ALIBABA CLOUD

阿里云

DataWorks

教程

文档版本：20220104

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.简单用户画像分析（MaxCompute版）

1.1. Workshop介绍

本模块为您介绍DataWorks的设计思路和核心功能，帮助您深入了解阿里云DataWorks。

教程概述

教程时长：2小时，采用在线学习的方式。

教程对象：面向Java工程师、产品运营等DataWorks所有的新老用户。只需要熟悉标准SQL，无需对数据仓库和MaxCompute的原理过多了解，即可快速掌握DataWorks的基本技能。建议您进一步学习DataWorks教程，深入了解DataWorks的基本概念及功能，详情请参见什么是DataWorks。

教程目标：以常见的真实的海量日志数据分析任务为教程背景，争取在完成教程后，您对DataWorks的主要功能有所了解。按照教程演示内容，独立通过MaxCompute计算引擎完成数据采集 、数据开发和任务运维等数据岗位常见的任务。

开发流程

Workshop教程涉及的具体开发流程如下：

1. 环境准备：准备操作过程中需要的MaxCompute、DataWorks等环境。详情请参见准备环境。
2. 数据采集：学习如何从不同的数据源同步数据至MaxCompute中、如何快速触发任务运行、如何查看任务日志等。详情请参见采集数据。
3. 数据加工：学习如何运行数据流程图、如何新建数据表、如何新建数据流程任务节点、如何配置任务的周期调度属性。详情请参见加工数据。
4. 数据质量监控：学习如何给任务配置数据质量的监控规则，以保证任务运行的质量问题。详情请参见配置数据质量监控。
5. 数据可视化展现：学习如何通过Quick BI创建网站用户分析画像的仪表板，实现所需数据的可视化展现。详情请参见数据可视化展现。
6. 通过Function Studio开发UDF：学习如何通过Function Studio开发UDF，并将其提交至DataStudio的开发环境。详情请参见通过Function Studio开发UDF。

DataWorks简介

DataWorks是一站式大数据研发平台，上层有机融合数据集成、数据建模、数据开发、运维监控、数据管理、数据安全和数据质量等产品功能，同时与算法平台PAI打通，完善了从大数据开发到数据挖掘、机器学习的完整链路。

学习答疑

如果您在学习过程中遇到问题，请申请加入钉钉群进行咨询。

1.2. 准备环境

为保证您可以顺利完成本次实验，请您首先确保云账号已开通大数据计算服务MaxCompute和数据工场DataWorks。

前提条件

- 阿里云账号注册，详情请参见。
- 实名认证，详情请参见或。

背景信息

本次实验涉及的阿里云产品如下：

- 大数据计算服务MaxCompute
- 数据工场DataWorks

开通大数据计算服务MaxCompute

> **说明** 如果您已经开通MaxCompute，请跳过该步骤，直接创建DataWorks工作空间。

1. 登录阿里云官网，单击右上角的**登录**，输入您的阿里云账号和密码。
2. 鼠标悬停至顶部菜单栏中的**产品**，单击**大数据 > 大数据计算与分析 > MaxCompute**，进入MaxCompute产品详情页。
3. 单击**立即开通**。
4. 在购买页面，选择**地域**，并选中**服务协议**，单击**确认订单并支付**。

> **说明**
> - 购买页面默认提供的规格类型为MaxCompute按量计费标准版和DataWorks基础版。
> - MaxCompute的项目管理和查询编辑集成DataWorks的功能，因此需要同时开通DataWorks服务。DataWorks基础版为0元开通，如果您不使用数据集成、不执行调度任务，则不会产生费用。
> - 选择地域时，您需要考虑的最主要因素是MaxCompute与其它阿里云产品之间的关系。例如，ECS所在地域、数据所在地域等。

创建工作空间

> **说明** 因本实验提供的数据资源都在华东2（上海），建议您将工作空间创建在华东2（上海），以避免工作空间创建在其它区域，添加数据源时出现网络不可达的情况。

1. 使用主账号登录DataWorks控制台。
2. 在**概览**页面，单击右侧的**创建工作空间**。

   您也可以单击左侧导航栏中的**工作空间列表**，切换至相应的区域后，单击**创建工作空间**。
3. 配置**创建工作空间**对话框中的**基本配置**，单击**下一步**。

   > **说明** 本教程以标准模式的工作空间为例进行操作。

4. 进入**选择引擎**界面，勾选MaxCompute引擎后，单击**下一步**。

   DataWorks已正式商用，如果该区域没有开通，需要首先开通正式商用的服务。默认选中**数据集成**、**数据开发**、**运维中心**和**数据质量**。
5. 进入**引擎详情**页面，配置选购引擎的参数。

分类	参数	描述
MaxCompute	实例显示名称	实例显示名称不能超过27个字符，仅支持字母开头，仅包含字母、数字和下划线（_）。
	Quota组切换	Quota用于实现计算资源和磁盘配额。
	MaxCompute数据类型	该选项设置后，将在5分钟内生效。详情请参见数据类型版本说明
	MaxCompute项目名称	默认与DataWorks工作空间的名称一致。
	MaxCompute访问身份	包括阿里云主账号和任务负责人。

6. 配置完成后，单击**创建工作空间**。

   工作空间创建成功后，即可在**工作空间列表**页面查看相应内容。

1.3. 采集数据

本文为您介绍如何通过DataWorks采集日志数据至MaxCompute。

背景信息

根据本次实验模拟的场景，您需要分别创建OSS数据源和RDS数据源，并准备好相应的数据表。

> 说明
> - 您可以直接使用本实验提供的数据源，也可以使用自己的数据源。
> - 因本实验提供的数据资源在**华东2(上海)**，建议您使用华东2（上海）的工作空间。以避免工作空间创建在其它区域，添加数据源时出现网络不可达的情况。

新建OSS数据源

1. 进入**数据源管理**页面。
   i. 登录DataWorks控制台。
   ii. 在左侧导航栏，单击**工作空间列表**。
   iii. 单击相应工作空间后的**进入数据集成**。

   如果您已在DataWorks的某个功能模块，请单击左上方的☰图标，选择**全部产品 > 数据汇聚 > 数据集成**，进入**数据集成**页面。
   iv. 在左侧导航栏，单击**数据源**，进入**工作空间管理 > 数据源管理**页面。
2. 在**数据源管理**页面，单击右上方的**新增数据源**。
3. 在**新增数据源**对话框中，选择数据源类型为**OSS**。
4. 在**新增OSS数据源**对话框，配置各项参数。

参数	描述
数据源名称	输入oss_workshop_log。
数据源描述	对数据源进行简单描述。
适用环境	勾选**开发**。 **说明** 开发环境的数据源创建完成后，需要勾选**生产**，以同样方式创建生产环境的数据源，否则任务生产执行会报错。
Endpoint	输入 `http://oss-cn-shanghai-internal.aliyuncs.com` 。
Bucket	输入new-dataworks-workshop。
AccessKey ID	输入LTAI4FvGT3iU4xjKotpUMAjS。
AccessKey Secret	输入9RSUoRmNxpRC9EhC4m9PjuG7Jzy7px。

5. 在资源组列表，单击相应资源组后的**测试连通性**。

   数据同步时，一个任务只能使用一种资源组。您需要测试每个资源组的连通性，以保证同步任务使用的数据集成资源组能够与数据源连通，否则将无法正常执行数据同步任务。如果您需要同时测试多种资源组，请选中相应资源组后，单击**批量测试连通性**。详情请参见选择网络连通方案。

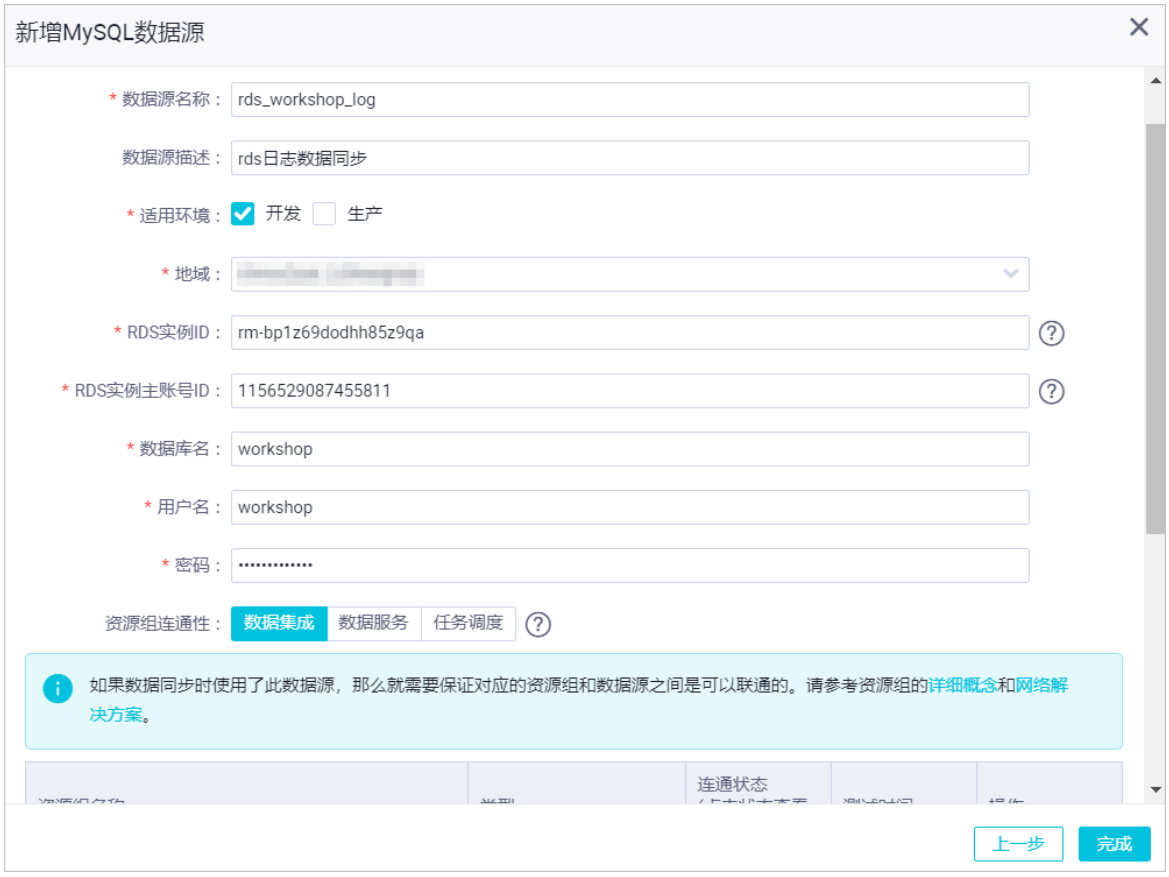
> 说明
> - （推荐）资源组列表默认仅显示独享数据集成资源组，为确保数据同步的稳定性和性能要求，推荐使用独享数据集成资源组。
> - 如果您需要测试公共资源组或自定义资源组的连通性，请在资源组列表右下方，单击**更多选项**，在**警告**对话框单击**确定**，资源组列表会显示可供选择的公共资源组和自定义资源组。

6. 连通性测试通过后，单击**完成**。

> 说明
> - 如果测试连通性失败，请检查您的AccessKey ID、AccessKey Secret和工作空间所在区域。
> - 如果您无法使用内网Endpoint连接数据源，请改用公网Endpoint。

新建RDS数据源

1. 单击当前页面左上角的图标，选择**全部产品 > 数据汇聚 > 数据集成**。
2. 在左侧导航栏，单击**数据源 > 数据源列表**，进入**工作空间管理 > 数据源管理**页面。
3. 在**数据源管理**页面，单击右上方的**新增数据源**。
4. 在**新增数据源**对话框中，选择数据源类型为**MySQL**。
5. 在**新增MySQL数据源**对话框中，配置各项参数。



参数	描述
数据源类型	选择**阿里云实例模式**。
数据源名称	输入rds_workshop_log。
数据源描述	输入RDS日志数据同步。
适用环境	勾选**开发**。 **说明** 开发环境的数据源创建完成后，需要勾选**生产**，以同样方式创建生产环境的数据源，否则任务生产执行会报错。
地区	选择RDS实例所在的区域。
RDS实例ID	输入rm-bp1z69dodhh85z9qa。
RDS实例主账号ID	输入1156529087455811。
数据库名	输入workshop。
用户名	输入workshop。
密码	输入workshop#2017。

6. 在资源组列表，单击相应资源组后的**测试连通性**。

   数据同步时，一个任务只能使用一种资源组。您需要测试每个资源组的连通性，以保证同步任务使用的数据集成资源组能够与数据源连通，否则将无法正常执行数据同步任务。如果您需要同时测试多种资源组，请选中相应资源组后，单击**批量测试连通性**。详情请参见选择网络连通方案。

   说明
   - （推荐）资源组列表默认仅显示独享数据集成资源组，为确保数据同步的稳定性和性能要求，推荐使用独享数据集成资源组。
   - 如果您需要测试公共资源组或自定义资源组的连通性，请在资源组列表右下方，单击**更多选项**，在**警告**对话框单击**确定**，资源组列表会显示可供选择的公共资源组和自定义资源组。

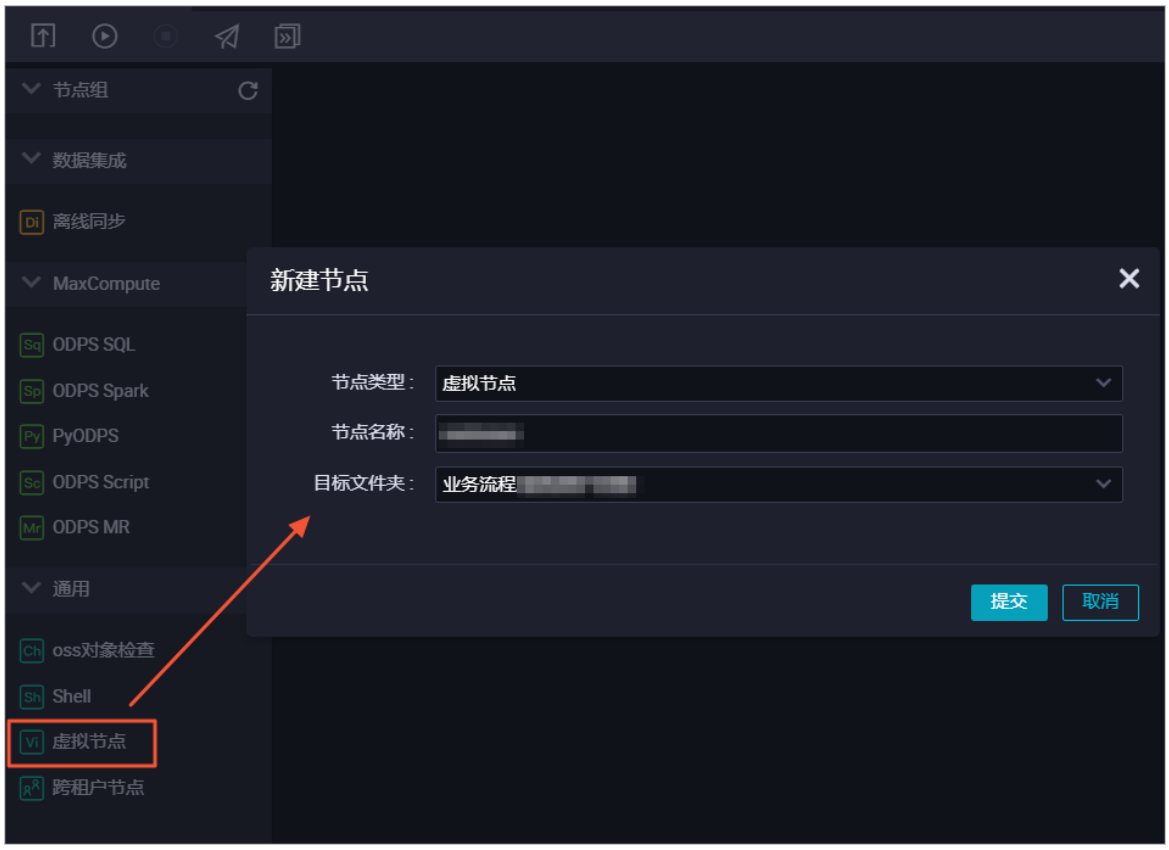
7. 测试连通性通过后，单击**完成**。

创建业务流程

1. 单击当前页面左上方的图标，选择**全部产品 > 数据开发 > DataStudio（数据开发）**。
2. 在**数据开发**面板，右键单击**业务流程**，选择**新建业务流程**。
3. 在**新建业务流程**对话框中，输入**业务名称**和**描述**。

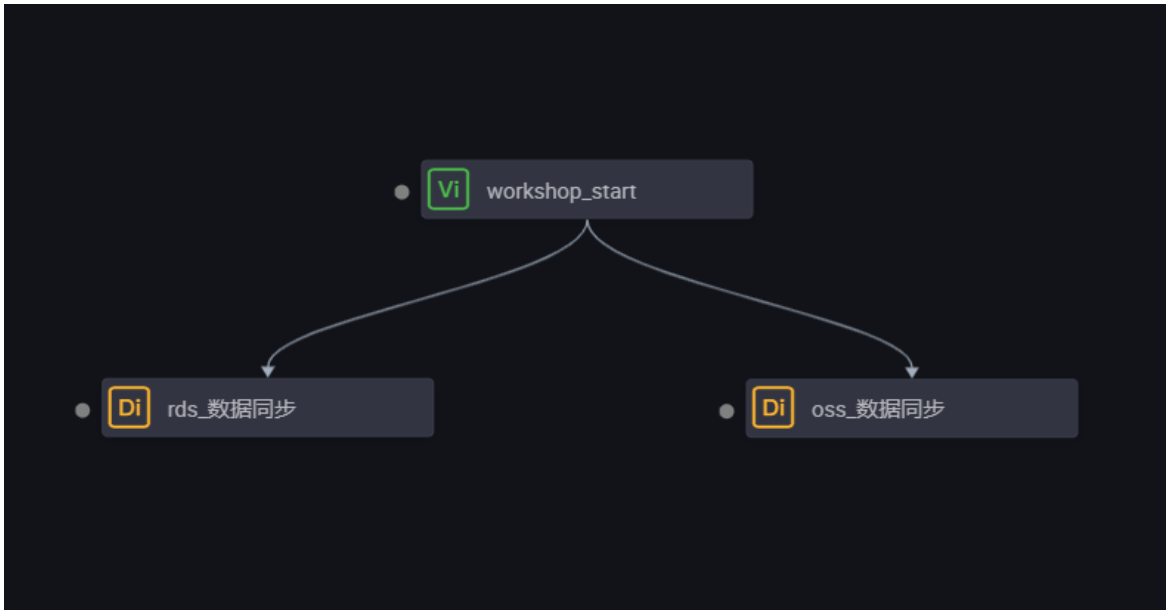
   注意 业务名称不能超过128个字符，且必须是大小写字母、中文、数字、下划线（_）以及小数点（.）。

4. 单击**新建**。
5. 进入业务流程开发面板，鼠标单击**虚拟节点**并拖拽至右侧的编辑页面。
6. 在**新建节点**对话框中，输入**节点名称**为workshop_start，单击**提交**。



以同样的方式新建两个离线同步节点，**节点名称**分别为oss_数据同步和rds_数据同步。

7. 通过拖拽连线，将workshop_start节点设置为两个离线同步节点的上游节点。



配置workshop_start节点

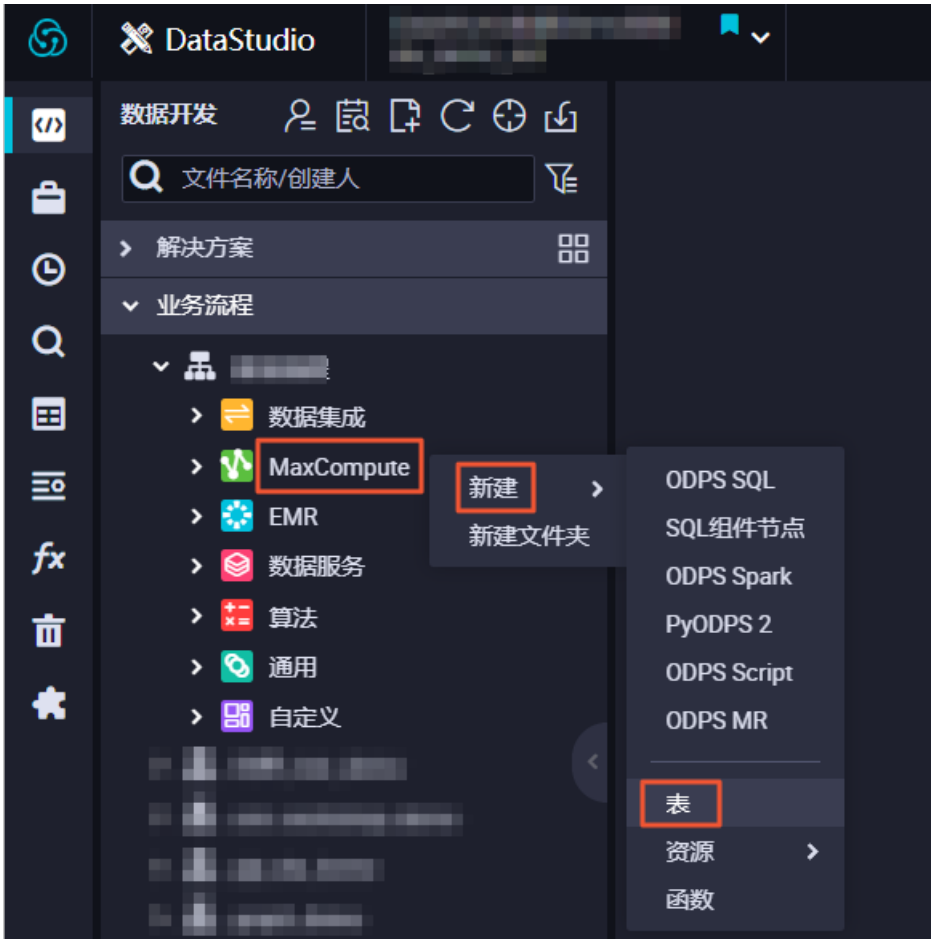
1. 在**数据开发**页面，双击相应业务流程下的虚拟节点。打开该节点的编辑页面，单击右侧的**调度配置**。
2. 在**调度依赖**区域，单击**使用工作空间根节点**，设置workshop_start节点的上游节点为工作空间根节点。

   由于新版本给每个节点都设置了输入输出节点，所以需要给workshop_start节点设置一个输入。此处设置其上游节点为工作空间根节点，通常命名为工作空间名称_root。

3. 配置完成后，单击工具栏中的图标。

新建表

1. 在**数据开发**页面打开新建的业务流程，右键单击**MaxCompute**，选择**新建 > 表**。



2. 在**新建表**对话框中，输入**表名**，单击**提交**。

   此处需要创建两张表（ods_raw_log_d和ods_user_info_d），分别存储同步过来的OSS日志数据和RDS日志数据。

   > **注意** 表名必须以字母开头，不能包含中文或特殊字符，且不能超过64个字符。

3. 通过DDL模式新建表。
   - 新建ods_raw_log_d表。

在表的编辑页面单击**DDL模式**，输入下述建表语句。


```
--创建OSS日志对应目标表
CREATE TABLE IF NOT EXISTS  ods_raw_log_d (
    col STRING
)
PARTITIONED BY (
    dt STRING
);
```

- 新建ods_user_info_d表。

  在表的编辑页面单击**DDL模式**，输入下述建表语句。

```
--创建RDS对应目标表
CREATE TABLE IF NOT  EXISTS ods_user_info_d (
    uid STRING COMMENT '用户ID',
    gender STRING COMMENT '性别',
    age_range STRING COMMENT '年龄段',
    zodiac STRING COMMENT '星座'
)
PARTITIONED BY (
    dt STRING
);
```

4. 单击**生成表结构**，并**确认**覆盖当前操作。

5. 返回建表页面，在**基本属性**中输入表的中文名。
6. 完成设置后，分别单击**提交到开发环境**和**提交到生产环境**。

配置离线同步节点

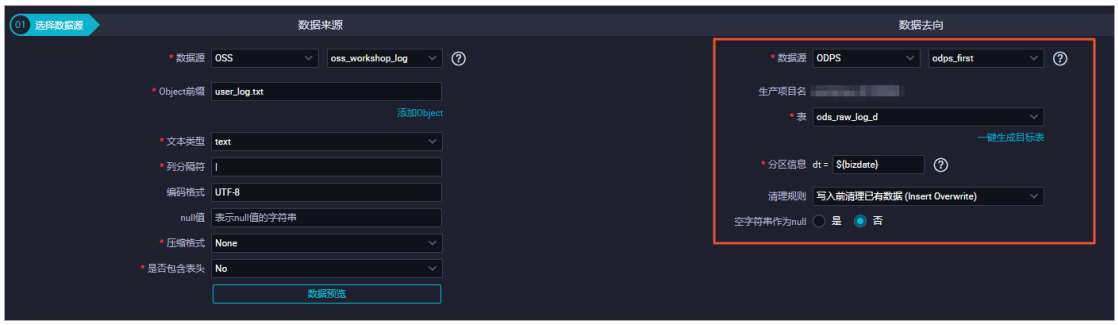
> **说明** 标准模式的工作空间下，不建议离线同步任务在开发环境下运行（开发面板直接运行），建议将其发布至生产环境后再测试运行，以获取完整的运行日志。
>
> 同时，数据产出至生产环境后，您可以申请数据权限，以读取写入开发环境中的表数据。

1. 配置oss_数据同步节点。
   i. 在**数据开发**页面，双击oss_数据同步节点，进入节点配置页面。

ii. 选择数据来源。

参数	描述
数据源	选择OSS > oss_workshop_log数据源。
Object前缀	输入OSS文件夹的路径，请勿填写Bucket的名称。示例为user_log.txt。
文本类型	选择text类型。
列分隔符	输入列分隔符为|。
编码格式	默认为UTF-8格式。
null值	表示null值的字符串。
压缩格式	包括None、Gzip、Bzip2和Zip四种类型，此处选择None。
是否包含表头	默认为No。

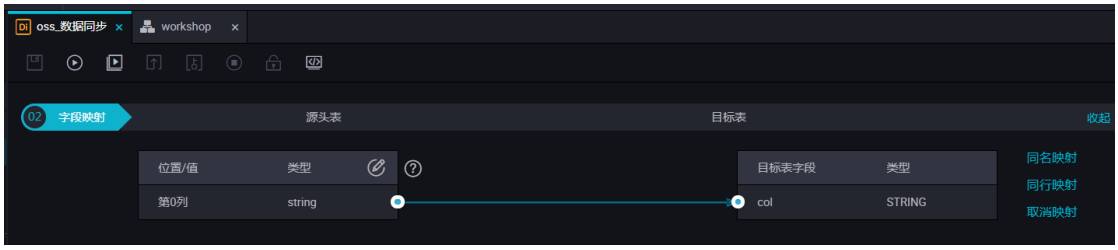
iii. 选择数据去向。



参数	描述
数据源	选择**ODPS > odps_first**数据源。
表	选择数据源中的ods_raw_log_d表。
分区信息	默认配置为${bizdate}。
清理规则	默认为**写入前清理已有数据**。
空字符串作为null	此处勾选**否**。

> 说明
> - odps_first数据源是工作空间绑定MaxCompute实例时，系统自动生成的默认数据源。
> - odps_first数据源写入至当前工作空间下的MaxCompute项目中。

iv. 配置字段映射。

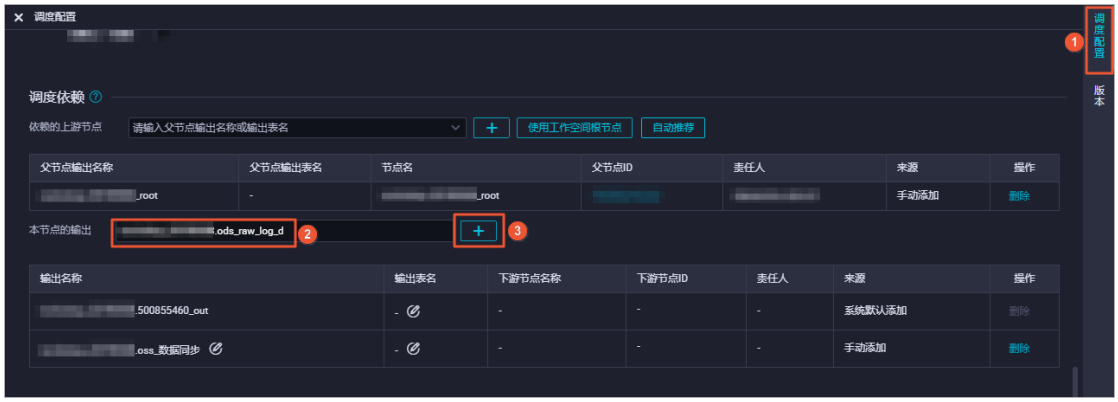


v. 配置**通道控制**。

参数	描述
任务期望最大并发数	数据同步任务内，可以从源并行读取或并行写入数据存储端的最大线程数。向导模式通过界面化配置并发数，指定任务所使用的并行度。
同步速率	设置同步速率可以保护读取端数据库，以避免抽取速度过大，给源库造成太大的压力。同步速率建议限流，结合源库的配置，请合理配置抽取速率。
错误记录数	错误记录数，表示脏数据的最大容忍条数。

vi. 单击页面右侧的**调度配置**，在**调度依赖 > 本节点的输出**区域，输入本节点的输出名称为工作空间名称.ods_raw_log_d，单击+图标。

> 注意　不建议您使用中文作为本节点输出名称，会减少自动推荐功能的准确性。



vii. 确认当前节点的配置无误后，单击工具栏中的图标。

viii. 关闭当前任务，返回业务流程配置面板。

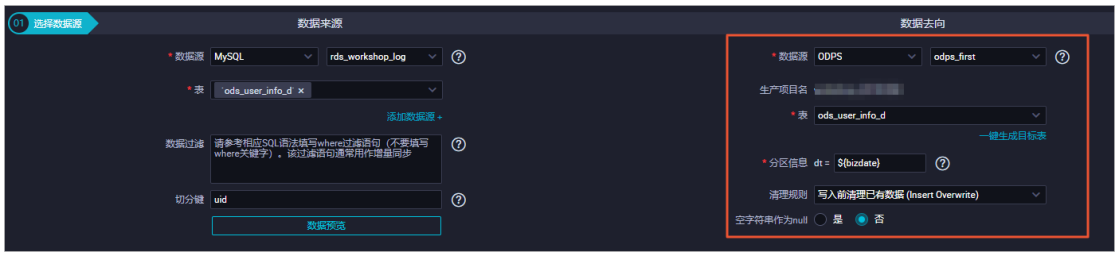
2. 配置rds_数据同步节点。

i. 在**数据开发**页面，双击rds_数据同步节点，进入节点配置页面。

ii. 选择数据来源。

参数	描述
数据源	选择MySQL > rds_workshop_log数据源。
表	选择数据源中的ods_user_info_d表。
数据过滤	该数据过滤语句通常用作增量同步，此处可以不填。
切分键	默认为uid。

iii. 选择数据去向。



参数	描述
数据源	选择ODPS > odps_first数据源。
表	选择数据源中的ods_user_info_d表。
分区信息	默认配置为${bizdate}。
清理规则	默认为**写入前清理已有数据**。
空字符串作为null	此处勾选**否**。

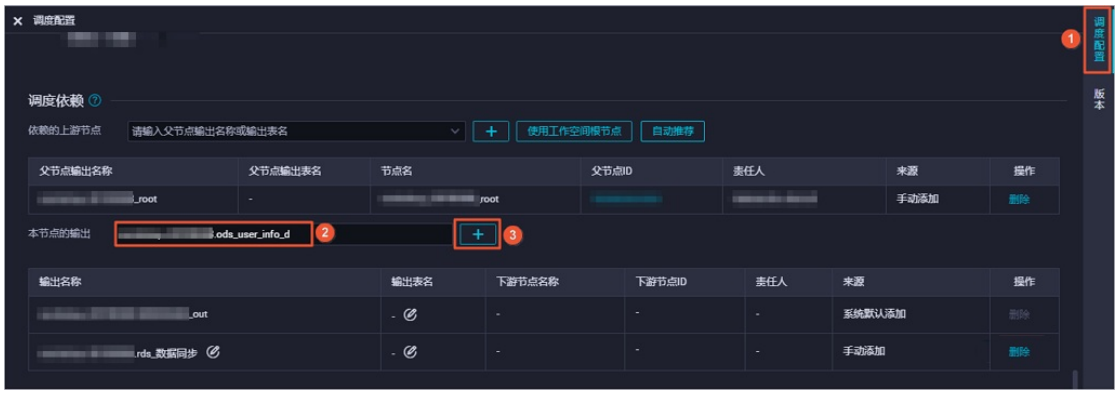
iv. 配置字段映射。

v. 配置**通道控制**。

vi. 单击页面右侧的**调度配置**，在**调度依赖 > 本节点的输出**区域，输入本节点的输出名称为工作空间名称.ods_user_info_d，单击图标。

添加成功后，您可以**删除**不规范的输出名称。

> 注意　不建议您使用中文作为本节点输出名称，会减少自动推荐功能的准确性。

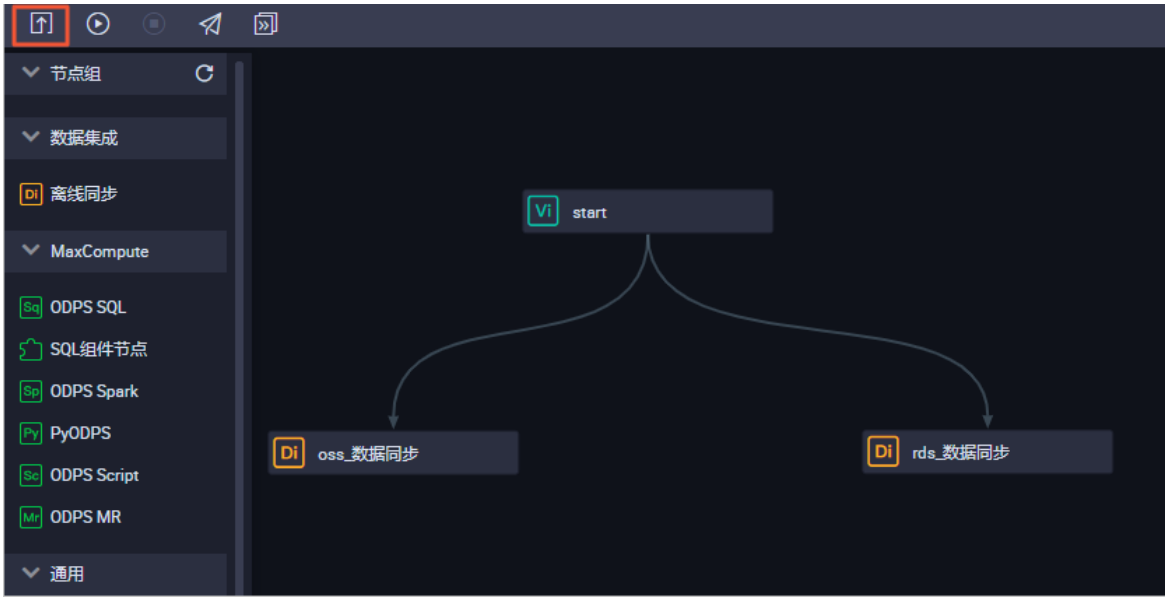


vii. 确认当前节点的配置无误后，单击工具栏中的图标。

viii. 关闭当前任务，返回业务流程配置面板。

提交业务流程

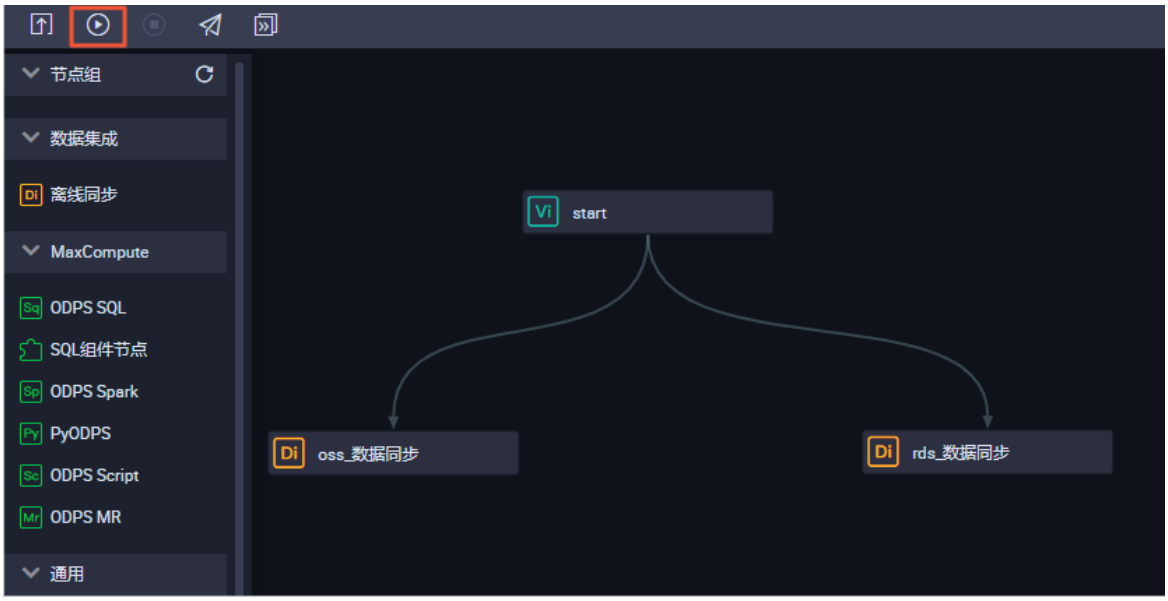
1. 在**数据开发**页面，双击相应的业务流程打开编辑页面，单击工具栏中的图标。



2. 选择**提交**对话框中需要提交的节点，输入**备注**，勾选**忽略输入输出不一致的告警**。
3. 单击**提交**，待显示**提交成功**即可。

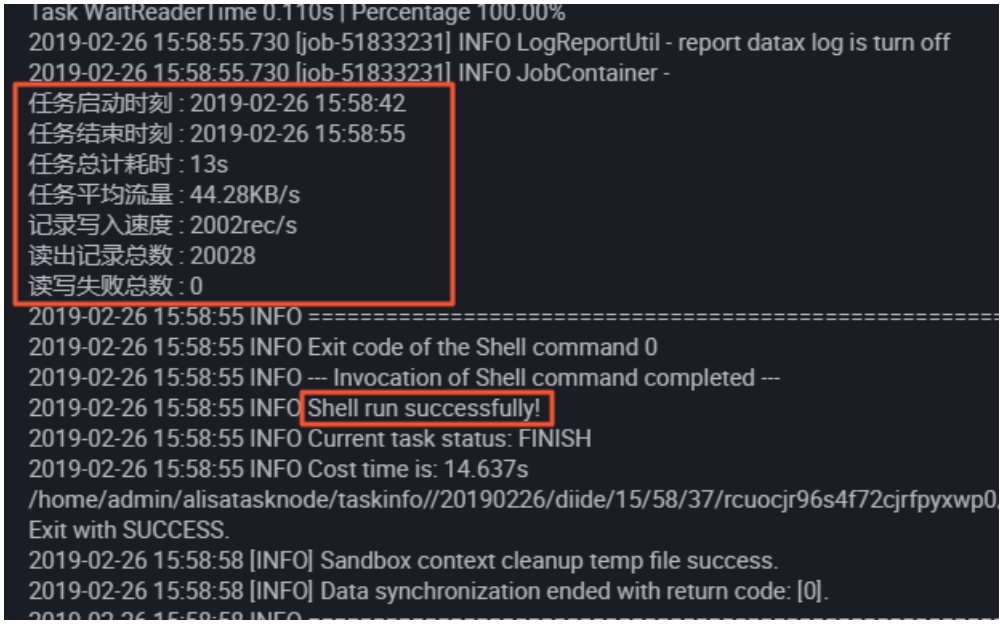
运行业务流程

1. 在**数据开发**页面，双击相应的业务流程打开编辑页面，单击工具栏中的图标。



2. 右键单击rds_数据同步节点，选择**查看日志**。

   当日志中出现如下字样，表示同步节点运行成功，并成功同步数据。



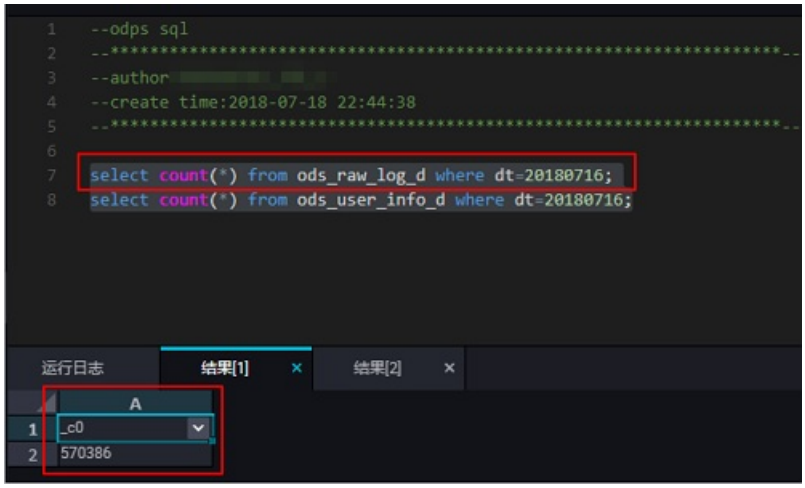
3. 右键单击oss_数据同步节点，选择**查看日志**，确认方法与rds_数据同步节点一致。

确认数据是否成功导入MaxCompute

1. 在**数据开发**页面的左侧导航栏，单击**临时查询**，进入**临时查询**面板。
2. 右键单击**临时查询**，选择**新建节点 > ODPS SQL**。
3. 编写并执行SQL语句，查看导入ods_raw_log_d和ods_user_info_d的记录数。

> **说明** SQL语句如下所示，其中分区列需要更新为业务日期。例如，任务运行的日期为20180717，则业务日期为20180716，即任务运行日期的前一天。

```
--查看是否成功写入MaxCompute
select count(*) from ods_raw_log_d where dt=业务日期;
select count(*) from ods_user_info_d where dt=业务日期;
```



后续步骤

现在，您已经学习了如何进行日志数据同步，完成数据的采集，您可以继续下一个教程。在该教程中，您将学习如何对采集的数据进行计算与分析。详情请参见数据加工。

1.4. 加工数据

本文为您介绍如何通过DataWorks计算和分析已采集的数据。

前提条件

开始本实验前，请首先完成采集数据中的操作。

新建数据表

1. 进入**数据开发**页面。
   i. 登录DataWorks控制台。
   ii. 在左侧导航栏，单击**工作空间列表**。
   iii. 选择工作空间所在地域后，单击相应工作空间后的**进入数据开发**。
2. 在**数据开发**页面，打开新建的业务流程。右键单击**MaxCompute**，选择**新建 > 表**。
3. 在**新建表**对话框中，输入**表名**，单击**提交**。

   此处需要创建三张表，分别为数据运营层表（ods_log_info_d）、数据仓库层表（dw_user_info_all_d）和数据产品层表（rpt_user_info_d）。
4. 通过DDL模式新建表。
   - 新建ods_log_info_d表。

     双击ods_log_info_d表，在右侧的编辑页面单击**DDL模式**，输入下述建表语句。

```
--创建数据运营层（ODS）表
CREATE TABLE IF NOT EXISTS ods_log_info_d (
  ip STRING COMMENT 'ip地址',
  uid STRING COMMENT '用户ID',
  time STRING COMMENT '时间yyyymmddhh:mi:ss',
  status STRING COMMENT '服务器返回状态码',
  bytes STRING COMMENT '返回给客户端的字节数',
  region STRING COMMENT '地域，根据ip得到',
  method STRING COMMENT 'http请求类型',
  url STRING COMMENT 'url',
  protocol STRING COMMENT 'http协议版本号',
  referer STRING COMMENT '来源url',
  device STRING COMMENT '终端类型 ',
  identity STRING COMMENT '访问类型 crawler feed user unknown'
)
PARTITIONED BY (
  dt STRING
);
```

- 新建dw_user_info_all_d表。

  双击dw_user_info_all_d表，在右侧的编辑页面单击**DDL模式**，输入下述建表语句。

```
--创建数据仓库层（DW）表
CREATE TABLE IF NOT EXISTS dw_user_info_all_d (
  uid STRING COMMENT '用户ID',
  gender STRING COMMENT '性别',
  age_range STRING COMMENT '年龄段',
  zodiac STRING COMMENT '星座',
  region STRING COMMENT '地域，根据ip得到',
  device STRING COMMENT '终端类型 ',
  identity STRING COMMENT '访问类型 crawler feed user unknown',
  method STRING COMMENT 'http请求类型',
  url STRING COMMENT 'url',
  referer STRING COMMENT '来源url',
  time STRING COMMENT '时间yyyymmddhh:mi:ss'
)
PARTITIONED BY (
  dt STRING
);
```

- 新建rpt_user_info_d表。

  双击rpt_user_info_d表，在右侧的编辑页面单击**DDL模式**，输入下述建表语句。

```
--创建数据产品层（RPT）表
CREATE TABLE IF NOT EXISTS rpt_user_info_d (
  uid STRING COMMENT '用户ID',
  region STRING COMMENT '地域，根据ip得到',
  device STRING COMMENT '终端类型 ',
  pv BIGINT COMMENT 'pv',
  gender STRING COMMENT '性别',
  age_range STRING COMMENT '年龄段',
  zodiac STRING COMMENT '星座'
)
PARTITIONED BY (
  dt STRING
);
```

5. 建表语句输入完成后，单击**生成表结构**并**确认**覆盖当前操作。
6. 返回建表页面后，在**基本属性**中输入表的中文名。
7. 完成设置后，分别单击**提交到开发环境**和**提交到生产环境**。

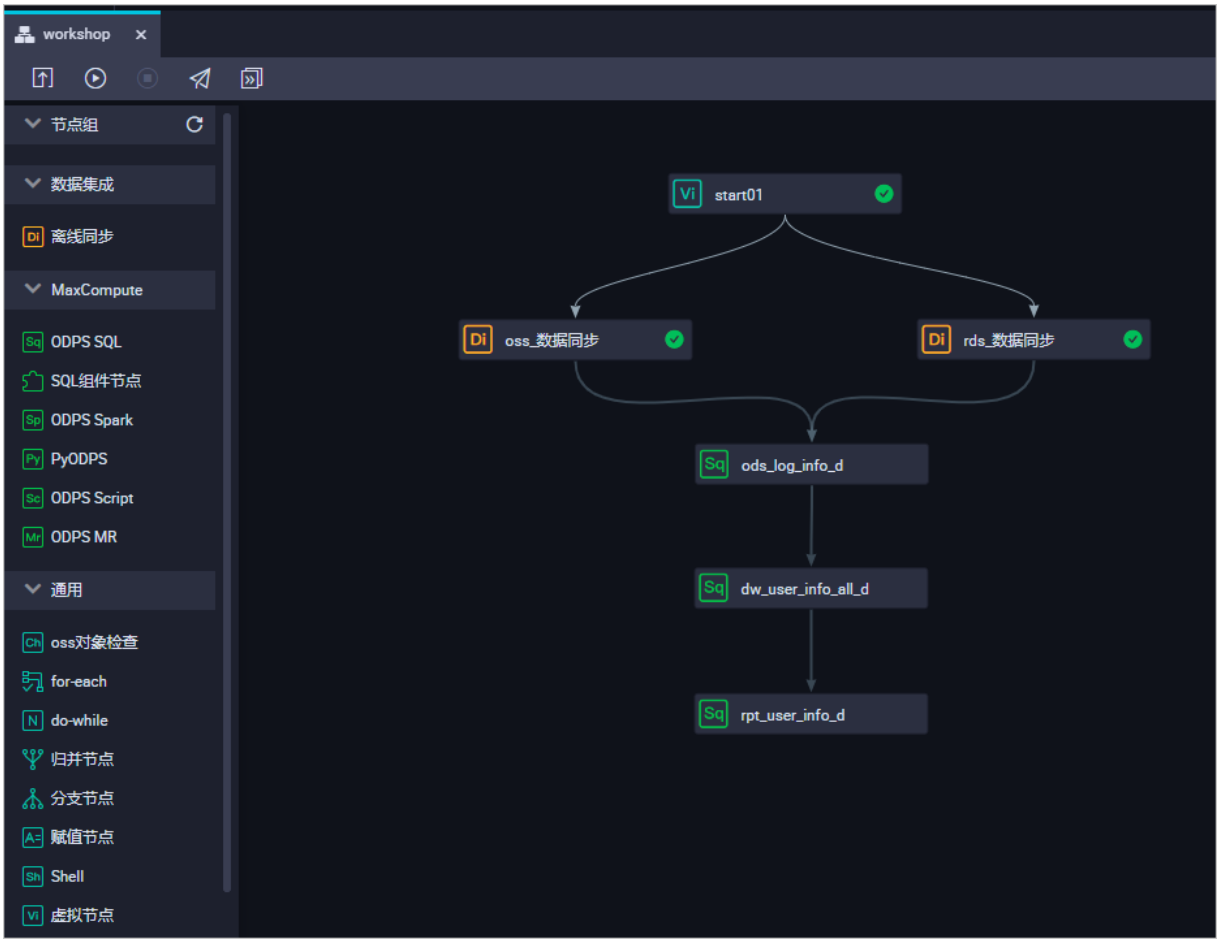
> **说明** 如果您使用的是简单模式的工作空间，页面仅显示**提交到生产环境**。

设计业务流程

业务流程节点间依赖关系的配置请参见采集数据。

双击新建的业务流程打开编辑页面，鼠标单击**ODPS SQL**并拖拽至右侧的编辑页面。在**新建节点**对话框中，输入**节点名称**，单击**提交**。

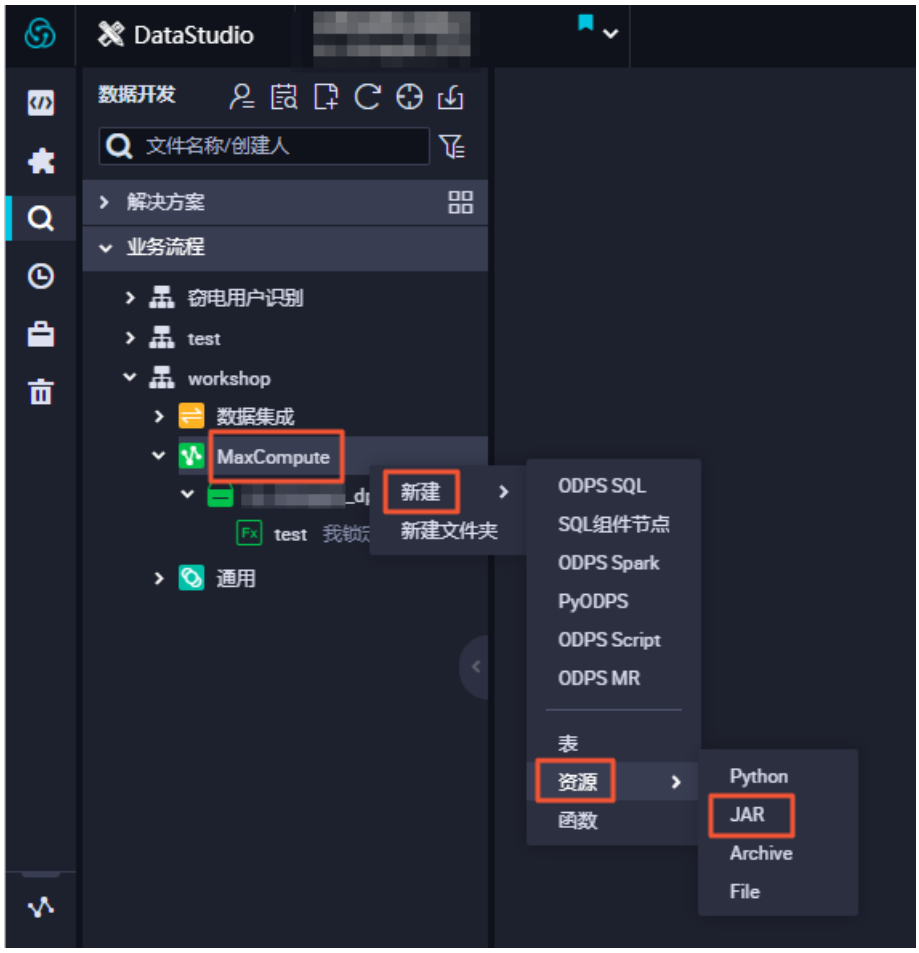
此处需要新建三个ODPS SQL节点，依次命名为ods_log_info_d、dw_user_info_all_d和rpt_user_info_d，并配置如下图所示的依赖关系。



创建用户自定义函数

1. 新建资源。
   i. 下载ip2region.jar。

ii. 在**数据开发**页面打开业务流程，右键单击**MaxCompute**，选择**新建 > 资源 > JAR**。



iii. 在**新建资源**对话框中，输入**资源名称**，并选择**目标文件夹**。

> 说明
> - 请选中**上传为ODPS资源**。
> - 资源名称无需与上传的文件名保持一致。
> - 资源名称命名规范：1~128个字符，字母、数字、下划线、小数点，大小写不敏感，JAR资源的后缀为.jar，Python资源的后缀为.py。

iv. 单击**点击上传**，选择已经下载至本地的*ip2region.jar*，单击**打开**。

v. 单击**确定**。

vi. 单击工具栏中的图标。

2. 注册函数。

i. 在**数据开发**页面打开业务流程，右键单击**MaxCompute**，选择**新建 > 函数**。

ii. 在**新建函数**对话框中，输入**函数名称**（示例为getregion），单击**提交**。

iii. 在**注册函数**对话框中，配置各项参数。

参数	描述
函数类型	选择函数类型。
MaxCompute引擎实例	默认不可以修改。
函数名	新建函数时输入的函数名称。
责任人	选择责任人。
类名	输入 `org.alidata.odps.udf.Ip2Region` 。
资源列表	输入 `ip2region.jar` 。
描述	输入IP地址转换地域。
命令格式	输入 `getregion('ip')` 。
参数说明	输入IP地址。

iv. 分别单击工具栏中的和图标。

配置ODPS SQL节点

1. 配置ods_log_info_d节点。
   i. 双击ods_log_info_d节点，进入节点配置页面。

ii. 在节点编辑页面，编写如下SQL语句。

```
INSERT OVERWRITE TABLE ods_log_info_d PARTITION (dt=${bdp.system.bizdate})
SELECT ip
  , uid
  , time
  , status
  , bytes
  , getregion(ip) AS region --使用自定义UDF通过IP得到地域。
  , regexp_substr(request, '(^[^ ]+ )') AS method --通过正则把request差分为3个字段。
  , regexp_extract(request, '^[^ ]+ (.*) [^ ]+$') AS url
  , regexp_substr(request, '([^ ]+$)') AS protocol
  , regexp_extract(referer, '^[^/]+://([^/]+){1}') AS referer --通过正则清晰refer，得到更精准的URL。
  , CASE
    WHEN TOLOWER(agent) RLIKE 'android' THEN 'android' --通过agent得到终端信息和访问形式。
    WHEN TOLOWER(agent) RLIKE 'iphone' THEN 'iphone'
    WHEN TOLOWER(agent) RLIKE 'ipad' THEN 'ipad'
    WHEN TOLOWER(agent) RLIKE 'macintosh' THEN 'macintosh'
    WHEN TOLOWER(agent) RLIKE 'windows phone' THEN 'windows_phone'
    WHEN TOLOWER(agent) RLIKE 'windows' THEN 'windows_pc'
    ELSE 'unknown'
  END AS device
  , CASE
    WHEN TOLOWER(agent) RLIKE '(bot|spider|crawler|slurp)' THEN 'crawler'
    WHEN TOLOWER(agent) RLIKE 'feed'
    OR regexp_extract(request, '^[^ ]+ (.*) [^ ]+$') RLIKE 'feed' THEN 'feed'
    WHEN TOLOWER(agent) NOT RLIKE '(bot|spider|crawler|feed|slurp)'
    AND agent RLIKE '^[Mozilla|Opera]'
    AND regexp_extract(request, '^[^ ]+ (.*) [^ ]+$') NOT RLIKE 'feed' THEN 'user'
    ELSE 'unknown'
  END AS identity
  FROM (
    SELECT SPLIT(col, '##@@')[0] AS ip
    , SPLIT(col, '##@@')[1] AS uid
    , SPLIT(col, '##@@')[2] AS time
    , SPLIT(col, '##@@')[3] AS request
    , SPLIT(col, '##@@')[4] AS status
    , SPLIT(col, '##@@')[5] AS bytes
    , SPLIT(col, '##@@')[6] AS referer
    , SPLIT(col, '##@@')[7] AS agent
  FROM ods_raw_log_d
  WHERE dt = ${bdp.system.bizdate}
) a;
```

iii. 单击工具栏中的图标。

2. 配置dw_user_info_all_d节点。

i. 双击dw_user_info_all_d节点，进入节点配置页面。

ii. 在节点编辑页面，编写如下SQL语句。

```
INSERT OVERWRITE TABLE dw_user_info_all_d PARTITION (dt='${bdp.system.bizdate}')
SELECT COALESCE(a.uid, b.uid) AS uid
  , b.gender
  , b.age_range
  , b.zodiac
  , a.region
  , a.device
  , a.identity
  , a.method
  , a.url
  , a.referer
  , a.time
FROM (
  SELECT *
  FROM ods_log_info_d
  WHERE dt = ${bdp.system.bizdate}
) a
LEFT OUTER JOIN (
  SELECT *
  FROM ods_user_info_d
  WHERE dt = ${bdp.system.bizdate}
) b
ON a.uid = b.uid;
```

iii. 单击工具栏中的图标。

3. 配置rpt_user_info_d节点。

i. 双击rpt_user_info_d节点，进入节点配置页面。

ii. 在节点编辑页面，编写如下SQL语句。

```
INSERT OVERWRITE TABLE rpt_user_info_d PARTITION (dt='${bdp.system.bizdate}')
SELECT uid
  , MAX(region)
  , MAX(device)
  , COUNT(0) AS pv
  , MAX(gender)
  , MAX(age_range)
  , MAX(zodiac)
FROM dw_user_info_all_d
WHERE dt = ${bdp.system.bizdate}
GROUP BY uid;
```

iii. 单击工具栏中的图标。

提交业务流程

1. 在业务流程的编辑页面，单击图标，提交业务流程中已配置完成的节点。
2. 在**提交**对话框中，选择需要提交的节点，选中**忽略输入输出不一致的告警**。
3. 单击**提交**。

