. 确保kyuubi的/conf/kyuubi-defaults.conf配置打开如下参数开关。
   ```
   kyuubi.engine.dli.set.conf.transform.to.annotation=true
   ```
2. 在sql语句的末尾添加注解参数。
   ```
   set 参数
   示例：
   set dli.sql.current.database=tpch
   set dli.sql.shuffle.partitions=10
   ```

在DLI的SQL编辑器的执行效果：Set参数会修改为注释提交到DLI侧执行。

图 7-46 在 DLI 的 SQL 编辑器查看配置的参数