运行业务流程

1. 在业务流程的编辑页面，单击工具栏中的图标，验证代码逻辑。
2. 待所有任务运行完成显示绿色箭头后，在左侧导航栏，单击**临时查询**。
3. 在**临时查询**面板，右键单击**临时查询**，选择**新建节点 > ODPS SQL**。
4. 编写并执行SQL语句，查询任务运行结果，确认数据产出。

```
--odps sql
--********************************************************************--
--author:
--create time:2019-06-28 19:42:29
--********************************************************************--
select * from rpt_user_info_d where dt=20190630 limit 10;
```

运行日志　结果[1]

	A	B	C	D	E	F	G	H
1	uid	region	device	pv	gender	age_range	zodiac	dt
2	0016359810821	湖北省	windows_pc	1	女	30-40岁	巨蟹座	20190630
3	0016359814159	未知	windows_pc	5	女	30-40岁	巨蟹座	20190630
4	001d9e7863049	浙江省	iphone	21	女	40-50岁	双鱼座	20190630
5	001d9e7866387	河南省	windows_pc	1	女	40-50岁	双鱼座	20190630
6	001d9e7869725	未知	windows_pc	1	女	40-50岁	双鱼座	20190630
7	001dce2983544	湖北省	unknown	2	女	20-30岁	水瓶座	20190630
8	001dce2986882	广东省	windows_pc	3	女	20-30岁	水瓶座	20190630
9	0026c84ad1206	台湾省	windows_pc	1	女	20岁以下	天秤座	20190630
10	0026c84ad4544	福建省	windows_pc	126	女	20岁以下	天秤座	20190630
11	0026c84ad7882	福建省	windows_pc	3	女	20岁以下	天秤座	20190630

查询语句如下所示，通常默认业务日期为运行日期的前一天。

```
---查看rpt_user_info_d数据情况。
select * from rpt_user_info_d where dt=业务日期 limit 10;
```

发布业务流程

提交业务流程后，表示任务已进入开发环境。由于开发环境的任务不会自动调度，您需要发布配置完成的任务至生产环境。

> **说明**
> - 发布任务至生产环境前，您需要对代码进行测试，确保其正确性。
> - 如果您使用的是简单模式的工作空间，则没有图标。您在提交任务后，单击图标，进入运维中心页面。

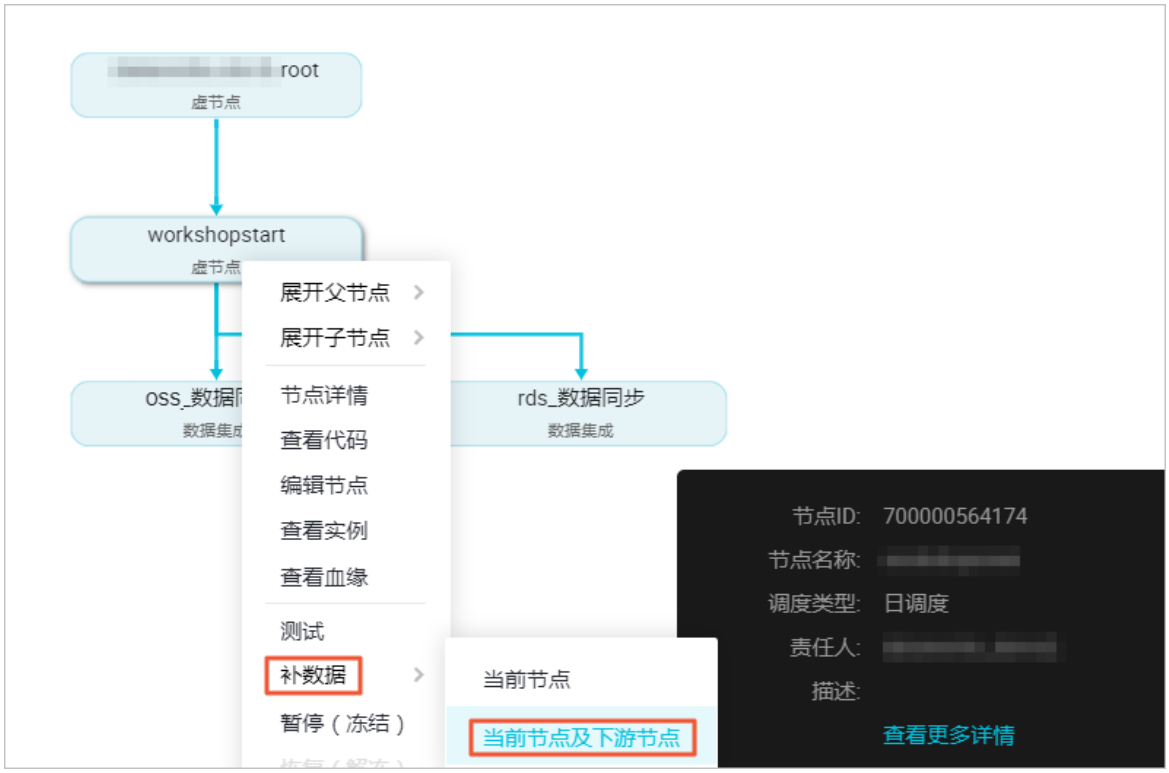
1. 在业务流程的编辑页面，单击工具栏中的图标，进入发布页面。
2. 选择待发布任务，单击**添加到待发布**。

3. 单击右上角的**待发布列表**，进入列表后，单击**全部打包发布**。
4. 在**确认发布**对话框中，单击**发布**。
5. 在左侧导航栏，单击**发布包列表**，查看发布状态。

在生产环境运行任务

1. 任务发布成功后，单击右上角的**运维中心**。

   您也可以进入业务流程的编辑页面，单击工具栏中的**前往运维**，进入**运维中心**页面。
2. 在左侧导航栏，单击**周期任务运维 > 周期任务**，进入**周期任务**页面，单击workshop业务流程。
3. 双击DAG图中的虚节点展开业务流程，右键单击workshop_start节点，选择**补数据 > 当前节点及下游节点**。



4. 选中需要补数据的任务，输入业务日期，单击**确定**，自动跳转至**补数据实例**页面。
5. 单击**刷新**，直至SQL任务全部运行成功即可。

后续步骤

现在，您已经学习了如何创建SQL任务、如何处理原始日志数据。您可以继续下一个教程，学习如何对开发完成的任务设置数据质量监控，保证任务运行的质量。详情请参见配置数据质量监控。

1.5. 配置数据质量监控

本文为您介绍如何监控数据质量、设置表的质量监控规则和监控提醒等。

前提条件

在进行本实验前，请确保已采集并加工数据。详情请参见采集数据和加工数据。

背景信息

数据质量是支持多种异构数据源的质量校验、通知、管理服务的一站式平台。数据质量以数据集（DataSet）为监控对象，目前支持MaxCompute数据表和DataHub实时数据流的监控。当离线MaxCompute数据发生变化时，数据质量会对数据进行校验，并阻塞生产链路，以避免问题数据污染扩散。同时，数据质量提供历史校验结果的管理，以便您对数据质量分析和定级。

在流式数据场景下，数据质量能够基于DataHub数据通道进行断流监控，第一时间告警给订阅用户，并且支持橙色、红色告警等级以及告警频次设置，最大限度减少冗余报警。

数据质量开发流程

1. 针对已有的表进行监控规则配置，配置完成后进行试跑，验证该规则是否适用。

   您可以根据试跑结果，确认此次任务产出的数据是否符合预期。建议每个表的监控规则配置完成后，都进行一次试跑操作，以验证表规则的适用性。

2. 试跑成功后，将该规则和调度任务进行关联。

   在监控规则配置完成且试跑成功的情况下，您需要将表和其产出任务进行关联，以便每次表的产出任务运行完成后，都会触发数据质量规则的校验，以保证数据的准确性。

3. 关联调度后，每次调度任务代码运行完成，都会触发数据质量的校验规则，以提升任务准确性。

   数据质量支持设置规则订阅，您可以针对重要的表及其规则设置订阅，设置订阅后会根据数据质量的校验结果进行告警，从而实现对校验结果的跟踪。如果数据质量校验结果异常，则会根据配置的告警策略进行通知。

> **说明**
> - 每张表在完成规则的配置后，都需要进行试跑、关联调度和规则订阅等操作。
> - 数据质量会产生额外的计算费用，更多详情请参见数据质量概述。

配置数据表的监控规则

如果已经完成数据采集和数据加工实验，请确认您已拥有数据表：ods_raw_log_d、ods_user_info_d、ods_log_info_d、dw_user_info_all_d和rpt_user_info_d。确认后，进行以下操作：

1. 进入表ods_raw_log_d的**监控规则**页面。
   i. 登录DataWorks控制台。
   ii. 在左侧导航栏，单击**工作空间列表**。
   iii. 选择工作空间所在地域后，单击相应工作空间后的**进入数据开发**。
   iv. 单击左上方的☰图标，选择**全部产品 > 数据治理 > 数据质量**。
   v. 在左侧导航栏，单击**监控规则**，从数据源下拉列表中选择**MaxCompute**。

vi. 在**引擎/数据库实例**下拉框中选择待配置监控规则表所在的引擎实例，在过滤后的表列表中找到待配置监控规则的表，例如本教程的ods_raw_log_d表。

vii. 单击ods_raw_log_d表后的**配置监控规则**。

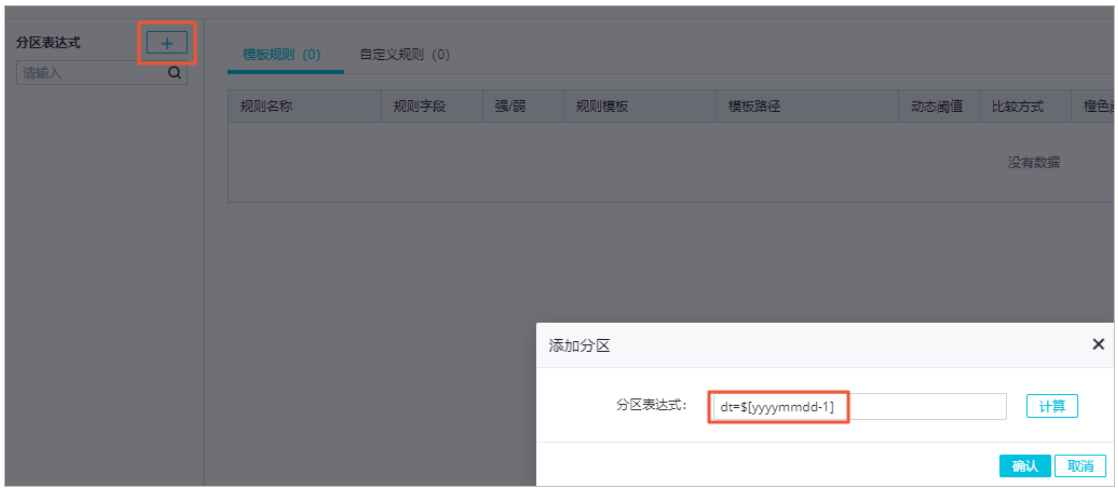
2. 配置表ods_raw_log_d的监控规则。

i. 在**已添加的分区表达式**模块，单击**+**，添加分区表达式。

ods_raw_log_d表的数据来源为oss_workshop_log，数据是从OSS中获取到的日志数据，其分区格式为${bdp.system.bizdate}（获取到前一天的日期）。

对于此类每天产出的日志数据，您可以配置表的分区表达式。在**添加分区**对话框中，选择dt=$[yyyymmdd-1]，单击**确认**。分区表达式的详情请参见配置调度参数。

> **说明** 如果表中无分区列，可以配置**无分区**，请根据真实的分区值配置对应的分区表达式。



ii. 单击**创建规则**，默认在**模板规则**对话框。

iii. 单击**添加监控规则**，选择**规则模板**为**表行数，固定值**，设置规则的强度为**强**、比较方式为**期望值大于0**。

表ods_raw_log_d的数据来源于OSS上传的日志文件，作为源头表，您需要尽早判断该表的分区中是否存在数据。如果该表没有数据，则需要阻止后续任务运行。如果来源表没有数据，后续任务运行无意义。

> **说明** 只有强规则下红色报警会导致任务阻塞，阻塞会将任务的实例状态置为失败。

配置完成后，单击**批量保存**。

> **说明** 该配置主要是为了避免分区中没有数据，导致下游任务的数据来源为空的问题。

iv. 单击**试跑**，在**试跑**对话框中，选择**调度时间**，单击**试跑**。

试跑可以立即触发数据质量的校验规则，对配置完成的规则进行校检。试跑完成后，单击**试跑成功！点击查看试跑结果**，即可跳转至试跑结果页面。

v. 进行关联调度。

数据质量支持和调度任务关联。在表规则和调度任务绑定后，任务实例运行完成都会触发数据质量的检查。您可以通过以下两种方式进行表规则和任务的关联调度：

- 在运维中心页面关联表规则

  单击左上方的图标，选择**全部产品 > 运维中心**。

  在左侧导航栏，单击**周期任务运维 > 周期任务**。在DAG图中，右键单击**oss_数据同步**任务，选择**配置质量监控**。

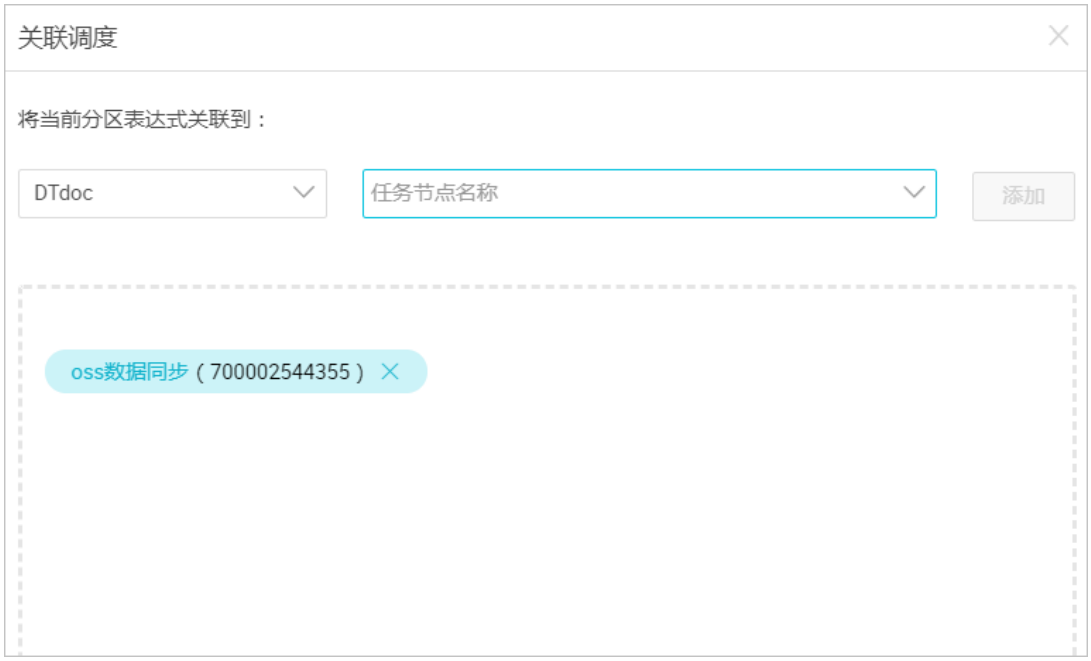
  在**配置质量监控**对话框中，选择**表名**（ods_raw_log_d）和**分区表达式**dt=$[yyyymmdd-1]），单击**添加**。

- 在数据质量页面关联表规则

  在表的**监控规则**页面，单击**关联调度**，配置规则与任务的绑定关系。

  单击**关联调度**，可以与已提交到调度的节点任务进行绑定，系统会根据血缘关系给出推荐绑定的任务，也支持自定义绑定。

  在**关联调度**对话框中，输入节点ID或节点名称，单击**添加**。添加完成后，即可完成与调度节点任务的绑定。



vi. 订阅任务。

在表的监控规则页面，单击**订阅管理**，设置接收人以及订阅方式。数据质量支持**邮件通知**、**邮件和短信通知**、**钉钉群机器人**和**钉钉群机器人@ALL**。

订阅管理设置完成后，在左侧导航栏，单击**我的订阅**，查看和修改已订阅的任务。

> **说明** 建议订阅全部规则，避免校验结果无法及时通知。

3. 配置ods_user_info_d表规则。

ods_user_info_d是用户信息表，您在配置规则时，需要配置表的行数校验和主键唯一性校验，避免数据重复。

i. 配置一个分区字段的监控规则，监控的时间表达式为dt=$[yyyymmdd-1]。配置成功后，在已添加的分区表达式中可以查看成功的分区配置记录。

ii. 分区表达式配置完成后，单击**创建规则**，配置数据质量的校验规则。

分别添加表级规则和列规则：

- 选择**规则字段**为**表级规则**。

选择**规则模板**为**表行数，固定值**、**强弱**为**强**、**比较方式**为**大于**以及**期望值**为**0**。

- 选择**规则字段**为uid。

添加列级规则，设置主键列（uid）为监控列。选择**模板类型**为**重复值个数，固定值**、**强弱**为**弱**、**比较方式**为**小于**以及**期望值**为1。

iii. 配置完成后，单击**批量保存**。

> **说明** 该配置主要是为了避免数据重复，导致下游数据被污染的情况。

4. 配置ods_log_info_d表规则。

ods_log_info_d数据主要来源于解析ods_raw_log_d表中的数据。鉴于日志中的数据无法配置过多监控，只需要配置表数据不为空的校验规则即可。

i. 配置表的分区表达式为dt=$[yyyymmdd-1]。

ii. 单击**创建规则**，在对话框中单击**添加监控规则**。

配置表数据不为空的校验规则，选择**规则强度**为**强**、**规则模板**为**表级规则**、**比较方式**为**不等于**、**期望值**为**0**。

iii. 配置完成后，单击**批量保存**。

5. 配置dw_user_info_all_d表规则。

   dw_user_info_all_d表是针对ods_user_info_d和ods_log_info_d表的数据汇总。由于流程较为简单，ODS层已配置了表行数不为空的规则，所以该表无需进行数据质量监控规则的配置，以节省计算资源。

6. 配置rpt_user_info_d表规则。

   rpt_user_info_d表是数据汇总后的结果表。根据该表的数据，您可以进行表行数波动监测和针对主键进行唯一值校验。

   i. 单击**已添加的分区表达式**模块的**+**，配置表的分区表达式为dt=$[yyyymmdd-1]。

   ii. 单击**创建规则**，在**添加监控规则**对话框中添加列级规则。设置主键列（uid）为监控列，选择**规则模板**为**重复值个数，固定值**、**强弱**为**弱**、**比较方式**为**小于**以及**期望值**为**1**。

iii. 继续添加监控规则和表级规则，选择**规则模板**为**表行数，7天波动率**、**强弱**为**弱**，设置**橙色阈值**为1%、**红色阈值**为50%（此处阈值范围根据业务逻辑进行设置）。

> 说明
> - **橙色阈值**和**红色阈值**必须大于0%。
> - 此处监控表行数是为了查看每日UV的波动，以便及时了解应用动态。

iv. 配置完成后，单击**批量保存**。

在设置表规则强度时，数据仓库中越底层的表，设置强规则的次数越多。这是因为ODS层的数据作为数仓中的原始数据，一定要保证其数据的准确性，避免因ODS层的数据质量太差而影响其它层的数据，及时止损。

数据质量还为您提供**任务查询**功能，以便查看已配置规则的校验结果，详情请参见查看监控任务。

1.6. 数据可视化展现

通过补数据完成数据表rpt_user_info_d加工后，您可以通过Quick BI创建网站用户分析画像的仪表板，实现该数据表的可视化。

前提条件

在开始试验前，请确认您已经完成了加工数据。单击进入Quick BI控制台。

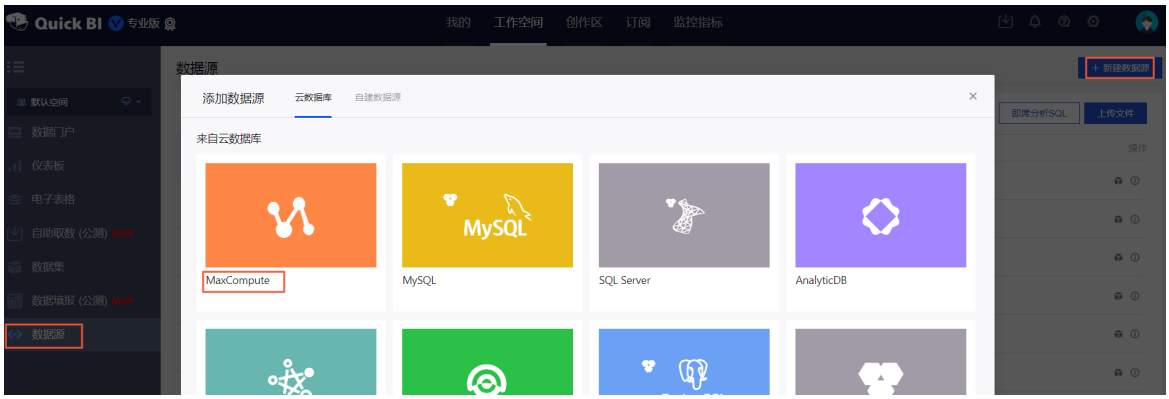
背景信息

rpt_user_info_d表包含了region、device、gender、age、zodiac等字段信息。您可以通过仪表板展示用户的核心指标、周期变化、用户地区分布、年龄与星座分布和记录。为查看数据在日期上的变化，建议您在补数据时至少选择一周的时间。

操作步骤

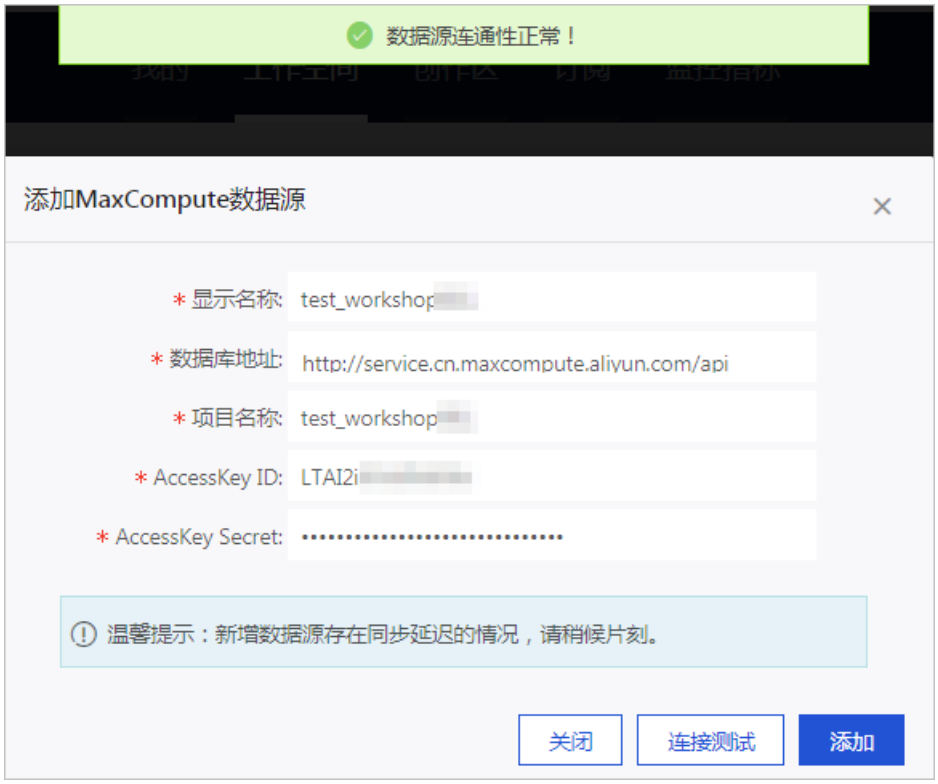
1. 单击进入**默认空间**，您也可以使用自己的**个人空间**。

2. 选择**数据源 > 新建数据源 > 云数据库 > MaxCompute**。

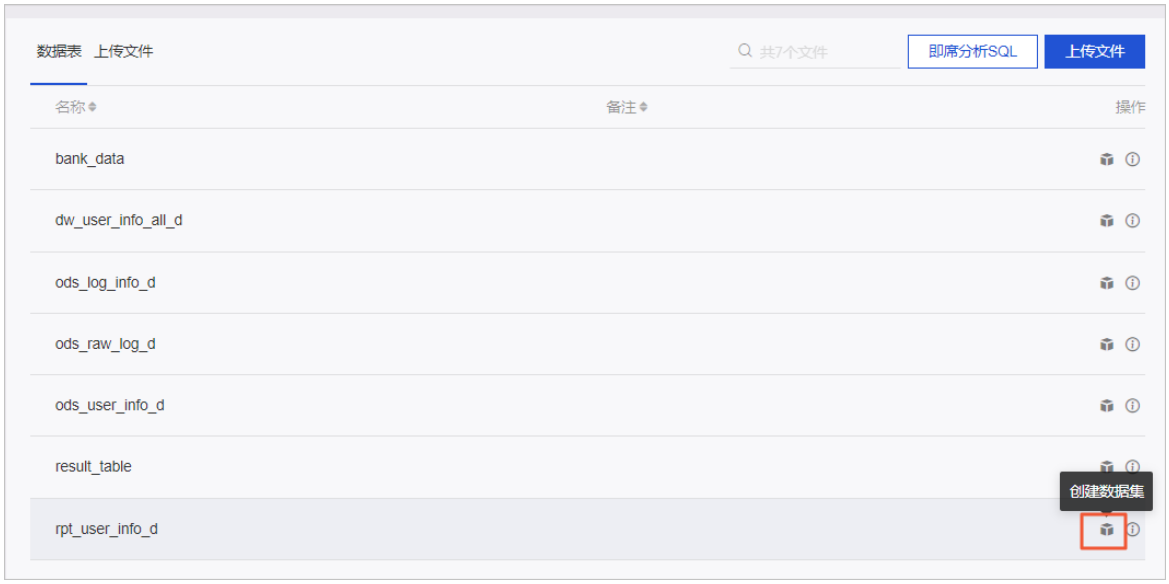


3. 输入您的MaxCompute项目名称以及您的AcessKey信息，数据库地址使用默认地址即可，关于数据库地址详情请参见Endpoint。

   完成填写后，单击**连接测试**，待显示**数据源连通性正常**后单击**添加**即可。



4. 找到您刚添加的数据源的rpt_user_info_d表，单击**创建数据集**。



选择您想放置的数据集位置，单击**确定**。

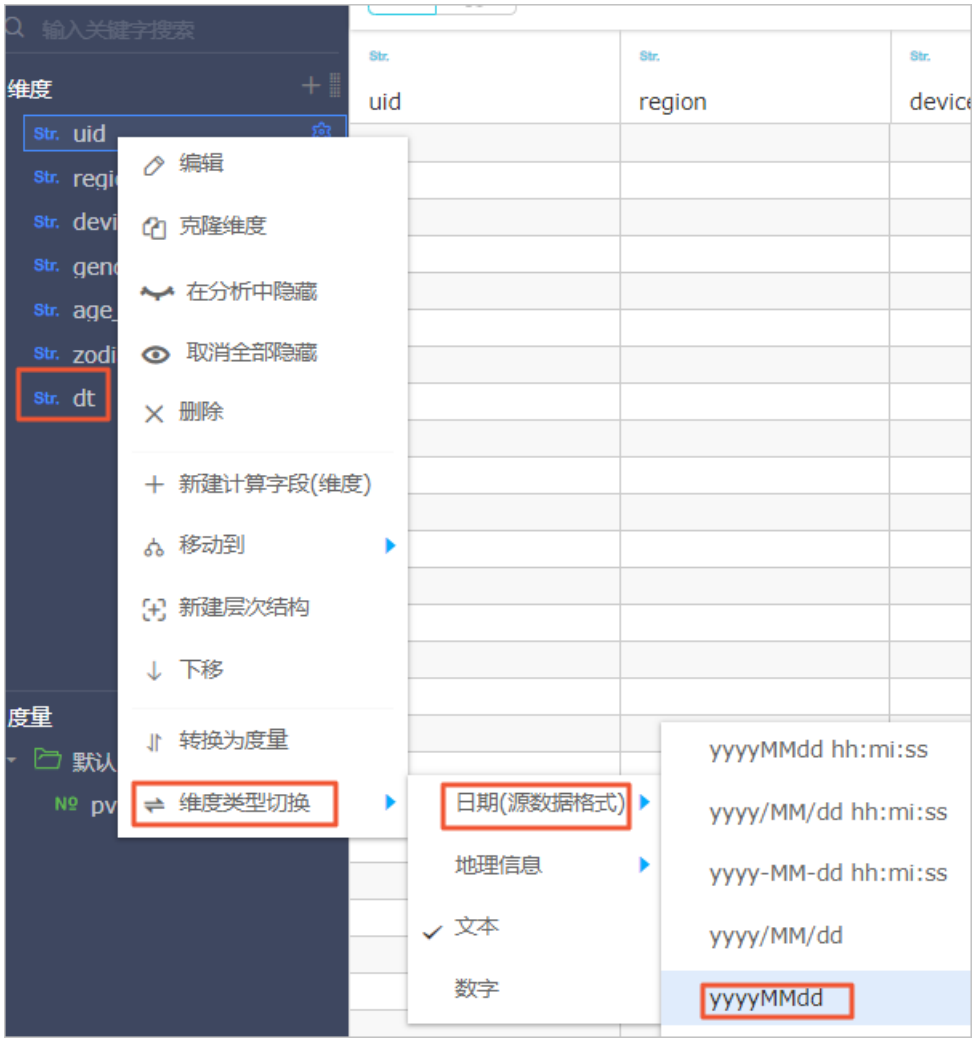
5. 进入数据集列表页，单击您刚刚创建的数据集，对数据集进行编辑。

常见的数据集加工包括：维度、度量的切换、修改维度的类型、增加计算字段、创建层次结构、修改字段的数据类型、更改度量聚合方式、制作关联模型。

6. 转换字段的维度类型。完成转换后，您可以根据字段中具体的数值进行过滤筛选。

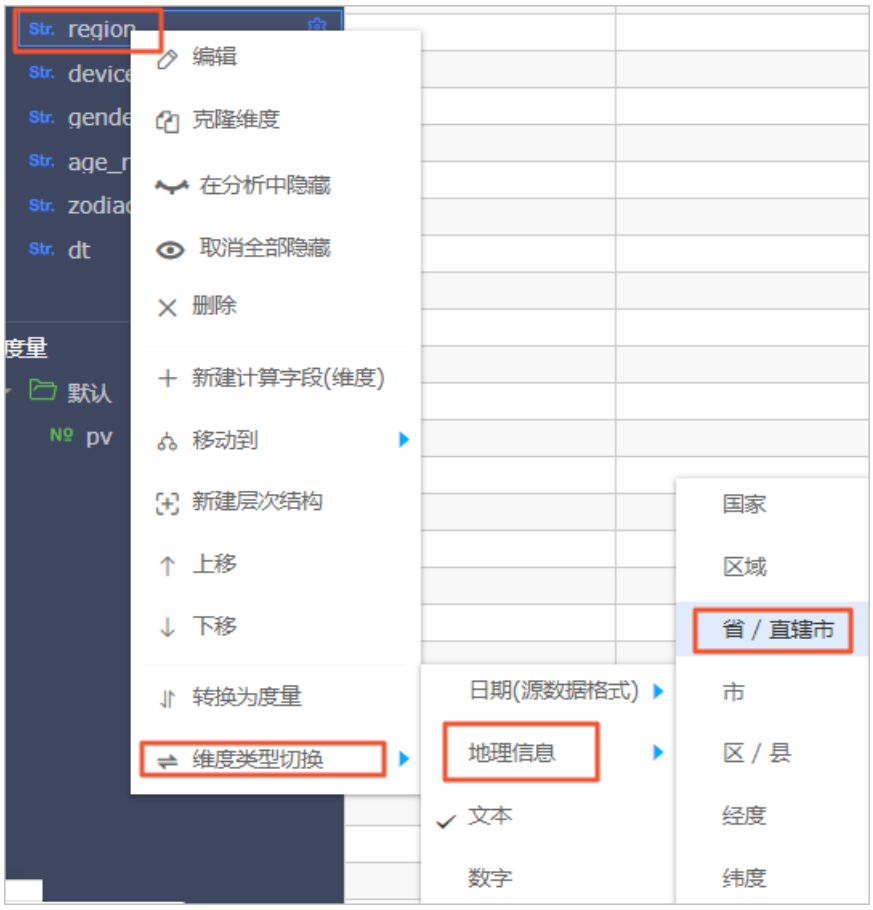
i. 转换日期字段的维度类型。

右键单击dt字段，选择**维度类型切换 > 日期（源数据格式） > yyyyMMdd**。



ii. 转换地理信息字段的维度类型。

右键单击region字段，选择**维度类型切换 > 地理信息 > 省/直辖市**。转换成功后，在左侧维度栏中会看到字段前多一个地理位置图标。



7. 制作仪表板。

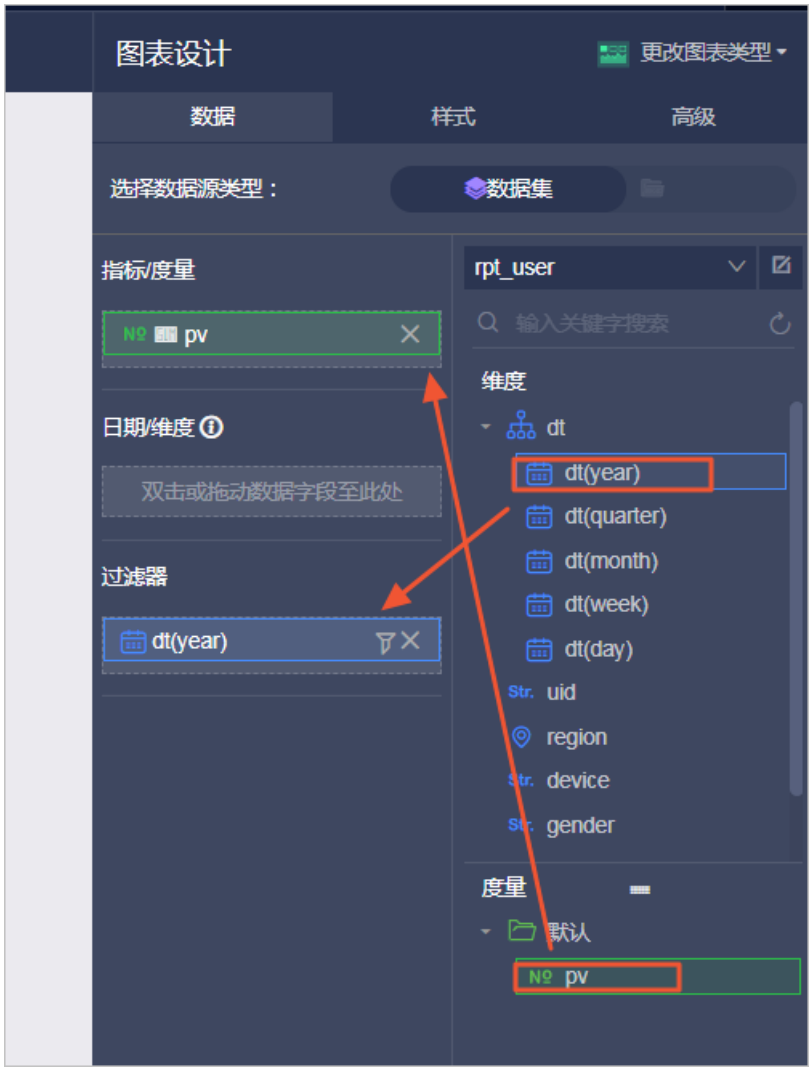
随着数据的更新，让报表可视化地展现最新数据，这个过程叫制作仪表板。仪表板的制作流程为： 确定内容、布局和样式，制作图表，完成动态联动查询。

i. 单击rpt_user数据集后的新建仪表板，选择**常规模式**，进入仪表板编辑页。

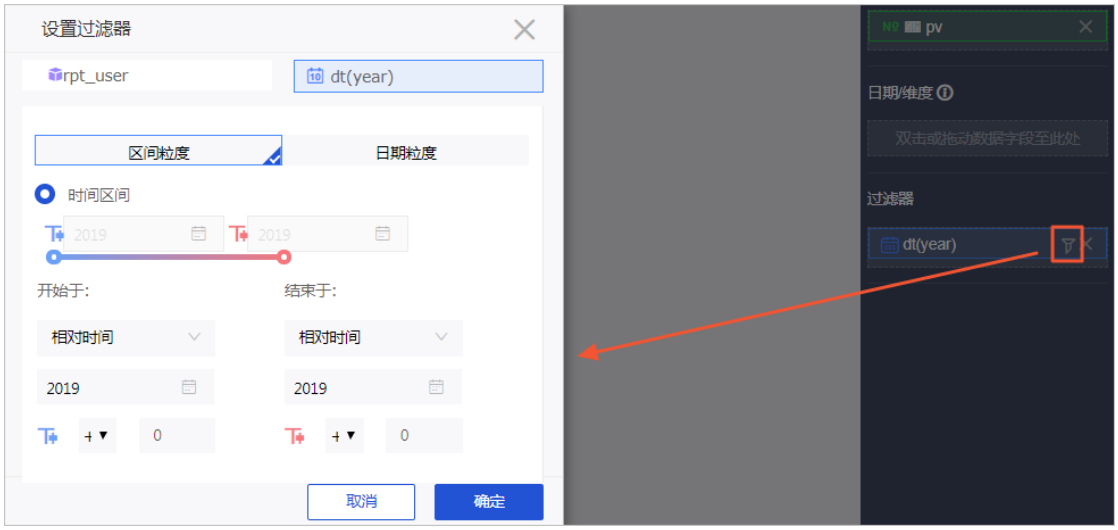
> **说明**
>
> - 如果您使用的是标准版或高级版的Quick BI，则单击新建仪表板后默认进入常规模式仪表板。
> - 如果您使用的是专业版Quick BI，则单击新建仪表板后可以选择进入常规模式或全屏模式仪表板。当前文档以常规模式仪表板示例。

ii. 从仪表板空间中向空白区拖入1个指标看板。

选择数据来源为数据集rpt_user，选择度量为pv。



由于数据表rpt_user_info_d为分区表，因此必须在**过滤器**处选择筛选的日期，本例中筛选为2019~2019年，完成设置后单击**更新**。

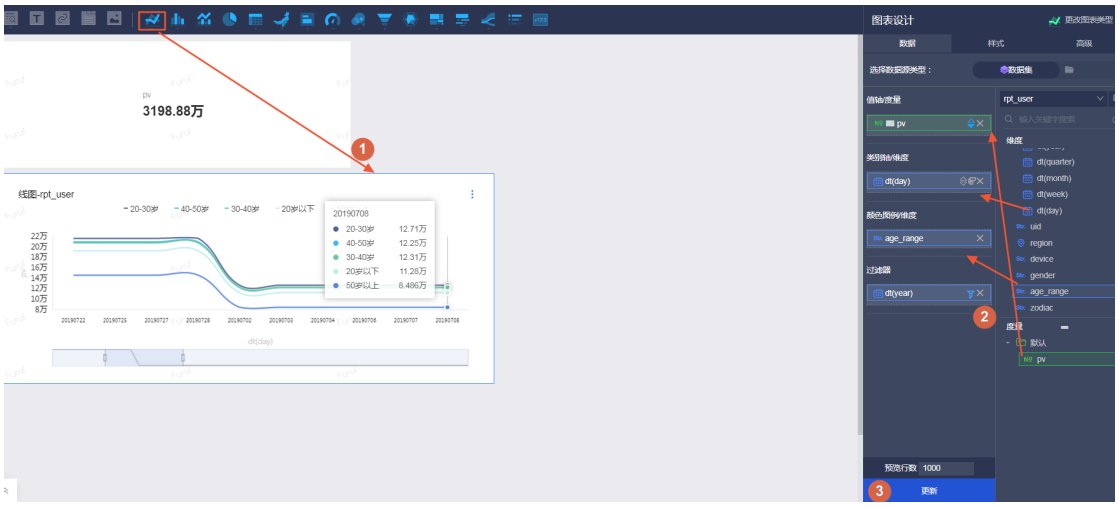


完成后可以看到当下数据。

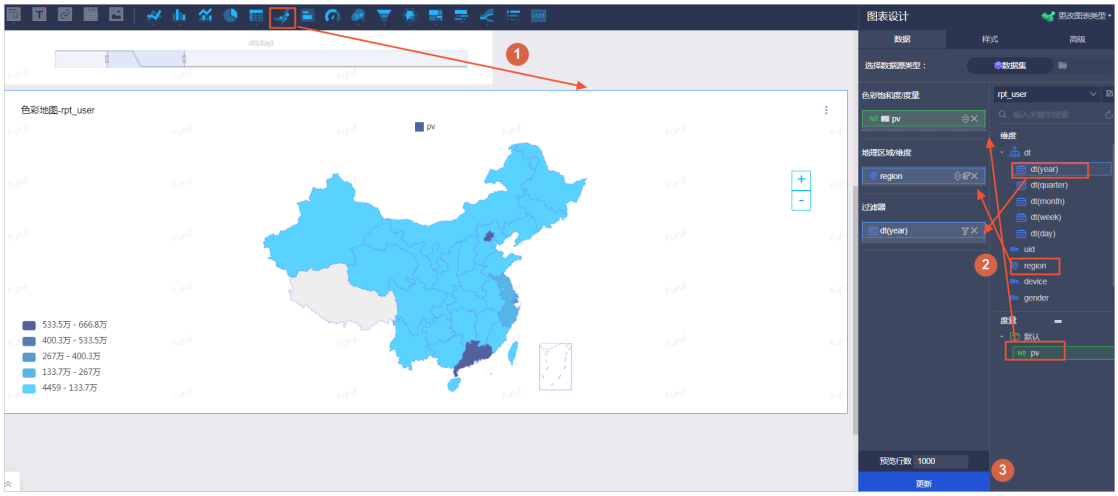
iii. 制作趋势图：将图表区域内的**线图**拖拽到左侧画布。

参数配置如下，完成之后单击**更新**：

- **值轴/度量**：pv
- **类别轴/维度**：dt（day）
- **颜色图例/维度**：age_range
- **过滤器**：dt（year）



iv. 制作色彩地图：单击图表区域内的**色彩地图**，并选择数据源来源为数据集rpt_user，选择**地理区域/维度**为region、**色彩饱和度/度量**为pv，选择完成后单击**更新**，结果如下。



v. 完成配置后，单击**保存**及**预览**，即可看到展示效果。

1.7. 通过Function Studio开发UDF

本文为您介绍如何通过Function Studio开发UDF，并将其提交至DataStudio的开发环境。

使用限制

目前仅华北2（北京）、华东2（上海）、华南1（深圳）和华东1（杭州）地域支持Function Studio。

新建工程

如果您已经有Git代码，可以直接导入Git代码创建工程。此处仅支持Code中的代码导入。

1. 进入**数据开发**页面。
   i. 登录DataWorks控制台。
   ii. 在左侧导航栏，单击**工作空间列表**。
   iii. 选择工作空间所在地域后，单击相应工作空间后的**进入数据开发**。
2. 单击左上方的图标，选择**全部产品 > 数据开发 > Function Studio**。
3. 在**工作空间**页面，单击**导入Git工程**。
4. 在**新建项目**对话框中，输入**Git地址**、**工程名**和**工程描述**，并**选择运行环境**。

   其中**Git地址**仅支持配置为阿里云的Git地址，您可以在云计算服务平台中创建。

   新创建的工程默认未关联Git服务，会弹出**设置**对话框，请首先进行**SSH KEY**、**Git Config**和**偏好设置**的配置，单击**保存**。
   - 选择**SSH Key**中的**service**为 `code.aliyun.com` ，单击**生成sshKey**，即可生成**Public key**，单击**保存**。
   - 填写**Git Config**中的**User Name**和**Email**，单击**保存**。
   - 根据自身需求选择**偏好设置**中的**编辑器字号**，单击**保存**。

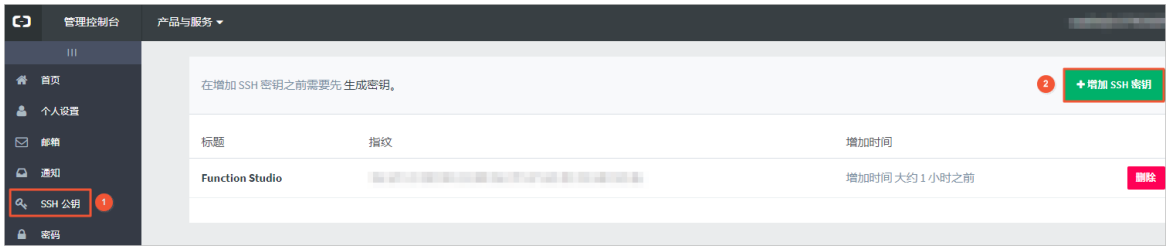
> **说明** 如果工程创建完成后，需要修改相关信息，可以鼠标悬停至顶部菜单栏中的**设置**进行修改。

5. 单击**提交**。
   工程创建完成后，Function Studio会自动拉取该工程。

新建SSH密钥

设置好**SSH KEY**、**Git Config**和**偏好设置**后，可以新增SSH密钥。

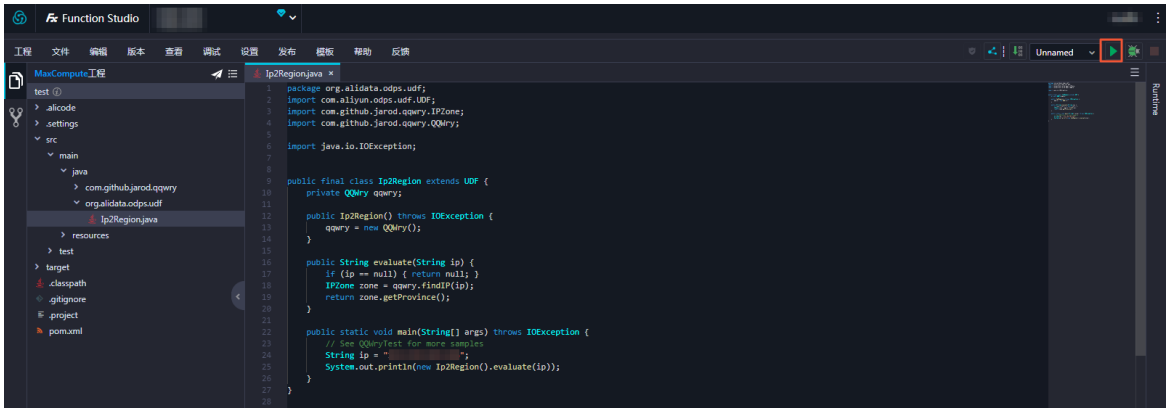
1. 访问Code页面，单击左侧导航栏中的**设置**。
2. 进入**设置**页面，选择**SSH公钥 > 增加SSH密钥**。



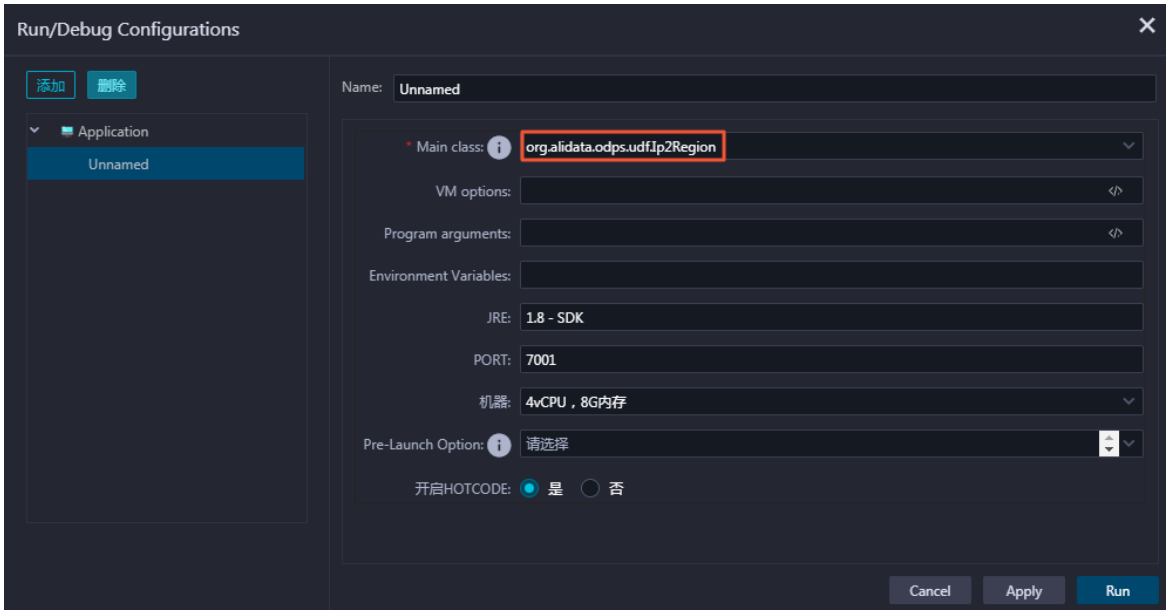
3. 在**增加SSH密钥**对话框中填写前文生成的**Public key**，单击**增加密钥**。

测试需要运行的类

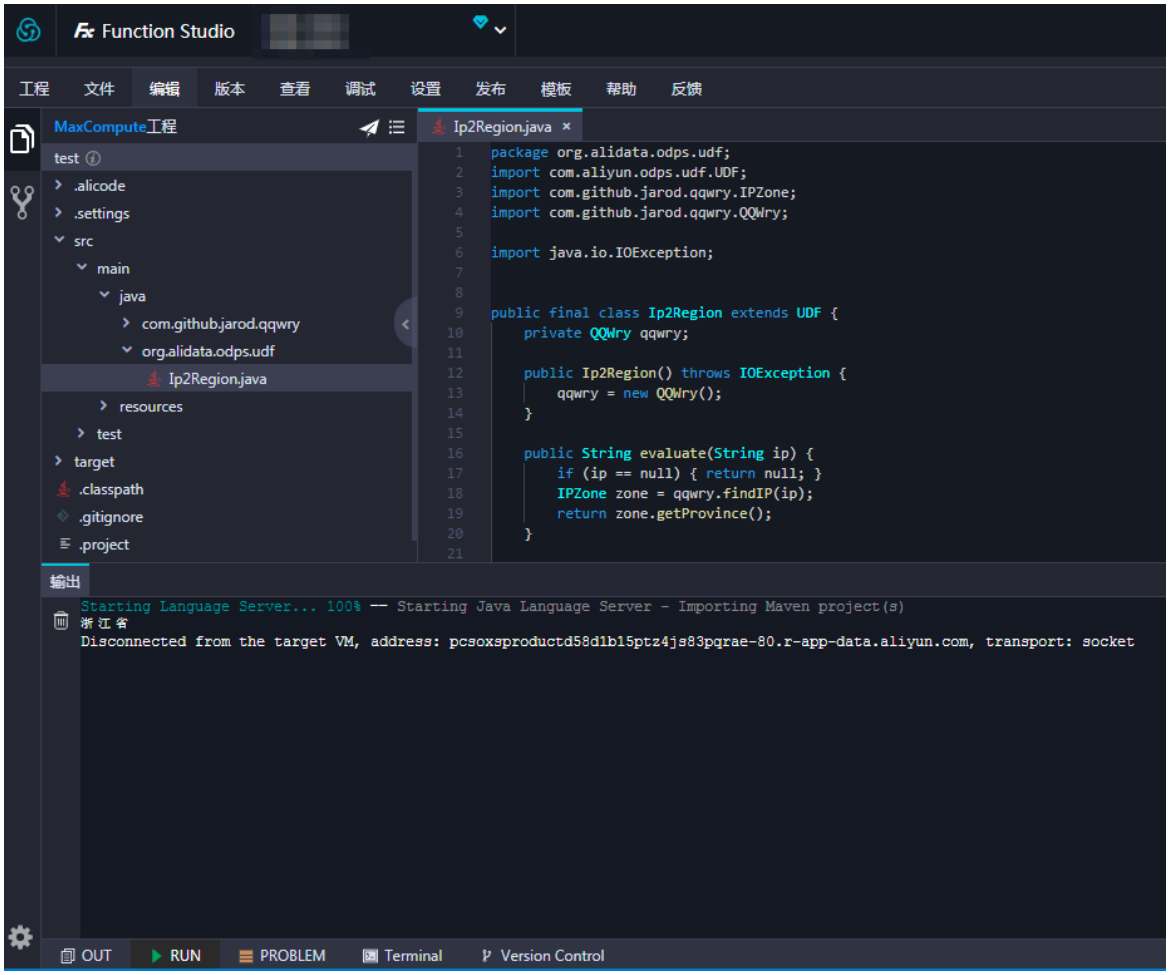
1. 打开需要运行的类，单击右上角的运行按钮进行测试。



2. 在**Run/Debug Configurations**对话框中，手动添加测试类的信息。



3. 添加完成后，单击**Run**，即可看到输出的测试信息。

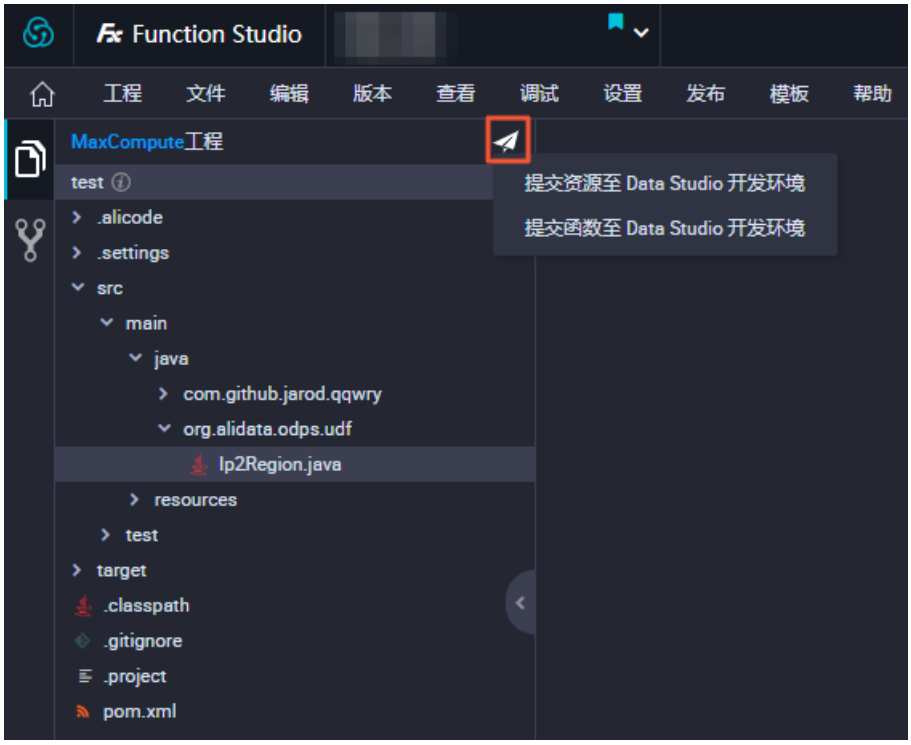


> 说明
> - 第一次启动时速度较慢，之后的启动速度会逐渐接近本地编辑器的体验。
> - 如果需要运行的类已经存在，直接在右上角进行选择，单击运行按钮即可。

提交函数和资源至DataStudio开发环境

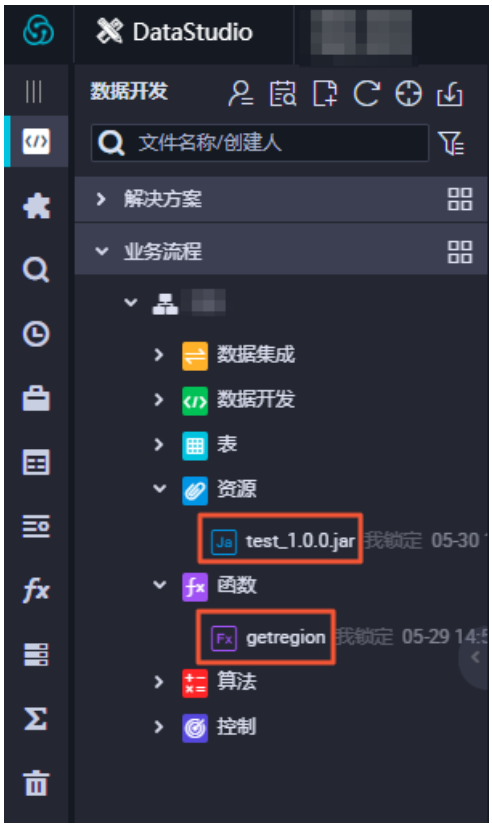
确认代码无误后，可以提交函数和资源至DataStudio开发环境。

- 提交资源至DataStudio开发环境。
  i. 鼠标悬停至**提交**按钮，单击**提交资源至DataStudio开发环境**。



  ii. 选择**提交资源至DataStudio开发环境**对话框中的**目标业务空间**和**目标业务流程**，并填写**资源**。
  iii. 单击**确认**。
- 提交函数至DataStudio开发环境。
  i. 鼠标悬停至**提交**按钮，单击**提交函数至DataStudio开发环境**。
  ii. 选择**提交函数至DataStudio开发环境**对话框中的**目标业务空间**、**目标业务流程**和**类名**，并填写**资源**和**函数名**。
  iii. 单击**确认**。

当资源和函数都提交至DataStudio开发环境后，即可直接在SQL节点中使用。



2.简单用户画像分析（EMR版）

2.1. 准备环境

为保证您可以顺利完成本次实验，请您首先确保云账号已开通E-MapReduce（简称为EMR）、数据工场DataWorks和数据存储OSS。

前提条件

- 阿里云账号注册，详情请参见。
- 实名认证，详情请参见或。
- 您在工作空间配置页面添加E-MapReduce计算引擎实例后，当前页面才会显示EMR目录。详情请参见配置工作空间。
- 您已创建阿里云EMR集群，且集群所在的安全组中入方向的安全策略包含以下策略。
  - 授权策略：允许
  - 协议类型：自定义 TCP
  - 端口范围：8898/8898
  - 授权对象：100.104.0.0/16
- 如果EMR启用了Ranger，则使用DataWorks进行EMR的作业开发前，您需要在EMR中修改配置，添加白名单配置并重启Hive，否则作业运行时会报错Cannot modify spark.yarn.queue at runtime或Cannot modify SKYNET_BIZDATE at runtime。
  i. 白名单的配置通过EMR的自定义参数，添加Key和Value进行配置，以Hive组件的配置为例，配置值如下。

  ```
  hive.security.authorization.sqlstd.confwhitelist.append=tez.*|spark.*|mapred.*|mapreduce.*|ALISA.*|SKYNET.*
  ```

  > **说明** 其中 `ALISA.*` 和 `SKYNET.*` 为DataWorks专有的配置。

  ii. 白名单配置完成后需要重启服务，重启后配置才会生效。重启服务的操作详情请参见重启服务。
- 已开通独享调度资源组，并且独享调度资源组需要绑定EMR所在的VPC专有网络，详情请参见新增和使用独享调度资源组。

  > **说明** 仅支持使用独享调度资源组运行该类型任务。

背景信息

本次实验涉及的阿里云产品如下：

- E-MapReduce
- 数据工场DataWorks
- 对象存储OSS

操作步骤

1. 创建EMR集群。
   i. 登录E-MapReduce控制台。

ii. 选择**华东2（上海）**区域，单击**创建集群**。

> **说明**
> - 由于源数据存储在华东2（上海），建议EMR集群创建在相同的区域。
> - 您可以通过**一键购买**和**自定义购买**两种方式创建EMR集群，本文以自定义购买为例。

iii. 在**自定义购买 > 软件配置**对话框中，选择**集群类型**为**Hadoop**，其它配置项默认无需修改。单击**下一步：硬件配置**。

iv. 在**硬件配置**对话框中，选择**付费类型**为**按量付费**，并进行**网络配置**和**实例配置**，单击**下一步：基础配置**。

v. 在**基础配置**对话框中，输入**集群名称**，并选择**密钥对**，单击**下一步：确定**。

EMR默认选项不开启挂载公网，创建集群后只能通过内网访问EMR集群。本次实验的Workshop操作中不涉及挂载公网，直接单击**挂载公网说明**对话框中的**继续下一步**即可。如果您需要公网访问，请进入ECS控制台挂载EIP。

vi. 在**确认**对话框中，确认订单无误后，勾选**《E-MapReduce服务条款》**，单击**创建**。

2. 初始化集群。

购买成功后，即可进入**集群管理**页面进行查看，集群初始化需要几分钟的时间。

i. 集群初始化成功后，单击顶部菜单栏中的**数据开发**。

ii. 在**数据开发**页面，单击**新建项目**。

iii. 在**新建项目**对话框中，输入**项目名称**和**项目描述**。

> **说明** 请使用主账号创建项目，该项目用于关联DataWorks工作空间。

iv. 单击**创建**。

3. 创建DataWorks工作空间。

> **说明** 因本实验提供的数据资源都在华东2（上海），建议您将工作空间创建在华东2（上海），以避免工作空间创建在其它区域，添加数据源时出现网络不可达的情况。

i. 鼠标悬停至EMR控制台左上角的图标，单击**产品与服务 > 大数据（数加） > DataWorks**。

ii. 在左侧导航栏，单击**工作空间列表**。

iii. 在**工作空间列表**页面，鼠标悬停至左上角的地域，单击需要创建工作空间的地域。

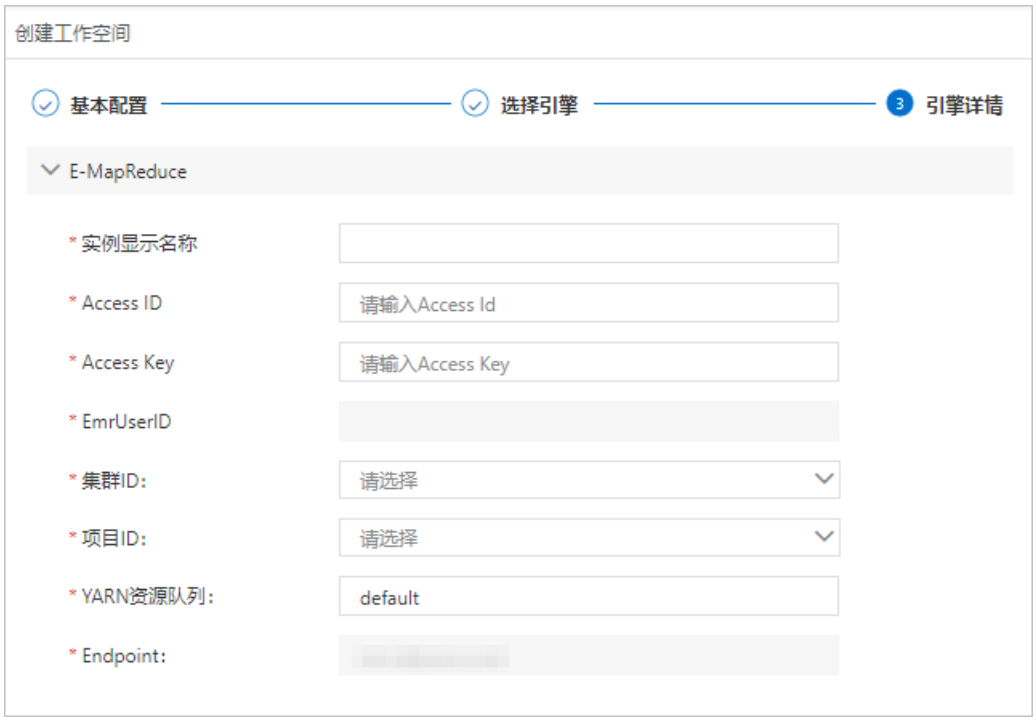
iv. 单击**创建工作空间**，进行**基本配置**，单击**下一步**。

分类	参数	描述
基本信息	**工作空间名称**	工作空间名称的长度需要在3~27个字符，以字母开头，且只能包含字母下划线和数字。
	显示名	显示名不能超过27个字符，只能字母、中文开头，仅包含中文、字母、下划线和数字。
	模式	包括**简单模式**和**标准模式**，本文以创建**简单模式**的工作空间为例。
	描述	对创建的工作空间进行简单描述。
高级设置	**能下载select结果**	设置是否允许下载数据开发中查询的数据结果。

v. 在**选择引擎**对话框中，选中**E-MapReduce**引擎，单击**下一步**。

DataWorks已正式商用，如果该地域没有开通，您需要首先开通正式商用服务。

vi. 在**引擎详情**对话框中，配置各项参数。



参数	描述
实例显示名称	自定义实例名称。
Access ID	已经授权可以访问EMR集群的账号的AccessKey ID。
Access Key	已经授权可以访问EMR集群的账号的AccessKey Secret。
EmrClusterID	集群ID，从EMR端获取。
集群ID	当前集群创建者的用户ID。
项目ID	当前集群下的项目ID。
YARN资源队列	当前集群下的队列名称。如果无特殊需求，请输入*default*。
Endpoint	EMR的Endpoint，从EMR端获取。

vii. 配置完成后，单击**创建工作空间**。

4. 购买OSS并创建Bucket。
   i. 购买OSS，详情请参见开通OSS服务。
   ii. 登录OSS控制台。
   iii. 在左侧导航栏，单击**Bucket列表**。
   iv. 在**Bucket列表**页面，单击**创建Bucket**。
   v. 在**创建Bucket**对话框中，配置各项参数，单击**确定**。

> **说明** 此处需要选择**区域**为**华东2（上海）**，更多参数说明请参见创建存储空间。

vi. 单击相应的**Bucket名称**，进入Bucket的**文件管理**页面。

vii. 在**新建目录**对话框中，输入**目录名**，单击**确定**。

> **说明** 此处需要新建三个目录，分别存放同步过来的外部OSS数据源、RDS数据源和JAR资源。

2.2. 采集数据

本文为您介绍如何通过DataWorks采集日志数据至EMR引擎。

前提条件

开始本文的操作前，请准备好需要使用的环境。详情请参见准备环境。

背景信息

根据本次实验模拟的场景，您需要分别新建OSS数据源、RDS数据源，用于存储数据。同时需要新建私有OSS数据源，用于存储同步后的数据。

新建OSS数据源

1. 进入**数据源管理**页面。
   i. 登录DataWorks控制台。
   ii. 在左侧导航栏，单击**工作空间列表**。
   iii. 单击相应工作空间后的**进入数据集成**。

   如果您已在DataWorks的某个功能模块，请单击左上方的图标，选择**全部产品 > 数据汇聚 > 数据集成**，进入**数据集成**页面。
   iv. 在左侧导航栏，单击**数据源**，进入**工作空间管理 > 数据源管理**页面。
2. 在**数据源管理**页面，单击右上方的**新增数据源**。
3. 在**新增数据源**对话框中，选择数据源类型为**OSS**。
4. 在**新增OSS数据源**对话框中，配置各项参数。此处您可以直接按照示例参数进行填写。

参数	描述
数据源名称	输入数据源名称，示例为oss_workshop_log。
数据源描述	对数据源进行简单描述。
Endpoint	输入Endpoint，示例为 `http://oss-cn-shanghai-internal.aliyuncs.com` 。
Bucket	输入Bucket名称，示例为new-dataworks-workshop。
AccessKey ID	输入访问密钥中的AccessKey ID，示例为LTAI4FvGT3iU4xjKotpUMAjS。
AccessKey Secret	输入访问密钥中的AccessKey Secret，示例为9RSUoRmNxpRC9EhC4m9PjuG7Jzy7px。

5.

6. 连通性测试通过后，单击**完成**。

新建RDS数据源

1. 在**数据源管理**页面，单击右上方的**新增数据源**。
2. 在**新增数据源**对话框中，选择数据源类型为**MySQL**。
3. 在**新增MySQL数据源**对话框中，配置各项参数。此处您可以直接按照示例参数进行填写。

参数	描述
数据源类型	选择**阿里云实例模式**。
数据源名称	输入数据源名称，示例为rds_workshop_log。
数据源描述	对数据源进行简单描述。
地区	选择RDS实例所在的区域，示例为华东2-上海。
RDS实例ID	输入RDS实例ID，示例为rm-bp1z69dodhh85z9qa。
RDS实例主账号ID	输入购买RDS实例的主账号ID，示例为1156529087455811。
数据库名	输入数据库名称，示例为workshop。
用户名	输入用户名，示例为workshop。
密码	输入密码，示例为workshop#2017。

4.
5. 连通性测试通过后，单击**完成**。

新建私有OSS数据源

本次实验将EMR引擎的数据存储在OSS数据源中。

1. 在**数据源管理**页面，单击右上方的**新增数据源**。
2. 在**新增数据源**对话框中，选择数据源类型为**OSS**。
3. 在**新增OSS数据源**对话框中，配置各项参数。

参数	描述
数据源名称	输入数据源的名称。
数据源描述	对数据源进行简单描述。
Endpoint	输入 `http://oss-cn-shanghai-internal.aliyuncs.com` 。
Bucket	您在环境准备中创建的OSS Bucket的名称，示例为dw-emr-demo。
AccessKey ID	当前登录账号的AccessKey ID，您可以进入安全信息管理页面复制AccessKey ID。

参数	描述
AccessKey Secret	输入当前登录账号的AccessKey Secret。

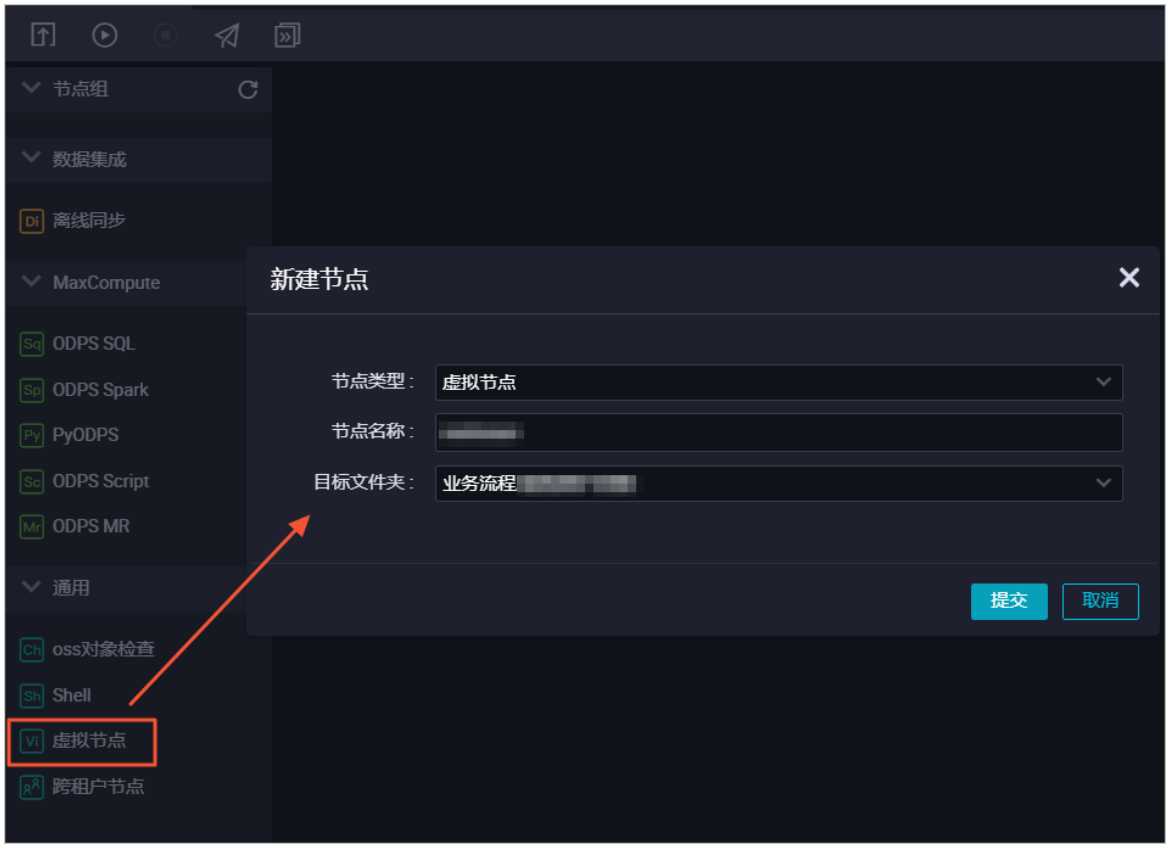
4.
5. 连通性测试通过后，单击**完成**。

新建业务流程

1. 单击左上方的图标，选择**全部产品 > 数据开发 > DataStudio（数据开发）**。
2. 在**数据开发**面板，右键单击**业务流程**，选择**新建业务流程**。
3. 在**新建业务流程**对话框中，输入**业务名称**和**描述**。

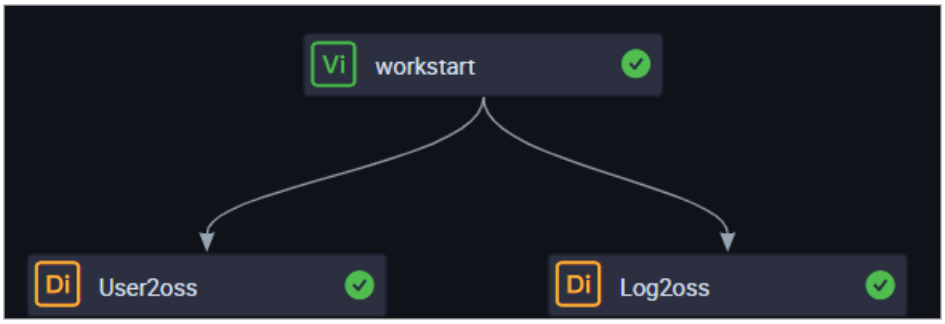
   > **注意** 业务名称不能超过128个字符，且必须是大小写字母、中文、数字、下划线（_）以及英文句号（.）。

4. 单击**新建**。
5. 进入业务流程开发面板，鼠标单击**虚拟节点**并拖拽至右侧的编辑页面。在**新建节点**对话框中，输入**节点名称**为workstart，单击**提交**。



以同样的方式新建两个离线同步节点，**节点名称**分别为Log2oss和User2oss。

6. 通过拖拽连线，将workstart节点设置为两个离线同步节点的上游节点。



配置workstart节点

1. 在**数据开发**页面，双击相应业务流程下的虚拟节点。打开该节点的编辑页面，单击右侧的**调度配置**。
2. 在**调度依赖**区域，单击**使用工作空间根节点**，设置workstart节点的上游节点为工作空间根节点。

   由于新版本给每个节点都设置了输入输出节点，所以需要给workstart节点设置一个输入。此处设置其上游节点为工作空间根节点，通常命名为工作空间名_root。

3. 配置完成后，单击左上方的图标。

配置离线同步节点

1. 同步RDS数据源的用户信息至自建的OSS。
   i. 在**数据开发**页面，双击User2oss节点，进入节点配置页面。

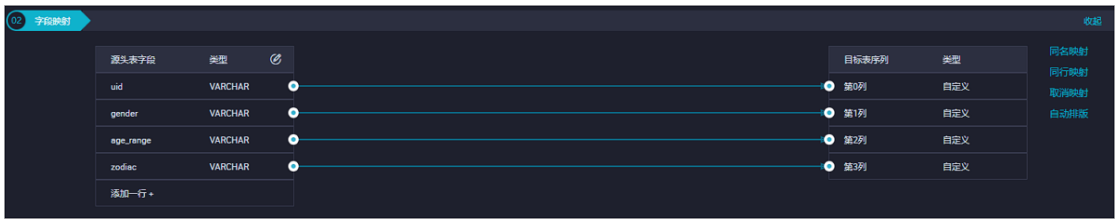
ii. 选择数据来源。

参数	描述
数据源	选择**MySQL > rds_workshop_log**数据源。
表	选择数据源中的ods_user_info_d。
数据过滤	您将要同步数据的筛选条件，暂时不支持limit关键字过滤。此处可以不填写。
切分键	建议使用主键或有索引的列作为切分键，仅支持类型为整型的字段。此处设置切分键为uid。

iii. 选择数据去向。

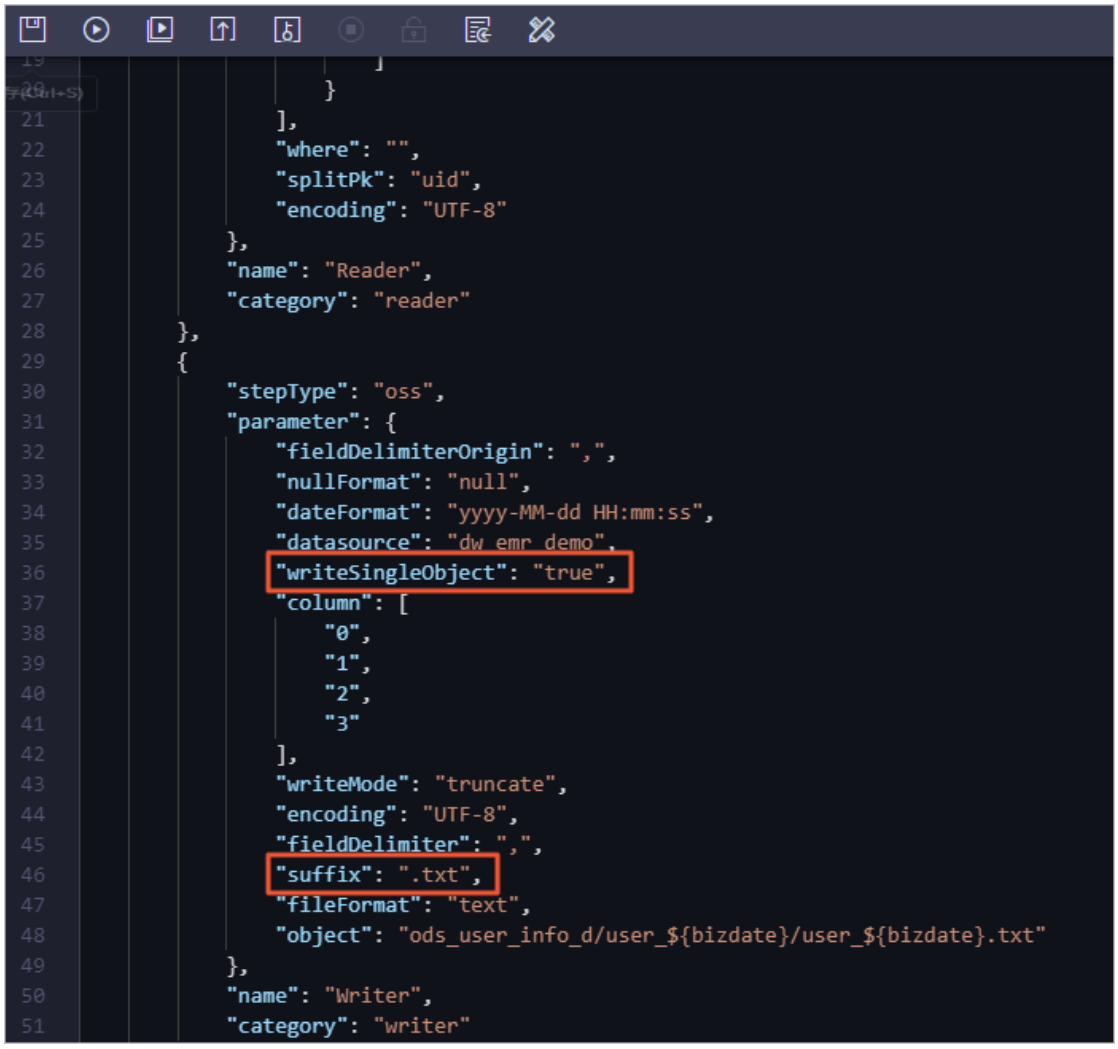
参数	描述
数据源	选择前文创建的OSS数据源，此处示例为**OSS > dw_emr_demo**数据源。
Object前缀	根据您自建OSS的目录进行输入，示例为ods_user_info_d/user_${bizdate}/user_${bizdate}.txt。
文本类型	选择text类型。
列分隔符	输入列分隔符为|。
编码格式	默认为UTF-8格式。
null值	表示null值的字符串，此处可以不填写。
时间格式	时间序列化格式，此处可以不填写。
前缀冲突	此处选择**替换原有文件**。

iv. 配置字段映射。



v. 配置**通道控制**，单击工具栏中的图标。

vi. 单击工具栏中的图标，在已有的脚本中手动添加参数"writeSingleObject": "true"和"suffix": ".txt"。



```
                    }
                ],
                "where": "",
                "splitPk": "uid",
                "encoding": "UTF-8"
            },
            "name": "Reader",
            "category": "reader"
        },
        {
            "stepType": "oss",
            "parameter": {
                "fieldDelimiterOrigin": ",",
                "nullFormat": "null",
                "dateFormat": "yyyy-MM-dd HH:mm:ss",
                "datasource": "dw_emr_demo",
                "writeSingleObject": "true",
                "column": [
                    "0",
                    "1",
                    "2",
                    "3"
                ],
                "writeMode": "truncate",
                "encoding": "UTF-8",
                "fieldDelimiter": ",",
                "suffix": ".txt",
                "fileFormat": "text",
                "object": "ods_user_info_d/user_${bizdate}/user_${bizdate}.txt"
            },
            "name": "Writer",
            "category": "writer"
```

> 说明
>
> - writeSingleObject和suffix参数仅支持脚本模式进行添加。
> - 存储的路径object需要与自建OSS中的目录一致。

vii. 配置完成后，单击工具栏中的图标。

2. 同步OSS数据源的日志信息至自建的OSS。

i. 在**数据开发**页面，双击Log2oss节点，进入节点配置页面。

ii. 选择数据来源。

参数	描述
数据源	选择**OSS > oss_workshop_log**数据源。
Object前缀	输入user_log.txt。
文本类型	选择text类型。
列分隔符	输入列分隔符为|。
编码格式	默认为UTF-8格式。
null值	表示null值的字符串，此处可以不填写。
压缩格式	包括None、Gzip、Bzip2和Zip四种类型，此处选择None。
是否包含表头	默认为No。

iii. 选择数据去向。

参数	描述
数据源	选择前文创建的OSS数据源，此处示例为**OSS > dw_emr_demo**数据源。
Object前缀	根据您自建OSS的目录进行输入，示例为ods_raw_log_d/user_log_${bizdate}/user_log_${bizdate}.txt。
文本类型	选择text类型。
列分隔符	输入列分隔符为|。
编码格式	默认为UTF-8格式。
null值	表示null值的字符串，此处可以不填写。
时间格式	时间序列化格式，此处可以不填写。
前缀冲突	此处选择**替换原有文件**。

iv. 配置字段映射。

> 注意 源数据表中只有一列数据，此处需要删除其它映射过来的空列。

v. 配置**通道控制**，单击工具栏中的图标。

vi. 单击工具栏中的图标，在已有的脚本中手动添加参数"writeSingleObject": "true"和"suffix": ".txt"。

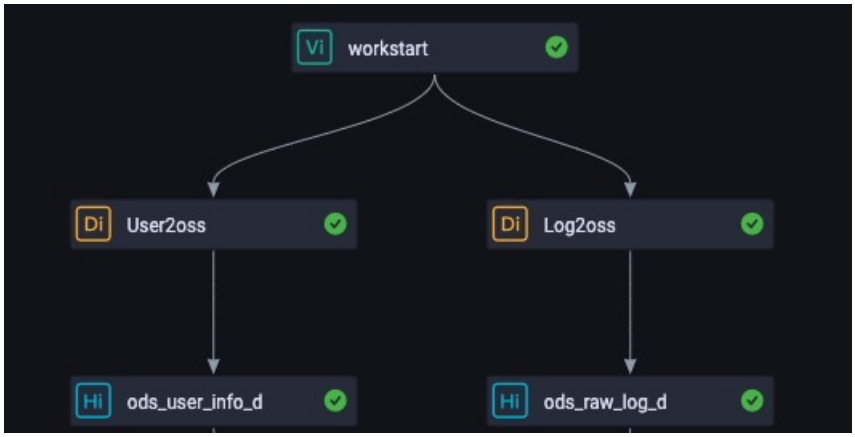
> 说明
> - writeSingleObject和suffix参数仅支持脚本模式进行添加。
> - 存储的路径object需要与自建OSS中的目录一致。

vii. 配置完成后，单击工具栏中的图标。

新建表

1. 在**数据开发**页面打开新建的业务流程，右键单击**EMR**，选择**新建 > EMR Hive**。
2. 在**新建节点**对话框中，输入**节点名称**，单击**提交**。

   此处需要新建两个EMR Hive节点（ods_user_info_d和ods_raw_log_d），分别新建存储同步过来的OSS日志数据和RDS日志数据的两张表。



3. 分别在EMR Hive节点中，**选择EMR引擎**并输入建表语句，单击**保存**并**运行**各建表语句。
   - 新建ods_user_info_d表。

     双击ods_user_info_d节点，在右侧的编辑页面输入下述建表语句。

```
--author
--create time:2020-06-12 14:10:02
--*************************************************************************
CREATE EXTERNAL TABLE IF NOT  EXISTS ods_user_info_d
(
    `uid` STRING COMMENT '用户ID',
    `gender` STRING COMMENT '性别',
    `age_range` STRING COMMENT '年龄段',
    `zodiac` STRING COMMENT '星座'
) PARTITIONED BY (
  dt STRING
)
ROW FORMAT  delimited fields terminated by '|'
LOCATION 'oss://dw-emr-demo/ods_user_info_d/';

ALTER ods_user_info_d ADD IF NOT EXISTS PARTITION (dt=${bizdate})
LOCATION 'oss://dw-emr-demo/ods_user_info_d/user_${bizdate}/';
```

```
CREATE EXTERNAL TABLE IF NOT EXISTS ods_user_info_d
(
    `uid` STRING COMMENT '用户ID',
    `gender` STRING COMMENT '性别',
    `age_range` STRING COMMENT '年龄段',
    `zodiac` STRING COMMENT '星座'
) PARTITIONED BY (
  dt STRING
)
ROW FORMAT  delimited fields terminated by '|'
LOCATION 'oss://dw-emr-demo/ods_user_info_d/';
ALTER TABLE ods_user_info_d ADD IF NOT EXISTS PARTITION (dt=${bizdate})
LOCATION 'oss://dw-emr-demo/ods_user_info_d/user_${bizdate}/';
```

> **说明** 上述代码中的location为示例路径，需要输入您建立的文件夹的路径名称。

- 新建ods_raw_log_d表。

  双击ods_raw_log_d节点，在右侧的编辑页面输入下述建表语句。

```
--创建OSS日志对应目标表
CREATE EXTERNAL TABLE IF NOT EXISTS ods_raw_log_d
(
  `col` STRING
) PARTITIONED BY (
  dt STRING
);
ALTER TABLE ods_raw_log_d ADD IF NOT EXISTS PARTITION (dt=${bizdate})
LOCATION 'oss://dw-emr-demo/ods_raw_log_d/user_log_${bizdate}/';
```

> **说明** 上述代码中的location为示例路径，需要输入您建立的文件夹的路径名称。

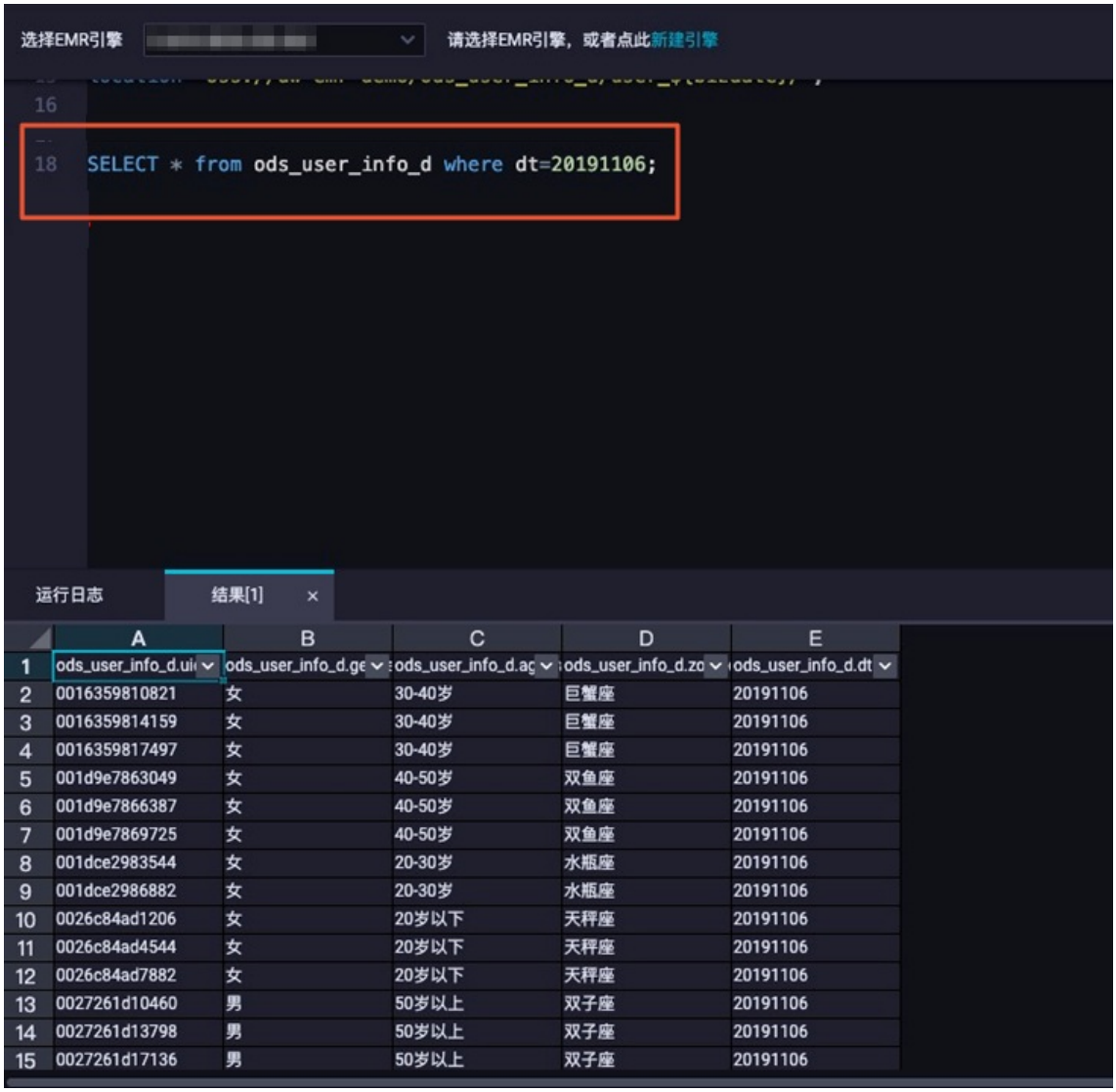
4. 查看数据同步结果。

   建表语句运行成功后，分别在两个EMR Hive节点中输入查询语句。

> **说明** 查询语句中的分区列需要更新为业务日期。例如，任务运行的日期为20191107，则业务日期为20191106，即任务运行日期的前一天。

- 查询ods_user_info_d表的数据。

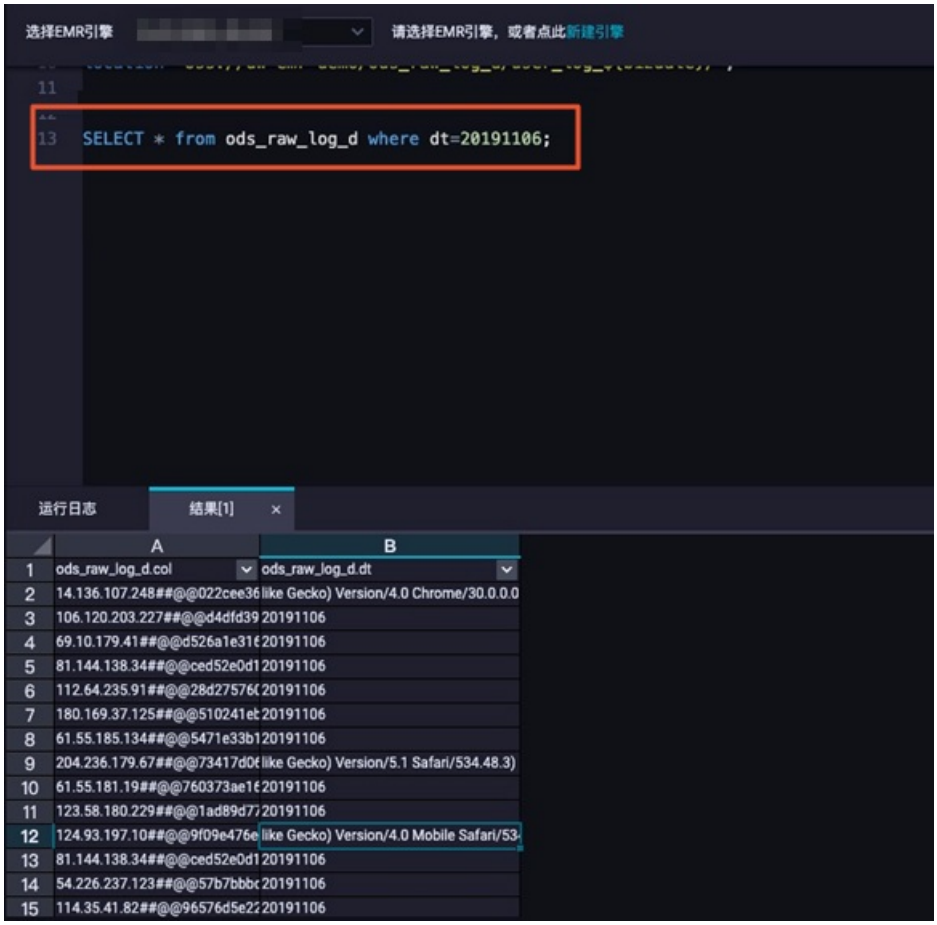
```
SELECT * from ods_user_info_d where dt=业务日期; --业务日期为任务运行日期的前一天。
```



	A	B	C	D	E
1	ods_user_info_d.ui	ods_user_info_d.ge	ods_user_info_d.ag	ods_user_info_d.zo	ods_user_info_d.dt
2	0016359810821	女	30-40岁	巨蟹座	20191106
3	0016359814159	女	30-40岁	巨蟹座	20191106
4	0016359817497	女	30-40岁	巨蟹座	20191106
5	001d9e7863049	女	40-50岁	双鱼座	20191106
6	001d9e7866387	女	40-50岁	双鱼座	20191106
7	001d9e7869725	女	40-50岁	双鱼座	20191106
8	001dce2983544	女	20-30岁	水瓶座	20191106
9	001dce2986882	女	20-30岁	水瓶座	20191106
10	0026c84ad1206	女	20岁以下	天秤座	20191106
11	0026c84ad4544	女	20岁以下	天秤座	20191106
12	0026c84ad7882	女	20岁以下	天秤座	20191106
13	0027261d10460	男	50岁以上	双子座	20191106
14	0027261d13798	男	50岁以上	双子座	20191106
15	0027261d17136	男	50岁以上	双子座	20191106

- 查询ods_raw_log_d表的数据。

```
SELECT * from ods_raw_log_d where dt=业务日期; --业务日期为任务运行日期的前一天。
```



后续步骤

现在，您已经学习了如何进行日志数据同步，完成数据的采集，您可以继续下一个教程。在该教程中，您将学习如何对采集的数据进行计算与分析。详情请参见加工数据。

2.3. 加工数据

本文将为您介绍如何通过DataWorks中的EMR Hive节点加工采集的日志数据。

前提条件

开始本实验前，请首先完成采集数据中的操作。

在OSS上传资源

1. 下载ip2region-emr.jar存放至本地。
2. 登录OSS控制台。
3. 在左侧**存储空间**列表中，单击目标存储空间（示例为dw-emr-demo）。
4. 单击**文件管理**，打开在 *环境准备* 章节新建的用于存储JAR资源的目录，示例的目录名为ip2region。
5. 单击**上传文件**，在**上传文件**对话框中，设置上传文件的参数。

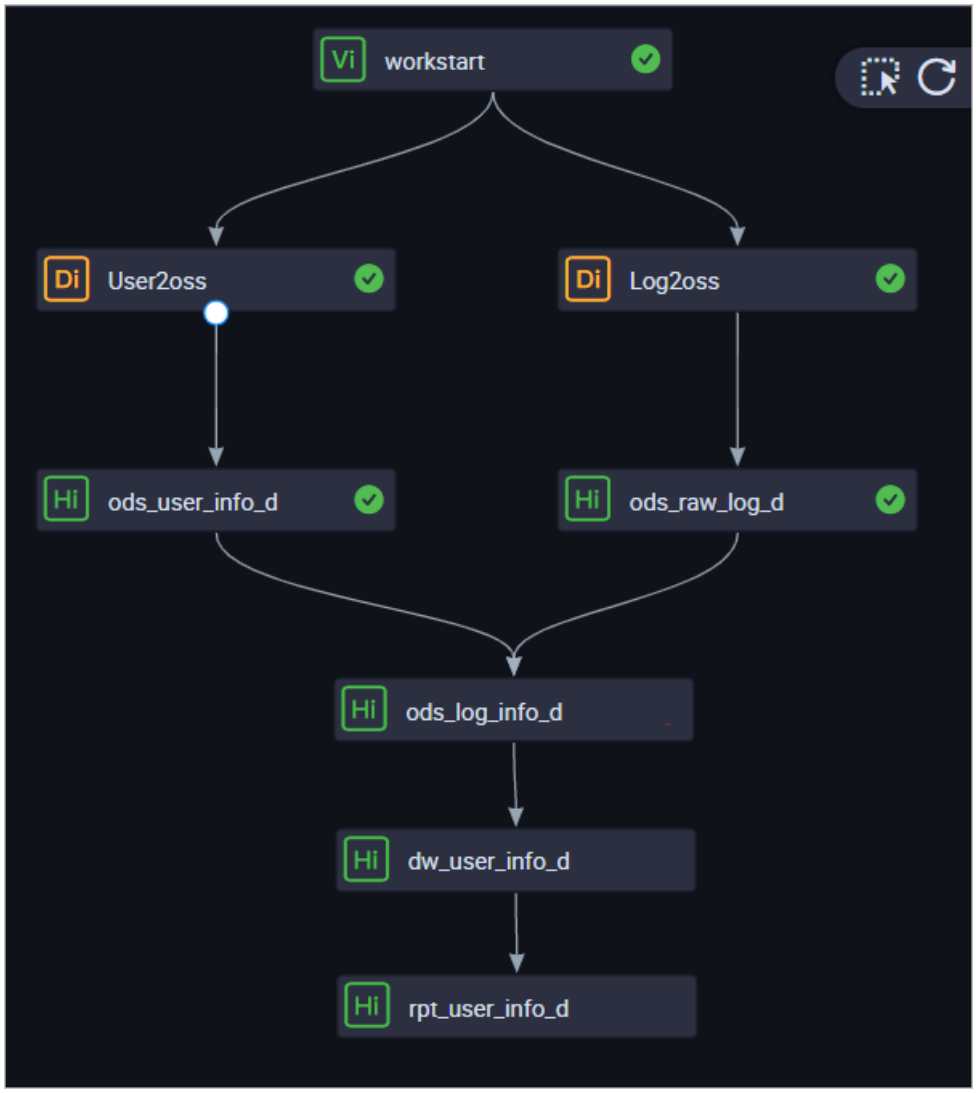
参数	描述
上传到	选择**当前目录**，示例的目录路径为*oss://dw-emr-demo/ip2region/*。
文件ACL	默认为**继承Bucket**，即单个文件的读写权限以Bucket的读写权限为准。
上传文件	单击**直接上传**，选择已下载的ip2region-emr.jar文件。

设计业务流程

业务流程节点间依赖关系的配置请参见采集数据。

双击新建的业务流程打开编辑页面，鼠标单击**EMR Hive**并拖拽至右侧的编辑页面。在**新建节点**对话框中，输入**节点名称**，单击**提交**。

此处需要新建3个EMR Hive节点，依次命名为ods_log_info_d、dw_user_info_all_d和rpt_user_info_d，并配置如下图所示的依赖关系。



配置EMR Hive节点

1. 配置ods_log_info_d节点。
   i. 双击ods_log_info_d节点，进入节点配置页面。
   ii. 在节点编辑页面，编写如下语句。

> **说明** 如果您的工作空间绑定多个EMR引擎，需要选择EMR引擎。如果仅绑定一个EMR引擎，则无需选择。

```
--创建ODS层表
CREATE TABLE IF NOT EXISTS ods_log_info_d (
  ip STRING COMMENT 'ip地址',
  uid STRING COMMENT '用户ID',
  `time` STRING COMMENT '时间yyyymmddhh:mi:ss',
  status STRING COMMENT '服务器返回状态码',
  bytes STRING COMMENT '返回给客户端的字节数',
  region STRING COMMENT '地域，根据ip得到',
  method STRING COMMENT 'http请求类型',
  url STRING COMMENT 'url',
  protocol STRING COMMENT 'http协议版本号',
```

```
  referer STRING COMMENT '来源url',
  device STRING COMMENT '终端类型 ',
  identity STRING COMMENT '访问类型 crawler feed user unknown'
)
PARTITIONED BY (
  dt STRING
);
create function  getregion as 'org.alidata.emr.udf.Ip2Region'
using jar 'oss://dw-emr-demo/ip2region/ip2region-emr.jar';
ALTER TABLE ods_log_info_d ADD IF NOT EXISTS PARTITION (dt=${bizdate});
set hive.vectorized.execution.enabled = false;
INSERT OVERWRITE TABLE ods_log_info_d PARTITION (dt=${bizdate})
SELECT ip
  , uid
  , tm
  , status
  , bytes
  , getregion(ip) AS region --使用自定义UDF通过ip得到地域。
  , regexp_extract(request, '(^[^ ]+) .*') AS method --通过正则把request差分为三个字段
。
  , regexp_extract(request, '^[^ ]+ (.*) [^ ]+$') AS url
  , regexp_extract(request, '.* ([^ ]+$)') AS protocol
  , regexp_extract(referer, '^[^/]+://([^/]+){1}') AS referer  --通过正则清洗refer，
得到更精准的url。
  , CASE
    WHEN lower(agent) RLIKE 'android' THEN 'android' --通过agent得到终端信息和访问形式
。
    WHEN lower(agent) RLIKE 'iphone' THEN 'iphone'
    WHEN lower(agent) RLIKE 'ipad' THEN 'ipad'
    WHEN lower(agent) RLIKE 'macintosh' THEN 'macintosh'
    WHEN lower(agent) RLIKE 'windows phone' THEN 'windows_phone'
    WHEN lower(agent) RLIKE 'windows' THEN 'windows_pc'
    ELSE 'unknown'
  END AS device
  , CASE
    WHEN lower(agent) RLIKE '(bot|spider|crawler|slurp)' THEN 'crawler'
    WHEN lower(agent) RLIKE 'feed'
    OR regexp_extract(request, '^[^ ]+ (.*) [^ ]+$') RLIKE 'feed' THEN 'feed'
    WHEN lower(agent) NOT RLIKE '(bot|spider|crawler|feed|slurp)'
    AND agent RLIKE '^[Mozilla|Opera]'
    AND regexp_extract(request, '^[^ ]+ (.*) [^ ]+$') NOT RLIKE 'feed' THEN 'user'
    ELSE 'unknown'
  END AS identity
  FROM (
    SELECT SPLIT(col, '##@@')[0] AS ip
    , SPLIT(col, '##@@')[1] AS uid
    , SPLIT(col, '##@@')[2] AS tm
    , SPLIT(col, '##@@')[3] AS request
    , SPLIT(col, '##@@')[4] AS status
    , SPLIT(col, '##@@')[5] AS bytes
    , SPLIT(col, '##@@')[6] AS referer
    , SPLIT(col, '##@@')[7] AS agent
    FROM ods_raw_log_d
  WHERE dt = ${bizdate}
```

```
) a;
```

iii. 单击工具栏中的。

2. 配置dw_user_info_all_d节点。

i. 双击dw_user_info_all_d节点，进入节点配置页面。

ii. 在节点编辑页面，编写如下语句。

> **说明** 如果您的工作空间绑定多个EMR引擎，需要**选择EMR引擎**。如果仅绑定一个EMR引擎，则无需选择。

```
--创建DW层表
CREATE TABLE IF NOT EXISTS dw_user_info_all_d (
  uid STRING COMMENT '用户ID',
  gender STRING COMMENT '性别',
  age_range STRING COMMENT '年龄段',
  zodiac STRING COMMENT '星座',
  region STRING COMMENT '地域，根据ip得到',
  device STRING COMMENT '终端类型 ',
  identity STRING COMMENT '访问类型 crawler feed user unknown',
  method STRING COMMENT 'http请求类型',
  url STRING COMMENT 'url',
  referer STRING COMMENT '来源url',
  `time` STRING COMMENT '时间yyyymmddhh:mi:ss'
)
PARTITIONED BY (
  dt STRING
);
ALTER TABLE dw_user_info_all_d ADD IF NOT EXISTS PARTITION (dt = ${bizdate});
INSERT OVERWRITE TABLE dw_user_info_all_d PARTITION (dt=${bizdate})
SELECT COALESCE(a.uid, b.uid) AS uid
  , b.gender
  , b.age_range
  , b.zodiac
  , a.region
  , a.device
  , a.identity
  , a.method
  , a.url
  , a.referer
  , a.`time`
FROM (
  SELECT *
  FROM ods_log_info_d
  WHERE dt = ${bizdate}
) a
LEFT OUTER JOIN (
  SELECT *
  FROM ods_user_info_d
  WHERE dt = ${bizdate}
) b
ON a.uid = b.uid;
```

iii. 单击工具栏中的。

3. 配置rpt_user_info_d节点。

i. 双击rpt_user_info_d节点，进入节点配置页面。

ii. 在节点编辑页面，编写如下语句。

> **说明** 如果您的工作空间绑定多个EMR引擎，需要**选择EMR引擎**。如果仅绑定一个EMR引擎，则无需选择。

```
--创建RPT层表
CREATE TABLE IF NOT EXISTS rpt_user_info_d (
  uid STRING COMMENT '用户ID',
  region STRING COMMENT '地域，根据ip得到',
  device STRING COMMENT '终端类型 ',
  pv BIGINT COMMENT 'pv',
  gender STRING COMMENT '性别',
  age_range STRING COMMENT '年龄段',
  zodiac STRING COMMENT '星座'
)
PARTITIONED BY (
  dt STRING
);
ALTER TABLE rpt_user_info_d ADD IF NOT EXISTS PARTITION (dt=${bizdate});
INSERT OVERWRITE TABLE rpt_user_info_d PARTITION (dt=${bizdate})
SELECT uid
  , MAX(region)
  , MAX(device)
  , COUNT(0) AS pv
  , MAX(gender)
  , MAX(age_range)
  , MAX(zodiac)
FROM dw_user_info_all_d
WHERE dt = ${bizdate}
GROUP BY uid;
```

iii. 单击工具栏中的。

提交业务流程

1. 在业务流程的编辑页面，单击，运行业务流程。
2. 待业务流程中的所有节点后出现，单击，提交运行成功的业务流程。
3. 选择**提交**对话框中需要提交的节点，勾选**忽略输入输出不一致的告警**。
4. 单击**提交**。

在生产环境运行任务

1. 任务发布成功后，单击右上角的**运维中心**。
   您也可以进入业务流程的编辑页面，单击工具栏中的**前往运维**，进入**运维中心**页面。
2. 单击左侧导航栏中的**周期任务运维 > 周期任务**，进入**周期任务**页面，单击workstart虚节点。
3. 在右侧的DAG图中，右键单击workstart节点，选择**补数据 > 当前节点及下游节点**。
4. 勾选需要补数据的任务，输入业务日期，单击**确定**，自动跳转至**补数据实例**页面。
5. 单击**刷新**，直至SQL任务全部运行成功即可。

后续步骤

现在，您已经学习了如何创建EMR Hive节点、如何处理原始日志数据。您可以继续下一个教程，学习如何在数据地图模块开启元数据收集功能，并查看数据表信息。详情请参见收集和查看元数据。

2.4. 收集和查看元数据

本文为您介绍如何在数据地图模块开启元数据收集功能，并查看数据表信息。

前提条件

开始本实验前，请首先完成加工数据中的操作。

背景信息

元数据是数据的描述数据，可以为数据说明其属性（名称、大小、数据类型等），或结构（字段、类型、长度等），或其相关数据（位于何处、拥有者、产出任务、访问权限等）。DataWorks中元数据主要指库、表相关的信息，元数据管理对应的主要应用是数据地图。

开启元数据收集

1.
2.
3.
4.
5.
6.
7.
8. 在**E-MapReduce元数据采集**页面，单击新建的采集器后的**运行全量获取**。

   单击页面右上角的**刷新**，待EMR采集实例的**运行状态**显示为**收集成功**即可。

   > **说明** 全量采集E-MapReduce元数据后，系统会开启自动增量采集，自动同步表中新增的元数据。

查看数据表信息

1. 在当前页面的顶部菜单栏，单击**全部数据**。
2. 在**全部数据**页面，单击**E-MapReduce**。
3. 在**E-MapReduce**页签下，单击表名（rpt_user_info_d），查看该表的详情。

   您也可以在顶部搜索框中输入关键字进行搜索，查看E-MapReduce表详情。
4. 单击**血缘信息**，查看该表的上下游血缘详情。

后续步骤

现在，您已经学习了如何在数据地图模块开启元数据收集功能，并查看数据表信息。您可以继续下一个教程，学习如何对开发完成的任务设置数据质量监控，保证任务运行的质量。详情请参见配置数据质量监控。

2.5. 配置数据质量监控

本文为您介绍如何配置表ods_log_info_d的数据质量监控规则。

前提条件

在进行本实验前，请首先采集元数据，详情请参见收集和查看元数据。

操作步骤

1. 进入**数据开发**页面。
   i. 登录DataWorks控制台。
   ii. 在左侧导航栏，单击**工作空间列表**。
   iii. 选择工作空间所在地域后，单击相应工作空间后的**进入数据开发**。
2. 进入ods_log_info_d表的**监控规则**页面。
   i. 单击左上角的图标，选择**全部产品 > 数据质量**。
   ii. 在左侧导航栏，单击**监控规则**，从数据源下拉列表中选择**EMR**。
   iii. 单击ods_log_info_d表后的**配置监控规则**。
3. 添加分区表达式。
   i. 在**已添加的分区表达式**模块，单击**+**。
   ii. **添加分区**对话框中，选择**分区表达式**为dt=$[yyyymmdd-1]，并选择相应的**数据质量插件**。
   iii. 单击**计算**，即可查看调度结果。
   iv. 确认无误后，单击**确认**。
4. 创建规则。
   i. 选中分区后，单击右上角的**创建规则**。
   ii. 在**模板规则**对话框中，单击**添加监控规则**。
   iii. 配置监控规则。

参数	描述
规则名称	新建规则的名称。
强弱	设置规则的强度为**强**。
动态阈值	根据自身需求，选择是否开启动态阈值。 **说明** 您需要购买DataWorks企业版及以上版本，才可以使用动态阈值功能。
规则来源	包括**内置模板**和**规则模板库**。 **说明** 您需要购买DataWorks企业版及以上版本，才可以选择**规则模板库**。
规则字段	请选择**表级规则（table）**。

参数	描述
规则模板	请选择**表行数，固定值**。
比较方式	请选择**大于**。
期望值	设置为0，即比较方式为期望值大于0。

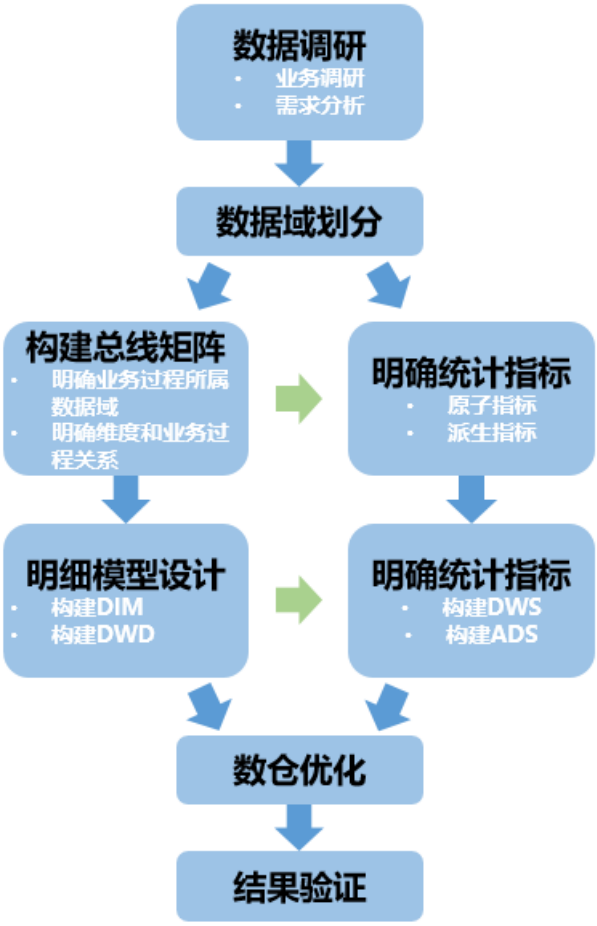
iv. 配置完成后，单击**批量添加**。

5. 进行试跑。
   i. 单击页面右上角的**试跑**。
   ii. 在**试跑**对话框中，选择**调度时间**和**资源组**，单击**试跑**。
   iii. 试跑完成后，单击**试跑成功！点击查看试跑结果**，即可跳转至试跑结果页面。
6. 进行关联调度。
   i. 在ods_log_info_d表的**监控规则**页面，单击**关联调度**。
   ii. 在**关联调度**对话框中，输入节点ID或节点名称，单击**添加**。
   iii. 添加完成后，即可完成与调度节点任务的绑定，则任务实例运行完成都会触发数据质量的检查。
7. 配置任务订阅。
   i. 在ods_log_info_d表的**监控规则**页面，单击**订阅管理**。
   ii. 在**订阅管理**对话框中，设置**订阅方式**和**接受对象**。
   目前支持的订阅方式包括**邮件通知**、**邮件和短信通知**、**钉钉群机器人**和**钉钉群机器人@ALL**。
   iii. 设置完成后，单击**保存**，您可以进入**我的订阅**页面进行查看和修改。

3.构建与优化数据仓库

3.1. 数仓构建流程

下图为MaxCompute数据仓库构建的整体流程。



基本概念

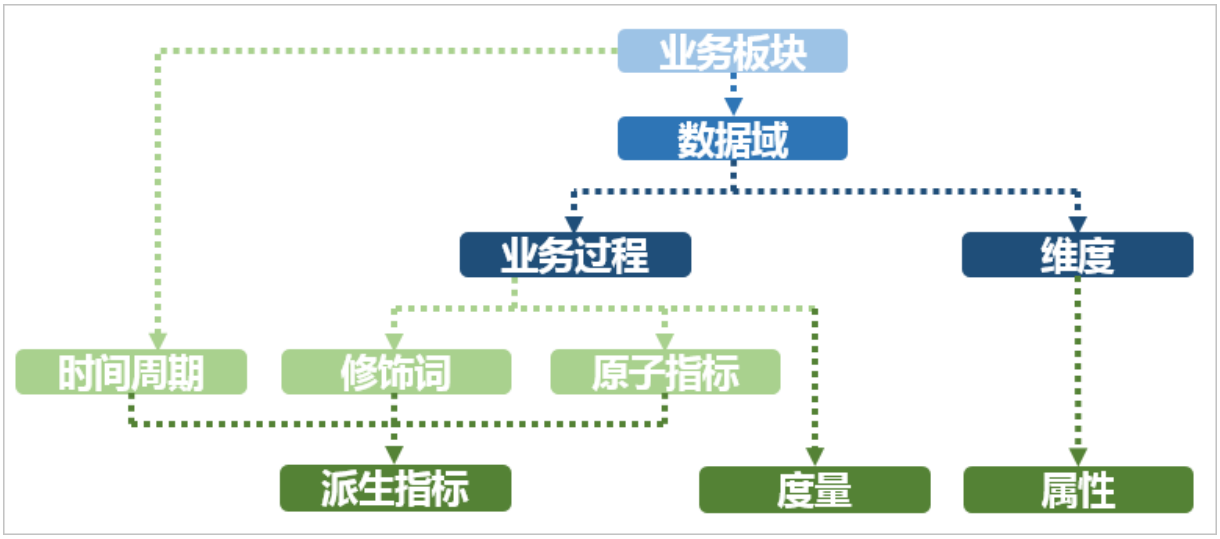
在正式学习本教程之前，您需要首先理解以下基本概念：

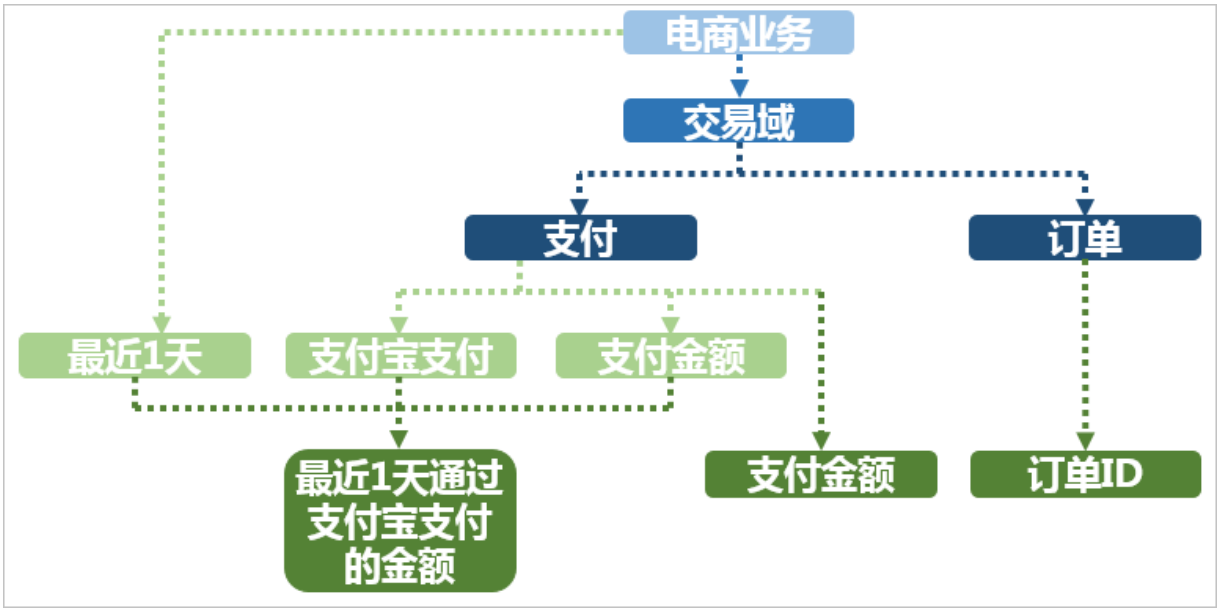
- 业务板块：比数据域更高维度的业务划分方法，适用于庞大的业务系统。
- 维度：维度建模由Ralph Kimball提出。维度模型主张从分析决策的需求出发构建模型，为分析需求服务。维度是度量的环境，是我们观察业务的角度，用来反映业务的一类属性 。属性的集合构成维度 ，也可以称为实体对象。例如， 在分析交易过程时，可以通过买家、卖家、商品和时间等维度描述交易发生的环境。
- 属性（维度属性）：维度所包含的表示维度的列称为维度属性。维度属性是查询约束条件、分组和报表标签生成的基本来源，是数据易用性的关键。
- 度量：在维度建模中，将度量称为事实 ，将环境描述为维度，维度是用于分析事实所需要的多样环境。度量通常为数值型数据，作为事实逻辑表的事实。
- 指标：指标分为原子指标和派生指标。原子指标是基于某一业务事件行为下的度量，是业务定义中不可再拆分的指标，是具有明确业务含义的名词 ，体现明确的业务统计口径和计算逻辑，例如支付金额。
  - 原子指标=业务过程+度量。

- 派生指标=时间周期+修饰词+原子指标，派生指标可以理解为对原子指标业务统计范围的圈定。

- 业务限定：统计的业务范围，筛选出符合业务规则的记录（类似于SQL中**where**后的条件，不包括时间区间）。
- 统计周期：统计的时间范围，例如最近一天，最近30天等（类似于SQL中**where**后的时间条件）。
- 统计粒度：统计分析的对象或视角，定义数据需要汇总的程度，可理解为聚合运算时的分组条件（类似于SQL中的**group by**的对象）。粒度是维度的一个组合，指明您的统计范围。例如，某个指标是某个卖家在某个省份的成交额，则粒度就是卖家、地区这两个维度的组合。如果您需要统计全表的数据，则粒度为全表。在指定粒度时，您需要充分考虑到业务和维度的关系。统计粒度常作为派生指标的修饰词而存在。

基本概念之间的关系和举例如下图所示。





3.2. 业务调研

3.2.1. 确定需求

您在构建数据仓库之前，首先需要确定构建数据仓库的目标与需求，并进行全面的业务调研。您需要了解真实的业务需求，以及确定数据仓库要解决的问题。

业务调研

充分的业务调研和需求分析是数据仓库建设的基石，直接决定数据仓库能否建设成功。在数仓建设项目启动前，您需要请相关的业务人员介绍具体的业务，以便明确各个团队的分析员和运营人员的需求，沉淀出相关文档。

您可以通过调查表和访谈等形式详细了解以下信息：

1. 用户的组织架构和分工界面。

   例如，用户可能分为数据分析、运营和维护部门人员，各个部门对数据仓库的需求不同，您需要对不同部门分别进行调研。

2. 用户的整体业务架构，各个业务板块之间的联系和信息流动的流程。

   您需要梳理出整体的业务数据框架。

3. 各个已有的业务板块的主要功能及获取的数据。

本教程中以A公司的电商业务为例，梳理出业务数据框架如下图所示。A公司的电商业务板块分为招商、供应链、营销和服务四个模块，每个板块的需求和数据应用都不同。您在构建数据仓库之前，首先需要明确构建数据仓库的业务板块和需要具体满足的业务需求。

A公司电商	招商	供应链	营销	服务
商业目标/业务需求	寻找优质商家并帮助快速入驻	优化进、销、存链路，降低成本	商家成长、行业增长、精准营销	提升用户体验和留存
数据需求	市场评估、商家成交分析、品牌成交分析	仓库选址、货品规划、货单跟踪	用户运营、营销分析、成交驱动	客户体验、服务质量、完美订单
核心数据	品牌分析、行业趋势、商家流量、商家成交	供应商分层、库存周转、财务结算、库存管理、物流时效	行业用户、行业流量、竞品监控、订单成交	退款纠纷、用户评价、投诉率
数据应用	销售预测、商家分层、生意参谋	物流时效、货品汰换、智能补货	用户画像、成交预测、品类分析、人群投放	假货感知、服务跟踪

此外，您还需要进一步了解各业务板块中已有的数据功能模块。数据功能模块通常和业务板块紧耦合，对应一个或多个表，可以作为构建数据仓库的数据源。下表展现的是一个营销业务板块的数据功能模块。

数据功能模块	A公司电商营销管理
商品管理	Y
用户管理	Y
购买流程	Y
交易订单	Y
用户反馈	Y

> **说明** Y代表包含该数据功能模块，N代表不包含。

本教程中，假设用户是电商营销部门的营销数据分析师。数据需求为最近一天某个类目（例如，厨具）商品在各省的销售总额、该类目Top10销售额商品名称和各省客户购买力分布（人均消费额）等，用于营销分析。最终的业务需求是通过营销分析完成该类目的精准营销，提升销售总额。通过业务调研，我们将着力分析营销业务板块的交易订单数据功能模块。

需求分析

在未考虑数据分析师和业务运营人员的数据需求的情况下，单纯根据业务调研结果构建的数据仓库可用性差。完成业务调研后，您需要进一步收集数据使用者的需求，进而对需求进行深度的思考和分析。

需求分析的途径有两种：

- 根据与分析师和业务运营人员的沟通获知需求。
- 对报表系统中现有的报表进行研究分析。

在需求分析阶段，您需要沉淀出业务分析或报表中的指标，以及指标的定义和粒度。粒度可以作为维度的输入。建议您思考下列问题，对后续的数据建模将有巨大的帮助：

- 业务数据是根据什么（维度、粒度）汇总的，衡量标准是什么？例如，成交量是维度，订单数是成交量的度量。
- 明细数据层和汇总数据层应该如何设计？公共维度层该如何设计？是否有公共的指标？
- 数据是否需要冗余或沉淀到汇总数据层中？

举例：数据分析师需要了解A公司电商业务中厨具类目的成交金额。当获知这个需求后，您需要分析：根据什么（维度）汇总、汇总什么（度量）以及汇总的范围多大（粒度）。例如，类目是维度，金额是度量，范围是全表。此外，还需要思考明细数据和汇总数据应该如何设计、是否是公共层的报表及数据是否需要沉淀到汇总表中等因素。

需求调研的分析产出通常是记录原子与派生指标的文档。

3.2.2. 分析业务过程

业务过程可以概括为一个个不可拆分的行为事件。用户的业务系统中，通过埋点或日常积累，通常已经获取了充足的业务数据。为理清数据之间的逻辑关系和流向，首先需要理解用户的业务过程，了解过程中涉及到的数据系统。

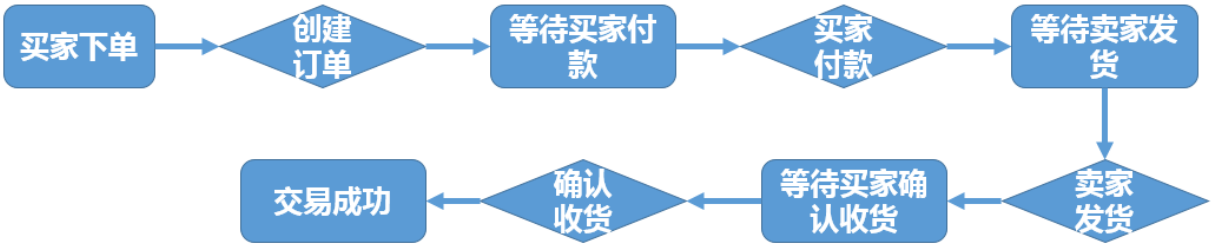
您可以采用过程分析法，将整个业务过程涉及的每个环节一一列清楚，包括技术、数据、系统环境等。在分析企业的工作职责范围（部门）后，您也可以借助工具通过逆向工程抽取业务系统的真实模型。您可以参考业务规划设计文档以及业务运行（开发、设计、变更等）相关文档，全面分析数据仓库涉及的源系统及业务管理系统：

- 每个业务会生成哪些数据，存在于什么数据库中。
- 对业务过程进行分解，了解过程中的每一个环节会产生哪些数据，数据的内容是什么。
- 数据在什么情况下会更新，更新的逻辑是什么。

业务过程可以是单个业务事件，例如交易的支付、退款等；也可以是某个事件的状态，例如当前的账户余额等；还可以是一系列相关业务事件组成的业务流程。具体取决于您分析的是某些事件过去发生情况、当前状态还是事件流转效率。

选择粒度：在业务过程事件分析中，您需要预判所有分析需要细分的程度和范围，从而决定选择的粒度。 识别维表、选择好粒度之后，您需要基于此粒度设计维表，包括维度属性等，用于分析时进行分组和筛选。最后，您需要确定衡量的指标。

本教程中，经过业务过程调研，我们了解到用户电商营销业务的交易订单功能模块的业务流程如下。



这是一个非常典型的电商交易业务流程图。在该业务流程图中，有创建订单、买家付款、卖家发货、确认收货四个核心业务步骤。由于确认收货代表交易成功，我们重点分析确认收货（交易成功）步骤即可。

在明确用户的业务过程之后，您可以根据需要进行分析决策的业务划分数据域。

3.2.3. 划分数据域

数据仓库是面向主题（数据综合、归类并进行分析利用的抽象）的应用。数据仓库模型设计除横向的分层外，通常也需要根据业务情况进行纵向划分数据域。数据域是联系较为紧密的数据主题的集合，是业务对象高度概括的概念层次归类，目的是便于数据的管理和应用。

划分数据域

通常，您需要阅读各源系统的设计文档、数据字典和数据模型设计文档，研究逆向导出的物理数据模型。进而，可以进行跨源的主题域合并，跨源梳理出整个企业的数据域。

数据域是指面向业务分析，将业务过程或者维度进行抽象的集合。为保障整个体系的生命力，数据域需要抽象提炼，并长期维护更新。在划分数据域时，既能涵盖当前所有的业务需求，又能让新业务在进入时可以被包含进已有的数据域或扩展新的数据域。数据域的划分工作可以在业务调研之后进行，需要分析各个业务模块中有哪些业务活动。

数据域可以按照用户企业的部门划分，也可以按照业务过程或者业务板块中的功能模块进行划分。例如A公司电商营销业务板块可以划分为如下数据域，数据域中每一部分都是实际业务过程经过归纳抽象之后得出的。

数据域	业务过程
会员店铺域	注册、登录、装修、开店、关店
商品域	发布、上架、下架、重发
日志域	曝光、浏览、点击
交易域	下单、支付、发货、确认收货
服务域	商品收藏、拜访、培训、优惠券领用
采购域	商品采购、供应链管理

3.2.4. 定义维度与构建总线矩阵

明确每个数据域下有哪些业务过程后，您需要开始定义维度，并基于维度构建总线矩阵。

定义维度

在划分数据域、构建总线矩阵时，需要结合对业务过程的分析定义维度。以本教程中A电商公司的营销业务板块为例，在交易数据域中，我们重点考察确认收货（交易成功）的业务过程。

在确认收货的业务过程中，主要有商品和收货地点（本教程中，假设收货和购买是同一个地点）两个维度所依赖的业务角度。从商品维度我们可以定义出以下维度的属性：

- 商品ID（主键）
- 商品名称
- 商品交易价格
- 商品新旧程度： 1 全新 2 闲置 3 二手
- 商品类目ID
- 商品类目名称
- 品类ID
- 品类名称
- 买家ID
- 商品状态： 0 正常 1 删除 2 下架 3 从未上架
- 商品所在城市
- 商品所在省份

从地域维度，我们可以定义出以下维度的属性：

- 城市code
- 城市名称
- 省份code
- 省份名称

作为维度建模的核心，在企业级数据仓库中必须保证维度的唯一性。以A公司的商品维度为例，有且只允许有一种维度定义。例如，省份code这个维度，对于任何业务过程所传达的信息都是一致的。

构建总线矩阵

明确每个数据域下有哪些业务过程后，即可构建总线矩阵。您需要明确业务过程与哪些维度相关，并定义每个数据域下的业务过程和维度。如下所示是A公司电商板块交易功能的总线矩阵，我们定义了购买省份、购买城市、类目名称、类目ID、品牌名称、品牌ID、商品名称、商品ID、成交金额等维度。

数据域/过程		一致性维度								
		购买省份	购买城市	类目ID	类目名称	品牌ID	品牌名称	商品ID	商品名称	成交金额
交易	下单	Y	Y	Y	Y	Y	Y	Y	Y	N
	支付	Y	Y	Y	Y	Y	Y	Y	Y	N
	发货	Y	Y	Y	Y	Y	Y	Y	Y	N
	确认收货	Y	Y	Y	Y	Y	Y	Y	Y	Y

> **说明** Y代表包含该维度，N代表不包含。

3.2.5. 明确统计指标

需求调研输出的文档中，含有原子指标与派生指标，此时我们需要在设计汇总层表模型前完成指标的设计。

指标定义注意事项

原子指标是明确的统计口径、计算逻辑： `原子指标=业务过程+度量` 。派生指标即常见的统计指标： `派生指标=时间周期+修饰词+原子指标` 。原子指标的创建需要在业务过程定义后才可以创建。派生指标的创建一般需要在了解具体报表需求之后展开，在新建派生指标前必须新建好原子指标。 注意事项如下：

- 原子指标、修饰类型及修饰词，直接归属在业务过程下，其中修饰词继承修饰类型的数据域。
- 派生指标可以选择多个修饰词，由具体的派生指标语义决定。例如，支付金额为原子指标，则最近一个月商品A的支付金额（时间周期为*最近一个月*，修饰词为*商品A*，原子指标为*支付金额*）为派生指标。
- 派生指标唯一归属一个原子指标，继承原子指标的数据域，与修饰词的数据域无关。

根据业务需求确定指标

本教程中，用户是电商营销部门的营销数据分析师。数据需求为最近一天厨具类目的商品在各省的销售总额、该类目Top10销售额商品名称、各省用户购买力分布（人均消费额）等，用于营销分析。

根据之前的分析，我们确认业务过程为：确认收货（交易成功），而度量为商品的销售金额。因此根据业务需求，我们可以定义出原子指标：商品成功交易金额。

派生指标为：

- 最近一天全省厨具类目各商品销售总额
- 最近一天全省厨具类目人均消费额（消费总额除以人数）

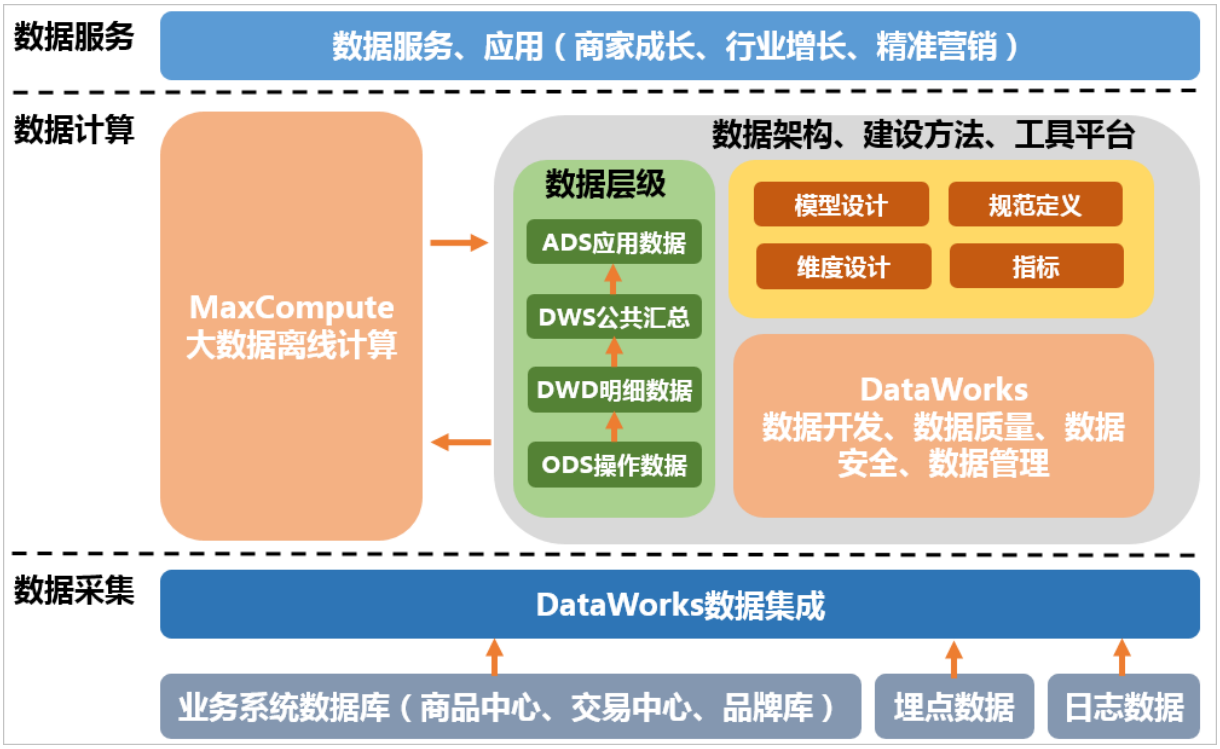
最近一天全省厨具类目各商品销售总额进行降序排序后取前10名的名称，即可得到该类目Top10销售额商品名称。

3.3. 架构与模型设计

3.3.1. 技术架构选型

在数据模型设计之前，您需要首先完成技术架构的选型。本教程中使用阿里云大数据产品MaxCompute配合DataWorks，完成整体的数据建模和研发流程。

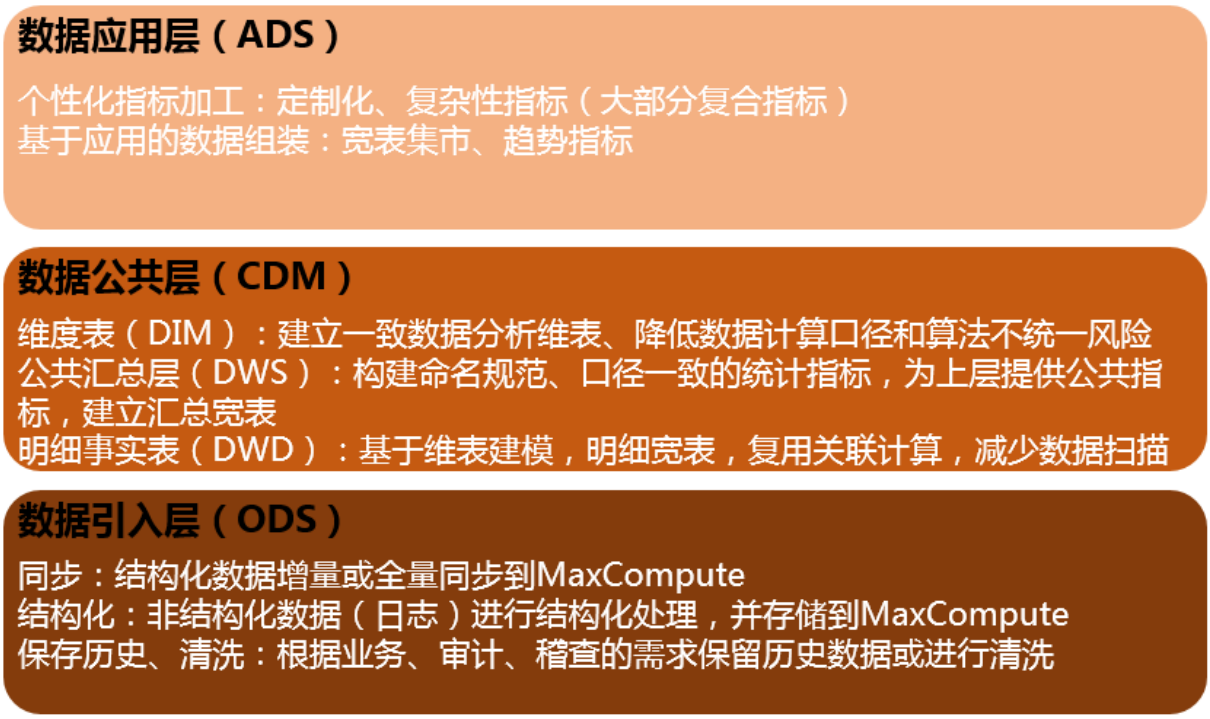
完整的技术架构图如下图所示。其中，DataWorks的数据集成负责完成数据的采集和基本的ETL。MaxCompute作为整个大数据开发过程中的离线计算引擎。DataWorks则包括数据开发、数据质量、数据安全、数据管理等在内的一系列功能。



3.3.2. 数仓分层

在阿里巴巴的数据体系中，我们建议将数据仓库分为三层，自下而上为：数据引入层（ODS，Operation Data Store）、数据公共层（CDM，Common Data Model）和数据应用层（ADS，Application Data Service）。

数据仓库的分层和各层级用途如下图所示。



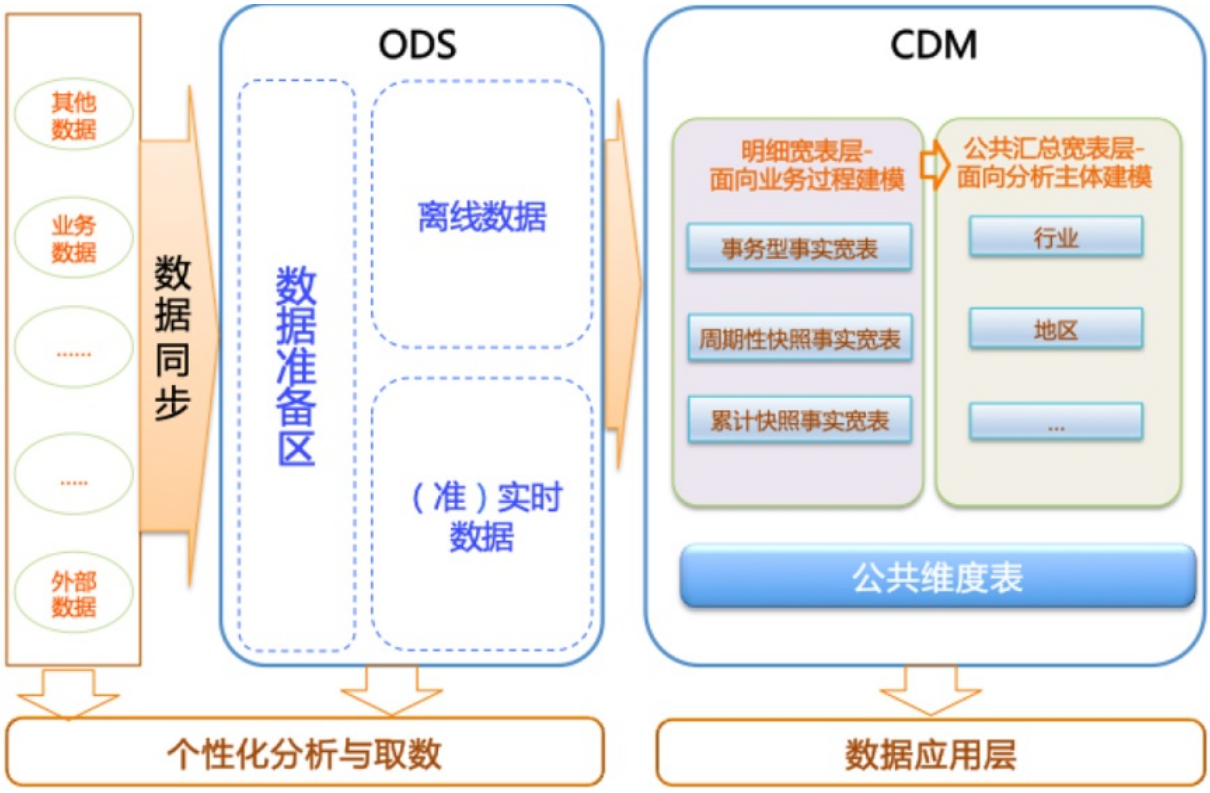
- 数据引入层ODS（Operation Data Store）：存放未经过处理的原始数据至数据仓库系统，结构上与源系统保持一致，是数据仓库的数据准备区。主要完成基础数据引入到MaxCompute的职责，同时记录基础数据的历史变化。
- 数据公共层CDM（Common Data Model，又称通用数据模型层），包括DIM维度表、DWD和DWS，由ODS层数据加工而成。主要完成数据加工与整合，建立一致性的维度，构建可复用的面向分析和统计的明细事实表，以及汇总公共粒度的指标。
  - 公共维度层（DIM）：基于维度建模理念思想，建立整个企业的一致性维度。降低数据计算口径和算法不统一风险。

    公共维度层的表通常也被称为逻辑维度表，维度和维度逻辑表通常一一对应。
  - 公共汇总粒度事实层（DWS）：以分析的主题对象作为建模驱动，基于上层的应用和产品的指标需求，构建公共粒度的汇总指标事实表，以宽表化手段物理化模型。构建命名规范、口径一致的统计指标，为上层提供公共指标，建立汇总宽表、明细事实表。

    公共汇总粒度事实层的表通常也被称为汇总逻辑表，用于存放派生指标数据。
  - 明细粒度事实层（DWD）：以业务过程作为建模驱动，基于每个具体的业务过程特点，构建最细粒度的明细层事实表。可以结合企业的数据使用特点，将明细事实表的某些重要维度属性字段做适当冗余，即宽表化处理。

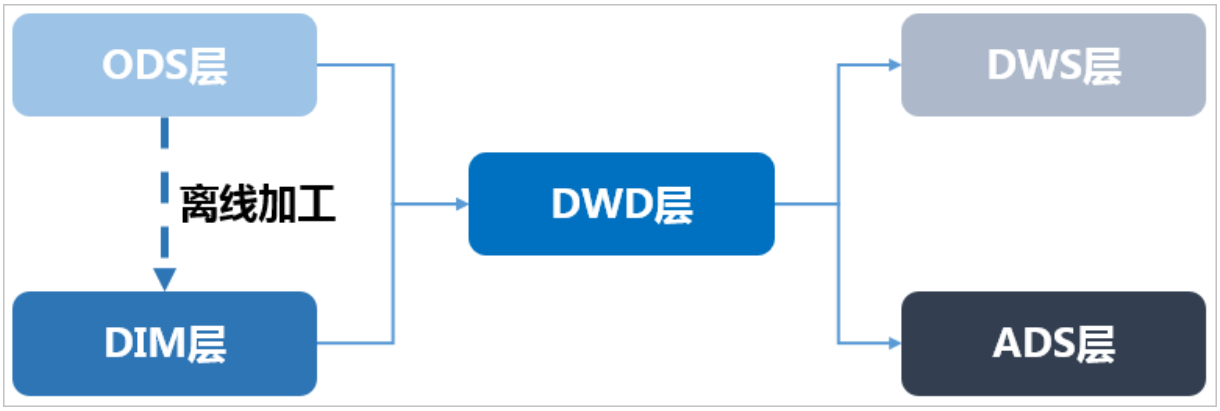
    明细粒度事实层的表通常也被称为逻辑事实表。
- 数据应用层ADS（Application Data Service）：存放数据产品个性化的统计指标数据。根据CDM与ODS层加工生成。

该数据分类架构在ODS层分为三部分：数据准备区、离线数据和准实时数据区。整体数据分类架构如下图所示。



在本教程中，从交易数据系统的数据经过DataWorks数据集成，同步到数据仓库的ODS层。经过数据开发形成事实宽表后，再以商品、地域等为维度进行公共汇总。

整体的数据流向如下图所示。其中，ODS层到DIM层的ETL（萃取（Extract）、转置（Transform）及加载（Load））处理是在MaxCompute中进行的，处理完成后会同步到所有存储系统。ODS层和DWD层会放在数据中间件中，供下游订阅使用。而DWS层和ADS层的数据通常会落地到在线存储系统中，下游通过接口调用的形式使用。



3.3.3. 数据模型

3.3.3.1. 数据引入层（ODS）

ODS（Operational Data Store）层存放您从业务系统获取的最原始的数据，是其他上层数据的源数据。业务数据系统中的数据通常为非常细节的数据，经过长时间累积，且访问频率很高，是面向应用的数据。

> **说明** 在构建MaxCompute数据仓库的表之前，您需要首先了解MaxCompute支持的数据类型版本说明。

数据引入层表设计

本教程中，在ODS层主要包括的数据有：交易系统订单详情、用户信息详情、商品详情等。这些数据未经处理，是最原始的数据。逻辑上，这些数据都是以二维表的形式存储。虽然严格的说ODS层不属于数仓建模的范畴，但是合理的规划ODS层并做好数据同步也非常重要。本教程中，使用了6张ODS表：

- 记录用于拍卖的商品信息：s_auction。
- 记录用于正常售卖的商品信息：s_sale。
- 记录用户详细信息：s_users_extra。
- 记录新增的商品成交订单信息：s_biz_order_delta。
- 记录新增的物流订单信息：s_logistics_order_delta。
- 记录新增的支付订单信息：s_pay_order_delta。

> **说明**
>
> - 表或字段命名尽量和业务系统保持一致，但是需要通过额外的标识来区分增量和全量表。例如，我们通过_delta来标识该表为增量表。
> - 命名时需要特别注意冲突处理，例如不同业务系统的表可能是同一个名称。为区分两个不同的表，您可以将这两个同名表的来源数据库名称作为后缀或前缀。例如，表中某些字段的名称刚好和关键字重名了，可以通过添加_col1后缀解决。

ODS层设计规范

ODS层表命名、数据同步任务命名、数据产出及生命周期管理及数据质量规范请参见ODS层设计规范。

建表示例

为方便您使用，集中提供建表语句如下。更多建表信息，请参见表操作。

```
CREATE TABLE IF NOT EXISTS s_auction
(
    id                              STRING COMMENT '商品ID',
    title                           STRING COMMENT '商品名',
    gmt_modified                    STRING COMMENT '商品最后修改日期',
    price                           DOUBLE COMMENT '商品成交价格，单位元',
    starts                          STRING COMMENT '商品上架时间',
    minimum_bid                     DOUBLE COMMENT '拍卖商品起拍价，单位元',
    duration                        STRING COMMENT '有效期，销售周期，单位天',
    incrementnum                    DOUBLE COMMENT '拍卖价格的增价幅度',
    city                            STRING COMMENT '商品所在城市',
    prov                            STRING COMMENT '商品所在省份',
    ends                            STRING COMMENT '销售结束时间',
    quantity                        BIGINT COMMENT '数量',
    stuff_status                    BIGINT COMMENT '商品新旧程度 0 全新 1 闲置 2 二手',
    auction_status                  BIGINT COMMENT '商品状态 0 正常 1 用户删除 2 下架 3 从未上架
',
    cate_id                          BIGINT COMMENT '商品类目ID',
    cate_name                         STRING COMMENT '商品类目名称',
```

```
    commodity_id                    BIGINT COMMENT '品类ID',
    commodity_name                   STRING COMMENT '品类名称',
    umid                             STRING COMMENT '买家umid'
)
COMMENT '商品拍卖ODS'
PARTITIONED BY (ds         STRING COMMENT '格式：YYYYMMDD')
LIFECYCLE 400;
CREATE TABLE IF NOT EXISTS s_sale
(
    id                              STRING COMMENT '商品ID',
    title                           STRING COMMENT '商品名',
    gmt_modified                    STRING COMMENT '商品最后修改日期',
    starts                          STRING COMMENT '商品上架时间',
    price                           DOUBLE COMMENT '商品价格，单位元',
    city                            STRING COMMENT '商品所在城市',
    prov                            STRING COMMENT '商品所在省份',
    quantity                        BIGINT COMMENT '数量',
    stuff_status                    BIGINT COMMENT '商品新旧程度 0 全新 1 闲置 2 二手',
    auction_status                  BIGINT COMMENT '商品状态 0 正常 1 用户删除 2 下架 3 从未上架
',
    cate_id                        BIGINT COMMENT '商品类目ID',
    cate_name                      STRING COMMENT '商品类目名称',
    commodity_id                   BIGINT COMMENT '品类ID',
    commodity_name                  STRING COMMENT '品类名称',
    umid                            STRING COMMENT '买家umid'
)
COMMENT '商品正常购买ODS'
PARTITIONED BY (ds      STRING COMMENT '格式：YYYYMMDD')
LIFECYCLE 400;
CREATE TABLE IF NOT EXISTS s_users_extra
(
    id                STRING COMMENT '用户ID',
    logincount        BIGINT COMMENT '登录次数',
    buyer_goodnum     BIGINT COMMENT '作为买家的好评数',
    seller_goodnum    BIGINT COMMENT '作为卖家的好评数',
    level_type        BIGINT COMMENT '1 一级店铺 2 二级店铺 3 三级店铺',
    promoted_num      BIGINT COMMENT '1 A级服务  2 B级服务  3 C级服务',
    gmt_create        STRING COMMENT '创建时间',
    order_id          BIGINT COMMENT '订单ID',
    buyer_id          BIGINT COMMENT '买家ID',
    buyer_nick        STRING COMMENT '买家昵称',
    buyer_star_id     BIGINT COMMENT '买家星级 ID',
    seller_id         BIGINT COMMENT '卖家ID',
    seller_nick       STRING COMMENT '卖家昵称',
    seller_star_id    BIGINT COMMENT '卖家星级ID',
    shop_id           BIGINT COMMENT '店铺ID',
    shop_name         STRING COMMENT '店铺名称'
)
COMMENT '用户扩展表'
PARTITIONED BY (ds       STRING COMMENT 'yyyymmdd')
LIFECYCLE 400;
CREATE TABLE IF NOT EXISTS s_biz_order_delta
(
    biz_order_id         STRING COMMENT '订单ID',
    pay_order_id         STRING COMMENT '支付订单ID'
```

```
    pay_order_id          STRING COMMENT '支付订单ID',
    logistics_order_id    STRING COMMENT '物流订单ID',
    buyer_nick            STRING COMMENT '买家昵称',
    buyer_id              STRING COMMENT '买家ID',
    seller_nick           STRING COMMENT '卖家昵称',
    seller_id             STRING COMMENT '卖家ID',
    auction_id            STRING COMMENT '商品ID',
    auction_title         STRING COMMENT '商品标题 ',
    auction_price         DOUBLE COMMENT '商品价格',
    buy_amount            BIGINT COMMENT '购买数量',
    buy_fee               BIGINT COMMENT '购买金额',
    pay_status            BIGINT COMMENT '支付状态 1 未付款  2 已付款 3 已退款',
    logistics_id          BIGINT COMMENT '物流订单ID',
    mord_cod_status       BIGINT COMMENT '物流状态 0 初始状态 1 接单成功 2 接单超时3 揽收成功 4揽
收失败 5 签收成功 6 签收失败 7 用户取消物流订单',
    status                BIGINT COMMENT '状态 0 订单正常 1 订单不可见',
    sub_biz_type          BIGINT COMMENT '业务类型 1 拍卖 2 购买',
    end_time              STRING COMMENT '交易结束时间',
    shop_id               BIGINT COMMENT '店铺ID'
)
COMMENT '交易成功订单日增量表'
PARTITIONED BY (ds       STRING COMMENT 'yyyymmdd')
LIFECYCLE 7200;
CREATE TABLE IF NOT EXISTS s_logistics_order_delta
(
    logistics_order_id STRING COMMENT '物流订单ID ',
    post_fee           DOUBLE COMMENT '物流费用',
    address            STRING COMMENT '收货地址',
    full_name          STRING COMMENT '收货人全名',
    mobile_phone       STRING COMMENT '移动电话',
    prov               STRING COMMENT '省份',
    prov_code          STRING COMMENT '省份ID',
    city               STRING COMMENT '市',
    city_code          STRING COMMENT '城市ID',
    logistics_status   BIGINT COMMENT '物流状态
1 - 未发货
2 - 已发货
3 - 已收货
4 - 已退货
5 - 配货中',
    consign_time       STRING COMMENT '发货时间',
    gmt_create         STRING COMMENT '订单创建时间',
    shipping           BIGINT COMMENT '发货方式
1，平邮
2，快递
3，EMS',
    seller_id          STRING COMMENT '卖家ID',
    buyer_id           STRING COMMENT '买家ID'
)
COMMENT '交易物流订单日增量表'
PARTITIONED BY (ds                 STRING COMMENT '日期')
LIFECYCLE 7200;
CREATE TABLE IF NOT EXISTS s_pay_order_delta
(
    pay_order_id     STRING COMMENT '支付订单ID',
```

```
    total_fee        DOUBLE COMMENT '应支付总金额 （数量*单价）',
    seller_id STRING COMMENT '卖家ID',
    buyer_id  STRING COMMENT '买家ID',
    pay_status       BIGINT COMMENT '支付状态
1等待买家付款,
2等待卖家发货,
3交易成功',
    pay_time         STRING COMMENT '付款时间',
    gmt_create       STRING COMMENT '订单创建时间',
    refund_fee       DOUBLE COMMENT '退款金额（包含运费）',
    confirm_paid_fee DOUBLE COMMENT '已经确认收货的金额'
)
COMMENT '交易支付订单增量表'
PARTITIONED BY (ds        STRING COMMENT '日期')
LIFECYCLE 7200;
```

数据引入层存储

为了满足历史数据分析需求，您可以在ODS层表中添加时间维度作为分区字段。实际应用中，您可以选择采用增量、全量存储或拉链存储的方式。

- 增量存储

  以天为单位的增量存储，以业务日期作为分区，每个分区存放日增量的业务数据。举例如下：

  - 1月1日，用户A访问了A公司电商店铺B，A公司电商日志产生一条记录t1。1月2日，用户A又访问了A公司电商店铺C，A公司电商日志产生一条记录t2。采用增量存储方式，t1将存储在1月1日这个分区中，t2将存储在1月2日这个分区中。
  - 1月1日，用户A在A公司电商网购买了B商品，交易日志将生成一条记录t1。1月2日，用户A又将B商品退货了，交易日志将更新t1记录。采用增量存储方式，初始购买的t1记录将存储在1月1日这个分区中，更新后的t1将存储在1月2日这个分区中。

  > **说明** 交易、日志等事务性较强的ODS表适合增量存储方式。这类表数据量较大，采用全量存储的方式存储成本压力大。此外，这类表的下游应用对于历史全量数据访问的需求较小（此类需求可通过数据仓库后续汇总后得到）。例如，日志类ODS表没有数据更新的业务过程，因此所有增量分区UNION在一起就是一份全量数据。

- 全量存储

  以天为单位的全量存储，以业务日期作为分区，每个分区存放截止到业务日期为止的全量业务数据。例如，1月1日，卖家A在A公司电商网发布了B、C两个商品，前端商品表将生成两条记录t1、t2。1月2日，卖家A将B商品下架了，同时又发布了商品D，前端商品表将更新记录t1，同时新生成记录t3。采用全量存储方式，在1月1日这个分区中存储t1和t2两条记录，在1月2日这个分区中存储更新后的t1以及t2、t3记录。

  > **说明** 对于小数据量的缓慢变化维度数据，例如商品类目，可直接使用全量存储。

- 拉链存储

  拉链存储通过新增两个时间戳字段（start_dt和end_dt），将所有以天为粒度的变更数据都记录下来，通常分区字段也是这两个时间戳字段。

  拉链存储举例如下。

商品	start_dt	end_dt	卖家	状态
B	20160101	20160102	A	上架
C	20160101	30001231	A	上架
B	20160102	30001231	A	下架

这样，下游应用可以通过限制时间戳字段来获取历史数据。例如，用户访问1月1日数据，只需限制 `start_dt<=20160101` 并且 `end_dt>20160101` 。

缓慢变化维度

MaxCompute不推荐使用代理键，推荐使用自然键作为维度主键，主要原因有两点：

1. MaxCompute是分布式计算引擎，生成全局唯一的代理键工作量非常大。当遇到大数据量情况下，这项工作就会更加复杂，且没有必要。
2. 使用代理键会增加ETL的复杂性，从而增加ETL任务的开发和维护成本。

在不使用代理键的情况下，缓慢变化维度可以通过快照方式处理。

快照方式下数据的计算周期通常为每天一次。基于该周期，处理维度变化的方式为每天一份全量快照。

例如商品维度，每天保留一份全量商品快照数据。任意一天的事实表均可以取到当天的商品信息，也可以取到最新的商品信息，通过限定日期，采用自然键进行关联即可。该方式的优势主要有以下两点：

- 处理缓慢变化维度的方式简单有效，开发和维护成本低。
- 使用方便，易于理解。数据使用方只需要限定日期即可取到当天的快照数据。任意一天的事实快照与任意一天的维度快照通过维度的自然键进行关联即可。

该方法的弊端主要是存储空间的极大浪费。例如某维度每天的变化量占总体数据量比例很低，极端情况下，每天无变化，这种情况下存储浪费严重。该方法主要实现了通过牺牲存储获取ETL效率的优化和逻辑上的简化。请避免过度使用该方法，且必须要有对应的数据生命周期制度，清除无用的历史数据。

数据同步加载与处理

ODS的数据需要由各数据源系统同步到MaxCompute，才能用于进一步的数据开发。本教程建议您使用DataWorks数据集成功能完成数据同步，详情请参见数据集成概述。在使用数据集成的过程中，建议您遵循以下规范：

- 一个系统的源表只允许同步到MaxCompute一次，保持表结构的一致性。
- 数据集成提供数据同步解决方案，您可以通过配置同步规则，实现离线数据全量及增量同步、增量数据实时写入、增量数据和全量数据定时自动合并写入新的全量表分区。详情请参见同步解决方案。
- ODS层的表建议以统计日期及时间分区表的方式存储，便于管理数据的存储成本和策略控制。

3.3.3.2. 公共维度汇总层（DIM）

公共维度汇总层（DIM）基于维度建模理念，建立整个企业的一致性维度。

公共维度汇总层（DIM）主要由维度表（维表）构成。维度是逻辑概念，是衡量和观察业务的角度。维表是根据维度及其属性将数据平台上构建的物理化的表，采用宽表设计的原则。因此，公共维度汇总层（DIM）首先需要定义维度。

定义维度

在划分数据域、构建总线矩阵时，需要结合对业务过程的分析定义维度。本教程以A电商公司的营销业务板块为例，在交易数据域中，我们重点考察确认收货（交易成功）的业务过程。

在确认收货的业务过程中，主要有商品和收货地点（本教程中，假设收货和购买是同一个地点）两个维度所依赖的业务角度。从商品角度可以定义出以下维度：

- 商品ID
- 商品名称
- 商品价格
- 商品新旧程度

  0表示全新，1表示闲置，2表示二手。
- 商品类目ID
- 商品类目名称
- 品类ID
- 品类名称
- 商品状态

  0表示正常，1表示用户删除，2表示下架，3表示从未上架。
- 商品所在城市
- 商品所在省份

从地域角度，可以定义出以下维度：

- 城市code
- 城市名称
- 省份code
- 省份名称

作为维度建模的核心，在企业级数据仓库中必须保证维度的唯一性。以A公司的商品维度为例，有且只允许有一种维度定义。例如，省份code这个维度，对于任何业务过程所传达的信息都是一致的。

设计维表

完成维度定义后，您可以对维度进行补充，进而生成维表。维表的设计需要注意：

- 建议维表单表信息不超过1000万条。
- 维表与其他表进行Join时，建议您使用Map Join。
- 避免过于频繁的更新维表的数据。

在设计维表时，您需要从下列方面进行考虑：

- 维表中数据的稳定性。

  例如，A公司电商会员通常不会出现消亡，但会员数据可能在任何时候更新，此时要考虑创建单个分区存储全量数据。如果存在不会更新的记录，您可能需要分别创建历史表与日常表。日常表用于存放当前有效的记录，保持表的数据量不会膨胀；历史表根据消亡时间插入对应分区，使用单个分区存放分区对应时间的消亡记录。
- 维表是否需要垂直拆分。

  如果一个维表存在大量属性不被使用，或由于承载过多属性字段导致查询变慢，则需要考虑对字段进行拆分，创建多个维表。
- 维表是否需要水平拆分。

如果记录之间有明显的界限，可以考虑拆成多个表或设计成多级分区。

- 核心维表的产出时间。通常有严格的要求。

设计维表的主要步骤如下：

1. 初步定义维度。

   保证维度的一致性。

2. 确定主维表（中心事实表，本教程中采用星型模型）。

   此处的主维表通常是数据引入层（ODS）表，直接与业务系统同步。例如，s_auction是与前台商品中心系统同步的商品表，此表即是主维表。

3. 确定相关维表。

   数据仓库是业务源系统的数据整合，不同业务系统或者同一业务系统中的表之间存在关联性。根据对业务的梳理，确定哪些表和主维表存在关联关系，并选择其中的某些表用于生成维度属性。以商品维度为例，根据对业务逻辑的梳理，可以得到商品与类目、卖家和店铺等维度存在关联关系。

4. 确定维度属性。

   主要包括两个阶段。第一个阶段是从主维表中选择维度属性或生成新的维度属性；第二个阶段是从相关维表中选择维度属性或生成新的维度属性。以商品维度为例，从主维表（s_auction）、类目、卖家和店铺等相关维表中选择维度属性或生成新的维度属性。维度属性的设计需要注意：

   - 尽可能生成丰富的维度属性。
   - 尽可能多地给出富有意义的文字性描述。
   - 区分数值型属性和事实。
   - 尽量沉淀出通用的维度属性。

公共维度汇总层（DIM）维表规范

公共维度汇总层（DIM）维表命名规范：dim_{业务板块名称/pub}_{维度定义}[_{自定义命名标签}]，pub是与具体业务板块无关或各个业务板块都可公用的维度。例如，时间维度，举例如下：

- 公共区域维表dim_pub_area
- A公司电商板块的商品全量表dim_asale_itm

建表示例

本例中，最终的维表建表语句如下所示。

```
CREATE TABLE IF NOT EXISTS dim_asale_itm
(
    item_id                          BIGINT COMMENT '商品ID',
    item_title                      STRING COMMENT '商品名称',
    item_price                      DOUBLE COMMENT '商品成交价格_元',
    item_stuff_status               BIGINT COMMENT '商品新旧程度_0全新1闲置2二手',
    cate_id                          BIGINT COMMENT '商品类目ID',
    cate_name                        STRING COMMENT '商品类目名称',
    commodity_id                     BIGINT COMMENT '品类ID',
    commodity_name                  STRING COMMENT '品类名称',
    item_status                     BIGINT COMMENT '商品状态_0正常1用户删除2下架3未上架',
    city                            STRING COMMENT '商品所在城市',
    prov                            STRING COMMENT '商品所在省份'
)
COMMENT '商品全量表'
PARTITIONED BY (ds        STRING COMMENT '日期,yyyymmdd');
CREATE TABLE IF NOT EXISTS dim_pub_area
(
    city_code      STRING COMMENT '城市code',
    city_name      STRING COMMENT '城市名称',
    prov_code      STRING COMMENT '省份code',
    prov_name      STRING COMMENT '省份名称'
)
COMMENT '公共区域维表'
PARTITIONED BY (ds             STRING COMMENT '日期分区,格式yyyymmdd')
LIFECYCLE 3600;
```

3.3.3.3. 明细粒度事实层（DWD）

明细粒度事实层DWD（Data Warehouse Detail）以业务过程驱动建模，基于每个具体的业务过程特点，构建最细粒度的明细层事实表。您可以结合企业的数据使用特点，将明细事实表的某些重要维度属性字段做适当冗余，即宽表化处理。

公共汇总粒度事实层（DWS）和明细粒度事实层（DWD）的事实表作为数据仓库维度建模的核心，需紧绕业务过程来设计。通过获取描述业务过程的度量来描述业务过程，包括引用的维度和与业务过程有关的度量。度量通常为数值型数据，作为事实逻辑表的依据。事实逻辑表的描述信息是事实属性，事实属性中的外键字段通过对应维度进行关联。

事实表中一条记录所表达的业务细节程度被称为粒度。通常粒度可以通过两种方式来表述：一种是维度属性组合所表示的细节程度，一种是所表示的具体业务含义。

作为度量业务过程的事实，通常为整型或浮点型的十进制数值，有可加性、半可加性和不可加性三种类型：

- 可加性事实是指可以按照与事实表关联的任意维度进行汇总。
- 半可加性事实只能按照特定维度汇总，不能对所有维度汇总。例如库存可以按照地点和商品进行汇总，而按时间维度把一年中每个月的库存累加则毫无意义。
- 完全不可加性，例如比率型事实。对于不可加性的事实，可分解为可加的组件来实现聚集。

事实表相对维表通常更加细长，行增加速度也更快。维度属性可以存储到事实表中，这种存储到事实表中的维度列称为维度退化，可加快查询速度。与其他存储在维表中的维度一样，维度退化可以用来进行事实表的过滤查询、实现聚合操作等。

明细粒度事实层（DWD）通常分为三种：事务事实表、周期快照事实表和累积快照事实表，详情请参见数仓建设指南。

- 事务事实表用来描述业务过程，跟踪空间或时间上某点的度量事件，保存的是最原子的数据，也称为原子事实表。
- 周期快照事实表以具有规律性的、可预见的时间间隔记录事实。
- 累积快照事实表用来表述过程开始和结束之间的关键步骤事件，覆盖过程的整个生命周期，通常具有多个日期字段来记录关键时间点。当累积快照事实表随着生命周期不断变化时，记录也会随着过程的变化而被修改。

明细粒度事实表设计原则

明细粒度事实表设计原则如下所示：

- 通常，一个明细粒度事实表仅和一个维度关联。
- 尽可能包含所有与业务过程相关的事实 。
- 只选择与业务过程相关的事实。
- 分解不可加性事实为可加的组件。
- 在选择维度和事实之前必须先声明粒度。
- 在同一个事实表中不能有多种不同粒度的事实。
- 事实的单位要保持一致。
- 谨慎处理Null值。
- 使用退化维度提高事实表的易用性。

明细粒度事实表整体设计流程如下图所示。

在一致性度量中已定义好了交易业务过程及其度量。明细事实表注意针对业务过程进行模型设计。明细事实表的设计可以分为四个步骤：选择业务过程、确定粒度、选择维度、确定事实（度量）。粒度主要是在维度未展开的情况下记录业务活动的语义描述。在您建设明细事实表时，需要选择基于现有的表进行明细层数据的开发，清楚所建表记录存储的是什么粒度的数据。

明细粒度事实层（DWD）规范

通常您需要遵照的命名规范为：dwd_{业务板块/pub}_{数据域缩写}_{业务过程缩写}[_{自定义表命名标签缩写}]_{单分区增量全量标识}，pub表示数据包括多个业务板块的数据。单分区增量全量标识通常为：i表示增量，f表示全量。例如： dwd_asale_trd_ordcrt_trip_di（A电商公司航旅机票订单下单事实表，日刷新增量）及dwd_asale_itm_item_df（A电商商品快照事实表，日刷新全量）。

本教程中，DWD层主要由三个表构成：

- 交易商品信息事实表：dwd_asale_trd_itm_di。
- 交易会员信息事实表：dwd_asale_trd_mbr_di。
- 交易订单信息事实表：dwd_asale_trd_ord_di。

DWD层数据存储及生命周期管理规范请参见CDM明细层设计规范。

建表示例

本教程中充分使用了维度退化以提升查询效率，建表语句如下所示。

```
CREATE TABLE IF NOT EXISTS dwd_asale_trd_itm_di
(
    item_id             BIGINT COMMENT '商品ID',
    item_title          STRING COMMENT '商品名称',
```

```
    item_price            DOUBLE COMMENT '商品价格',
    item_stuff_status     BIGINT COMMENT '商品新旧程度_0全新1闲置2二手',
    item_prov             STRING COMMENT '商品省份',
    item_city             STRING COMMENT '商品城市',
    cate_id               BIGINT COMMENT '商品类目ID',
    cate_name             STRING COMMENT '商品类目名称',
    commodity_id          BIGINT COMMENT '品类ID',
    commodity_name        STRING COMMENT '品类名称',
    buyer_id              BIGINT COMMENT '买家ID'
)
COMMENT '交易商品信息事实表'
PARTITIONED BY (ds     STRING COMMENT '日期')
LIFECYCLE 400;
CREATE TABLE IF NOT EXISTS dwd_asale_trd_mbr_di
(
    order_id          BIGINT COMMENT '订单ID',
    bc_type           STRING COMMENT '业务分类',
    buyer_id          BIGINT COMMENT '买家ID',
    buyer_nick        STRING COMMENT '买家昵称',
    buyer_star_id     BIGINT COMMENT '买家星级ID',
    seller_id         BIGINT COMMENT '卖家ID',
    seller_nick       STRING COMMENT '卖家昵称',
    seller_star_id    BIGINT COMMENT '卖家星级ID',
    shop_id           BIGINT COMMENT '店铺ID',
    shop_name         STRING COMMENT '店铺名称'
)
COMMENT '交易会员信息事实表'
PARTITIONED BY (ds     STRING COMMENT '日期')
LIFECYCLE 400;
CREATE TABLE IF NOT EXISTS dwd_asale_trd_ord_di
(
    order_id              BIGINT COMMENT '订单ID',
    pay_order_id          BIGINT COMMENT '支付订单ID',
    pay_status            BIGINT COMMENT '支付状态_1未付款2已付款3已退款',
    succ_time             STRING COMMENT '订单交易结束时间',
    item_id               BIGINT COMMENT '商品ID',
    item_quantity         BIGINT COMMENT '购买数量',
    confirm_paid_amt      DOUBLE COMMENT '订单已经确认收货的金额',
    logistics_id          BIGINT COMMENT '物流订单ID',
    mord_prov             STRING COMMENT '收货人省份',
    mord_city             STRING COMMENT '收货人城市',
    mord_lgt_shipping     BIGINT COMMENT '发货方式_1平邮2快递3EMS',
    mord_address          STRING COMMENT '收货人地址',
    mord_mobile_phone     STRING COMMENT '收货人手机号',
    mord_fullname         STRING COMMENT '收货人姓名',
    buyer_nick            STRING COMMENT '买家昵称',
    buyer_id              BIGINT COMMENT '买家ID'
)
COMMENT '交易订单信息事实表'
PARTITIONED BY (ds       STRING COMMENT '日期')
LIFECYCLE 400;
```

3.3.3.4. 公共汇总粒度事实层（DWS）

公共汇总粒度事实层以分析的主题对象作为建模驱动，基于上层的应用和产品的指标需求构建公共粒度的汇总指标事实表。公共汇总层的一个表通常会对应一个派生指标。

公共汇总事实表设计原则

聚集是指针对原始明细粒度的数据进行汇总。DWS公共汇总层是面向分析对象的主题聚集建模。在本教程中，最终的分析目标为：最近一天某个类目（例如：厨具）商品在各省的销售总额、该类目Top10销售额商品名称、各省用户购买力分布。因此，我们可以以最终交易成功的商品、类目、买家等角度对最近一天的数据进行汇总。

> 注意
>
> - 聚集是不跨越事实的。聚集是针对原始星形模型进行的汇总。为获取和查询与原始模型一致的结果，聚集的维度和度量必须与原始模型保持一致，因此聚集是不跨越事实的。
> - 聚集会带来查询性能的提升，但聚集也会增加ETL维护的难度。当子类目对应的一级类目发生变更时，先前存在的、已经被汇总到聚集表中的数据需要被重新调整。

此外，进行DWS层设计时还需遵循以下原则：

- 数据公用性：需考虑汇总的聚集是否可以提供给第三方使用。您可以判断，基于某个维度的聚集是否经常用于数据分析中。如果答案是肯定的，就有必要把明细数据经过汇总沉淀到聚集表中。
- 不跨数据域。数据域是在较高层次上对数据进行分类聚集的抽象。数据域通常以业务过程进行分类，例如交易统一划到交易域下， 商品的新增、修改放到商品域下。
- 区分统计周期。在表的命名上要能说明数据的统计周期，例如_1d表示最近1天，td表示截至当天，nd表示最近N天。

公共汇总事实表规范

公共汇总事实表命名规范：dws_{业务板块缩写/pub}_{数据域缩写}_{数据粒度缩写}[_{自定义表命名标签缩写}]_{统计时间周期范围缩写}。

- 关于统计实际周期范围缩写，缺省情况下，离线计算应该包括最近一天（_1d），最近N天（_nd）和历史截至当天（_td）三个表。如果出现_nd的表字段过多需要拆分时，只允许以一个统计周期单元作为原子拆分。即一个统计周期拆分一个表，例如最近7天（_1w）拆分一个表。不允许拆分出来的一个表存储多个统计周期。
- 对于小时表（无论是天刷新还是小时刷新），都用_hh 来表示。
- 对于分钟表（无论是天刷新还是小时刷新），都用_mm来表示。

举例如下：

- dws_asale_trd_byr_subpay_1d（A电商公司买家粒度交易分阶段付款一日汇总事实表）
- dws_asale_trd_byr_subpay_td（A电商公司买家粒度分阶段付款截至当日汇总表）
- dws_asale_trd_byr_cod_nd（A电商公司买家粒度货到付款交易汇总事实表）
- dws_asale_itm_slr_td（A电商公司卖家粒度商品截至当日存量汇总表）
- dws_asale_itm_slr_hh（A电商公司卖家粒度商品小时汇总表）---维度为小时
- dws_asale_itm_slr_mm（A电商公司卖家粒度商品分钟汇总表）---维度为分钟

DWS层数据存储及生命周期管理规范请参见CDM汇总层设计规范。

建表示例

满足业务需求的DWS层建表语句如下。

```
CREATE TABLE IF NOT EXISTS dws_asale_trd_byr_ord_1d
(
    buyer_id                BIGINT COMMENT '买家ID',
    buyer_nick              STRING COMMENT '买家昵称',
    mord_prov               STRING COMMENT '收货人省份',
    cate_id                 BIGINT COMMENT '商品类目ID',
    cate_name               STRING COMMENT '商品类目名称',
    confirm_paid_amt_sum_1d DOUBLE COMMENT '最近一天订单已经确认收货的金额总和'
)
COMMENT '买家粒度所有交易最近一天汇总事实表'
PARTITIONED BY (ds         STRING COMMENT '分区字段YYYYMMDD')
LIFECYCLE 36000;
CREATE TABLE IF NOT EXISTS dws_asale_trd_itm_ord_1d
(
    item_id                 BIGINT COMMENT '商品ID',
    item_title               STRING COMMENT '商品名称',
    cate_id                 BIGINT COMMENT '商品类目ID',
    cate_name               STRING COMMENT '商品类目名称',
    mord_prov               STRING COMMENT '收货人省份',
    confirm_paid_amt_sum_1d DOUBLE COMMENT '最近一天订单已经确认收货的金额总和'
)
COMMENT '商品粒度交易最近一天汇总事实表'
PARTITIONED BY (ds         STRING COMMENT '分区字段YYYYMMDD')
LIFECYCLE 36000;
```

3.3.3.5. 附录：ODS层示例数据

本文为您提供ODS层各表格的示例数据，仅供您测试参考。

- s_auction.csv
- s_biz_order_delta.csv
- s_logistics_order_delta.csv
- s_pay_order_delta.csv
- s_sale.csv
- s_users_extra.csv

3.3.4. 层次调用规范

在完成数据仓库的分层后，您需要对各层次的数据之间的调用关系作出约定。

层次调用规范

ADS应用层优先调用数据仓库公共层数据。如果已经存在CDM层数据，不允许ADS应用层跨过CDM中间层从ODS层重复加工数据。CDM中间层应该积极了解应用层数据的建设需求，将公用的数据沉淀到公共层，为其他数据层次提供数据服务。同时，ADS应用层也需积极配合CDM中间层进行持续的数据公共建设的改造。避免出现过度的ODS层引用、不合理的数据复制和子集合冗余。总体遵循的层次调用原则如下：

- ODS层数据不能直接被应用层任务引用。如果中间层没有沉淀的ODS层数据，则通过CDM层的视图访问。CDM层视图必须使用调度程序进行封装，保持视图的可维护性与可管理性。

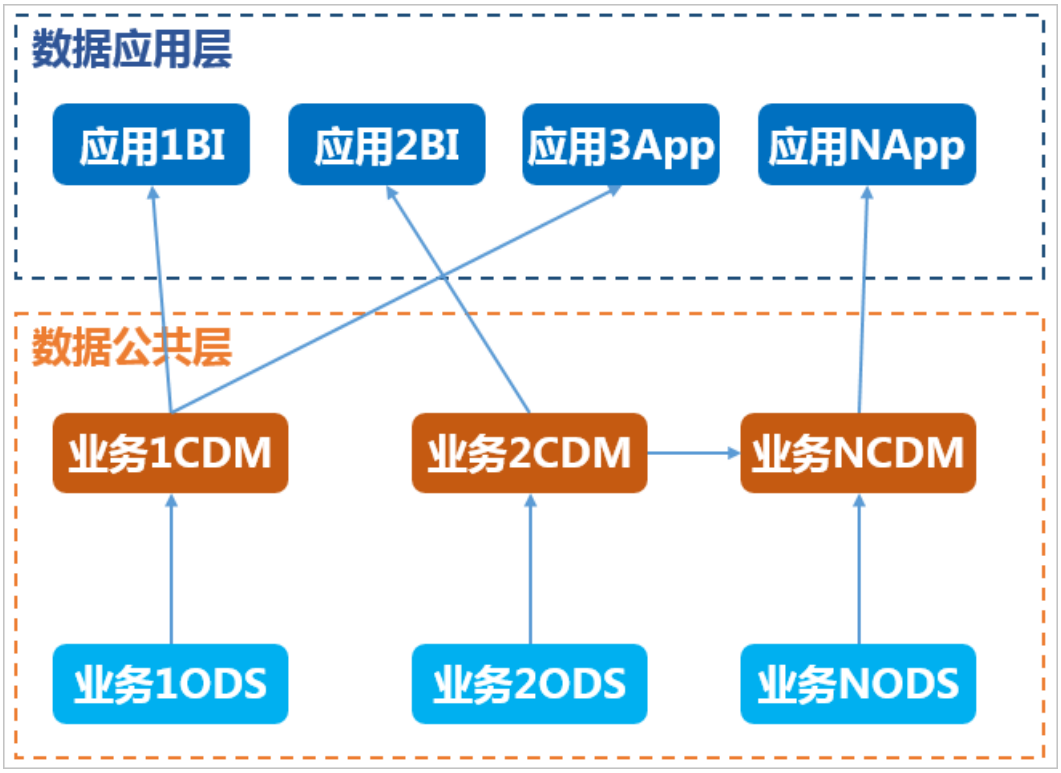
- CDM层任务的深度不宜过大（建议不超过10层）。
- 一个计算刷新任务只允许一个输出表，特殊情况除外。
- 如果多个任务刷新输出一个表（不同任务插入不同的分区），DataWorks上需要建立一个虚拟任务，依赖多个任务的刷新和输出。通常，下游应该依赖此虚拟任务。
- CDM汇总层优先调用CDM明细层，可累加指标计算。CDM汇总层尽量优先调用已经产出的粗粒度汇总层，避免大量汇总层数据直接从海量的明细数据层中计算得出。
- CDM明细层累计快照事实表优先调用CDM事务型事实表，保持数据的一致性产出。
- 有针对性地建设CDM公共汇总层，避免应用层过度引用和依赖CDM层明细数据。

3.4. 项目分配与安全

在为企业级大数据平台创建项目时，建议您对ODS层、DWD及DWS层的数据按照业务板块的粒度建立项目，对于ADS层的数据，按照应用的粒度建立项目。

项目分配

在本教程中，建议参考下图建立您的MaxCompute项目，图中的每一个方块代表一个项目。



- 对于ODS层项目，建议以 `ods` 结尾，例如 `asale_ods` 。
- 对于CDM层项目，建议以 `cdm` 结尾，例如 `asale_cdm` 。
- ADS应用层数据分为两类：
  - 数据报表、数据分析等以 `bi` 结尾，例如 `asale_bi` 。
  - 数据产品应用以 `App` 结尾，例如 `asale_app` 。

考虑到本教程仅聚焦于电商业务板块中交易成功的业务流程，您可以为ODS、CDM和ADS层分别仅建立一个项目。

项目模式选择

标准模式是指一个DataWorks项目对应两个MaxCompute项目，可设置开发和生产双环境，提升代码开发规范，并能够对表权限进行严格控制，禁止随意操作生产环境的表，保证生产表的数据安全。

当您在DataWorks建立项目时，建议您使用标准模式以保证生产环境项目安全，详情请参见简单模式和标准模式的区别。完成项目创建后，您会得到一个生产环境项目和以_dev结尾的开发环境项目。例如 `asaleods` 和 `asaleods_dev` 。

项目权限配置

您需要重点考虑为项目中的不同成员角色赋予不同的权限，例如生产任务如何保障不可随意变更、哪些成员可以进行代码编辑调试、哪些成员可以进行发布生产任务等。同时要为在数据开发过程中的资源使用赋权，并做好数据安全隔离。

关于MaxCompute数仓安全和权限配置详情，请参见安全模型。

3.5. 建立性能基准

MaxCompute性能表现优劣，主要取决您的表设计是否符合规范。为方便您衡量MaxCompute表的性能表现，建议您在优化性能之前首先建立性能基准。

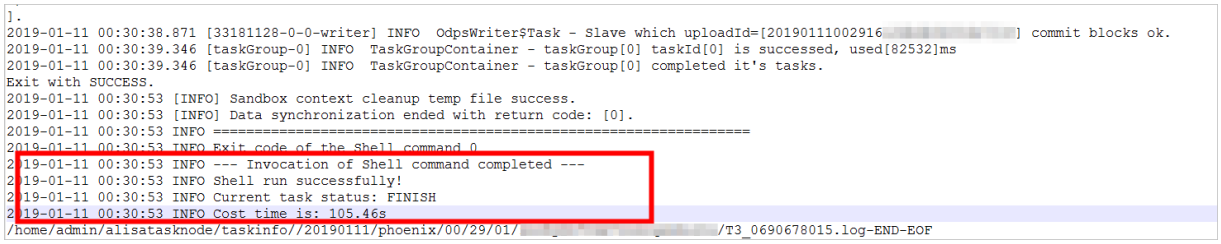
> **说明** MaxCompute表设计规范详情请参见表设计规范。

在优化表前后测试系统性能时，您需要记录每张表的数据同步时间、占用存储大小以及查询性能的详细信息。如果您使用的是包年包月方式购买的MaxCompute项目资源，还需要记录购买数。

测试项	测试值
数据同步时间	无
占用存储大小	无
查询执行时间	无
查询费用预估	无

记录数据同步时间

在您执行数据同步任务后，可以在**运维中心 > 周期实例**页面右键查看用户任务运行时间，如下图所示。



记录占用存储大小

登录DataWorks控制台。

您可以使用**describe**命令查看全表或表中某个分区占用物理存储的大小。

```
--odps sql
--********************************************************************
--author:dataphin
--create time:2019-05-13 16:08:04
--********************************************************************
DESCRIBE s_sale;
```

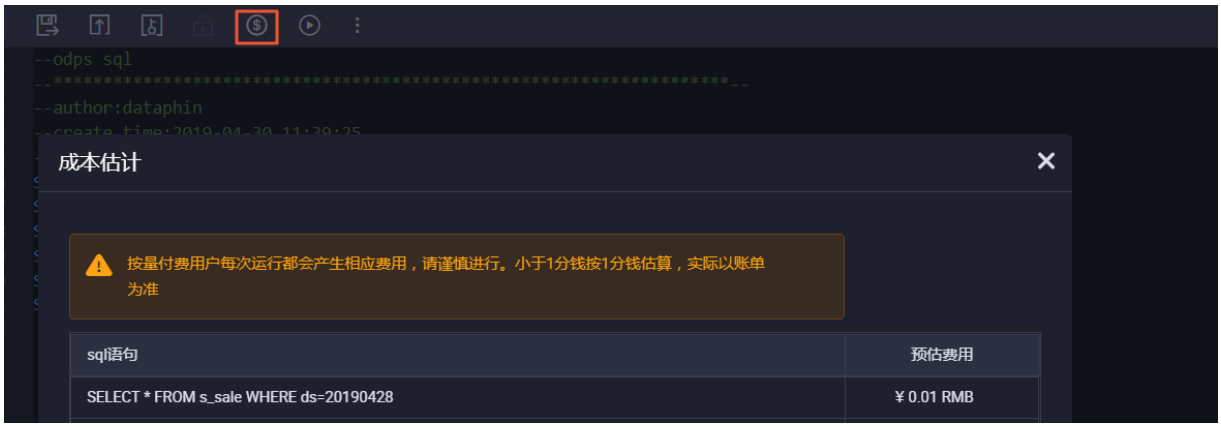
运行日志

```
+------------------------------------------------------------------------------------+
| Owner: ALIYUN$[redacted]      | Project: test_asale_dev                            |
| TableComment: 正常购买ods                                                          |
+------------------------------------------------------------------------------------+
| CreateTime:               2019-04-30 13:29:03                                      |
| LastDDLTime:              2019-04-30 13:29:03                                      |
| LastModifiedTime:         2019-04-30 19:26:46                                      |
| Lifecycle:                400                                                      |
+------------------------------------------------------------------------------------+
| InternalTable: YES        | Size: 9408                                             |
+------------------------------------------------------------------------------------+
| Native Columns:                                                                    |
+------------------------------------------------------------------------------------+
| Field           | Type          | Label | Comment                                  |
+------------------------------------------------------------------------------------+
```

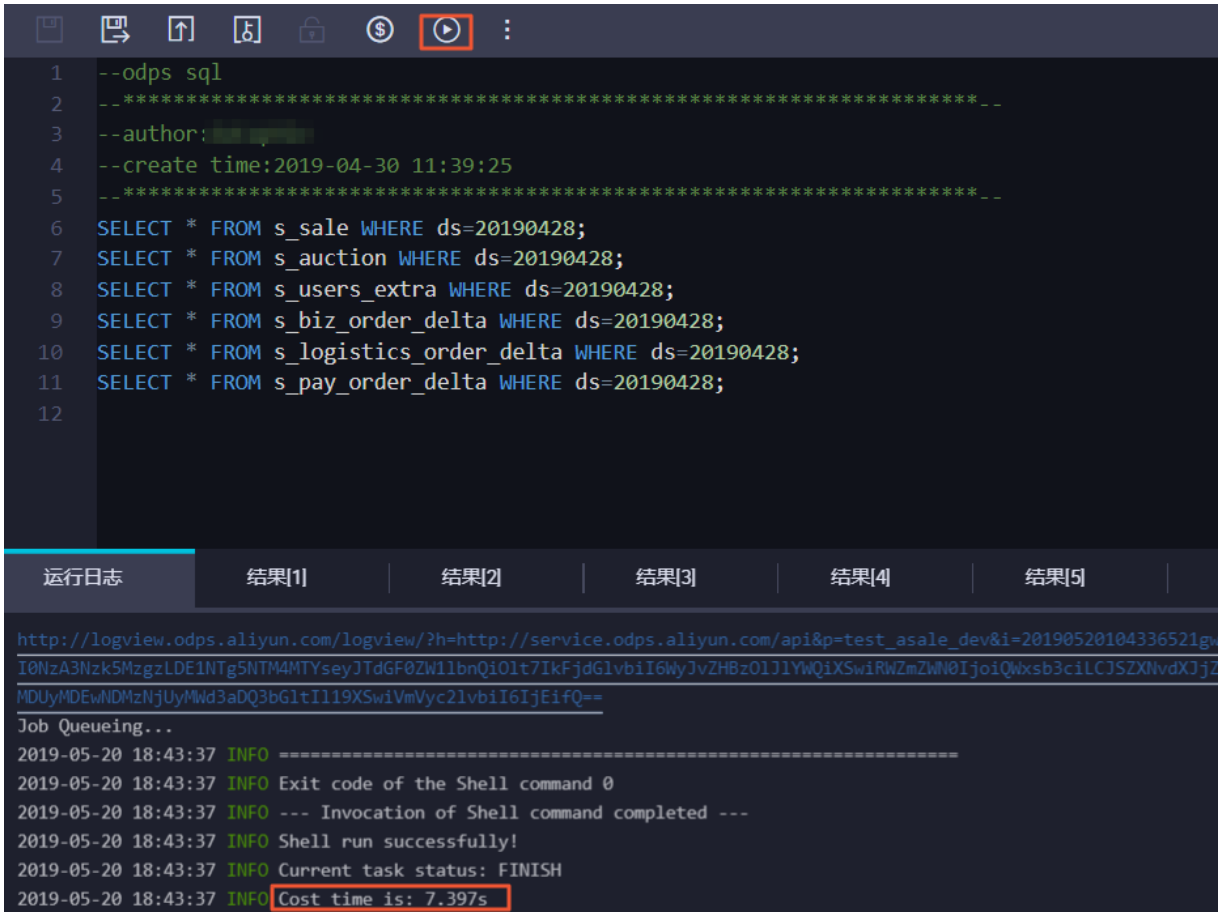
记录查询执行时间及预估费用

登录DataWorks控制台，进入**数据开发**页面，创建**ODPS sql**节点。

您可以在运行任务时或通过单击图标直接通过图形页面查看预估费用。



任务完成运行后，可在运行日志中查看到运行时间。



3.6. 数仓性能优化

针对数仓的性能优化，主要是针对表和数据分布的优化。

表设计的最佳实践请参见表设计最佳实践。

Hash Clustering

Hash Clustering表的优势在于可以实现Bucket Pruning优化、Aggregation优化以及存储优化。在创建表时，使用**clustered by**指定Hash Key后，MaxCompute将对指定列进行Hash运算，按照Hash值分散到各个Bucket里。Hash Key值的选择原则为选择重复键值少的列。Hash Clustering表的使用方法详情请参见表操作。

如何转化为Hash Clustering表：

```
ALTER TABLE table_name [CLUSTERED BY (col_name [, col_name, ...]) [SORTED BY (col_name [ASC | DESC] [, col_name [ASC | DESC] ...])] INTO number_of_buckets BUCKETS]
```

`ALTER TABLE` 语句适用于存量表，在增加了新的聚集属性之后，新的分区将做Hash Clustering存储。

创建完Hash Clustering表后，您可以使用 `INSERT OVERWRITE` 语句将源表转化为Hash Clustering表。

> **说明** Hash Clustering表存在以下限制：
> - 不支持 `INSERT INTO` 语句，只能通过 `INSERT OVERWRITE` 来添加数据。
> - 不支持直接使用**tunnel upload**命令将数据导入到range cluster表，因为tunnel上传的数据是无序的。

表的其他优化技巧

建议您严格遵循表设计规范。此外，您还可以利用下列技巧完成表的优化：

- 中间表的利用：适用于数据量非常大，下游任务很多的表。
- 拆表：适用于个别字段产出极慢的情况，您可以将字段拆分为单独的表。
- 合表：随着数仓的发展，针对业务重叠或重复的表，您可以进行任务和数据合并。
- 拉链表：合理利用拉链表能减少您的存储消耗，关于拉链存储的详情请参见拉链存储。
- 利用MaxCompute表的特殊功能：详情请参见MaxCompute表的高级功能。

3.7. 结果验证

完成数仓的优化后，您需要对结果进行评估验证，确认优化的有效性。

如果您在优化过程中改变了表结构，您需要删除原有的表，并根据优化策略新建表和分区。本教程中提供的测试数据也需要进行对应的结构调整，方便您完成数据的导入。

在重新创建表并导入数据后，您需要重新测试数仓性能。您可以通过下列表格记录相关数据，并与性能基准进行比对，性能基准详情请参见建立性能基准。

测试项	测试值
数据同步时间	无
占用存储大小	无
查询执行时间	无
查询费用预估	无

4.搭建互联网在线运营分析平台

4.1. 业务场景与开发流程

本教程基于大数据时代在线运营分析平台的基础需求，为开发者提供从数据高并发写入存储、便捷高效的数据加工处理到数据分析与展示的全链路解决方案。本教程帮助您了解并操作阿里云的大数据产品，完成在线运营分析平台的搭建。

业务场景

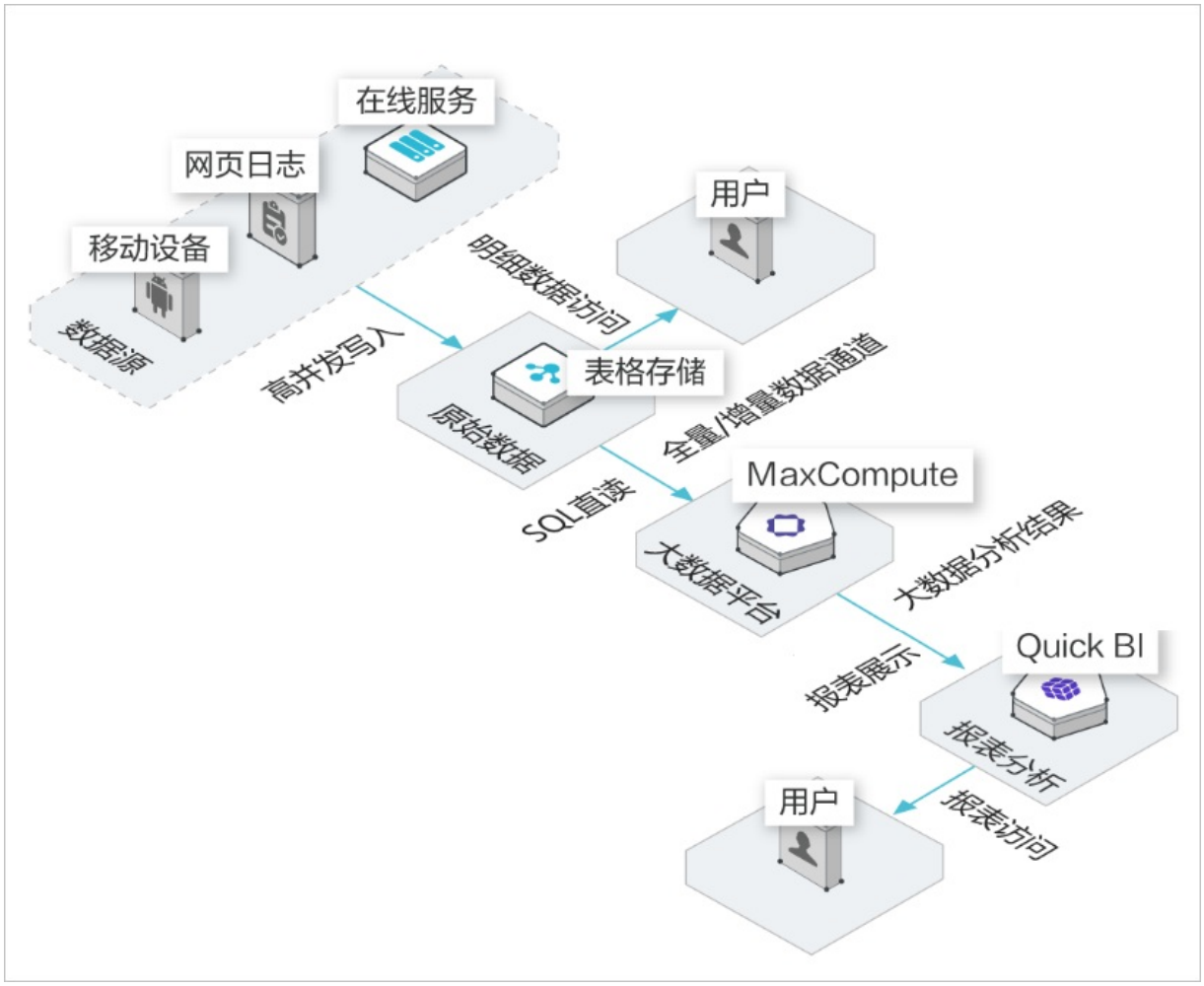
本文的示例基于真实的网站日志数据集，数据来源于某网站上的HTTP访问日志数据。基于这份网站日志，您可以实现如下分析需求：

- 统计并展现网站的PV和UV，并能够按照用户的终端类型（例如，Android、iPad、iPhone和PC等）分别统计。

  > **说明** 浏览次数（PV）和独立访客（UV）是衡量网站流量的两项最基本指标。用户每打开一个网站页面，记录一个PV，多次打开同一页面PV累计多次。独立访客（UV）是指一天内访问网站的不重复用户数，一天内同一访客多次访问网站只计算一次。

- 统计并展现网站的流量来源地域。

开发流程



本教程涉及的具体开发流程如下：

- 步骤一：环境准备。
- 步骤二：数据准备。
- 步骤三：新建数据表。
- 步骤四：设计工作流。
- 步骤五：节点配置。
- 步骤六：任务提交与测试。
- 步骤七：数据可视化展现。

整体数仓研发的规划建议请参见数据仓库研发规范概述。

4.2. 环境准备

本文为您介绍开始本教程前的环境准备工作，需要开通表格存储（TableStore）、大数据计算服务（MaxCompute）、数据工厂（DataWorks）和智能分析套件（Quick BI）。

前提条件

- 已注册阿里云账号。如果您还没有注册阿里云账号，请进入阿里云官网，单击**免费注册**，即可进入阿里云账号注册页面创建新的阿里云账号。
- 已实名认证。如果您还没有实名认证，请进入实名认证页面对账号进行实名认证。

背景信息

本教程涉及的阿里云产品如下：

- 表格存储TableStore
- 大数据计算服务MaxCompute
- 数据工场DataWorks
- 智能分析套件Quick BI

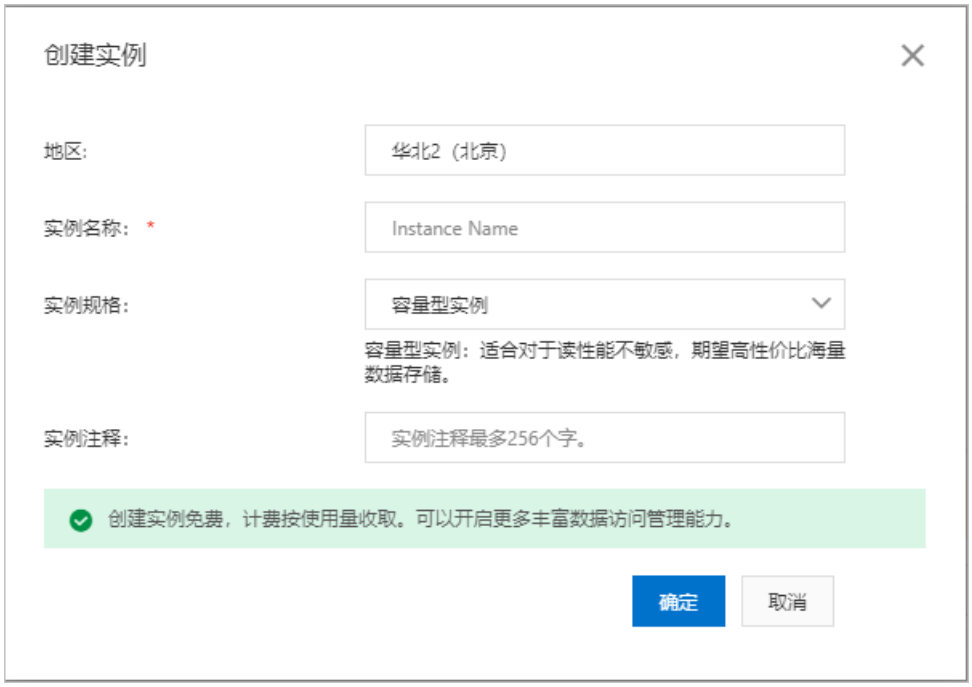
> **说明** 在本教程中，表格存储服务选择华北2（北京）。

操作步骤

1. 创建表格存储实例。
   i. 进入表格存储TableStore产品详情页，单击**立即开通**。
   ii. 在**云产品开通页**页面，勾选**我已阅读并同意表格存储（按量付费）服务协议**并单击**立即开通**。

   iii. 单击**管理控制台**。

iv. 单击**创建实例**。在**创建实例**页面，选择地区为**华北2（北京）**。填写**实例名称**，**实例规格**请选择**容量型实例**，单击**确定**。



> **说明** 实例名称在表格存储同一个区域内必须全局唯一，建议您选用自己可辨识且符合规则的名称。实例名称在MaxCompute数据处理中也会被使用，本例中为workshop-bj-001，关于实例的详细解释请参见实例。

v. 完成创建后，单击左侧导航栏**全部实例**可以看到您刚刚创建的实例，状态为**运行中**。

2. 开通大数据计算服务MaxCompute。

i. 进入MaxCompute产品详情页，单击**立即购买**。

ii. 选择**按量计费**，选择区域为**华东2（上海）**，**规格类型**为默认的**标准版**，单击**立即购买**。

> **说明** MaxCompute区域与表格存储区域相同可以节省您的流量费用，因此您可以选择区域为**华北2（北京）**。本教程中MaxCompute区域选择为**华东2（上海）**，以便为您展示跨地域的外部表使用过程。

3. 开通DataWorks。

i. 进入DataWorks产品详情页，单击**立即购买**。

ii. 选择区域为**华东2（上海）**，单击**立即购买**。

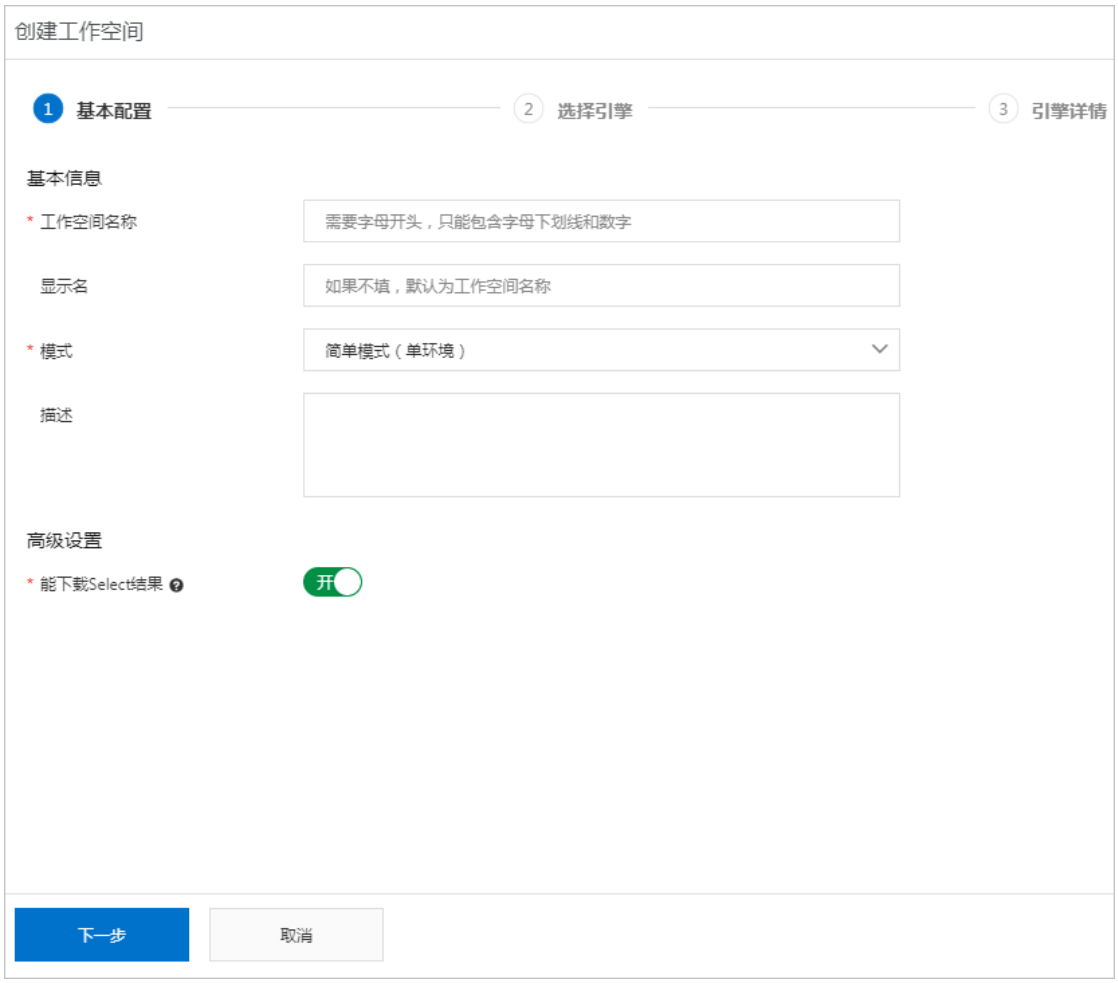
> **说明** MaxCompute区域与表格存储区域相同可以节省您的流量费用，因此您也可以选择区域为**华北2（北京）**。本教程中MaxCompute区域选择为**华东2（上海）**，以便为您展示跨地域访问数据的使用过程。

4. 创建DataWorks工作空间。

i. 进入DataWorks工作空间列表，选择区域为**华东1（杭州）**，单击**创建工作空间**。

ii. 填写**创建工作空间**对话框中的**基本配置**，单击**下一步**。

为方便使用，本教程中DataWorks工作空间模式为**简单模式（单环境）**。在简单模式下，DataWorks工作空间与MaxCompute项目一一对应，详情请参见简单模式和标准模式的区别。

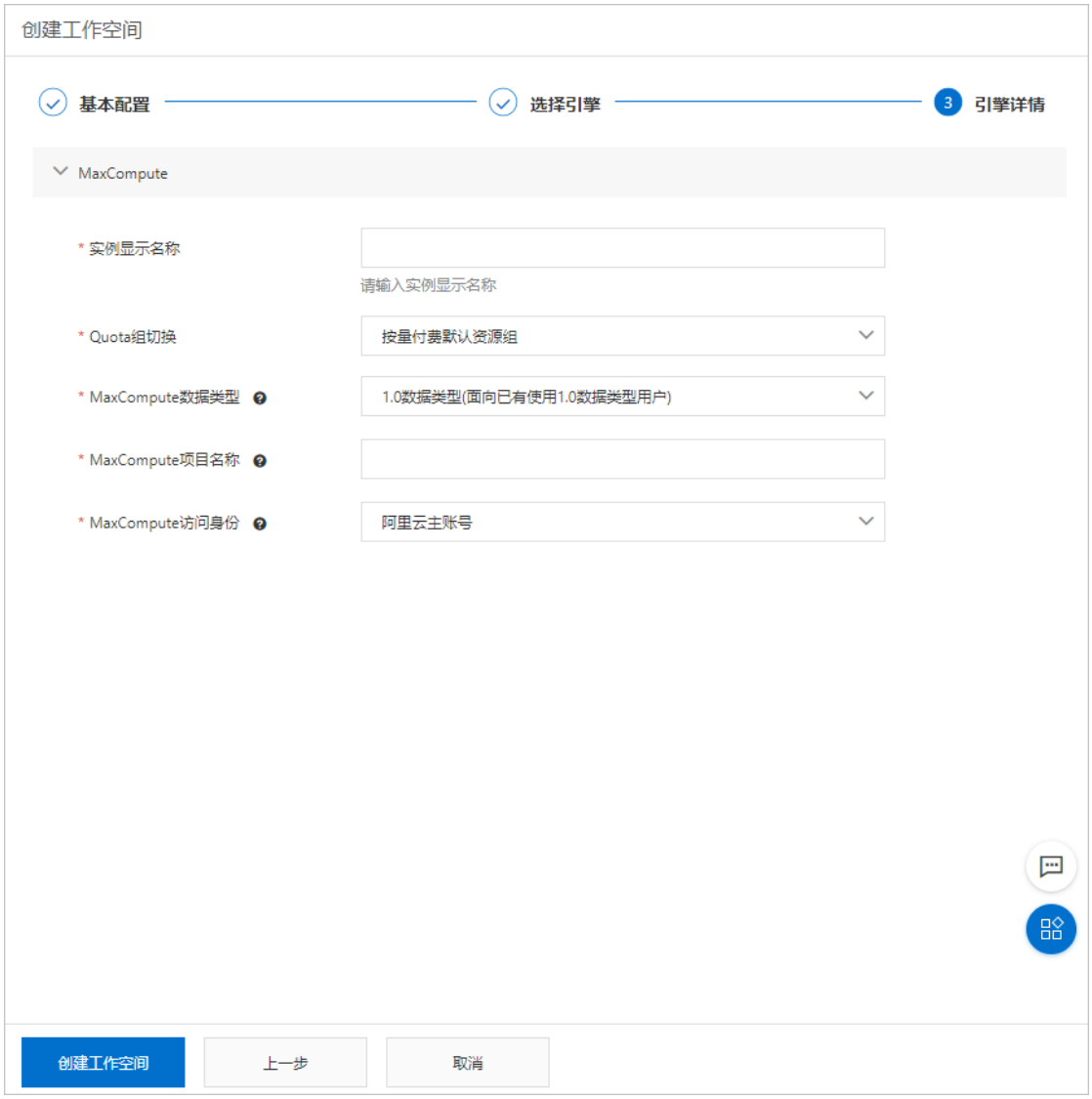


> **说明** 工作空间名称全局唯一，建议您使用易于区分的名称。

iii. 进入**选择引擎**界面，选择相应引擎后，单击**下一步**。

选择**计算引擎服务**为**MaxCompute**、**按量付费**。

iv. 进入**引擎详情**页面，填写选购引擎的配置。



分类	配置	说明
MaxCompute	**实例显示名称**	实例名称不能超过27个字符，仅支持字母开头，仅包含字母、下划线和数字。
	Quota组切换	Quota用来实现计算资源和磁盘配额。
	MaxCompute数据类型	MaxCompute项目的数据类型版本。
	MaxCompute项目名称	默认与DataWorks工作空间的名称一致。
	MaxCompute访问身份	包括**个人账号**和**工作空间所有者**，开发环境默认为**个人账号**，生产环境推荐使用**工作空间所有者**。

v. 配置完成后，单击**创建工作空间**。

工作空间创建成功后，即可在**工作空间列表**页面查看相应内容。

5. 开通Quick BI。

i. 进入Quick BI产品详情页，单击**管理控制台**

ii. 进入控制台后，单击**高级版30天试用申请**或**专业版30天试用申请**。勾选**同意Quick BI服务协议**，单击**免费试用**。

> **说明** 您可以选择使用**个人空间**或**默认空间**，推荐您使用**默认空间**。

4.3. 数据准备

在数据准备阶段，您需要通过数据Demo包生成模拟真实环境的数据，以便后续数据开发使用。

前提条件

- 创建华北2（北京）区域的表格存储实例，同时记录实例名称和实例访问地址。单击表格存储控制台中的实例名称，即可获得实例访问地址。对于跨区域的访问，建议您使用公网地址。详细操作请参见环境准备。
- 使用主账号登录安全信息管理控制台，获取并记录您的AccessKey ID和AccessKey Secret信息。

> **说明** AccessKey ID和AccessKey Secret是您访问阿里云API的密钥，具有该账户完全的权限，请您妥善保管。

操作步骤

1. 下载数据Demo包。

   数据Demo包下载地址如下，本例中使用环境为Windows7 64位：

   - Mac下载地址
   - Linux下载地址
   - Windows7 64位下载地址

2. 配置Demo环境。

   完成下载后，解压下载包，编辑conf文件夹内的app.conf文件。

   app.conf文件内容示例如下。

   ```
   endpoint = "https://workshop-bj-001.cn-beijing.ots.aliyuncs.com"
   instanceName = "workshop-bj-001"
   accessKeyId = "LTAIF24u7g******"
   accessKeySecret = "CcwFeF3sWTPy0wsKULMw34Px******"
   usercount = "200"
   daysCount = "7"
   ```

   其中，需要配置的参数如下：

   - endpoint：表格存储实例的访问网络地址，建议您使用公网地址。
   - instanceName：表格存储实例的名称。

- accessKeyId和accessKeySecret：访问阿里云的密钥。

3. 启动Demo准备测试数据。

   i. 启动Windows CMD命令行工具，进入您解压缩Demo包的路径，执行如下语句查看Demo包命令用法。

   ```
   workshop_demo.exe -h
   ```

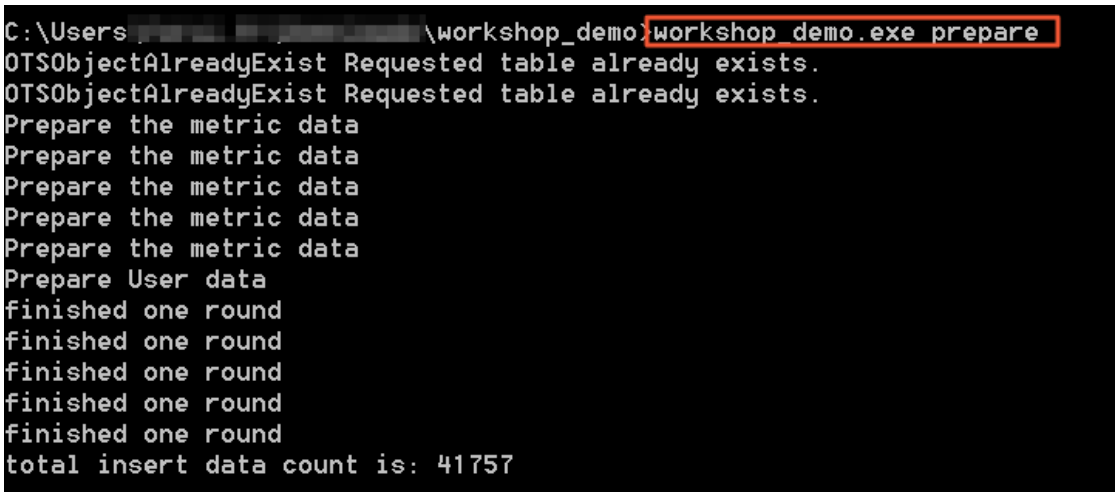
   该命令会列出该demo的相关命令，如下。

   ```
   workshop_demo.exe -h
   * prepare  准备测试数据，创建数据表，根据conf中的用户数量，为用户生成一周的行为日志数据。
   * raw ${userid} ${date} ${Top条数} 查询指定用户的日志明细。
   * new/day_active/month_active/day_pv/month_pv 在结果表中查询上述几种类型的报表数据（新增：new，日活：day_active，月活：month_active，日PV：day_pv，月PV：month_pv）。
   ```

   ii. 执行如下命令生成准备数据。

   ```
   workshop_demo.exe prepare
   ```

   结果如下。

在此过程中，Demo包会自动帮助您在表格存储中创建表，结构如下：

- 原始日志数据表：user_trace_log

列名	类型	说明
md5	STRING	用户uid的md5值undefined前8位，表格存储主键。
uid	STRING	用户uid，表格存储主键。
ts	BIGINT	用户操作时间戳，表格存储主键。
ip	STRING	IP地址。
status	BIGINT	服务器返回状态码。
bytes	BIGINT	返回给客户端的字节数。

列名	类型	说明
device	STRING	终端型号。
system	STRING	系统版本：ios xxx/android xxx。
customize_event	STRING	自定义事件：登录/退出/购买/注册/点击/后台/切换用户/浏览。
use_time	BIGINT	APP单次使用时长，当事件为退出、后台、切换用户时有该项。
customize_event_content	STRING	用户关注的内容信息。

- 分析结果表： analysis_result

列名	类型	说明
metric	STRING	报表的类型：'new'、'day_active'、'month_active'、'day_pv'、'month_pv'，表格存储主键。
ds	STRING	时间yyyy-mm-dd或yyyy-mm，表格存储主键。
num	BIGINT	对应的数据值。

4. 数据验证。
   - 用户明细查询

     通过如下语句查询指定用户在某一日期指定条数的明细数据。表格数据的日期对应于您创建表格的时间。

     ```
     raw ${userid} ${date} ${Top条数}
     ```

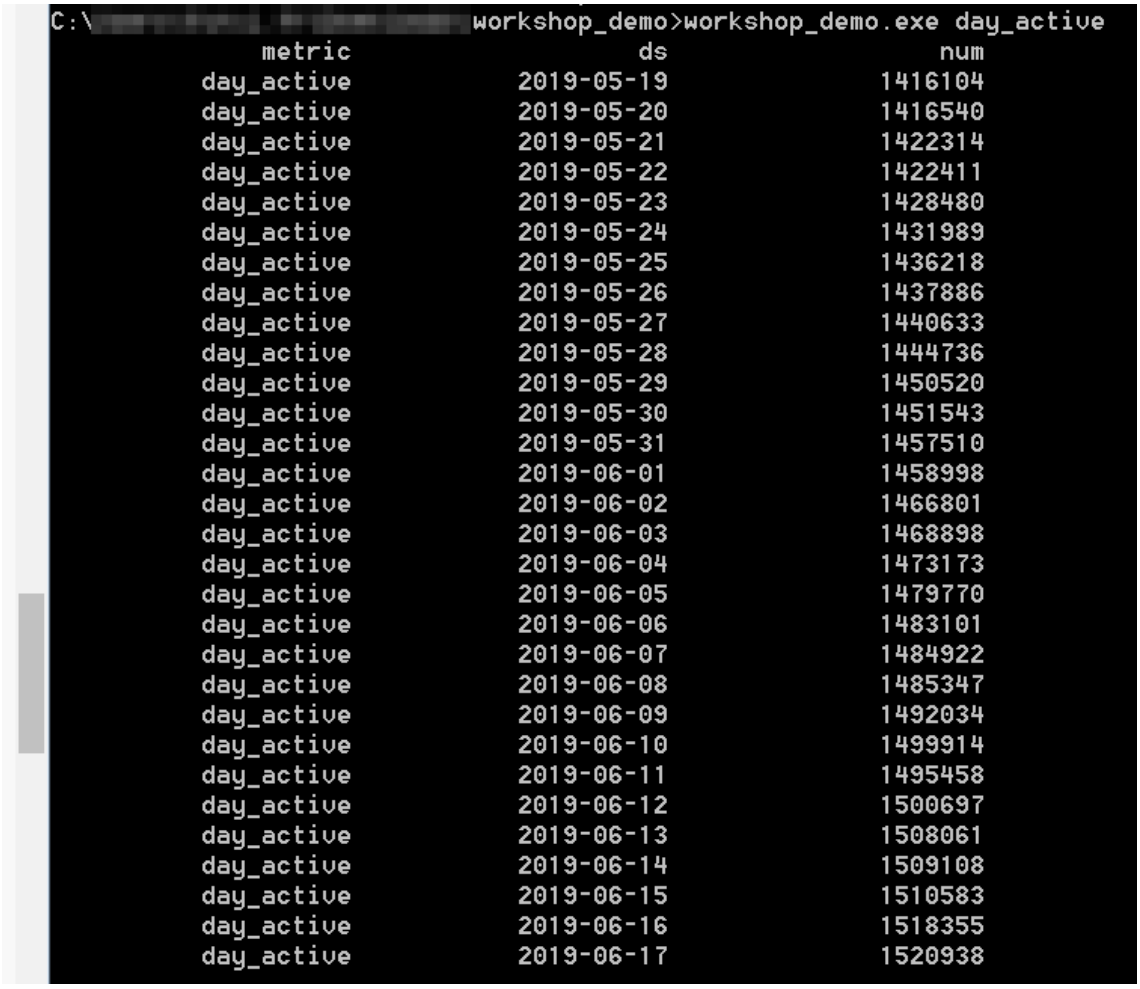
     其中，${userid}为用户ID，${date}为指定日期，${Top条数}为指定查询条数。例如，您创建数据时间为2019年6月15日，则可以使用 `workshop_demo.exe raw 00010 "2019-06-15" 20` 查看20条用户明细数据。

```
C [redacted]nloads\workshop_demo>workshop_demo.exe raw 00010 "2019-06-1
5" 20
       uid                          Date    bytes      customize_event
      device                   ip   status          system
     00010         2019-06-14 11:56:47 PM      759               regist
iPhone7 Plus        61.103.79.217      200          ios11
     00010         2019-06-14 11:26:34 PM      252     backstage    369
   iPad min2       157.249.67.241      200          ios11
     00010         2019-06-14 11:21:30 PM      427        browse travel
    iPhone6s      222.133.108.234      200          ios10
     00010         2019-06-14 11:16:03 PM      764        switch    185
iPhone7 Plus        61.103.79.217      200          ios11
     00010         2019-06-14 11:06:03 PM      436               click
iPhone7 Plus        61.103.79.217      200          ios11
     00010         2019-06-14 10:36:54 PM      131               click
iPhone7 Plus        61.103.79.217      200          ios11
     00010         2019-06-14 10:22:26 PM      778        switch     73
    iPhone6s      222.133.108.234      200          ios10
     00010         2019-06-14 10:06:29 PM      535     backstage    179
   iPad min2       157.249.67.241      200          ios11
     00010         2019-06-14 09:56:11 PM      668               click
   iPad min2       157.249.67.241      200          ios11
     00010         2019-06-14 09:20:45 PM      354               regist
    iPhone6s      222.133.108.234      200          ios10
     00010         2019-06-14 09:15:37 PM      989               click
   iPad min2       157.249.67.241      200          ios11
     00010         2019-06-14 08:51:17 PM      460        logout    462
    iPhone6s      222.133.108.234      200          ios10
     00010         2019-06-14 08:26:06 PM      887       comment  funny
   iPad min2       157.249.67.241      200          ios11
     00010         2019-06-14 08:10:34 PM      278       browse finance
    iPhone6s      222.133.108.234      200          ios10
     00010         2019-06-14 07:56:00 PM      480               click
iPhone7 Plus        61.103.79.217      200          ios11
     00010         2019-06-14 07:30:11 PM       68               click
    iPhone6s      222.133.108.234      200          ios10
     00010         2019-06-14 07:15:09 PM      398        browse   news
iPhone7 Plus        61.103.79.217      200          ios11
     00010         2019-06-14 07:11:21 PM       21               click
    iPhone6s      222.133.108.234      200          ios10
     00010         2019-06-14 06:35:07 PM      207        browse  photo
iPhone7 Plus        61.103.79.217      200          ios11
     00010         2019-06-14 06:24:43 PM      261               regist
iPhone7 Plus        61.103.79.217      200          ios11
```

说明 由于表格存储是SchemaFree结构，表的属性列不需要预先定义。Customize_Event中不同的事件对应了不同的内容，因此Demo中将事件、内容进行对齐显示。

- 报表结果查询

您可以使用 `workshop_demo.exe day_active` 命令查看日活数据。



```
C:\[redacted]workshop_demo>workshop_demo.exe day_active
          metric                  ds                 num
      day_active          2019-05-19             1416104
      day_active          2019-05-20             1416540
      day_active          2019-05-21             1422314
      day_active          2019-05-22             1422411
      day_active          2019-05-23             1428480
      day_active          2019-05-24             1431989
      day_active          2019-05-25             1436218
      day_active          2019-05-26             1437886
      day_active          2019-05-27             1440633
      day_active          2019-05-28             1444736
      day_active          2019-05-29             1450520
      day_active          2019-05-30             1451543
      day_active          2019-05-31             1457510
      day_active          2019-06-01             1458998
      day_active          2019-06-02             1466801
      day_active          2019-06-03             1468898
      day_active          2019-06-04             1473173
      day_active          2019-06-05             1479770
      day_active          2019-06-06             1483101
      day_active          2019-06-07             1484922
      day_active          2019-06-08             1485347
      day_active          2019-06-09             1492034
      day_active          2019-06-10             1499914
      day_active          2019-06-11             1495458
      day_active          2019-06-12             1500697
      day_active          2019-06-13             1508061
      day_active          2019-06-14             1509108
      day_active          2019-06-15             1510583
      day_active          2019-06-16             1518355
      day_active          2019-06-17             1520938
```

4.4. 数据建模与开发

4.4.1. 新建数据表

本文为您介绍如何在MaxCompute上建立数据表，用于承载原始数据及加工后的数据。

前提条件

- 已开通MaxCompute服务并创建DataWorks工作空间（本教程使用为简单模式工作空间），详情请参见环境准备。
- 已具备访问Tablestore数据的权限。当MaxCompute和Tablestore的所有者是同一个账号时，您可以单击此处一键授权。如果不是，您可以自定义授权，详情请参见OTS外部表。

操作步骤

1. 进入DataWorks数据开发界面。
   i. 进入DataWorks工作空间列表，选择区域为**华东2（上海）**。
   ii. 单击已创建好的工作空间后的**进入数据开发**，进入工作空间的数据开发界面。
2. 新建业务流程。

i. 右键单击**业务流程**，选择**新建业务流程**。

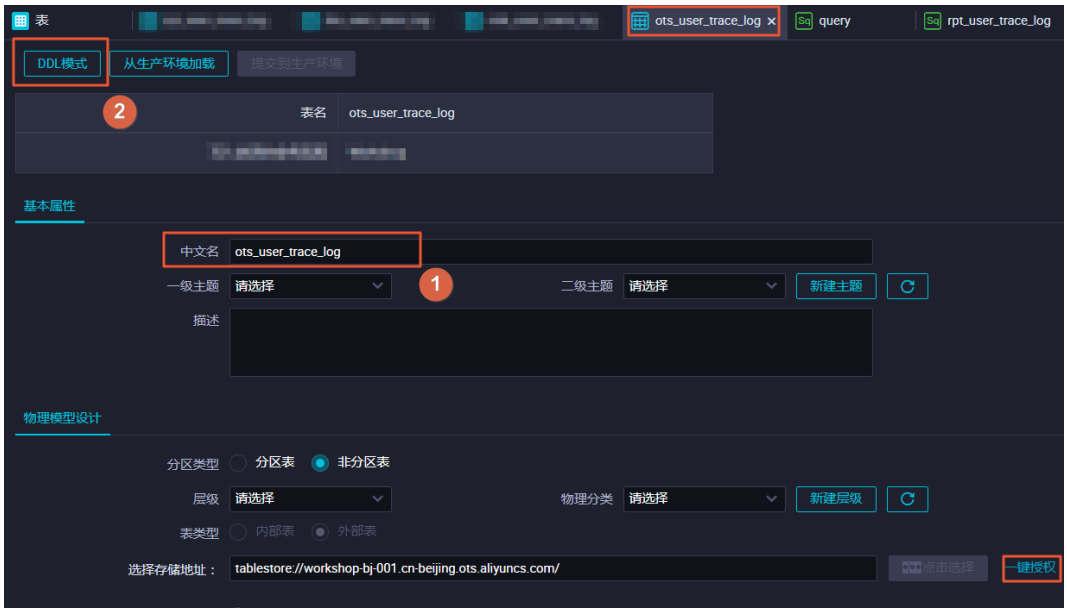
ii. 填写**业务名称**和**描述**，单击**新建**。本教程中，业务流程名为Workshop。

3. 新建数据表。

i. 创建外部表ots_user_trace_log。

a. 单击新建的业务流程Workshop，右键单击**MaxCompute**，选择**新建 > 表**，输入表名ots_user_trace_log，单击**提交**。

b. 填写创建表的**中文名**，然后单击**DDL模式**。



c. 在**DDL模式**页面，输入建表语句，单击**生成表结构**。

外部表ots_user_trace_log的建表语句如下。

```
CREATE EXTERNAL TABLE ots_user_trace_log (
    md5 string COMMENT '用户uid的md5值前8位',
    uid string COMMENT '用户uid',
    ts bigint COMMENT '用户操作时间戳',
    ip string COMMENT 'ip地址',
    status bigint COMMENT '服务器返回状态码',
    bytes bigint COMMENT '返回给客户端的字节数',
    device string COMMENT '终端型号',
    system string COMMENT '系统版本ios xxx/android xxx',
    customize_event string COMMENT '自定义事件：登录/退出/购买/注册/点击/后台/切换用户/浏览/评论',
    use_time bigint COMMENT 'APP单次使用时长，当事件为退出、后台、切换用户时有该项',
    customize_event_content string COMMENT '用户关注内容信息，在customize_event为浏览和评论时，包含该列'
)
STORED BY 'com.aliyun.odps.TableStoreStorageHandler'
WITH SERDEPROPERTIES (
    'tablestore.columns.mapping'=':md5,:uid,:ts, ip,status,bytes,device,system,customize_event,use_time,customize_event_content',
    'tablestore.table.name'='user_trace_log'
)
LOCATION 'tablestore://workshop-bj-001.cn-beijing.ots.aliyuncs.com/';
```

- STORED BY：必选参数，值为 `com.aliyun.odps.TableStoreStorageHandler` ，是MaxCompute内置处理Tablestore数据的StorageHandler，定义了MaxCompute和Tablestore的交互。

- SERDEPROPERITES：必选参数，是提供参数选项的接口，在使用TableStoreStorageHandler时，以下选项必须指定：
  - tablestore.columns.mapping：用于描述MaxCompute将访问的Tablestore表的列，包括主键和属性列。

    > **说明**
    > - 以冒号（：）开头的参数值为Tablestore主键，例如示例中的`:md5`和`:uid`，其它参数值均为属性列。
    > - 在指定映射时，您必须提供指定Tablestore表的所有主键，只需提供需要通过MaxCompute访问的属性列。提供的属性列必须是Tablestore表的列，否则即使外表可以创建成功，查询时也会报错。

  - tablestore.table.name：需要访问的Tablestore表名。如果指定的Tablestore表名错误（不存在），则会报错，MaxCompute不会主动创建Tablestore表。
- LOCATION：用来指定Tablestore的访问地址。请您根据环境准备，将自己的表格存储实例访问地址参数填写在此。

  > **说明** 如果您使用公网地址LOCATION 'tablestore://workshop-bj-001.cn-beijing.ots.aliyuncs.com/'报错，显示网络不同，可尝试更换为经典网地址LOCATION 'tablestore://workshop-bj-001.cn-beijing.ots-internal.aliyuncs.com/'。

d. 单击**提交到生产环境**，完成表的创建。

> **说明** 如果您使用的是标准模式工作空间，请先单击**提交到开发环境**，然后单击**提交到生产环境**。

ii. 创建ods_user_trace_log表。

建表方法同上，建表语句如下。ods_user_trace_log为ODS层表，相关数仓模型定义请参见数据引入层（ODS）。

```
CREATE TABLE IF NOT EXISTS ods_user_trace_log (
    md5 STRING COMMENT '用户uid的md5值前8位',
    uid STRING COMMENT '用户uid',
    ts BIGINT COMMENT '用户操作时间戳',
    ip STRING COMMENT 'ip地址',
    status BIGINT COMMENT '服务器返回状态码',
    bytes BIGINT COMMENT '返回给客户端的字节数',
    device STRING COMMENT '终端型号',
    system STRING COMMENT '系统版本ios xxx/android xxx',
    customize_event STRING COMMENT '自定义事件：登录/退出/购买/注册/点击/后台/切换用户/浏览/评论',
    use_time BIGINT COMMENT 'APP单次使用时长，当事件为退出、后台、切换用户时有该项',
    customize_event_content STRING COMMENT '用户关注内容信息，在customize_event为浏览和评论时，包含该列'
)
PARTITIONED BY (
    dt STRING
);
```

iii. 创建dw_user_trace_log表。

建表方法同上，建表语句如下。dw_user_trace_log为DW层表，相关数仓模型定义请参见明细粒度事实层（DWD）。

```
CREATE TABLE IF NOT EXISTS dw_user_trace_log (
    uid STRING COMMENT '用户uid',
    region STRING COMMENT '地域，根据ip得到',
    device_brand string comment '设备品牌',
    device STRING COMMENT '终端型号',
    system_type STRING COMMENT '系统类型，Android、IOS、ipad、Windows_phone',
    customize_event STRING COMMENT '自定义事件：登录/退出/购买/注册/点击/后台/切换用户/浏览/评论',
    use_time BIGINT COMMENT 'APP单次使用时长，当事件为退出、后台、切换用户时有该项',
    customize_event_content STRING COMMENT '用户关注内容信息，在customize_event为浏览和评论时，包含该列'
)
PARTITIONED BY (
    dt STRING
);
```

iv. 创建rpt_user_trace_log表。

建表方法同上，建表语句如下。rpt_user_trace_log为ADS层表，相关数仓模型定义请参见数仓分层。

```
CREATE TABLE IF NOT EXISTS rpt_user_trace_log (
    country STRING COMMENT '国家',
    province STRING COMMENT '省份',
    city STRING COMMENT '城市',
    device_brand string comment '设备品牌',
    device STRING COMMENT '终端型号',
    system_type STRING COMMENT '系统类型，Android、IOS、ipad、Windows_phone',
    customize_event STRING COMMENT '自定义事件：登录/退出/购买/注册/点击/后台/切换用户/浏览/评论',
    use_time BIGINT COMMENT 'APP单次使用时长，当事件为退出、后台、切换用户时有该项',
    customize_event_content STRING COMMENT '用户关注内容信息，在customize_event为浏览和评论时，包含该列',
    pv bigint comment '浏览量',
    uv bigint comment '独立访客'
)
PARTITIONED BY (
    dt STRING
);
```

4. 验证建表结果。

i. 完成建表后，您可以在Workshop业务流程**MaxCompute > 表**下看到新建的4张表。

ii. 右键单击业务流程中**MaxCompute**下的**数据开发**，选择**新建 > ODPS SQL**。

iii. 在**新建节点**页面，输入**节点名称**，单击**提交**新建ODPS SQL节点。

iv. 在新建的ODPS SQL节点中输入如下SQL语句，单击图标。

```
DESCRIBE ots_user_trace_log;
DESCRIBE ods_user_trace_log;
DESCRIBE dw_user_trace_log;
DESCRIBE rpt_user_trace_log;
```

返回表的结构信息如下：

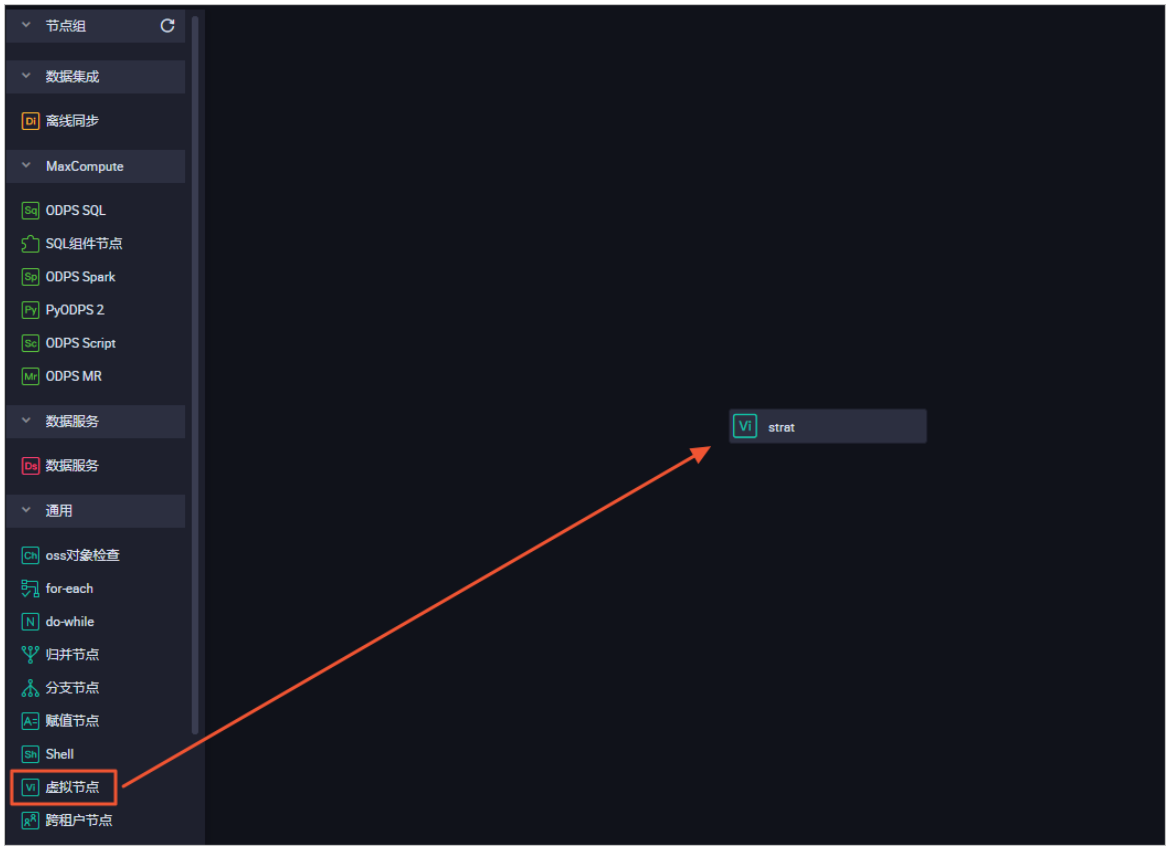
```
+------------------------------------------------------------------------------------+
| Owner: [redacted]                  | Project: [redacted]                           |
| TableComment:                                                                      |
+------------------------------------------------------------------------------------+
| CreateTime:              2020-06-16 18:56:46                                       |
| LastDDLTime:             2020-06-16 18:56:46                                       |
| LastModifiedTime:        2020-06-16 18:56:46                                       |
+------------------------------------------------------------------------------------+
| InternalTable: YES       | Size: 0                                                 |
+------------------------------------------------------------------------------------+
| Native Columns:                                                                    |
+------------------------------------------------------------------------------------+
| Field           | Type       | Label | Comment                                     |
+------------------------------------------------------------------------------------+
| country         | string     |       | 国家                                        |
| province        | string     |       | 省份                                        |
| city            | string     |       | 城市                                        |
| device_brand    | string     |       | 设备品牌                                    |
| device          | string     |       | 终端型号                                    |
| system_type     | string     |       | 系统类型，Android、IOS、ipad、Windows_phone |
| customize_event | string     |       | 自定义事件：登录/退出/购买/注册/点击/后台/切换用户/浏览 |
| use_time        | bigint     |       | APP单次使用时长，当事件为退出、后台、切换用户时有该项 |
| customize_event_content | string |       | 用户关注内容信息，在customize_event为浏览和评论时 包含该列 |
| pv              | bigint     |       | 浏览量                                      |
| uv              | bigint     |       | 独立访客                                    |
+------------------------------------------------------------------------------------+
| Partition Columns:                                                                 |
+------------------------------------------------------------------------------------+
| dt              | string     |                                                     |
+------------------------------------------------------------------------------------+
OK
2020-06-16 19:56:10 INFO ================================================================
2020-06-16 19:56:10 INFO Exit code of the Shell command 0
2020-06-16 19:56:10 INFO --- Invocation of Shell command completed ---
```

4.4.2. 设计工作流

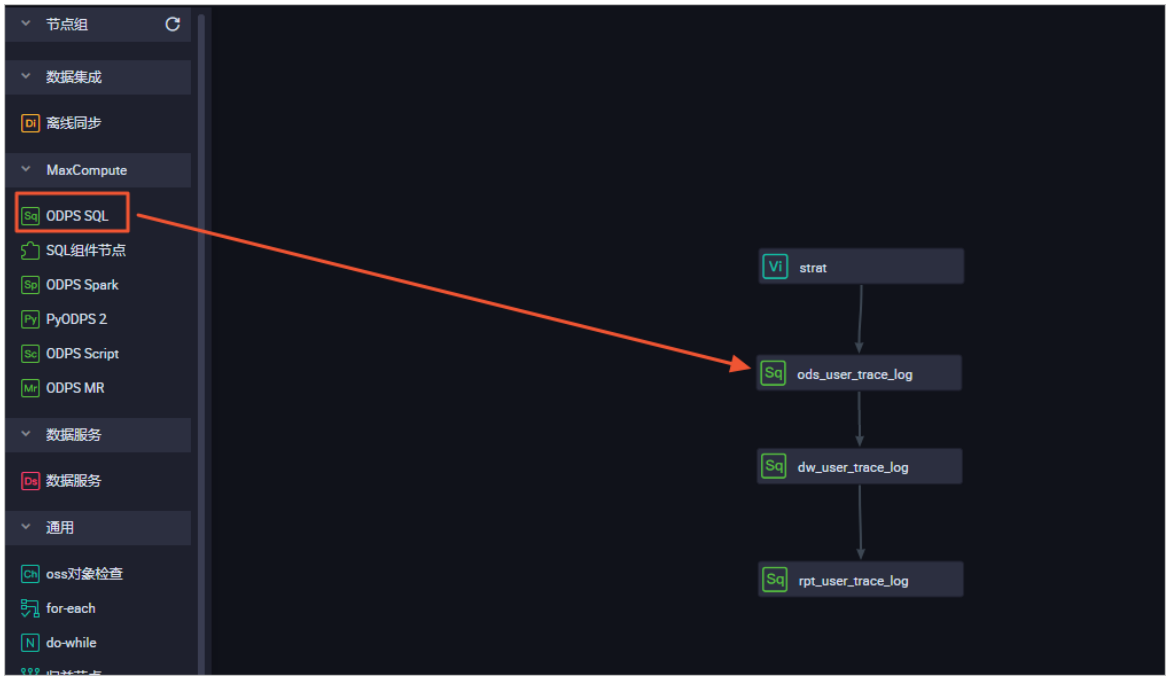
通过设计工作流，您可以明确在整体数据开发过程中各任务节点的排布。对于本教程中这种较为简单的单数据流场景，您可以选择每个数据表（数仓层次）对应一个工作流。

操作步骤

1. 双击您的业务流程，打开画布面板。
2. 向画布中拖入1个虚拟节点，命名为start。



3. 向画布中拖入3个ODPS SQL节点，依次命名为ods_user_trace_log、dw_user_trace_log、rpt_user_trace_log。通过连接不同节点，配置依赖关系如下。



> **说明** ods_user_trace_log、dw_user_trace_log、rpt_user_trace_log分别代表数据仓库的ODS、CDM和ADS层，详情请参见数仓分层。

4.4.3. 节点配置

完成工作流设计后，您需要对每个数据开发节点进行配置，填写SQL语句。

前提条件

本次数据开发过程中需要使用UDF自定义函数，您首先需要完成自定义函数的注册，详细请参见注册自定义函数。

注册自定义函数

1. 添加资源
   i. 下载用于IP地转换的自定义函数Java包getaddr.jar以及地址库ip.dat。

      关于IP地址转换的自定义函数，详情请参见MaxCompute中实现IP地址归属地转换。

   ii. 右键单击WorkShop业务流程下的**MaxCompute**，选择**新建 > 资源**。需要分别新建**File**和**JAR**类型的资源。

- File类型上传地址库ip.dat。
  a. 输入**资源名称**，选中**大文件（内容超过500KB）**及**上传为ODPS资源**，然后单击**点击上传**。

  b. 单击**提交**。

- JAR类型对应Java包getaddr.jar。
  a. 您需要勾选**上传为ODPS资源**，然后单击**点击上传**。

  b. 上传完成后，单击**提交**。

> **说明** 提交时，请忽略血缘不一致信息。

2. 注册函数
   i. 在业务流程下右键单击**MaxCompute**，选择**新建 > 函数**，将函数命名为getregion。
   ii. 在**注册函数**页面，依次填写类名为odps.test.GetAddr，资源列表为getaddr.jar,ip.dat，命令格式为getregion(ip string)，保存后单击 提交函数注册。

配置节点

1. 配置虚拟节点start。
    i. 双击start节点，进入节点配置页面。

ii. 单击右侧的**调度配置**，在**调度依赖**区域下单击**使用工作空间根节点**完成配置。

iii. 在时间属性区域选择**重跑属性**为**运行成功或失败后皆可重跑**。

iv. 单击 按钮，完成节点提交。

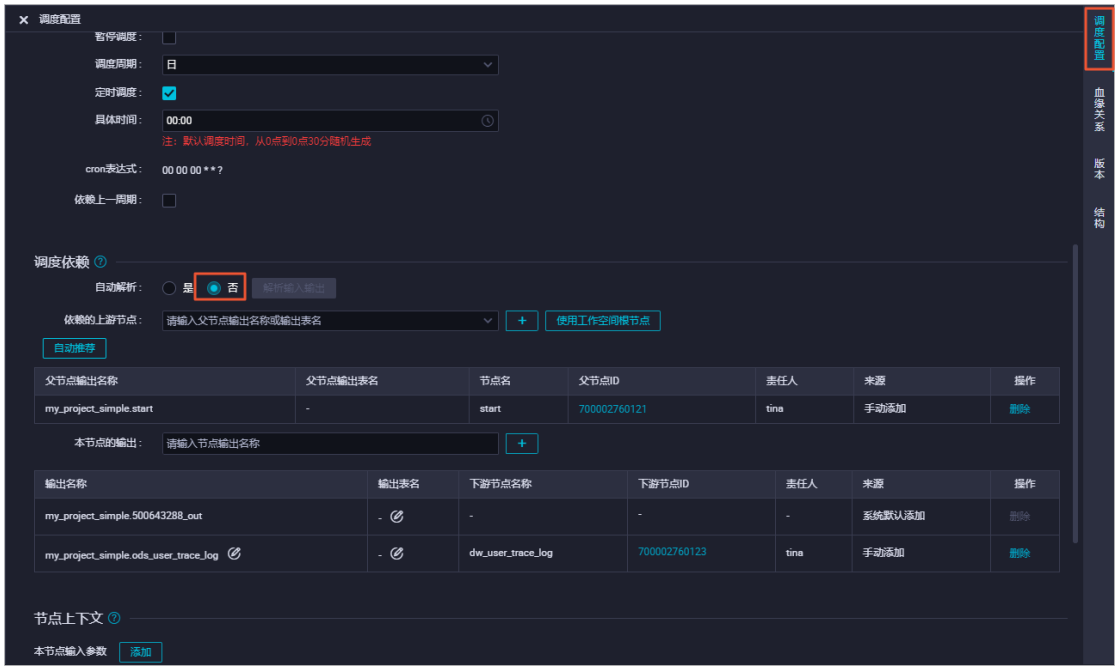
2. 配置ODPS SQL节点ods_user_trace_log

i. 双击ods_user_trace_log节点，进入节点配置界面，编写处理逻辑。SQL代码如下。

```
insert overwrite table ods_user_trace_log partition (dt=${bdp.system.bizdate})
select
    md5,
    uid ,
    ts,
    ip,
    status,
    bytes,
    device,
    system,
    customize_event,
    use_time,
    customize_event_content
    from ots_user_trace_log
    where to_char(FROM_UNIXTIME(ts),'yyyymmdd')=${bdp.system.bizdate};
```

> **说明** 关于${bdp.system.bizdate}释义请参见配置调度参数。

ii. 完成代码编写后，单击右侧的**调度配置**，选择**自动解析**为否。

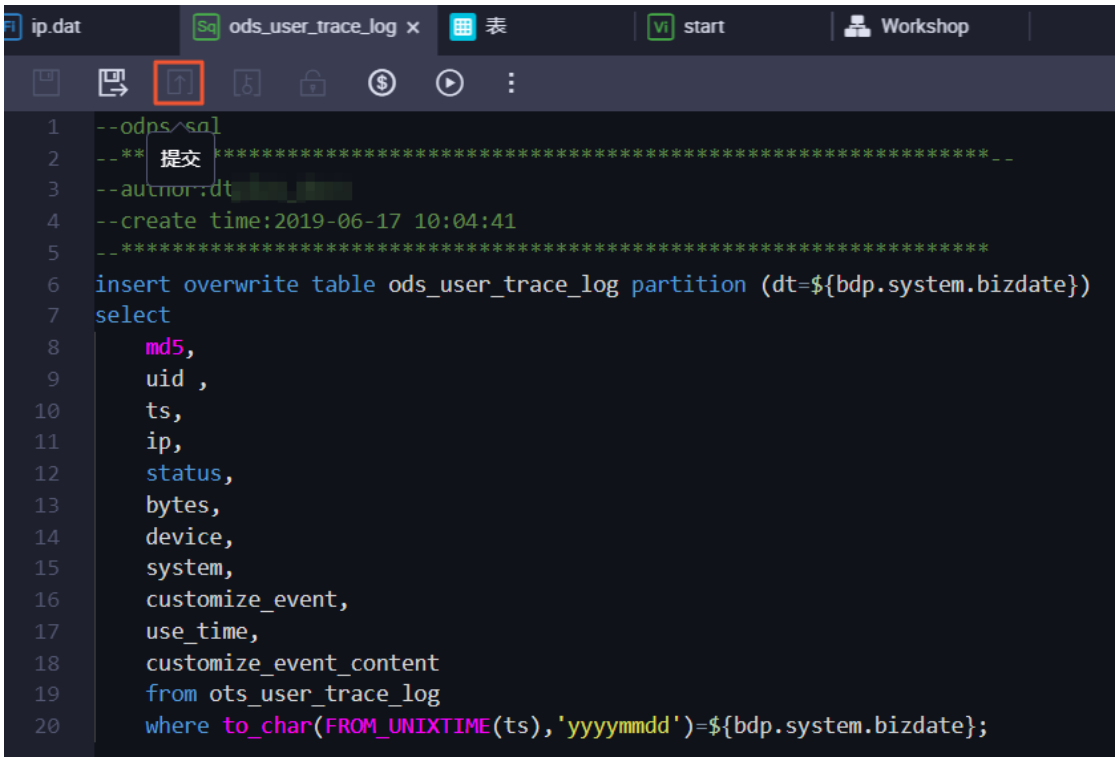


iii. 手动删除错误的依赖关系。

iv. 按照业务流程顺序搜索正确的上游节点，例如此处为start，并单击**添加**。

v. 在时间属性区域选择**重跑属性**为**运行成功或失败后皆可重跑**。

vi. 完成后，单击**提交**。



3. 配置ODPS SQL节点dw_user_trace_log

   您可以使用与ods_user_trace_log节点一样的方法配置dw_user_trace_log节点，SQL代码如下。

```
INSERT OVERWRITE TABLE dw_user_trace_log PARTITION (dt=${bdp.system.bizdate})
SELECT uid, getregion(ip) AS region
    , CASE
        WHEN TOLOWER(device) RLIKE 'xiaomi' THEN 'xiaomi'
        WHEN TOLOWER(device) RLIKE 'meizu' THEN 'meizu'
        WHEN TOLOWER(device) RLIKE 'huawei' THEN 'huawei'
        WHEN TOLOWER(device) RLIKE 'iphone' THEN 'iphone'
        WHEN TOLOWER(device) RLIKE 'vivo' THEN 'vivo'
        WHEN TOLOWER(device) RLIKE 'honor' THEN 'honor'
        WHEN TOLOWER(device) RLIKE 'samsung' THEN 'samsung'
        WHEN TOLOWER(device) RLIKE 'leeco' THEN 'leeco'
        WHEN TOLOWER(device) RLIKE 'ipad' THEN 'ipad'
        ELSE 'unknown'
    END AS device_brand, device
    , CASE
        WHEN TOLOWER(system) RLIKE 'android' THEN 'android'
        WHEN TOLOWER(system) RLIKE 'ios' THEN 'ios'
        ELSE 'unknown'
    END AS system_type, customize_event, use_time, customize_event_content
FROM ods_user_trace_log
WHERE dt = ${bdp.system.bizdate};
```

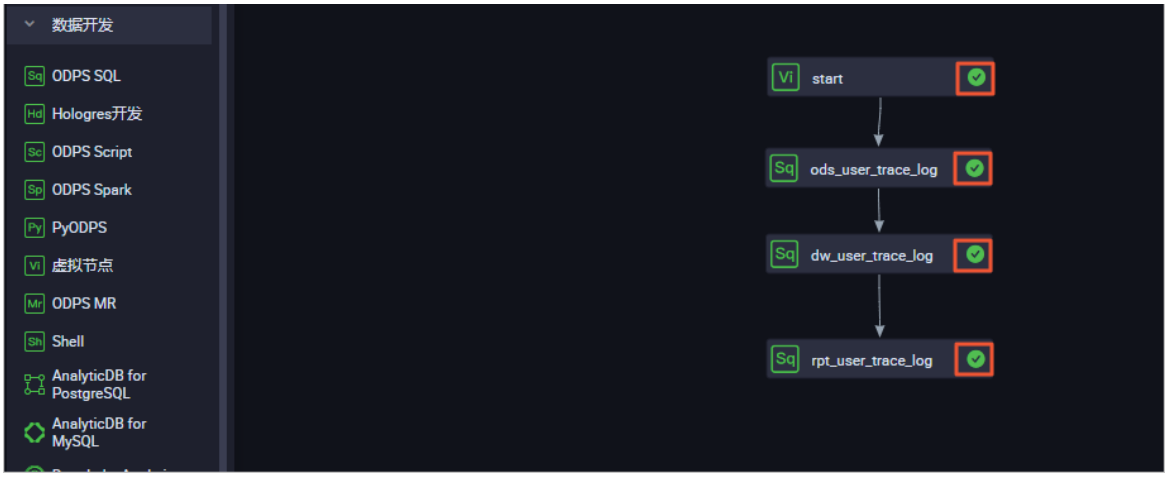
4. 配置ODPS SQL节点rpt_user_trace_log

   您可以使用与ods_user_trace_log节点一样的方法配置rpt_user_trace_log节点，SQL代码如下。

```
INSERT OVERWRITE TABLE rpt_user_trace_log PARTITION (dt=${bdp.system.bizdate})
SELECT split_part(split_part(region, ',', 1),'[',2) AS country
    , trim(split_part(region, ',', 2)) AS province
    , trim(split_part(region, ',', 3)) AS city
    , MAX(device_brand), MAX(device)
    , MAX(system_type), MAX(customize_event)
    , FLOOR(AVG(use_time / 60))
    , MAX(customize_event_content), COUNT(uid) AS pv
    , COUNT(DISTINCT uid) AS uv
FROM dw_user_trace_log
WHERE dt = ${bdp.system.bizdate}
GROUP BY uid,
    region;
```

5. 验证配置结果。

   双击业务流Workshop，打开画布面板。单击 按钮。运行成功如下图所示。



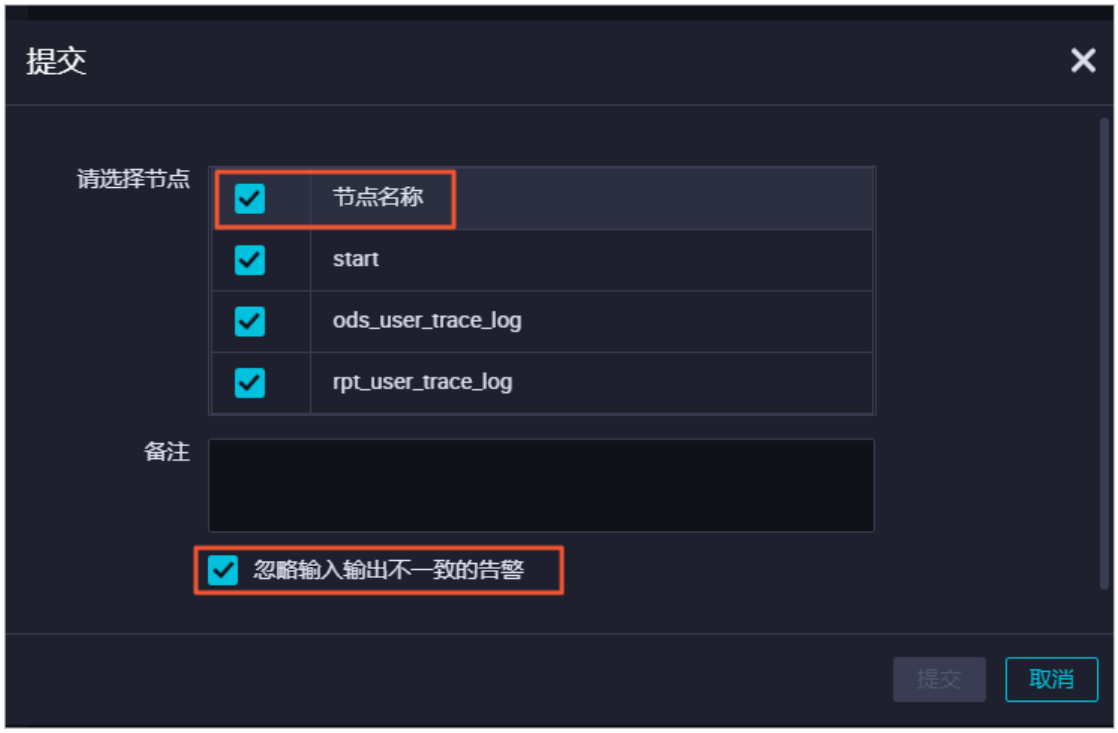
   如果运行状态异常，请右键单击出错节点，单击**查看运行日志**进行排查。

4.4.4. 任务提交与测试

您完成节点配置后，需要将任务提交到运维中心进行测试。

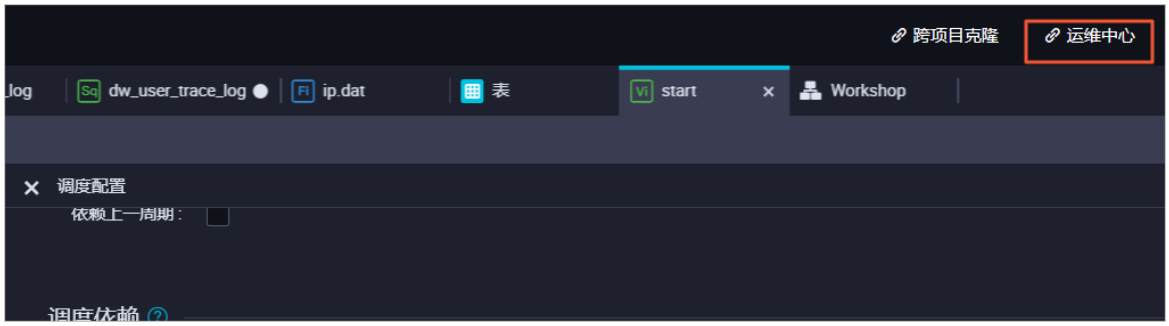
操作步骤

1. （可选）提交业务流程。如果您的节点在配置完成后已经提交完毕且无更新，请跳过本步骤。
   i. 双击业务流程名称Workshop，单击图标。
   ii. 勾选所有可提交节点及**忽略输入输出不一致的告警**，单击**提交**。



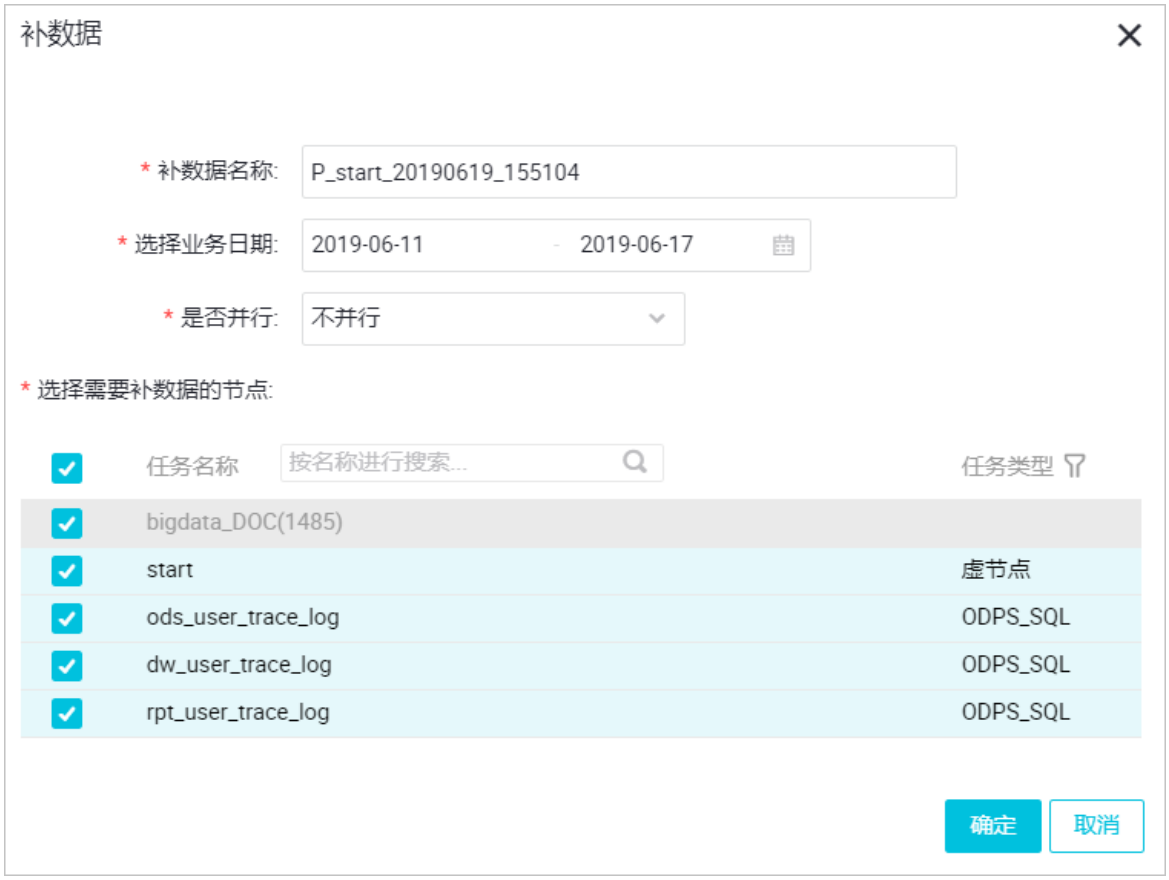
> **说明** 标准空间模式下，提交通过后，需要单击**发布**将任务发布至生产环境。

2. 单击右上角的**运维中心**。



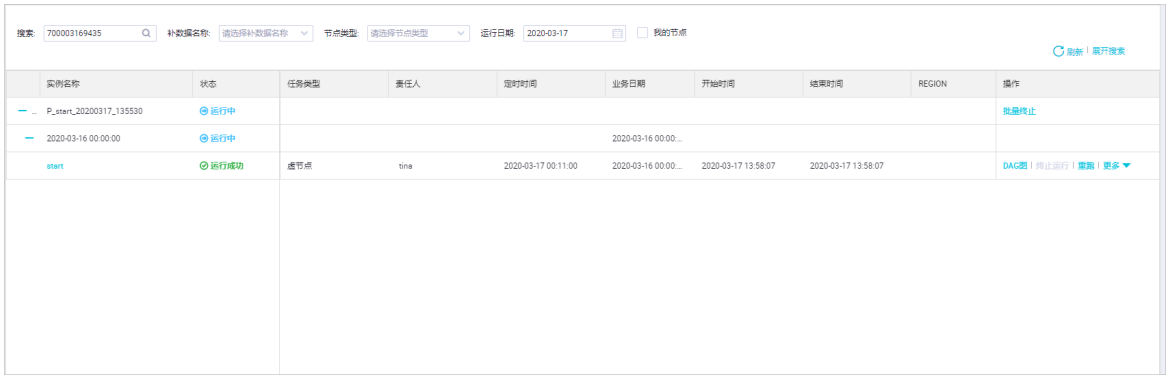
3. 在左侧导航栏，单击**周期任务运维 > 周期任务**，双击节点列表中的虚拟节点start。
4. 在右侧流程图上，右键单击虚拟节点start，选择**补数据 > 当前节点及下游节点**。

5. 在**补数据**页面，选中所有需要补数据的节点，选择业务日期为过去一周，单击**确定**。



> **说明** 关于补数据实例的详情请参见执行补数据并管理补数据实例。

6. 在左侧导航栏，单击**补数据实例**，查看补数据实例的运行情况，并通过单击**刷新**查看实时状态。



如果运行状态异常，右键单击出错节点，选择**查看运行日志**进行排查。

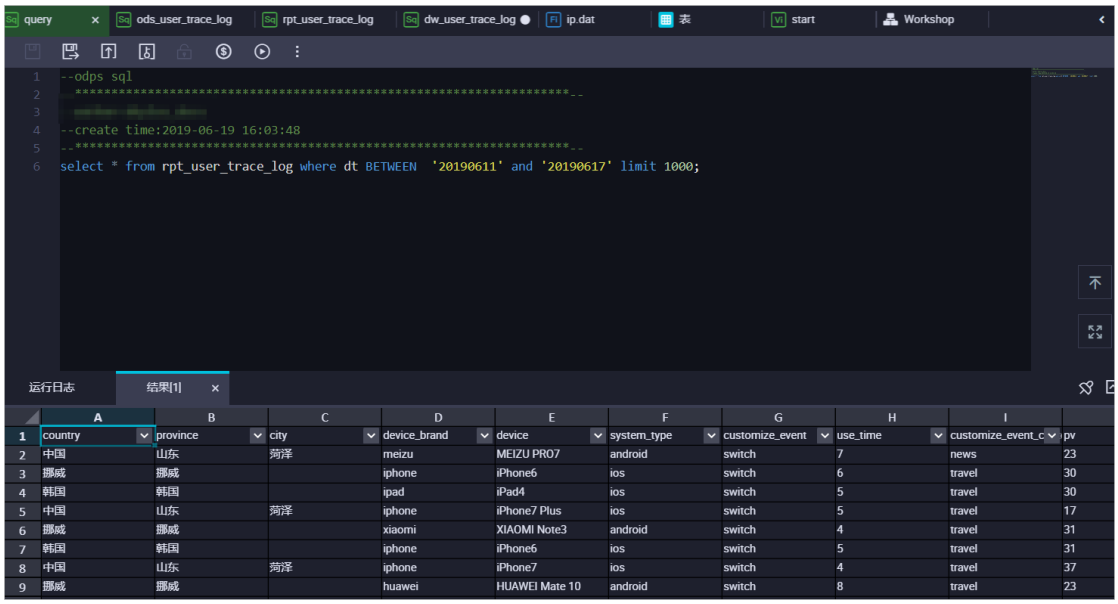
7. 补数据实例运行完成后，验证结果。
   i. 在左侧导航栏，单击业务流程**Workshop > MaxCompute**，右键单击**数据开发**，选择**新建 > ODPS SQL**，新建名为query的SQL节点。

ii. 输入如下SQL语句，查询2019年6月11日到2019年6月17日之间表rpt_user_trace_log中的数据，确认数据是否成功写入rpt_user_trace_log表。

```
select * from rpt_user_trace_log where dt BETWEEN '20190611' and '20190617' limit 1000;
```

iii. 单击图标。

查询结果如下。



	country	province	city	device_brand	device	system_type	customize_event	use_time	customize_event_c	pv
1	中国	山东	菏泽	meizu	MEIZU PRO7	android	switch	7	news	23
2	挪威	挪威		iphone	iPhone6	ios	switch	6	travel	30
3	韩国	韩国		ipad	iPad4	ios	switch	5	travel	30
4	中国	山东	菏泽	iphone	iPhone7 Plus	ios	switch	5	travel	17
5	挪威	挪威		xiaomi	XIAOMI Note3	android	switch	4	travel	31
6	韩国	韩国		iphone	iPhone6	ios	switch	5	travel	31
7	中国	山东	菏泽	iphone	iPhone7	ios	switch	4	travel	37
8	挪威	挪威		huawei	HUAWEI Mate 10	android	switch	8	travel	23

4.5. 数据可视化展现

数据表rpt_user_trace_log加工完成后，您可以通过Quick BI创建网站用户分析画像的仪表板，实现该数据表的可视化。

前提条件

在开始实验前，请确认您已经完成了环境准备和数据建模与开发的全部步骤。进入Quick BI控制台。

背景信息

rpt_user_trace_log表包含了country、province、city、device_brand、use_time、pv等字段信息。您可以通过仪表板展示用户的核心指标、周期变化、用户地区分布和记录。

操作步骤

1. 单击进入**默认空间**，您也可以使用自己的**个人空间**。

2. 新建MaxCompute数据源。
   i. 单击左侧导航栏上的**数据源**，进入**数据源**页面。
   ii. 单击右上角的**新建数据源**。选择**云数据库 > MaxCompute**。
   iii. 在**添加MaxCompute数据源**页面，配置数据源连接参数。

   完成填写后，单击**连接测试**，待显示**数据源连通性正常**后单击**添加**即可。

- 显示名称：数据源配置列表的显示名称。
- 数据库地址：此处有默认地址，通常无需修改。

  > **说明** 数据库地址根据Region不同而变化，详细对应信息请参见Endpoint。

- 项目名称：MaxCompute项目名称。
- AccessKey ID：您账号的AccessKey ID。
- AccessKey Secret：您账号的AccessKey Secret。

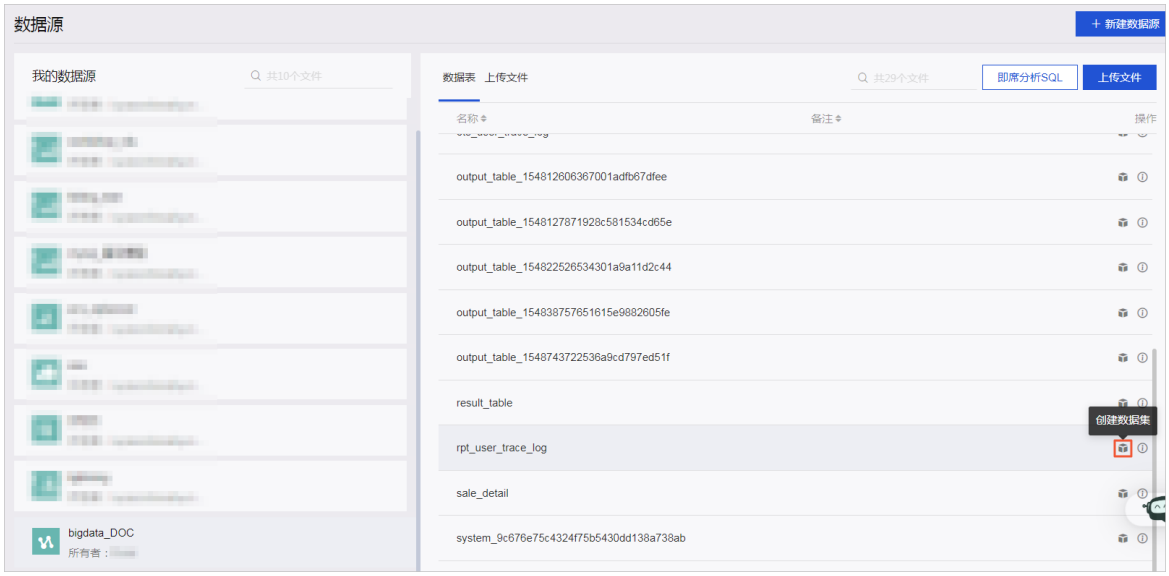
iv. 单击**连接测试**，进行数据源连通性测试。

> **说明** 如果连通正常，系统会给出连通成功提示。

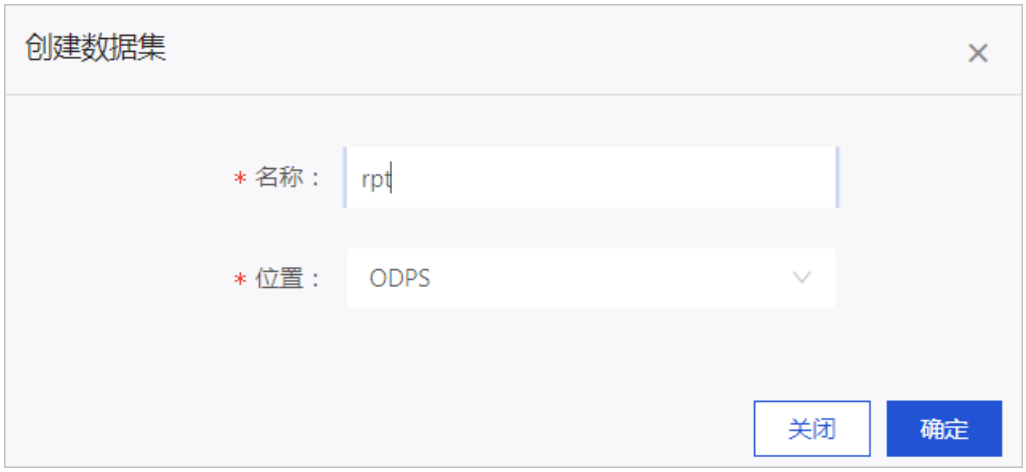
v. 单击**添加**，完成数据源添加。

成功添加完成后，页面自动跳转到**数据源**管理页面，并在页面右侧展示出数据源所包含的所有数据表。

3. 在**数据源**管理页面找到rpt_user_trace_log表，单击**创建数据集**。



输入数据集的**名称**和**位置**，单击**确定**。



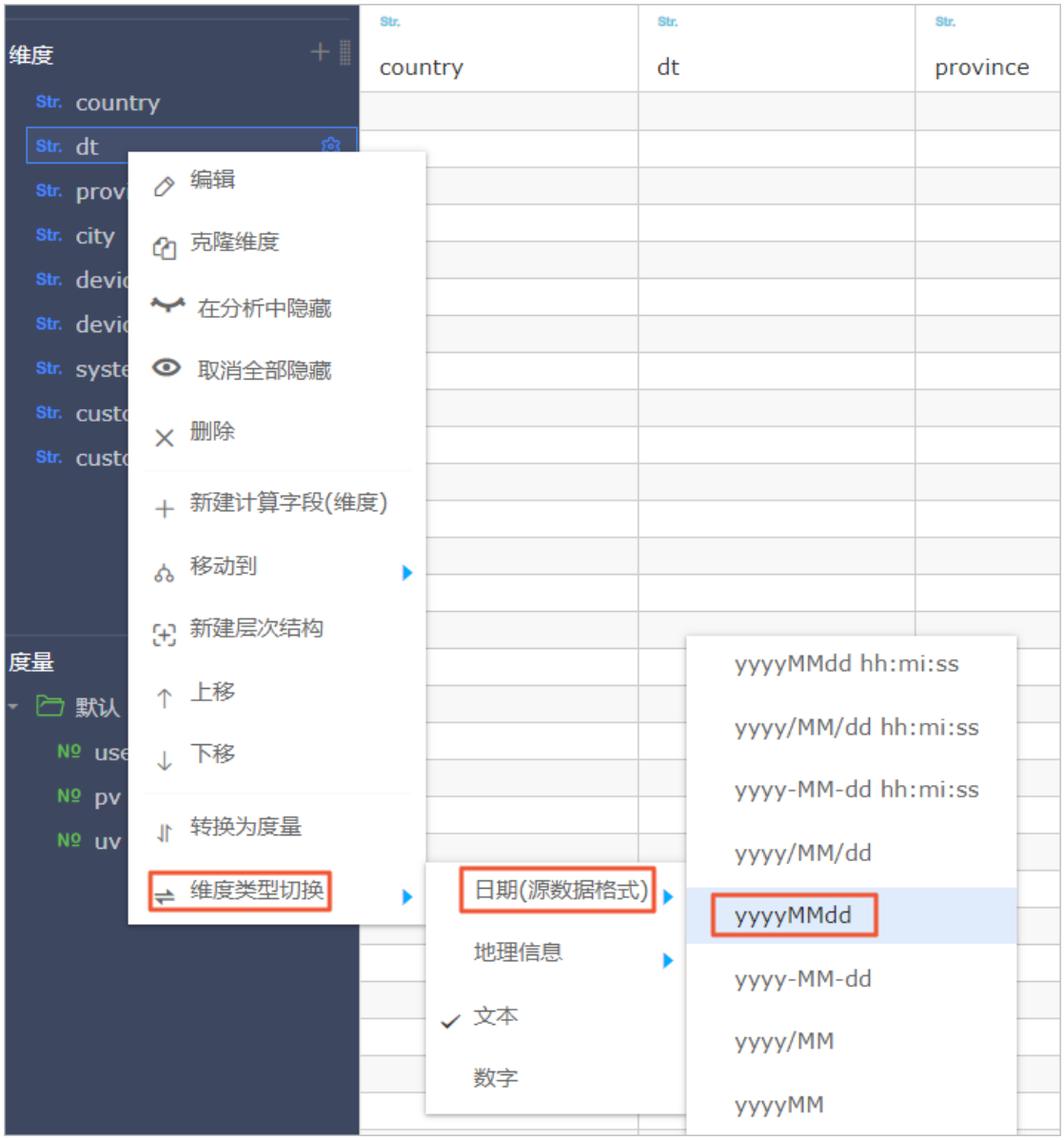
4. 单击左侧导航栏上的**数据集**，进入**数据集**页面。单击您刚刚创建的数据集，对数据集进行编辑。

   常见的数据集加工包括维度、度量的切换、修改维度的类型、增加计算字段、创建层次结构、修改字段的数据类型、更改度量聚合方式、制作关联模型。详情请参见概述。

5. 转换字段的维度类型。

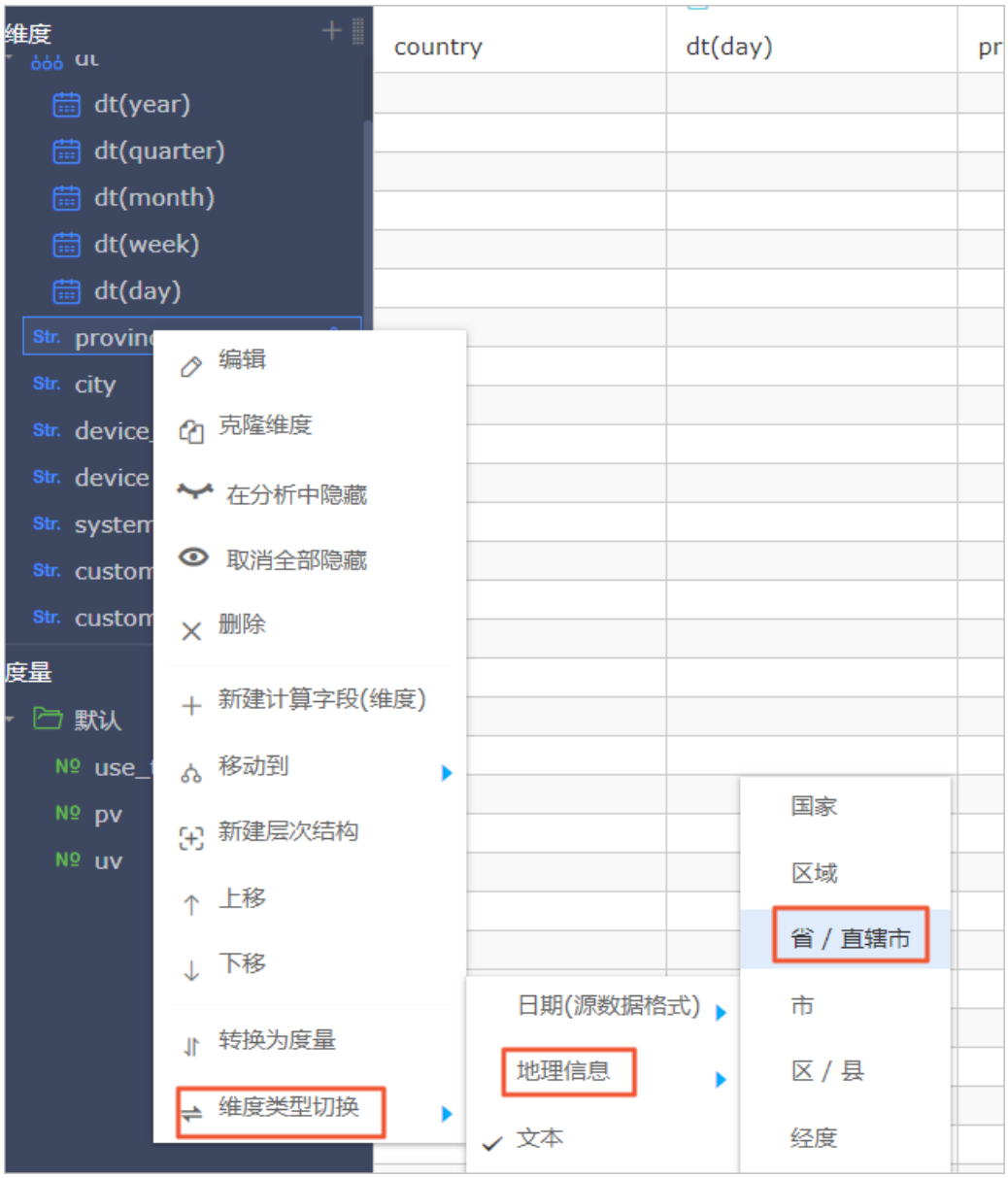
i. 转换日期字段的维度类型。

右键单击dt字段，选择**维度类型切换 > 日期（源数据格式） > yyyyMMdd**。

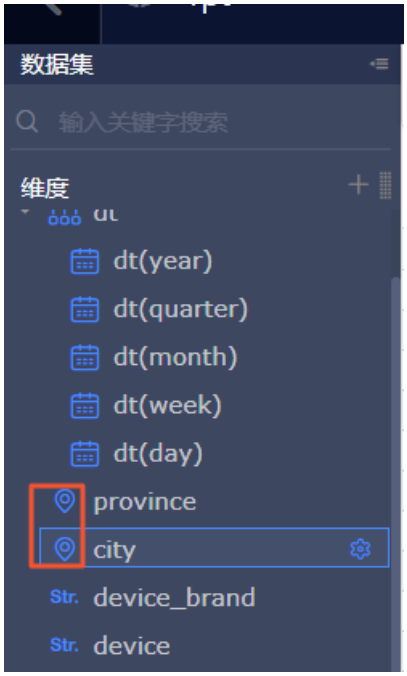


ii. 转换地理信息字段的维度类型。

a. 右键单击province字段，选择**维度类型切换 > 地理信息 > 省/直辖市**。

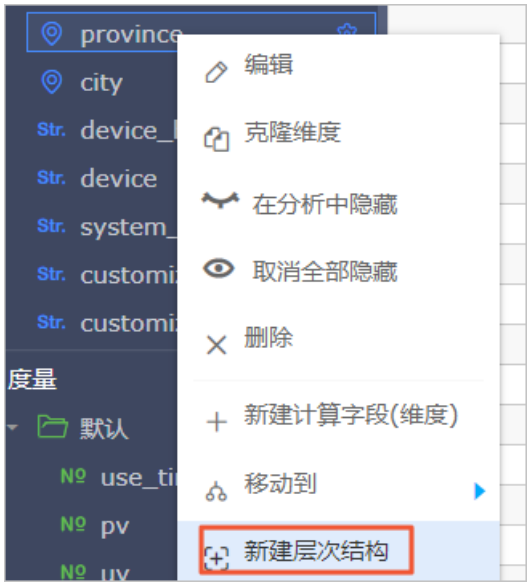


b. 右键单击city字段，选择**维度类型切换 > 地理信息 > 市**。转换成功后，在左侧维度栏中会看到字段前多一个地理位置图标。



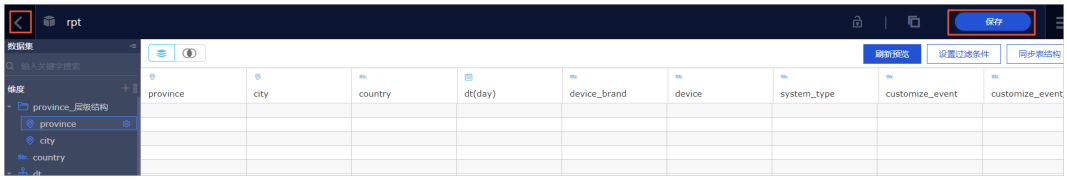
iii. 新建层次结构。

a. 右键province，单击**新建层次结构**，在弹框中单击**确定**。



b. 将city字段移到province层次结构的树下。

c. 完成上述操作后，单击**保存**，返回数据集列表。



6. 制作仪表板。

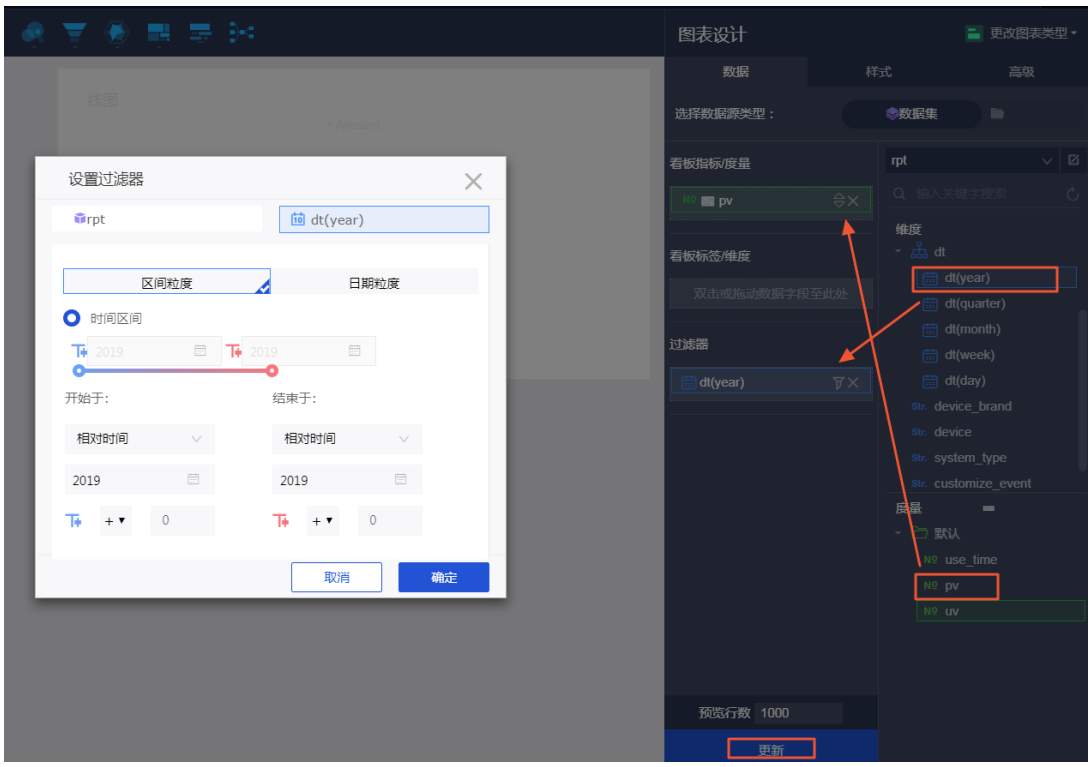
即随着数据的更新，让报表可视化地展现最新数据。

i. 单击rpt数据集后的新建仪表板图标，选择**常规模式**，进入仪表板编辑页。

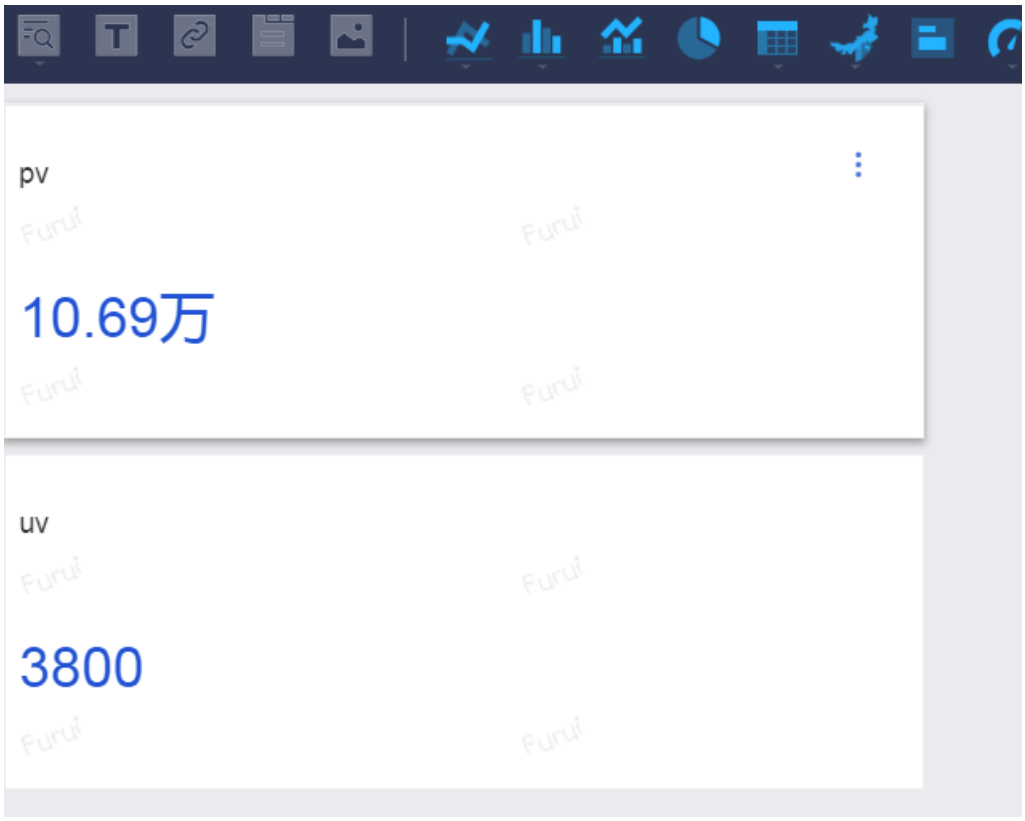
ii. 从仪表板空间中向空白区拖入2个指标看板，调整布局成一排。


未命名
指标看板
指标看板-rpt
Visitors count
1513
指标看板: 缺少维度或度量项
Visitors MoM
-15.43%
Visitors YoY
-
指标看板-rpt
Visitors count
1513
指标看板: 缺少维度或度量项
Visitors MoM
-15.43%
Visitors YoY
-

- 指标看板一：选择数据来源为数据集rpt，选择度量为pv。由于数据表rpt_user_trace_log为分区表，因此必须在**过滤器**处选择筛选的日期，本例中筛选为2019~2019年，完成设置后单击**更新**。



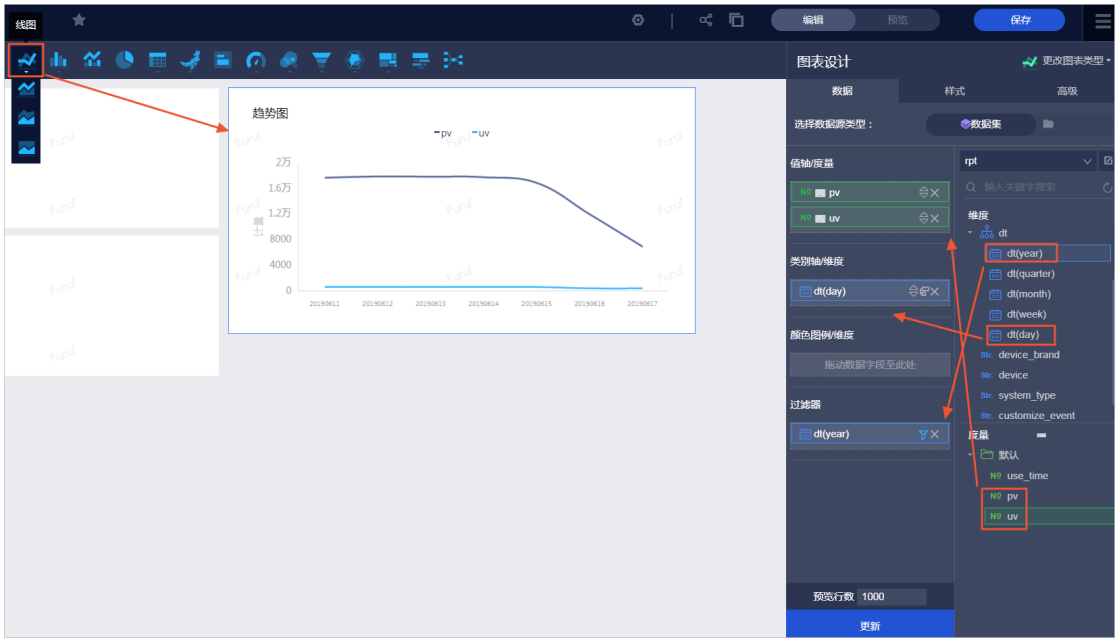
- 指标看板二：选择数据来源为来自数据集rpt，选择度量为uv，其他操作同上。完成设置后单击**更新样式**处设置指标看板显示的名称，显示效果如下。



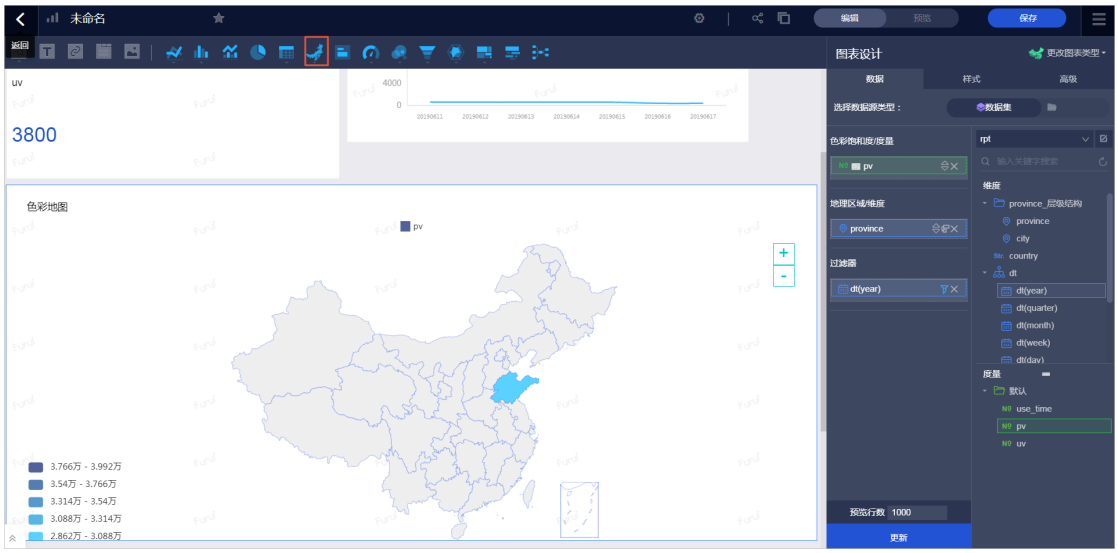
iii. 制作趋势图：将图表区域内的**线图**拖拽到左侧画布。

参数配置如下，完成之后单击**更新**：

- **值轴/度量**：pv、uv
- **类别轴/维度**：dt（day）
- **过滤器**：dt（year）



iv. 制作色彩地图：单击图表区域内的色彩地图，并选择数据源来源为数据集rpt_user_trace_log，选择**地理区域/维度**为province（地区）、**色彩饱和度/度量**为pv，选择完成后单击**更新**，结果如下。



v. 完成配置后，单击**保存**及**预览**，即可看到展示效果。

5.数据质量保障教程

5.1. 数据质量教程概述

数据质量是数据分析结论有效性和准确性的基础。本文为您介绍数据质量保障教程的业务场景以及如何衡量数据质量的高低。

前提条件

在开始本教程前，请您首先完成搭建互联网在线运行分析平台教程，详情请参见业务场景与开发流程。

业务场景

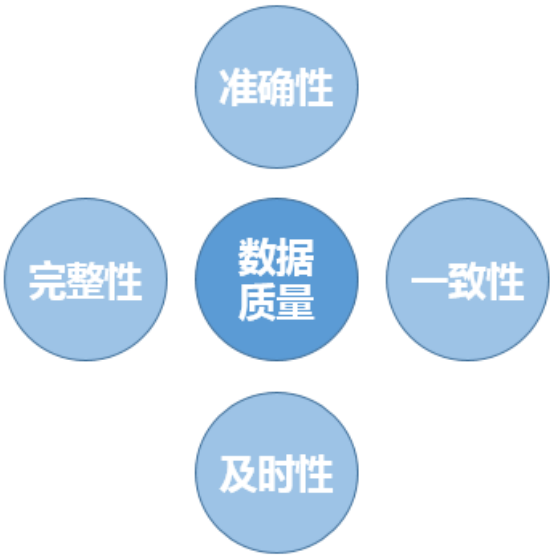
为保证业务数据质量，首先您需要明确数据的消费场景和加工链路。

本教程使用的数据来源于某网站上的HTTP访问日志。基于这份网站日志，您可以统计并展现网站的浏览次数（PV）和独立访客（UV），并能够按照用户的终端类型（如Android、iPad、iPhone、PC等）和地域分别统计。

在整体数据链路的处理过程中，为保证最终产出数据的质量，您需要对数据仓库ODS、CDM和ADS层的数据分别进行监控。数据仓库分层的定义请参见数仓分层。本教程基于搭建互联网在线运行分析平台教程，ods_user_trace_log、dw_user_trace_log、rpt_user_trace_log分别代表数据仓库的ODS、CDM和ADS层，详情请参见设计工作流。

数据质量的评估

数据质量可以从完整性、准确性、一致性和及时性共四个角度进行评估，详情请参见数据质量评估标准。



在本教程中，您将学会通过数据质量风险监控，保证数据的完整性、准确性、一致性；通过数据及时性监控，保证数据的及时性。

- 完整性

  完整性是指数据的记录和信息是否完整、不缺失。数据的缺失包括数据记录的缺失（表行数异常）和记录中某字段信息的缺失（字段出现空值）。在本教程中，您需要重点关注数据的生产环节（MaxCompute外部表引用的表格存储数据）和加工环节（数据仓库CDM及ADS层）中表行数是否大于0、表行数波动是否正常以及字段是否出现空值或重复的情况。

- 准确性

  准确性是指数据记录中信息和数据是否准确、不存在错误或异常。例如，在本教程中，如果UV、PV数值小于0，则明显是错误数据。

- 一致性

  对于不同的业务流程和节点，同一份数据必须保持一致性。例如表 `province` 字段中如果有浙江、ZJ两种表述，在您group by province时会出现两条记录。

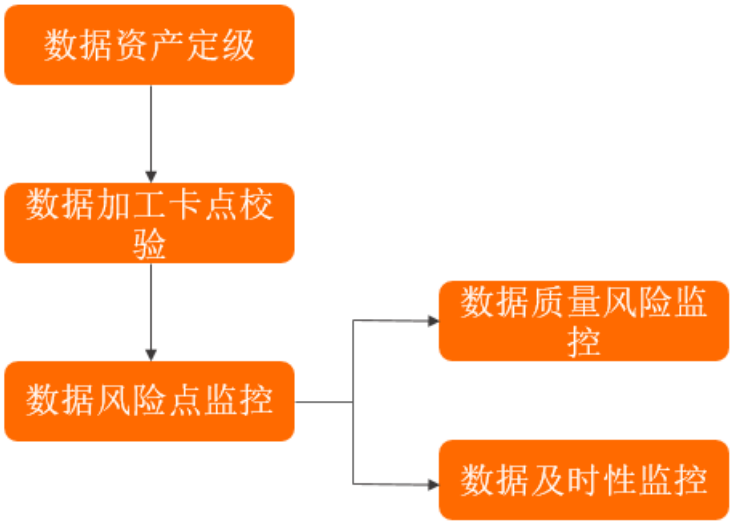
- 及时性

  及时性主要体现在最终ADS层的数据可以及时产出。为保证及时性，您需要确保整条数据加工链路上的每个环节都可以及时产出数据。本教程将利用DataWorks智能监控功能保证数据加工每个环节的及时性。

5.2. 数据质量管理流程

数据质量的管理流程包括业务数据资产定级、加工卡点、风险点监控和及时性监控，您可以构建属于自己的数据质量保障体系。

数据质量管理的流程图如下。



数据质量管理的流程说明如下：

1. 分析业务场景，对数据流转链路上的整个依赖关系，进行资产定级。详情请参见数据资产定级。
2. 在业务系统的数据生成过程中进行卡点校验。详情请参见离线数据加工卡点校验。
3. 对数据风险点进行监控，包括数据的质量风险和及时性。详情请参见：
   - 数据质量风险监控
   - 数据及时性监控

5.3. 数据资产定级

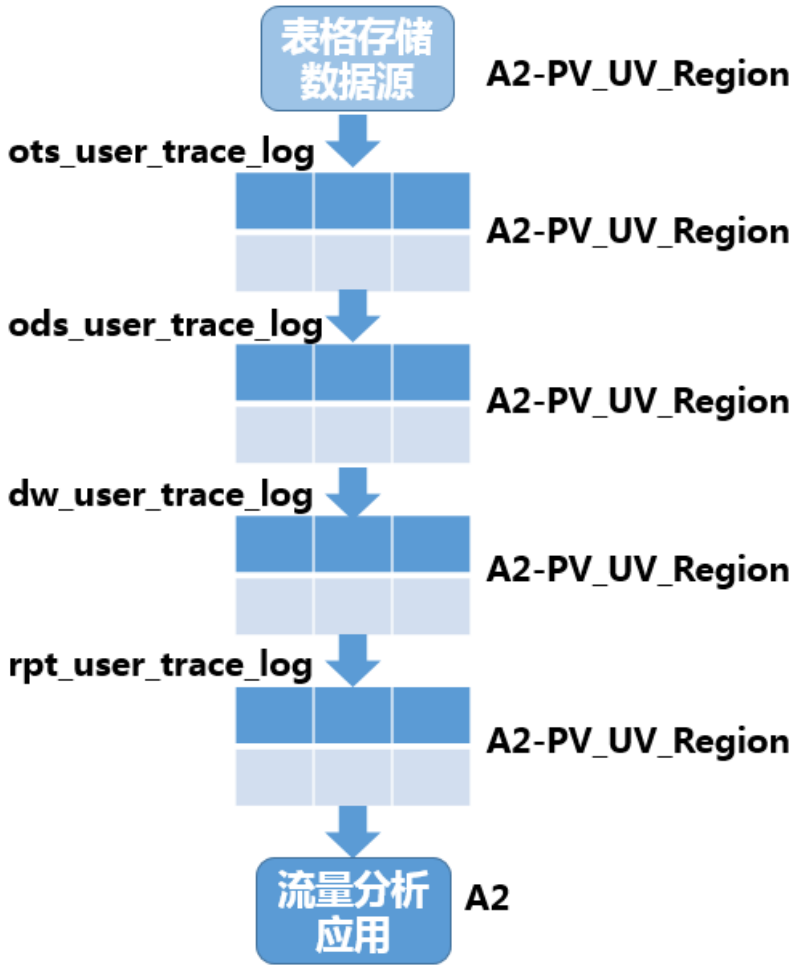
数据的资产等级，可以根据数据质量不满足完整性、准确性、一致性、及时性对业务的影响程度进行划分。

数据等级定义如下：

- 毁灭性质：数据一旦出错，将会引起重大资产损失，面临重大收益损失等。标记为A1。
- 全局性质：数据直接或间接用于企业级业务、效果评估和重要决策等。标记为A2。
- 局部性质：数据直接或间接用于某些业务线的运营、报告等，如果出现问题会给业务线造成一定的影响或造成工作效率降低。标记为A3。
- 一般性质：数据主要用于日常数据分析，出现问题带来的影响极小。标记为A4。
- 未知性质：无法明确数据的应用场景。标记为Ax。

资产等级标记包含毁灭性质为A1、全局性质为A2、局部性质为A3、一般性质为A4、未知性质为Ax。重要程度为A1>A2>A3>A4>Ax。

在数据流转链路上，您需要整理消费各个表的应用业务。通过给这些应用业务划分数据资产等级，结合数据的上下游依赖关系，将整个链路打上某一类资产等级的标签。在本教程中，互联网在线运营分析平台只存在一个应用，统计并展现网站的PV和UV，并能够按照用户的终端类型和地域进行统计，命名为PV_UV_Region。假设该应用会直接影响整个企业的重要业务决策，您可以定级应用为A2，从而整个数据链路上的表的数据等级，都可以标记为A2-PV_UV_Region。



> **说明** 当前MaxCompute暂无配套资产等级打标工具，您可以使用第三方工具完成打标。

5.4. 离线数据加工卡点

离线数据加工卡点，主要指在业务系统的数据生成过程中进行的卡点校验。

代码提交的卡点校验

代码提交卡点主要包括您在提交代码时，手动或自动进行SQL扫描，检查您的SQL逻辑。校验规则分类如下：

- 代码规范类规则。

  例如，表命名规范、生命周期设置及表注释等。

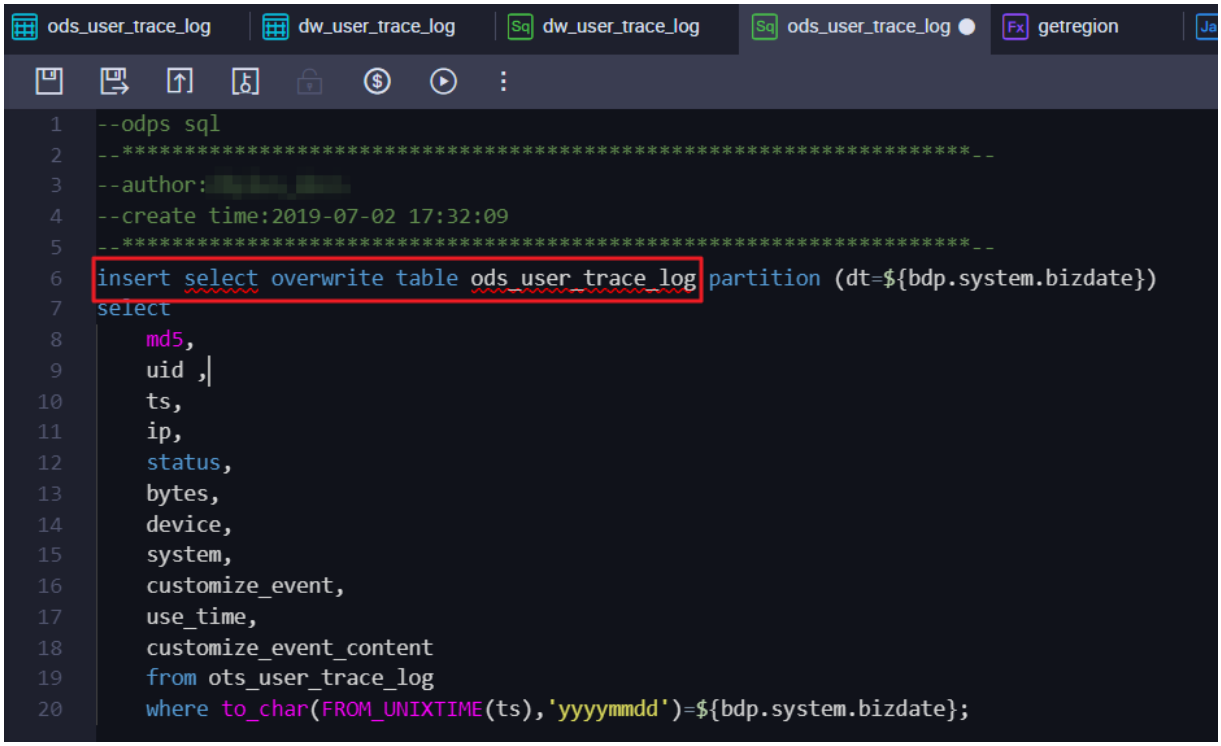
- 代码质量类规则。

  例如，分母为0提醒、NULL值参与计算影响结果提醒及插入字段顺序错误等。

- 代码性能类规则。

  例如，分区裁剪失效、扫描大表提醒及重复计算检测等。

您在使用DataWorks数据开发功能时，如果代码中有语法错误，会出现如下红色波浪线提示。



关于SQL代码、表命名、生命周期、注释的其他规范，请参见表设计规范及MaxCompute数据开发规范。

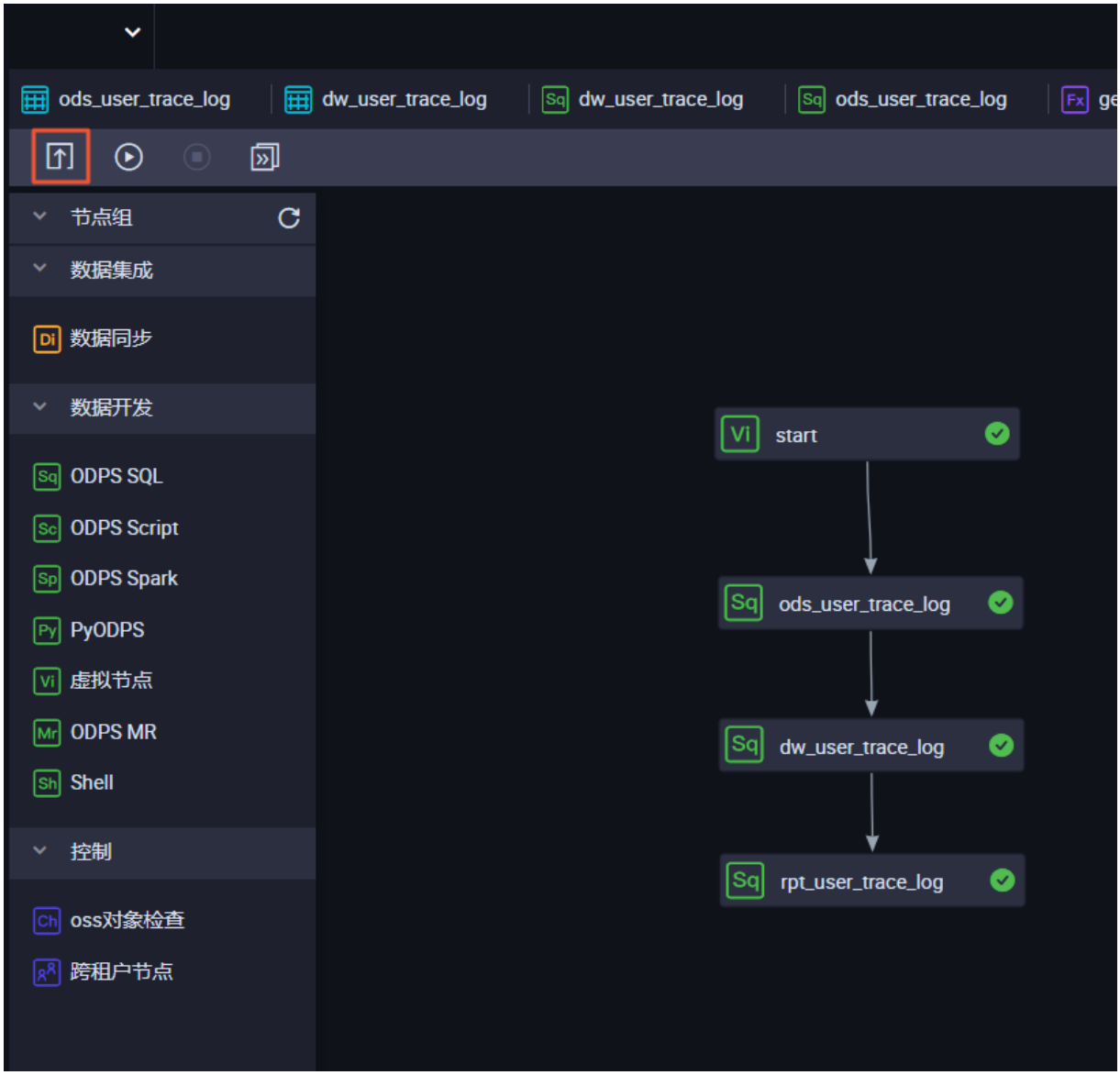
任务发布卡点

为保证线上数据的准确性，每次变更都需要经过测试再发布到线上生产环境，且生产环境测试通过后才算发布成功。发布上线前的测试包括代码审查和回归测试。对于资产等级较高的应用，必须在完成回归测试之后，才允许任务发布，本教程中应用为A2等级，属于高资产级别应用。

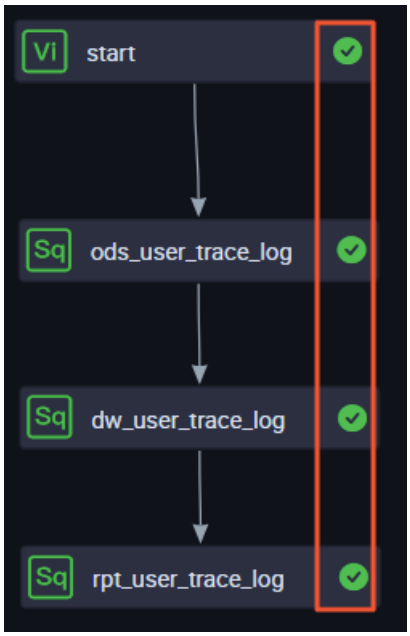
回归测试需要您能充分模拟真实环境进行测试：

- 对于标准模式项目，您可使用SQL语句将数据从生产环境复制开发环境，运行业务流程，观察是否存在报错。
- 对于简单模式的项目，您可以直接运行业务流程，观察是否存在报错。

由于本教程使用简单模式，您直接提交任务运行业务流程即可。


ods_user_trace_log
dw_user_trace_log
dw_user_trace_log
ods_user_trace_log
节点组
数据集成
数据同步
数据开发
ODPS SQL
ODPS Script
ODPS Spark
PyODPS
虚拟节点
ODPS MR
Shell
控制
oss对象检查
跨租户节点
start
ods_user_trace_log
dw_user_trace_log
rpt_user_trace_log

完成运行后，如果所有节点都显示绿色图标，则表示业务流程测试通过。



相关人员通告

在进行更新操作前，需要通知下游变更原因、变更逻辑、变更时间等信息。下游对此次变更没有异议后，再按照约定时间执行发布变更，将变更对下游的影响降到最小。例如，在本教程中，如果表格存储上数据源的表结构发生了变更，您需要通知ots_user_trace_log、ods_user_trace_log、dw_user_trace_log、rpt_user_trace_log表的责任人，及时更新表结构。

5.5. 数据质量风险监控

数据质量风险监控主要针对数据的准确性、一致性和完整性。本教程使用DataWorks数据质量（DQC）功能，完成数仓各层次的数据质量监控。

前提条件

首先您需要完成教程搭建互联网在线运营分析平台，并保证您的DataWorks工作空间创建区域为华东2（上海），详情参见业务场景与开发流程。您需要完成数据资产定级，本教程中定义为A2，详情请参见数据资产定级。

> **说明** 数据质量风险监控理论规范，请参见数据风险点监控。

背景信息

数据质量监控和数据资产等级对应，您可以根据以下因素细化您的监控配置，数据质量的详情请参见数据质量概述。

- 监控分类：数据量、主键、离散值、汇总值、业务规则和逻辑规则。
- 监控粒度：字段级别、表级别。
- 监控层次：ODS、CDM、ADS三层数据，其中ODS和DWD层主要偏重数据的完整性和一致性。DWS和ADS层数据量较小、逻辑复杂，偏重数据的准确性。

> **说明** 如需了解各分层的详细说明，请参见数仓分层。

以下为不同数据资产等级和数仓层次数据的数据质量监控建议，仅供参考。

数据质量DQC监控规范								
监控分类					数据量	主键	离散值	汇总值	业务逻辑、规则
适合场景					所有非临时表都建议配置该项监控。	对于存在业务主键、逻辑主键的表需配置该监控。	维表、事实表中的维度值、状态值、可枚举的值需配置该监控。	汇总统计表中的汇总值需配置该监控。	1、重要指标的异常值监控。例如，正常UID长度是否为32位。 2、字段间的平衡值监控。例如，字段a与字段b满足一一对应关系等。 3、多表关联监控。例如两张表左关联，关联不上的记录数应等于0。
监控粒度					表级数据量监控	字段级	字段级	字段级	字段级/表级
常用监控规则					表行数波动/自助规则表行数>固定值	模板规则的字段空值、重复值/自定义规则监控联合主键空值、重复值情况	离散值分组个数/离散值分组个数波动/离散值状态值波动	模板规则的单字段大于0/自定义规则判断字段等于0所占的比例等	自定义规则
层次	表类型				规则配置				
ODS/DWD	离线表	A2	增量表	有周期规律	模板表行数波动率	空值、重复值唯一性	需监控	不涉及	需监控
				无周期规律	自助表行数>固定值	空值、重复值唯一性	需监控	不涉及	需监控
			全量表		模板表行数波动率	空值、重复值唯一性	需监控	不涉及	需监控
		A3	增量表	有周期规律	模板表行数波动率	空值、重复值唯一性	需监控	不涉及	不涉及
				无周期规律	自助表行数>固定值	空值、重复值唯一性	需监控	不涉及	不涉及
			全量表		模板表行数波动率	空值、重复值唯一性	需监控	不涉及	不涉及
		A4	增量表	有周期规律	模板表行数波动率	空值、重复值唯一性	需监控	不涉及	不涉及
				无周期规律	自助表行数>固定值	空值、重复值唯一性	需监控	不涉及	不涉及
			全量表		模板表行数波动率	空值、重复值唯一性	需监控	不涉及	不涉及
		Ax	增量表	有周期规律	模板表行数波动率	空值、重复值唯一性	不涉及	不涉及	不涉及
				无周期规律	自助表行数>固定值	空值、重复值唯一性	不涉及	不涉及	不涉及
			全量表		模板表行数波动率	空值、重复值唯一性	不涉及	不涉及	不涉及
DWS/ADS	离线表	A2	增量表	有周期规律	模板表行数波动率	空值、重复值唯一性	需监控	需监控	需监控
				无周期规律	自助表行数>固定值	空值、重复值唯一性	需监控	需监控	需监控
			全量表		模板表行数波动率	空值、重复值唯一性	需监控	需监控	需监控
		A3	增量表	有周期规律	模板表行数波动率	空值、重复值唯一性	需监控	需监控	不涉及
				无周期规律	自助表行数>固定值	空值、重复值唯一性	需监控	需监控	不涉及
			全量表		模板表行数波动率	空值、重复值唯一性	需监控	需监控	不涉及
		A4	增量表	有周期规律	模板表行数波动率	空值、重复值唯一性	需监控	不涉及	不涉及
				无周期规律	自助表行数>固定值	空值、重复值唯一性	需监控	不涉及	不涉及
			全量表		模板表行数波动率	空值、重复值唯一性	需监控	不涉及	不涉及
		Ax	增量表	有周期规律	模板表行数波动率	空值、重复值唯一性	不涉及	不涉及	不涉及
				无周期规律	自助表行数>固定值	空值、重复值唯一性	不涉及	不涉及	不涉及
			全量表		模板表行数波动率	空值、重复值唯一性	不涉及	不涉及	不涉及

操作步骤

1. ODS层数据质量监控。

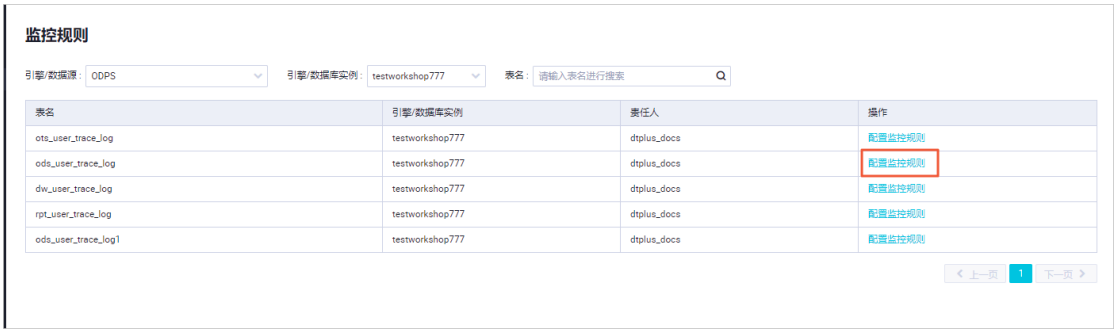
   ODS层表中的数据来源于OSS上的日志文件，作为源头表，您需要尽早判断此表分区中是否有数据。如果这张表中没有数据，则后续任务运行无意义，需要阻止后续任务运行。

   i. 进入**数据质量**页面。

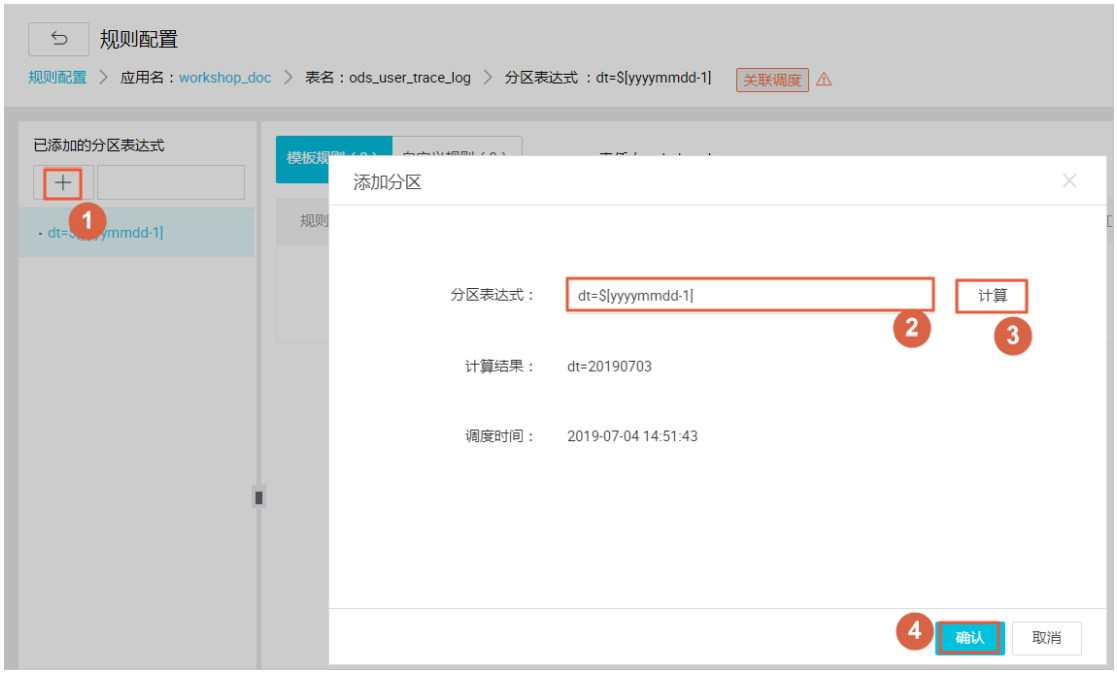
   在**数据开发**页面，单击左上角图标，选择**数据质量**。

   ii. 进入ods_user_trace_log监控规则页面。

   单击左侧导航栏上的**监控规则**，在**监控规则**页面找到代表外部数据源的ODS层表ods_user_trace_log，单击其后的**配置监控规则**。



iii. 添加分区。



a. 单击**+**，选择**分区表达式**为 dt=$[yyyymmdd-1]，对应表ods_user_trace_log的分区格式为${bdp.system.bizdate}（即获取到前一天的日期）。分区表达式的详细信息请参见配置调度参数。如果表中无分区列，可以配置无分区。

b. 单击**计算**，验证计算结果是否正确。

c. 单击**确认**，完成分区的添加。

iv. 创建规则确保ODS层表分区内存在数据。

a. 单击**创建规则**。

b. 单击**模板规则 > 添加监控规则**。

c. 输入配置参数。

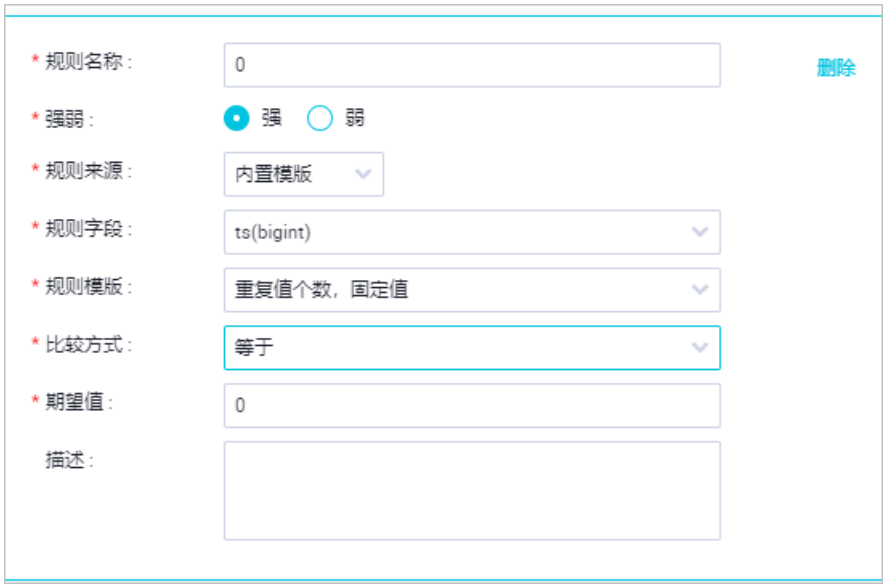
参数	描述
规则名称	请输入规则名称。您可以自定义。
强弱	设置为强规则。强弱规则说明如下： ■ 如果设置强规则，红色异常报警并阻塞下游任务节点，橙色异常报警不阻塞。 ■ 如果设置弱规则，红色异常报警不阻塞下游任务节点，橙色异常不报警不阻塞。
规则来源	选择**内置模板**。
规则字段	选择**表级规则**。
规则模板	选择**表行数，固定值**。

参数	描述
比较方式	选择**大于**。
期望值	设置为0。

v. 监控重复数据。

a. 单击**添加监控规则**。

b. 输入配置参数。



参数	描述
规则名称	请输入规则名称。您可以自定义。
强弱	设置为强规则。强弱规则说明如下： ■ 如果设置强规则，红色异常报警并阻塞下游任务节点，橙色异常报警不阻塞。 ■ 如果设置弱规则，红色异常报警不阻塞下游任务节点，橙色异常不报警不阻塞。
规则来源	选择**内置模板**。
规则字段	选择**ts(bigint)**。ts(bigint)值为用户时间戳，目的是避免ODS层出现重复的数据。
规则模板	选择**重复值个数**、**固定值**。
比较方式	选择**等于**。
期望值	设置为0。

vi. 监控空值数据。

a. 单击**添加监控规则**。

b. 输入配置参数。

参数	描述
规则名称	请输入规则名称。您可以自定义。
强弱	设置为强规则。强弱规则说明如下： ■ 如果设置强规则，红色异常报警并阻塞下游任务节点，橙色异常报警不阻塞。 ■ 如果设置弱规则，红色异常报警不阻塞下游任务节点，橙色异常不报警不阻塞。
规则来源	选择**内置模板**。
规则字段	选择**uid(string)**。uid(string)值为用户ID，目的是避免出现用户ID为空值的脏数据。
规则模板	选择**空值个数**、**固定值**。
比较方式	选择**等于**。
期望值	设置为0。

vii. 批量保存规则。完成上述操作后，单击**批量保存**。

viii. 规则试跑。

单击**试跑**，进行数据质量的校验规则。

ix. 查看试跑结果。

试跑完成后，单击**试跑成功！点击查看试跑结果**查看试跑结果。

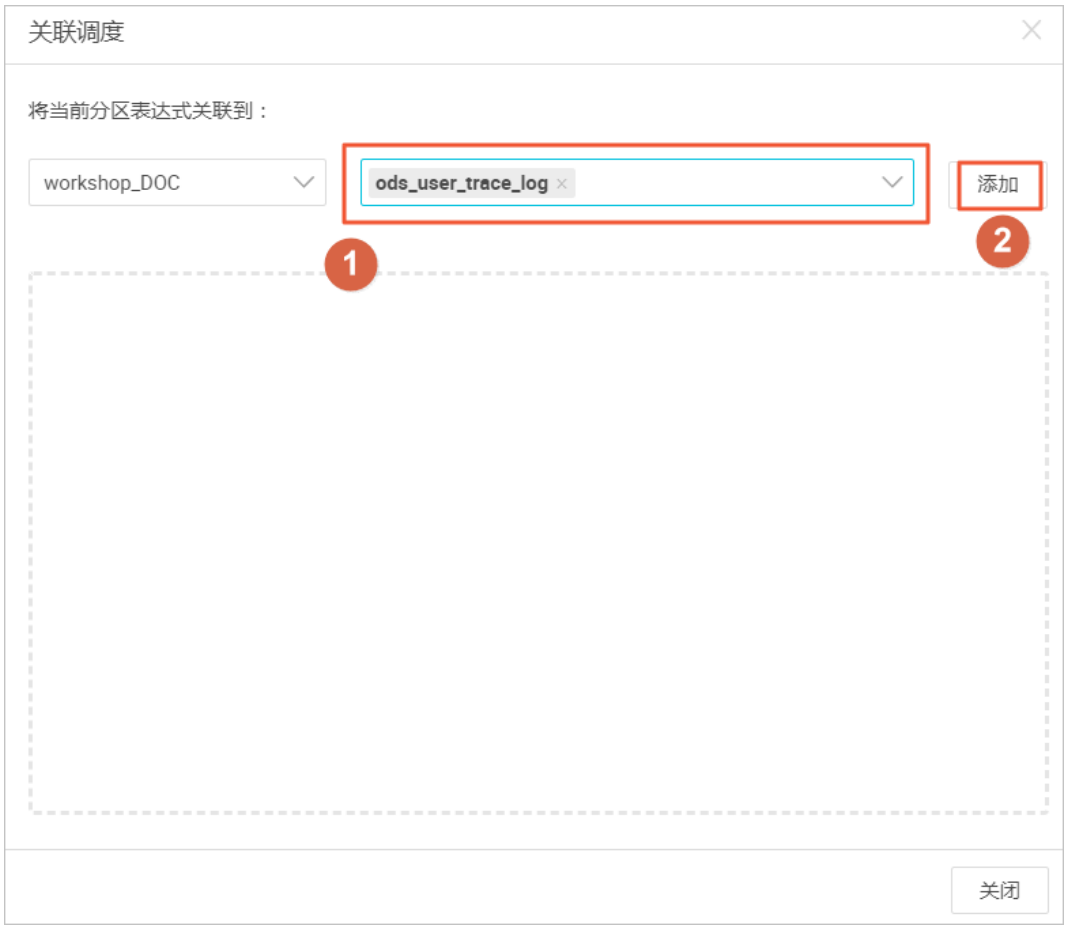
在弹出的页面中，您可以查看表数据是否已符合您的规则。根据试跑结果，可以确认此次任务产出的数据是否符合预期。建议每个表规则配置完毕后，都进行一次试跑操作，以验证表规则的适用性。

x. 关联调度。

在规则配置完毕，且试跑成功的情况下，您需要将表和其产出任务进行关联，这样每次表的产出任务运行完毕后，都会触发数据质量规则的校验，以保证数据的准确性。

a. 在表监控规则页面，单击**关联调度**，配置规则与任务的绑定关系。

b. 在**关联调度**弹框中输入您需要关联的任务节点名称，单击**添加**。

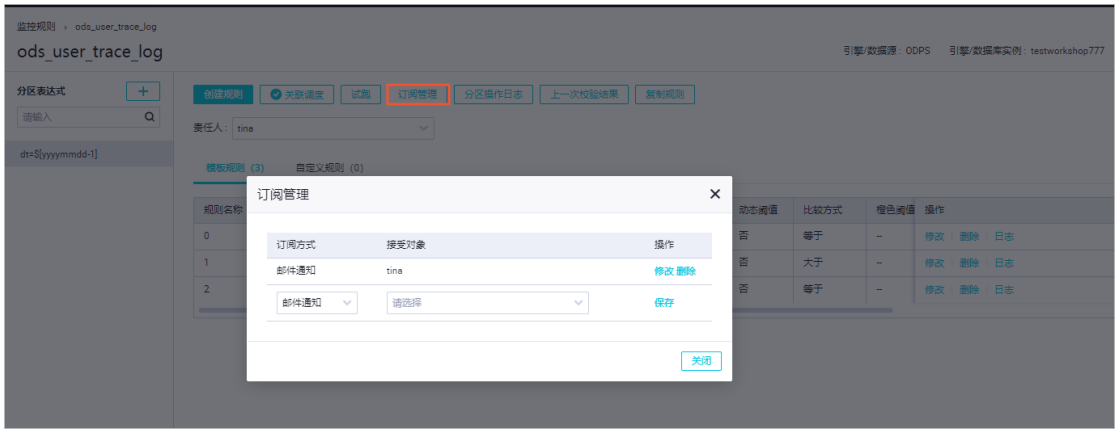


c. 单击**关闭**退出**关联调度**页面。如下图所示，关联调度配置成功。

xi. 配置任务订阅。

关联调度后，每次调度任务运行完毕，都会触发数据质量的校验。数据质量支持设置规则订阅，可以针对重要的表及其规则设置订阅，设置订阅后会根据数据质量的校验结果进行告警，从而实现对校验结果的跟踪。

单击**订阅管理**，设置接收人以及订阅方式。目前支持**邮件通知**、**邮件和短信通知**、**钉钉群机器人**和**钉钉群机器人@ALL**四种方式。



订阅管理设置完毕后，单击左侧导航栏上的**我的订阅**进行查看及修改，建议您订阅所有规则。

2. CDM层数据质量监控。

CDM层数据质量监控配置方法与ODS层相同，区别在于监控规则不同。

i. 添加分区表达式。

进入dw_user_trace_log表的规则配置页面，与ODS层一样配置分区为**dt=$[yyyymmdd-1]**，确保分区内存在表数据。

ii. 监控表行数及空值数据。表行数和空值数据的监控规则配置与ODS层相同。

iii. 监控表行数波动率。

参数	描述
规则名称	请输入规则名称。您可以自定义。
强弱	设置为强规则。强弱规则说明如下： ■ 如果设置强规则，红色异常报警并阻塞下游任务节点，橙色异常报警不阻塞。 ■ 如果设置弱规则，红色异常报警不阻塞下游任务节点，橙色异常不报警不阻塞。
规则来源	选择**内置模板**。
规则字段	选择**表级规则（table）**。
规则模板	选择**表行数、上周期波动率**。
比较方式	选择**绝对值**。
波动值比较	**橙色阈值**为10，**红色阈值**为50，代表当表行数波动率到达50%时，会产生红色报警。

iv. 规则试跑并关联调度。方法和ODS层一致。

3. ADS层数据质量监控。

ADS层数据质量监控配置方法与ODS层相同，区别在于监控规则的不同。

i. 添加分区表达式。

进入rpt_user_trace_log表的规则配置页面，同样配置分区为**dt=$[yyyymmdd-1]**。

ii. 监控表行数、波动率及空值数据。

监控表行数、波动率和空值数据的监控规则配置与CDM层相同。由于在数仓分层中，越靠近应用层数据越少、约束性越低，**强弱**选择为**弱**。

iii. 监控表异常PV。

您可以利用自定义规则功能监控ADS层的应用数据。

a. 单击**自定义规则 > 添加监控规则**。

b. 配置自定义规则参数。

参数	描述
规则名称	请输入规则名称。您可以自定义。
强弱	设置为弱规则。强弱规则说明如下： ■ 如果设置强规则，红色异常报警并阻塞下游任务节点，橙色异常报警不阻塞。 ■ 如果设置弱规则，红色异常报警不阻塞下游任务节点，橙色异常不报警不阻塞。
规则字段	选择规则字段为**pv(bigint)**。
采样方式	选择**sum**。
校验类型	选择**数值型**。
校验方式	选择**与固定值比较**。
比较方式	选择**大于**。
期望值	设置为100。当PV和异常锐减到100时，您可以及时收到告警。

c. 完成配置后，单击**批量保存**。

iv. 规则试跑并关联调度。方法与ODS层一致。

5.6. 数据及时性监控

基于MaxCompute的离线任务对数据产出有严格的时间要求，在确保数据准确性的前提下，还需要让数据能够及时提供服务。本文为您介绍如何使用DataWorks智能监控的规则管理功能监控数据的及时性。

前提条件

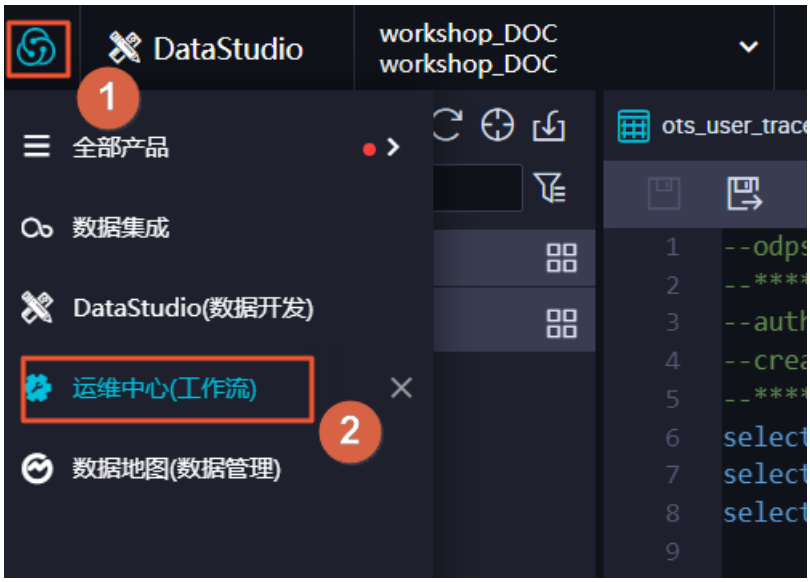
如果您想使用完整的智能监控功能，需要购买标准版及以上版本DataWorks，详情请参见DataWorks各版本详解。关于DataWorks智能监控功能详情请参见智能监控概述。

背景信息

在对数据产出及时性监控前，首先需要确定调度任务的优先级。数据资产等级越高的任务节点，优先级越高，您可以给予更加严格的数据及时性监控和告警规则。

操作步骤

1. 进入**规则管理**页面。
   i. 在DataStudio页面单击**运维中心（工作流）**。



   ii. 在**运维中心**页面，单击左侧菜单栏上的**智能监控 > 规则管理**，关于**规则管理**的详情请参见自定义规则。
2. 新建自定义规则。

   单击右上角的**新建自定义规则**，输入参数后单击**确定**即可。在本例中，监控整个业务流程每次运行时间不可超过30分钟。如果运行时间超过30分钟，则上报一次告警。连续上报3次告警，系统自动以邮件及短信的方式来上报。

分类	参数	描述
基本信息	**规则名称**	输入新建自定义规则的名称。
	对象类型	控制监控的粒度，包括**任务节点**、**基线**、**工作空间**、**业务流程**、**独享调度资源组**和**独享数据集成资源组**。
	规则对象	如果**对象类型**选择任务节点、基线、工作空间和业务流程，则需要填写**规则对象**。输入监控对象的名称或者ID后，在列表中选择需要添加的对象，单击图标。
	任务白名单	当**对象类型**为**基线**、**工作空间**、**业务流程**时，支持您输入**节点名称/ID**，**添加**至白名单列表中。白名单中的任务将不受监控。

分类	参数	描述
	资源组名称	如果**对象类型**选择独享调度资源组和独享数据集成资源组，则需要选择**资源组名称**。
触发方式	**触发条件**	如果**对象类型**选择任务节点、基线、工作空间和业务流程，此时触发条件取值如下： ○ **完成** 表示从实例任务运行的起始时间点开始监控，在任务运行成功时系统发送报警。 ○ **未完成** 表示从实例任务运行的起始时间点开始监控，到指定的目标时间点任务仍未结束运行，则系统发送报警。例如，实例任务的定时调度时间为1点，设置的未完成时间为2点，则1点时该任务开始运行，在2点时任务仍未结束运行，则发送报警。 ○ **出错** 表示从实例任务运行的起始时间点开始监控，如果任务运行出错，则系统发送报警。 实例任务运行出错即在**运维中心 > 周期任务运维 > 周期实例**的基本信息列，目标实例显示⊗状态。 ○ **周期未完成** 表示在指定的周期内，实例任务仍未结束运行，则系统发送报警。通常用于监控以小时为周期单位的实例任务。 例如，任务A每2小时调度一次，运行一次耗时25min。运行起始时间为每日0点0分，则该任务一天（24小时）共有12个任务周期，0点为第一个周期，2点为第二个周期，依次类推，22点为第12个周期。该任务正常运行时，会在每日0点25分、2点25分等时间节点执行完毕。如果在任意周期结束时间点该任务仍未结束运行，则发送报警。

分类	参数	描述
		说明 周期未完成可用于监控业务流程等任务。 当业务流程设置了**周期未完成**监控后，系统会根据您设置的周期N，对业务流程中的节点任务（例如，天任务、小时任务、分钟任务等），进行第N个周期任务的监控。如果任务实例数少于N时，则会忽略该任务的告警。 例如，设置的周期为3，业务流程中有如下两个节点任务，则告警监控情况如下： ■ 小时任务A：每2小时调度一次，运行一次耗时25min。运行起始时间为每日0点0分，则该任务一天（24小时）共有12个任务周期，0点为第一个周期，则第3个周期为4点。该任务正常运行时，第3个周期任务会在4点25分执行完毕。如果在该周期结束时间点该任务仍未结束运行，则发送报警。 ■ 分钟任务B：每10分钟调度一次，运行一次耗时2min。运行起始时间为每日0点0分，则该任务一小时共有6个任务周期，0点0分为第一个周期，则第3个周期为0点20分。该任务正常运行时，第3个周期任务会在0点22分执行完毕。如果在该周期结束时间点该任务仍未结束运行，则发送报警。
	触发条件	如果**对象类型**选择**独享调度资源组**和**独享数据集成资源组**，触发条件取值为： ○ **利用率**大于某个数值并持续多长时间。 例如：**利用率大于50%并持续15分钟**。 ○ **等资源实例数**大于某个数值并持续多长时间。 例如：**等资源实例数大于70并持续15分钟**。
报警行为	**报警方式**	包括**邮件**、**短信**、**电话钉钉群机器人**和**WebHook**。您可以添加钉钉群机器人接收报警，请参见下文的操作，将报警消息发送到**钉钉群**。如果您需要多个钉钉群接收报警信息，请添加多个Webhook地址。 ○ **超时** 表示从实例任务运行的起始时间点开始监控，到指定的运行时长后，该任务仍未结束运行，则系统发送报警。适用于监控实例任务的运行时长。 ○ **自动重跑后仍出错** 报警方式为短信、邮件、电话时，您可以单击**校验联系方式**，校验手机号/邮箱是否填写正确。 表示从实例任务运行的起始时间点开始监控，如果任务运行出错且自动重跑后仍出错，则系统发送报警。 注意 ○ 您需要购买DataWorks专业版及以上版本，才可以使用电话告警功能。 ○ 如果您选择**报警方式**为**电话**，则需要选中**为了避免短时间内产生大量报警电话，DataWorks会对报警电话进行过滤，同一个用户在20分钟内最多接受到一通报警电话，其余报警电话将被降级为短信，请知悉。** ○ 仅支持钉钉Webhook地址。
	接收人	报警的对象，包括**任务责任人**、**值班表**和**其他**。
疲劳度控制	**最大报警次数**	报警的最大次数，超过设置的次数后，不再产生报警。
	最小报警间隔	两次报警之间的最小时间间隔。
	免打扰时间	在设置的时间段内不会发送报警。

对于重要的任务节点，您还可以单独设置任务节点规则，并定义其他**触发条件**。

3. 数据及时性优化。

   通常，影响数据按时产出的主要原因和优化方式如下表所示。

问题原因	问题优化
计算资源不足 ○ 资源总量不足。例如，资源上限为500，但您提交了需要1000资源的任务。 ○ 资源分配不合理，重要任务未优先分配资源。	扩容计算资源，或让核心计算任务独占资源。
代码执行效率低 ○ 代码冗余。例如，扫描所有分区。 ○ 节点任务配置不合理。例如，出现长尾问题。	分级错峰，高峰时段让低优先级任务延迟启动。

问题原因	问题优化
缺少问题紧急预案，运维人员无法应对。	在任务正式运行前，进行充分的测试。

6.实现窃电用户自动识别教程

6.1. 窃电用户自动识别概述

本教程为您介绍如何通过DataWorks配合机器学习的方式，实现窃电用户的自动识别，保障用户的安全用电。

传统的识别窃电或计量装置故障的方法包括定期巡检、定期校验电表、用户举报窃电等，对人的依赖性较强，且查找窃电漏电的目标不明确。

目前，很多供电局的营销稽查、用电检查和计量工作人员，利用计量异常报警和电能量数据查询功能来在线监控用电情况。通过采集电量异常、负荷异常、线损异常、终端报警、主站报警信息，建立数据分析模型，工作人员可以实时监测窃漏电情况并发现计量装置故障。根据报警事件发生前后，客户计量点有关的电流、电压和负荷等数据情况，构建基于指标的用电异常分析模型，检查是否存在窃电、违章用电及计量装置故障等情况。

虽然上述防窃电漏电的查询方法可以获得用电异常信息，但由于终端误报或漏报过多，无法真正快速精确地定位窃电漏电用户。同时，采用上述方法建模时，需要专家根据其知识和经验，来判断模型各输入指标权重，主观性较强。

现有的电力计量自动化系统，能够采集到各项电流、电压、功率等用电负荷数据及用电异常等终端报警信息。此外，稽查工作人员还可以通过在线稽查系统和现场稽查，查找窃电漏电用户数据并录入系统。

通过上述数据信息，提取出窃电漏电用户的关键特征，构建窃漏电用户的识别模型，即可自动判断用户是否存在窃电漏电行为，降低稽查工作人员的工作量，并保障用户的正常、安全用电。

窃电用户自动识别教程涉及的具体开发流程如下：

1. 准备环境
2. 准备数据
3. 加工数据
4. 数据建模

6.2. 准备环境

为保证您可以顺利完成本次实验，请您首先确保自己云账号已开通大数据计算服务MaxCompute、数据工场DataWorks和机器学习PAI。

前提条件

- 注册阿里云账号，详情请参见。
- 实名认证，详情请参见或。

背景信息

本次实验涉及的阿里云产品如下：

- 大数据计算服务MaxCompute
- 数据工场DataWorks
- 机器学习PAI

开通大数据计算服务MaxCompute

> **说明** 如果您已经开通MaxCompute，请跳过该步骤，直接创建DataWorks工作空间。

1. 登录阿里云官网，单击右上角的**登录**，输入您的阿里云账号和密码。
2. 鼠标悬停至顶部菜单栏中的**产品**，单击**大数据 > 大数据计算与分析 > MaxCompute**，进入MaxCompute产品详情页。
3. 单击**立即开通**。
4. 在购买页面，选择**地域**，并选中**服务协议**，单击**确认订单并支付**。

> **说明**
> - 购买页面默认提供的规格类型为MaxCompute按量计费标准版和DataWorks基础版。
> - MaxCompute的项目管理和查询编辑集成DataWorks的功能，因此需要同时开通DataWorks服务。DataWorks基础版为0元开通，如果您不使用数据集成、不执行调度任务，则不会产生费用。
> - 选择地域时，您需要考虑的最主要因素是MaxCompute与其它阿里云产品之间的关系。例如，ECS所在地域、数据所在地域等。

创建工作空间

1. 使用主账号登录DataWorks控制台。
2. 在**概览**页面，单击右侧的**创建工作空间**。

   您也可以单击左侧导航栏中的**工作空间列表**，切换至相应的区域后，单击**创建工作空间**。
3. 在**创建工作空间**对话框，配置各项参数，单击**下一步**。

分类	参数	描述
基本信息	**工作空间名称**	工作空间名称的长度需要在3~23个字符，以字母开头，且只能包含字母、下划线（_）和数字。
	显示名	显示名不能超过23个字符，只能字母、中文开头，仅包含中文、字母、下划线（_）和数字。
	模式	工作空间模式是DataWorks新版推出的新功能，分为**简单模式**和**标准模式**： ○ **简单模式**：指一个DataWorks工作空间对应一个MaxCompute项目，无法设置开发和生产环境，只能进行简单的数据开发，无法对数据开发流程以及表权限进行强控制。 ○ **标准模式**：指一个DataWorks工作空间对应两个MaxCompute项目，可以设置开发和生产两种环境，提升代码开发规范，并能够对表权限进行严格控制，禁止随意操作生产环境的表，保证生产表的数据安全。 详情请参见简单模式和标准模式的区别。
	描述	对创建的工作空间进行简单描述。

分类	参数	描述
高级设置	能下载select结果	控制数据开发中查询的数据结果是否能够下载，如果关闭无法下载select的数据查询结果。此参数在工作空间创建完成后可以在工作空间配置页面进行修改，详情可参考文档：安全设置。

4. 在**选择引擎**界面，选择相应引擎后，单击**下一步**。

   DataWorks已正式商用，如果该区域没有开通，需要首先开通正式商用的服务。默认选中**数据集成**、**数据开发**、**运维中心**和**数据质量**。

   > **说明** 此处需要同时勾选机器学习PAI和MaxCompute。

5. 进入**引擎详情**页面，配置选购引擎的参数。

分类	参数	描述
MaxCompute	实例显示名称	实例显示名称需要以字母开头，只能包含字母、数字和下划线（_）。
	Quota组切换	Quota用于实现计算资源和磁盘配额。
	MaxCompute数据类型	该选项设置后将在5分钟内生效，数据类型模式的详情请参见数据类型版本说明。如果您不清楚模式的选择，建议与工作空间管理员确认后再进行选择。
	是否加密	您可以设置**不加密**和**加密**。
	MaxCompute项目名称	开发环境的默认名称为DataWorks工作空间的名称_dev，生产环境的默认名称与DataWorks工作空间名称一致。
	MaxCompute访问身份	开发环境的MaxCompute访问身份默认为**任务负责人**，不可以修改。 生产环境的MaxCompute访问身份包括**阿里云主账号**和**阿里云子账号**。

6. 配置完成后，单击**创建工作空间**。

   工作空间创建成功后，即可在**工作空间列表**页面查看相应内容。

6.3. 准备数据

在数据准备阶段，您需要同步原始数据至MaxCompute。

准备数据源

1. 通过RDS创建MySQL实例，获取RDS实例ID。详情请参见创建RDS MySQL实例。
2. 在RDS控制台添加白名单，详情请参见添加白名单。

> **说明** 如果是通过自定义资源组调度RDS的数据同步任务，必须把自定义资源组的机器IP也加入RDS的白名单中。

3. 下载本教程使用的原始数据indicators_data、steal_flag_data和trend_data。
4. 上传原始数据至RDS数据源，详情请参见将Excel的数据导入数据库。

新增数据源

> **说明** 本次实验需要创建MySQL数据源。

1. 进入**数据源管理**页面。
   i. 登录DataWorks控制台。
   ii. 在左侧导航栏，单击**工作空间列表**。
   iii. 选择工作空间所在地域后，单击相应工作空间后的**进入数据集成**。
   iv. 在左侧导航栏，单击**数据源**，进入**工作空间管理 > 数据源管理**页面。
2. 在**数据源管理**页面，单击右上角的**新增数据源**。
3. 在**新增数据源**对话框中，选择数据源类型为**MySQL**。
4. 在**新增MySQL数据源**对话框中，配置各项参数。

参数	描述
数据源类型	当前选择的数据源类型为**MySQL > 阿里云实例模式**。
数据源名称	数据源名称必须以字母、数字、下划线组合，且不能以数字和下划线开头。
数据源描述	对数据源进行简单描述，不得超过80个字符。
适用环境	可以选择**开发**或**生产**环境。 **说明** 仅标准模式工作空间会显示该配置。
地区	选择相应的区域。
RDS实例ID	您可以进入RDS控制台，查看RDS实例ID。
RDS实例主账号ID	实例购买者登录控制台，进入**安全设置**页面，即可查看实例账号ID。
数据库名	数据库的名称。
用户名	数据库对应的用户名。
密码	数据库对应的密码。

5. 单击**测试连通性**。

6.

7. 测试连通性通过后，单击**完成**。

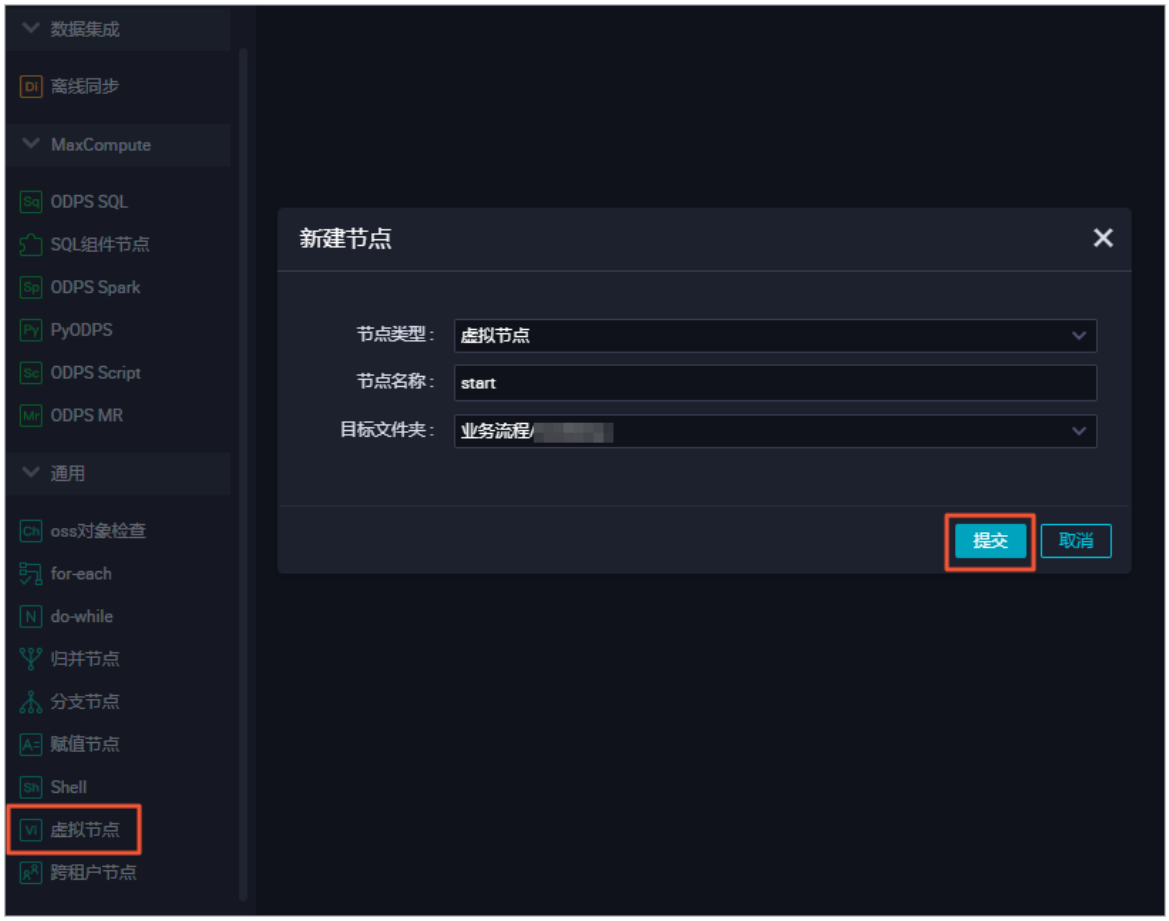
新建业务流程

1. 单击当前页面左上角的图标，选择**全部产品 > 数据开发 > DataStudio（数据开发）**。
2. 右键单击**业务流程**，选择**新建业务流程**。
3. 在**新建业务流程**对话框中，输入**业务名称**和**描述**。

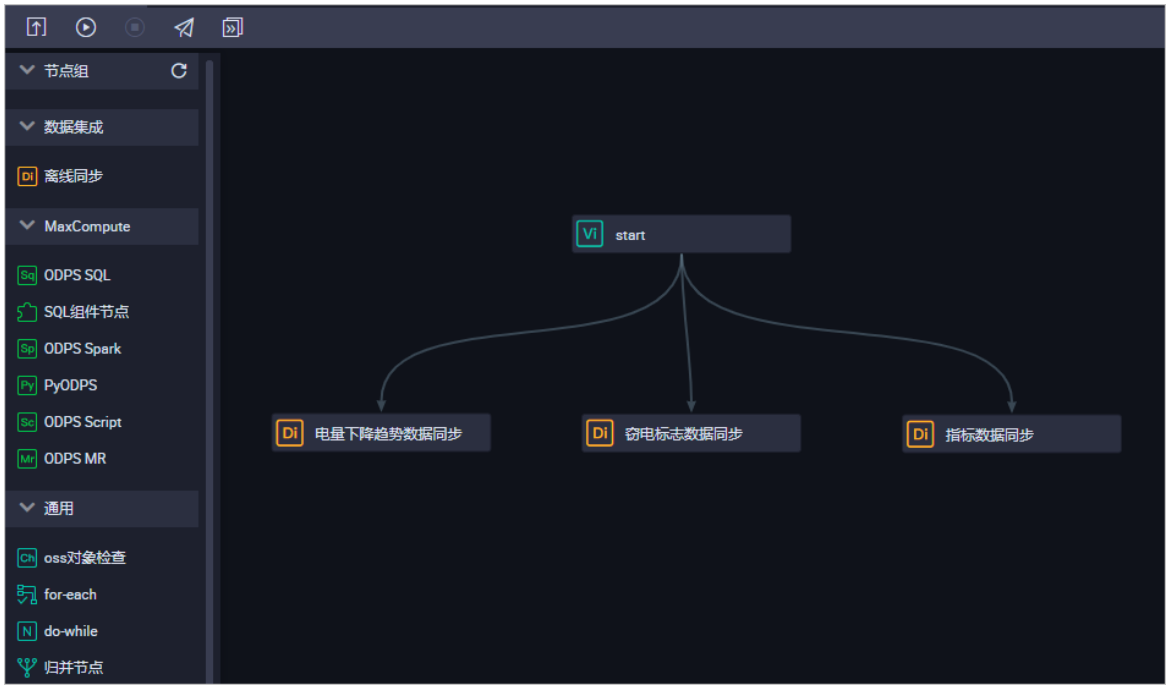
   数据开发

   > **说明** 业务名称的长度不能超过128个字符，且必须是大小写字母、中文、数字、下划线（_）以及小数点（.）。

4. 单击**新建**。
5. 进入业务流程开发面板，并向面板中拖入一个虚拟节点（start）和三个离线同步节点（电量下降趋势数据同步、窃电标志数据同步和指标数据同步）分别填写相应的配置后，单击**提交**。



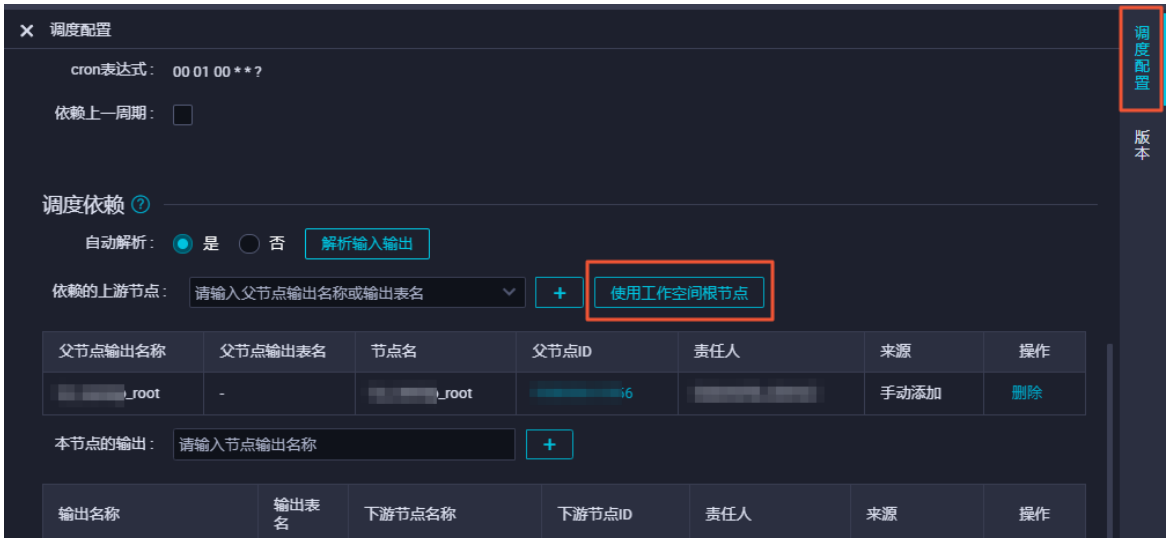
6. 拖拽连线将start节点设置为三个离线同步节点的上游节点。



配置start节点

1. 双击虚拟节点，单击右侧的**调度配置**。
2. 设置start节点的上游节点为工作空间根节点。

   由于新版本给每个节点都设置了输入输出节点，所以需要给start节点设置一个输入。此处设置其上游节点为工作空间根节点，通常命名为工作空间名_root。



3. 配置完成后，单击左上角的图标。

新建表

1. 打开新建的业务流程，单击**MaxCompute**左侧的展开图标，打开MaxCompute。
2. 右键单击**MaxCompute**下的**表**，单击**新建表**。
3. 在**新建表**对话框中，输入**表名**，单击**提交**。

此处需要创建3张表，分别存储同步过来的电量下降趋势数据、指标数据和窃电标志数据（trend_data、indicators_data和steal_flag_data）。

> **说明** 表名不能超过64个字符，且必须以字母开头，不能包含中文或特殊字符。

4. 打开创建的表，单击**DDL模式**，分别输入以下相应的建表语句。

```
--电量下降趋势表
CREATE TABLE trend_data (
    uid bigint,
    trend bigint
)
PARTITIONED BY (dt string);
```

```
--指标数据
CREATE TABLE indicators_data (
    uid bigint,
    xiansun bigint,
    warnindicator bigint
)
COMMENT '*'
PARTITIONED BY (ds string)
LIFECYCLE 36000;
```

```
--窃电标志数据
CREATE TABLE steal_flag_data (
    uid bigint,
    flag bigint
)
COMMENT '*'
PARTITIONED BY (ds string)
LIFECYCLE 36000;
```

5. 建表语句输入完成后，单击**生成表结构**并**确认**覆盖当前操作。
6. 返回建表页面后，在**基本属性**中输入表的中文名。
7. 完成设置后，分别单击**提交到开发环境**和**提交到生产环境**。

配置离线同步节点

1. 配置电量下降趋势数据同步节点。
   i. 双击电量下降趋势数据同步节点，进入节点配置页面。

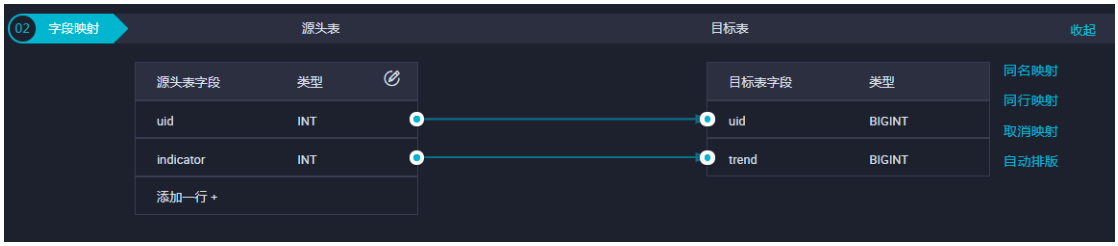
ii. 选择数据来源。

参数	描述
数据源	选择MySQL > workshop。
表	选择MySQL数据源中的表trending。
数据过滤	您将要同步数据的筛选条件，暂时不支持limit关键字过滤。SQL语法与选择的数据源一致，此处可以不填。
切分键	读取数据时，根据配置的字段进行数据分片，实现并发读取，可以提升数据同步效率。此处可以不填。

iii. 选择数据去向。

参数	描述
数据源	选择**ODPS > odps_first**。
表	选择ODPS数据源中的表trend_data。
分区信息	输入要同步的分区列，此处默认为 `dt=${bdp.system.bizdate}` 。
清理规则	选择**写入前清理已有数据**。
空字符串作为null	选择**否**。

iv. 配置字段映射。



v. 配置**通道控制**。

参数	描述
任务期望最大并发数	数据同步任务内，可以从源并行读取或并行写入数据存储端的最大线程数。向导模式通过界面化配置并发数，指定任务所使用的并行度。
同步速率	设置同步速率可以保护读取端数据库，以避免抽取速度过大，给源库造成太大的压力。同步速率建议限流，结合源库的配置，请合理配置抽取速率。
错误记录数	错误记录数，表示脏数据的最大容忍条数。

vi. 确认当前节点的配置无误后，单击左上角的图标。

提交业务流程

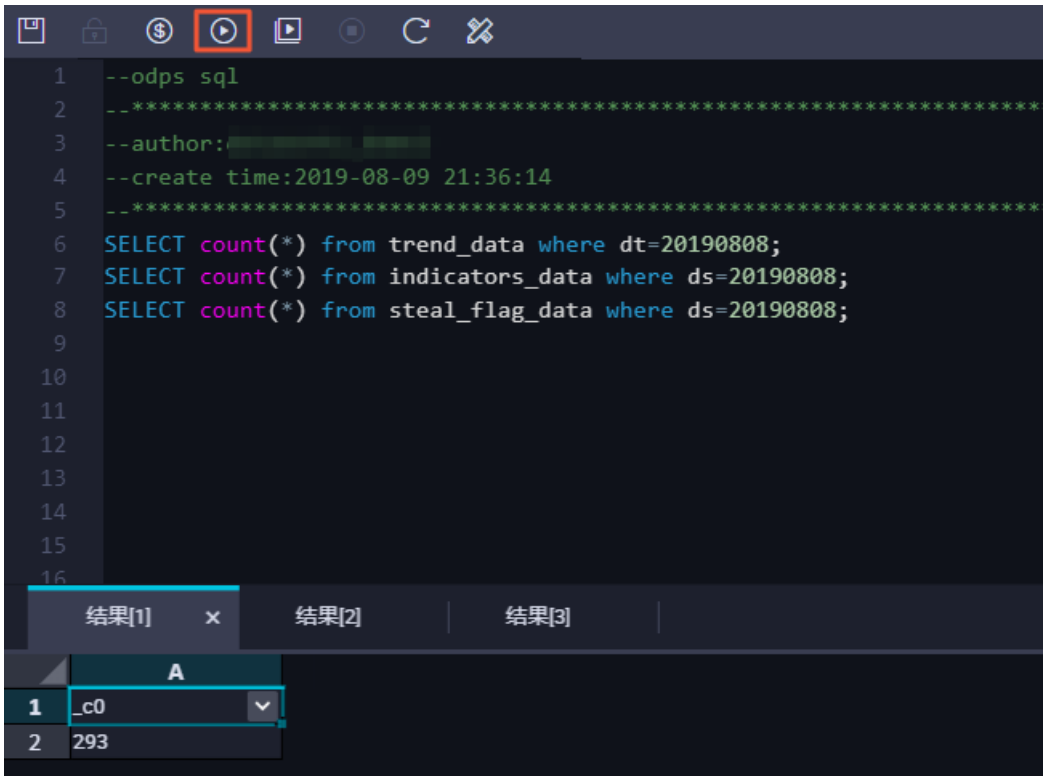
1. 打开业务流程配置面板，单击左上角的进行提交。
2. 选择**提交**对话框中需要提交的节点，输入**备注**，勾选**忽略输入输出不一致的告警**。

3. 单击**提交**，待显示**提交成功**即可。

确认数据是否成功导入MaxCompute

1. 在**数据开发**页面的左侧导航栏，单击**临时查询**，进入**临时查询**面板。
2. 右键单击**临时查询**，选择**新建节点 > ODPS SQL**。

3. 编写并执行SQL语句，查看导入表trend_data、indicators_data和steal_flag_data的记录数。



SQL语句如下所示，其中分区列需要更新为业务日期。例如，任务运行的日期为20190809，则业务日期为201900808。

```
--查看是否成功写入MaxCompute
SELECT count(*) from trend_data where dt=业务日期;
SELECT count(*) from indicators_data where ds=业务日期;
SELECT count(*) from steal_flag_data where ds=业务日期;
```

后续步骤

现在，您已经学习了如何通过数据同步采集数据，您可以继续下一个教程。在该教程中，您将学习如何对采集的数据进行计算与分析。详情请参见数据加工。

6.4. 加工数据

本文为您介绍如何通过DataWorks加工采集至MaxCompute的数据，并获取清洗后的数据。

前提条件

开始本文的操作前，请首先完成准备数据中的操作。

新建表

1. 进入**数据开发**页面。
   i. 登录DataWorks控制台。
   ii. 在左侧导航栏，单击**工作空间列表**。
   iii. 选择工作空间所在地域后，单击相应工作空间后的**进入数据开发**。
2. 在**数据开发**页面，单击相应业务流程左侧的图标，展开该业务流程。

3. 右键单击MaxCompute，选择**新建 > 表**。
4. 在**新建表**对话框中，输入**表名**，单击**提交**。

   > **注意** 表名必须以字母开头，不能包含中文或特殊字符，且不能超过64个字符。

   此处需要创建的数据表，如下所示：

   - 创建三张表，分别存储同步过来的电量下降趋势数据、指标数据和窃电标志数据清洗之后的数据（clean_trend_data、clean_indicators_data和clean_steal_flag_data）。
   - 创建表data4ml，存储汇聚后的数据。
5. 打开创建的表，单击**DDL模式**，分别输入以下相应的建表语句。

```
--清洗后的电量下降趋势数据
CREATE TABLE clean_trend_data (
    uid bigint,
    trend bigint
)
PARTITIONED BY (dt string)
LIFECYCLE 7;
```

```
--清洗后的指标数据
CREATE TABLE clean_indicators_data (
    uid bigint,
    xiansun bigint,
    warnindicator bigint
)
COMMENT '*'
PARTITIONED BY (ds string)
LIFECYCLE 36000;
```

```
--清洗后的窃电标志数据
CREATE TABLE clean_steal_flag_data (
    uid bigint,
    flag bigint
)
COMMENT '*'
PARTITIONED BY (ds string)
LIFECYCLE 36000;
```

```
--汇聚后的数据
CREATE TABLE data4ml (
    uid bigint,
    trend bigint,
    xiansun bigint,
    warnindicator bigint,
    flag bigint
)
COMMENT '*'
PARTITIONED BY (ds string)
LIFECYCLE 36000;
```

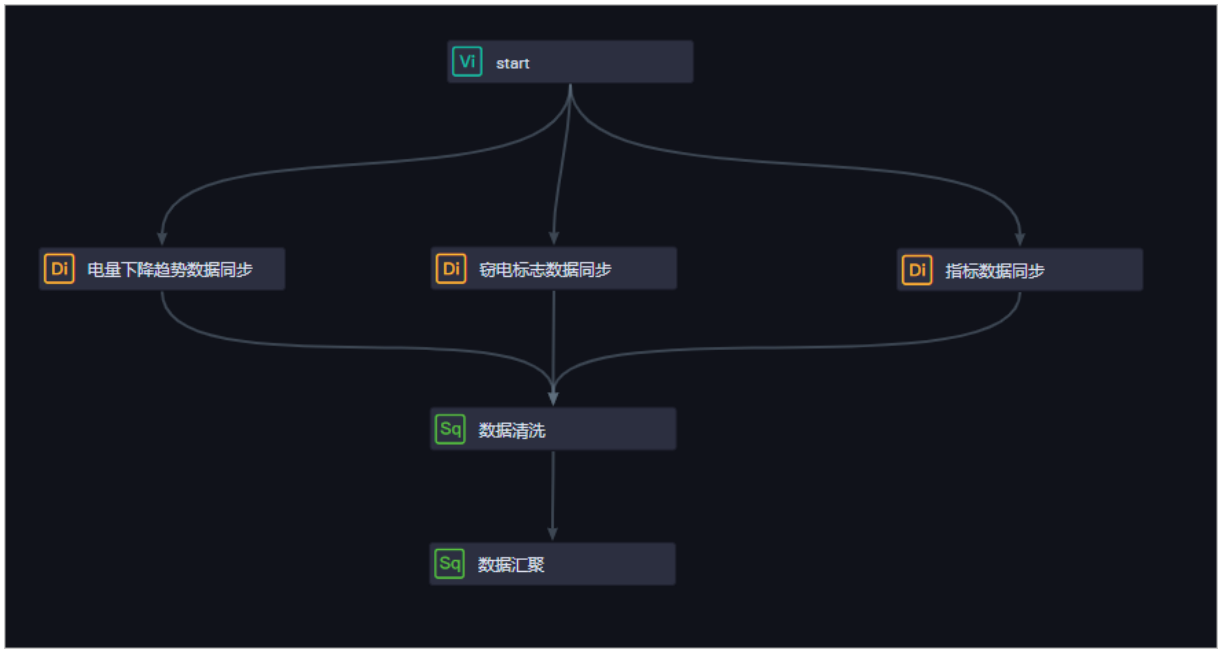
6. 建表语句输入完成后，单击**生成表结构**并**确认**覆盖当前操作。

7. 返回建表页面后，在**基本属性**中输入表的中文名。
8. 完成设置后，分别单击**提交到开发环境**和**提交到生产环境**。

设计业务流程

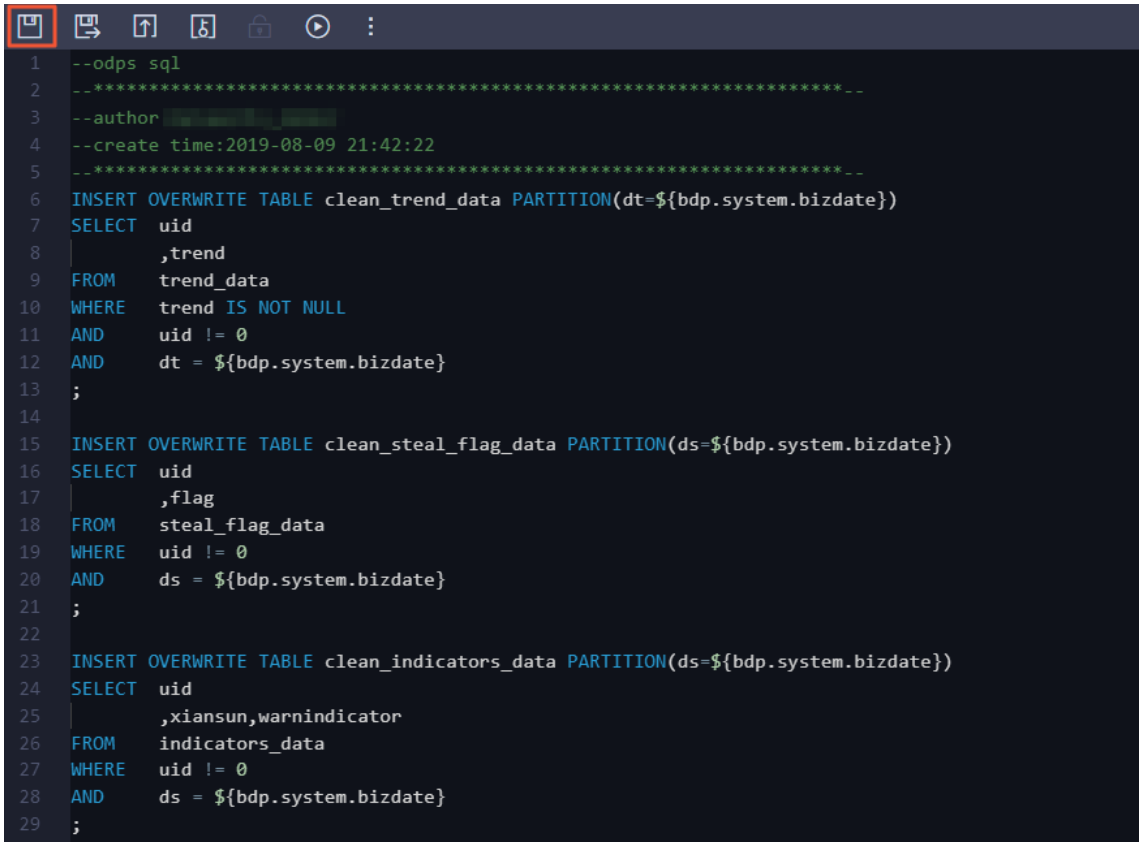
业务流程的新建及依赖关系的配置请参见新建业务流程。

进入业务流程开发面板，并向面板中拖入两个ODPS SQL节点，依次命名为**数据清洗**和**数据汇聚**，并配置如下图所示的依赖关系。



配置ODPS SQL节点

- 配置数据清洗节点。
  i. 双击数据清洗节点，进入节点配置页面。
  ii. 编写处理逻辑。



```
--odps sql
--********************************************************************--
--author
--create time:2019-08-09 21:42:22
--********************************************************************--
INSERT OVERWRITE TABLE clean_trend_data PARTITION(dt=${bdp.system.bizdate})
SELECT  uid
        ,trend
FROM    trend_data
WHERE   trend IS NOT NULL
AND     uid != 0
AND     dt = ${bdp.system.bizdate}
;

INSERT OVERWRITE TABLE clean_steal_flag_data PARTITION(ds=${bdp.system.bizdate})
SELECT  uid
        ,flag
FROM    steal_flag_data
WHERE   uid != 0
AND     ds = ${bdp.system.bizdate}
;

INSERT OVERWRITE TABLE clean_indicators_data PARTITION(ds=${bdp.system.bizdate})
SELECT  uid
        ,xiansun,warnindicator
FROM    indicators_data
WHERE   uid != 0
AND     ds = ${bdp.system.bizdate}
;
```

SQL逻辑如下所示。

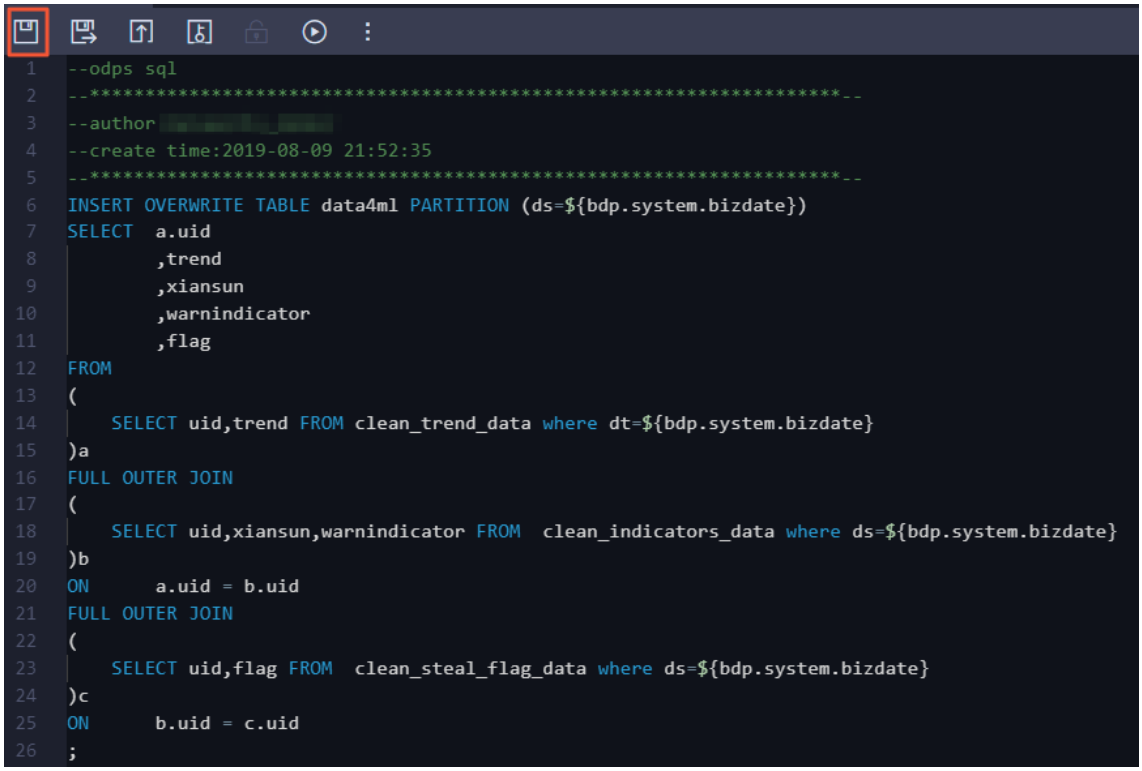
```
INSERT OVERWRITE TABLE clean_trend_data PARTITION(dt=${bdp.system.bizdate})
SELECT  uid
        ,trend
FROM    trend_data
WHERE   trend IS NOT NULL
AND     uid != 0
AND     dt = ${bdp.system.bizdate}
;
INSERT OVERWRITE TABLE clean_steal_flag_data PARTITION(ds=${bdp.system.bizdate})
SELECT  uid
        ,flag
FROM    steal_flag_data
WHERE   uid != 0
AND     ds = ${bdp.system.bizdate}
;
INSERT OVERWRITE TABLE clean_indicators_data PARTITION(ds=${bdp.system.bizdate})
SELECT  uid
        ,xiansun,warnindicator
FROM    indicators_data
WHERE   uid != 0
AND     ds = ${bdp.system.bizdate}
;
```

iii. 单击工具栏中的图标。

- 配置数据汇聚节点。

i. 双击数据汇聚节点，进入节点配置页面。

ii. 编写处理逻辑。



```
--odps sql
--********************************************************************--
--author [redacted]
--create time:2019-08-09 21:52:35
--********************************************************************--
INSERT OVERWRITE TABLE data4ml PARTITION (ds=${bdp.system.bizdate})
SELECT  a.uid
        ,trend
        ,xiansun
        ,warnindicator
        ,flag
FROM
(
    SELECT uid,trend FROM clean_trend_data where dt=${bdp.system.bizdate}
)a
FULL OUTER JOIN
(
    SELECT uid,xiansun,warnindicator FROM  clean_indicators_data where ds=${bdp.system.bizdate}
)b
ON      a.uid = b.uid
FULL OUTER JOIN
(
    SELECT uid,flag FROM  clean_steal_flag_data where ds=${bdp.system.bizdate}
)c
ON      b.uid = c.uid
;
```

SQL逻辑如下所示。

```
INSERT OVERWRITE TABLE data4ml PARTITION (ds=${bdp.system.bizdate})
SELECT  a.uid
        ,trend
        ,xiansun
        ,warnindicator
        ,flag
FROM
(
    SELECT uid,trend FROM clean_trend_data where dt=${bdp.system.bizdate}
)a
FULL OUTER JOIN
(
    SELECT uid,xiansun,warnindicator FROM  clean_indicators_data where ds=${bdp.syste
m.bizdate}
)b
ON      a.uid = b.uid
FULL OUTER JOIN
(
    SELECT uid,flag FROM  clean_steal_flag_data where ds=${bdp.system.bizdate}
)c
ON      b.uid = c.uid
;
```

iii. 单击工具栏中的图标。

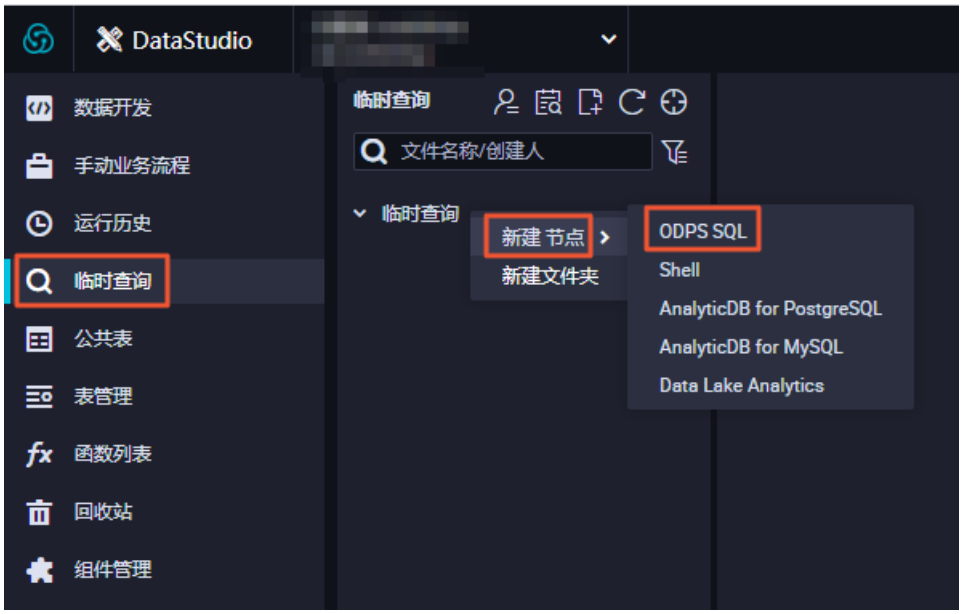
提交业务流程

1. 打开业务流程配置面板，单击工具栏中的图标。
2. 选择**提交**对话框中需要提交的节点，输入**备注**，并选中**忽略输入输出不一致的告警**。

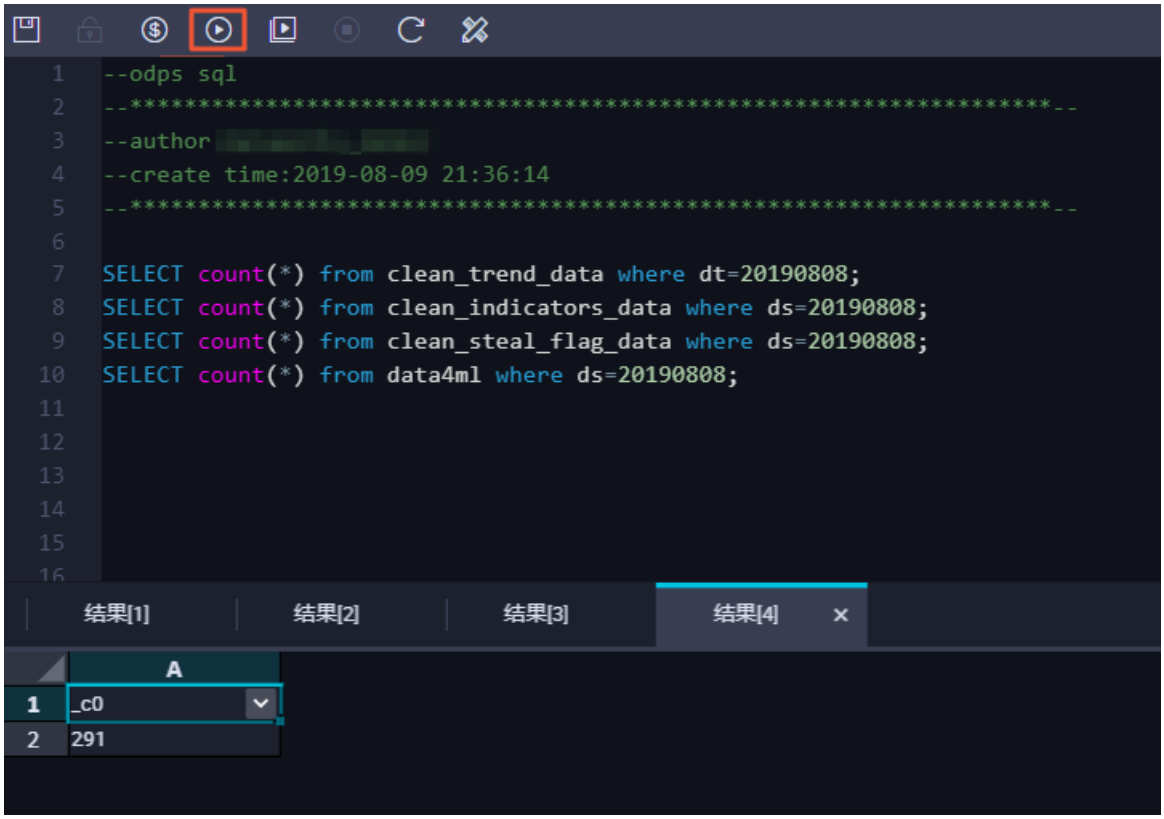
3. 单击**提交**，待显示**提交成功**即可。

运行业务流程

1. 打开业务流程配置面板，单击工具栏中的图标。
2. 在左侧导航栏，单击**临时查询**。
3. 在**临时查询**页面，右键单击**临时查询**，选择**新建节点 > ODPS SQL**。



4. 编写并执行SQL语句，查看导入表clean_trend_data、clean_indicators_data、clean_steal_flag_data和data4ml的记录数。



SQL语句如下所示，其中分区列需要更新为业务日期。例如，任务运行的日期为20190809，则业务日期为20190808。

```
--查看是否成功写入MaxCompute
SELECT count(*) from clean_trend_data where dt=业务日期;
SELECT count(*) from clean_indicators_data where ds=业务日期;
SELECT count(*) from clean_steal_flag_data where ds=业务日期;
SELECT count(*) from data4ml where ds=业务日期;
```

发布业务流程

提交业务流程后，表示任务已进入开发环境。由于开发环境的任务不会自动调度，您需要将配置完成的任务发布至生产环境。

> **说明** 将任务发布至生产环境前，您需要对代码进行测试，确保其正确性。

1. 打开业务流程配置面板，单击工具栏中的图标。
2. 在**创建发布包**页面，选中待发布的任务，单击**添加到待发布**。

3. 进入右上角的**待发布列表**，单击**全部打包发布**。

4. 在**发布包列表**页面查看已发布的内容。

在生产环境运行任务

1. 任务发布成功后，单击右上角的**运维中心**。
2. 选择**周期任务运维 > 周期任务**中的相应节点。

3. 右键单击DAG图中的start节点，选择**补数据 > 当前节点及下游节点**。

4. 选中需要补数据的任务，并**选择业务日期**。

5. 单击**确定**
6. 在**补数据实例**页面，单击**刷新**，直至SQL任务都运行成功即可。

后续步骤

现在，您已经学习了如何创建SQL任务、如何处理原始数据。您可以继续下一个教程，学习如何通过机器学习，载入处理好的数据并构建窃漏电用户的识别模型。详情请参见数据建模。

6.5. 数据建模

本文将为您介绍如何载入DataWorks中处理好的数据到机器学习中，构建窃漏电用户的识别模型。

前提条件

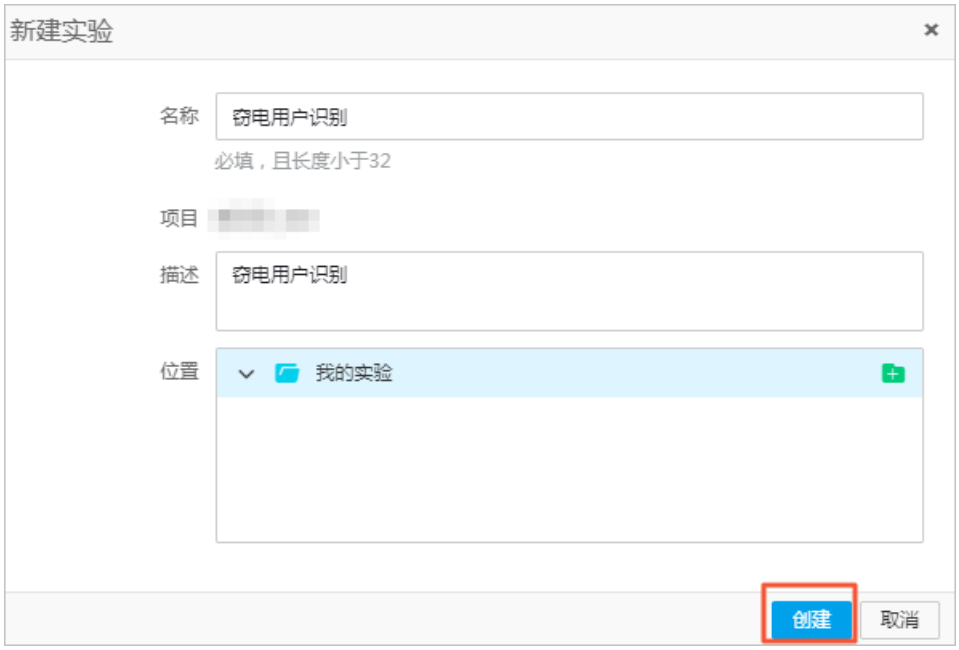
开始本文的操作前，请首先完成加工数据中的操作。

新建实验

1. 进入机器学习控制台，单击左侧导航栏中的**Studio-可视化建模**。
2. 单击相应工作空间后的**进入机器学习**。

3. 单击左侧菜单栏中的**实验**，右键单击**我的实验**，选择**新建空白实验**。

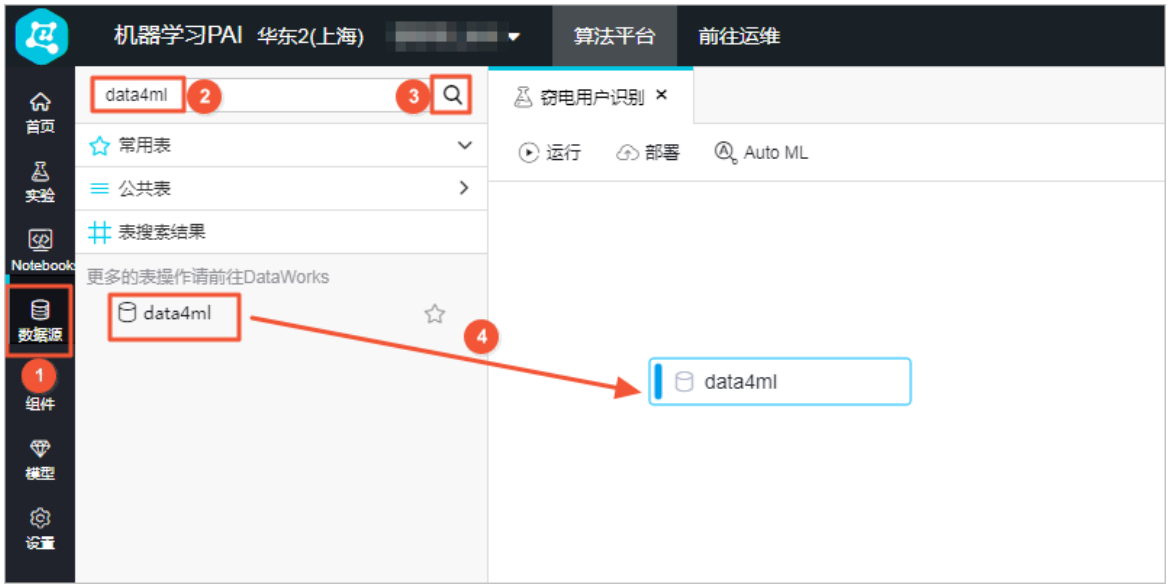
4. 填写**新建实验**对话框中的**名称**和**描述**。



5. 单击**创建**。

载入数据集

1. 单击左侧导航栏中的**数据源**。
2. 在搜索框输入加工数据中最终输出的data4ml表，单击搜索图标。
3. 拖拽**表搜索结果**下的data4ml表至右侧画布。



右键单击读数据表，选择**查看数据**，即可查看载入的结果数据。数据包括电量趋势下降指标、线损指标和告警类指标数量等窃电漏电指标，以及用户是否真实窃电漏电的数据。

数据探查 - data4ml - (仅显示前一百条)

序号	uid	trend	xiansun	warnindicator	flag	ds
1	1	4	1	1	1	20190808
2	2	4	0	4	1	20190808
3	3	2	1	1	1	20190808
4	4	9	0	0	0	20190808
5	5	3	1	0	0	20190808
6	6	2	0	0	0	20190808
7	7	5	0	2	1	20190808
8	8	3	1	3	1	20190808
9	9	3	0	0	0	20190808
10	10	4	1	0	0	20190808
11	11	10	1	2	1	20190808
12	12	10	1	3	1	20190808
13	13	2	0	3	0	20190808
14	14	4	0	2	0	20190808
15	15	3	0	0	0	20190808
16	16	0	0	3	0	20190808
17	17	9	0	3	1	20190808

复制 关闭

进行数据探索

1. 相关性分析

   i. 单击左侧导航栏中的**组件**，拖拽**统计分析 > 相关系数矩阵**至右侧画布。

   ii. 连线**读数据表**中ODPS源的输出和**相关系数矩阵**的输入。

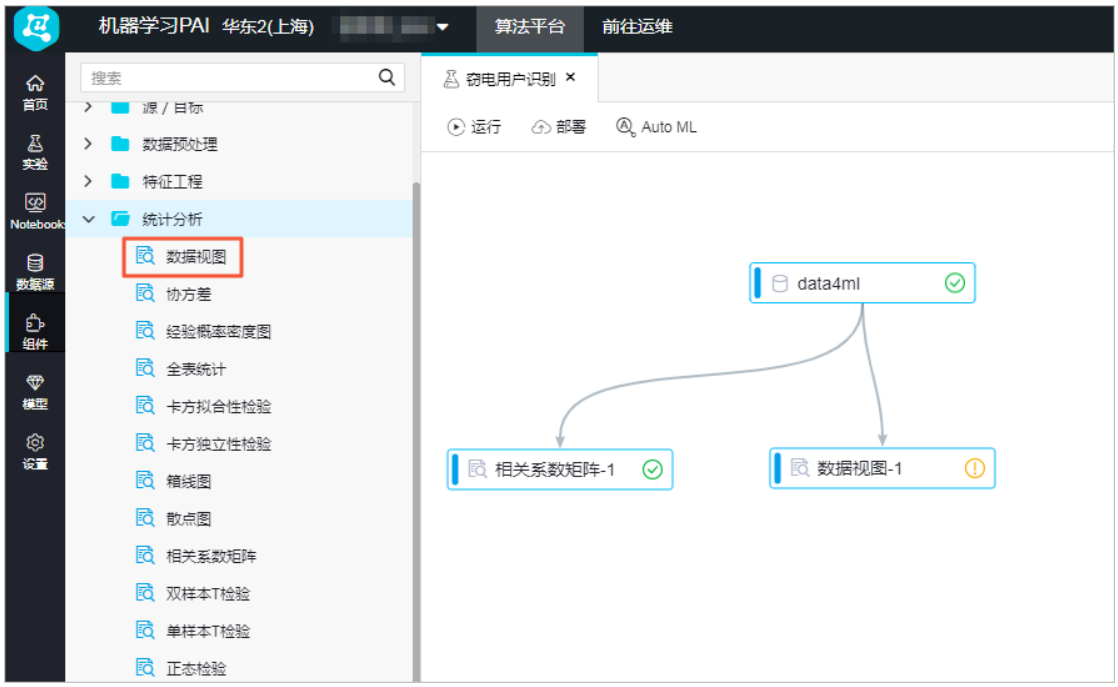
   iii. 右键单击**相关系数矩阵**，选择**从此处开始执行**。

iv. 待运行完成后，右键单击**相关系数矩阵**，选择**查看分析报告**。

如相关系数矩阵图所示，3个窃电漏电指标本身和最终是否为窃电用户的关系都不太明显，即用于判断用户是否为窃电用户的特征并不具有单一性。

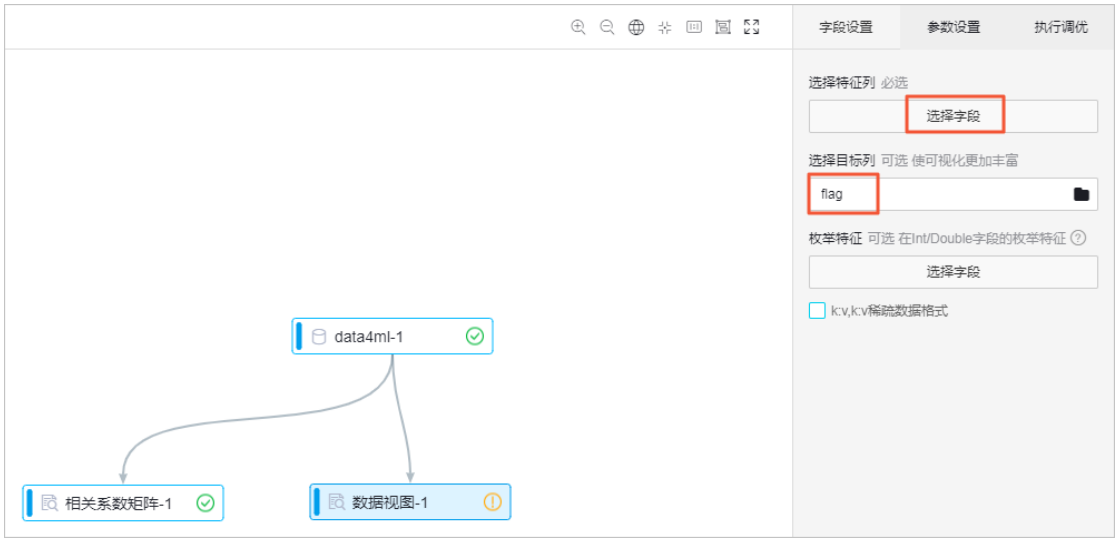
2. 特征分析

i. 单击左侧导航栏中的**组件**，拖拽**统计分析 > 数据视图**至右侧画布。

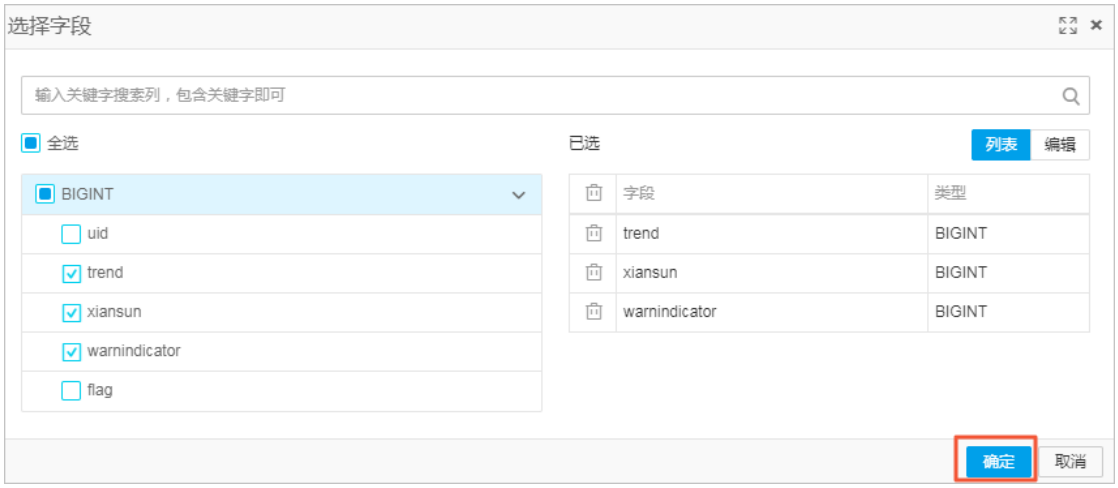


ii. 连线**读数据表**中ODPS源的输出和**数据视图**的输入。

iii. 双击**数据视图**，选择右侧的**字段设置 > 选择特征列**，单击**选择字段**，并**选择目标列**为flag。

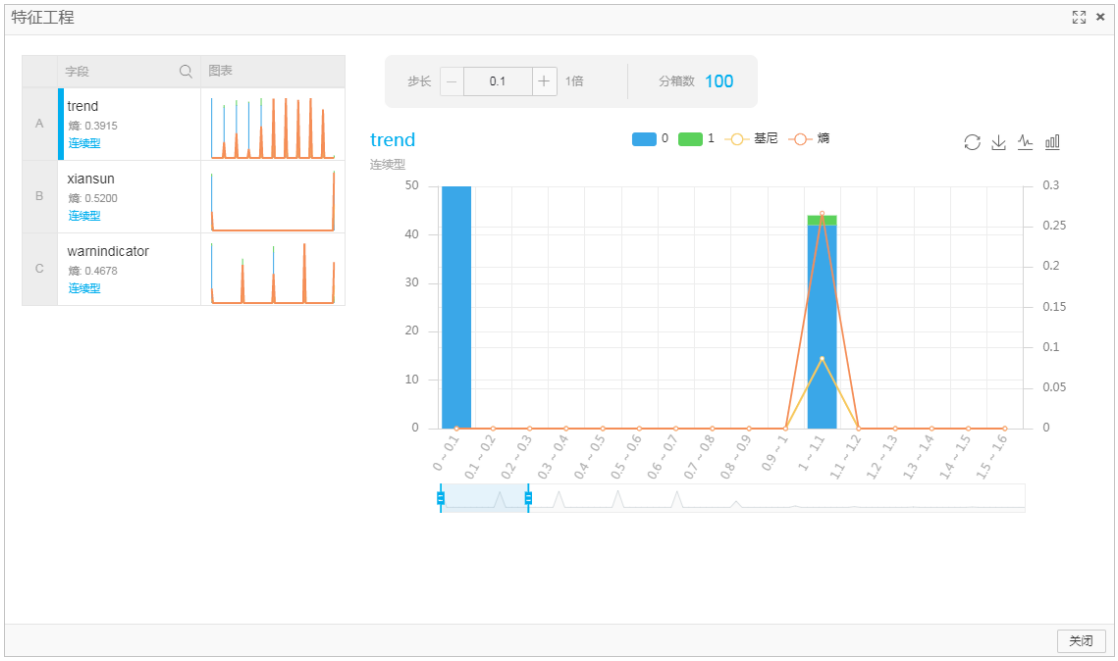


iv. 在**选择字段**对话框中，选择trend、xiansun和warnindicator3个字段，单击**确定**。



v. 右键单击 **数据视图**，选择**从此处开始执行**。

vi. 执行完成后，选择**查看分析报告**，即可查看各个特征和标签列在数据分布上的关系。

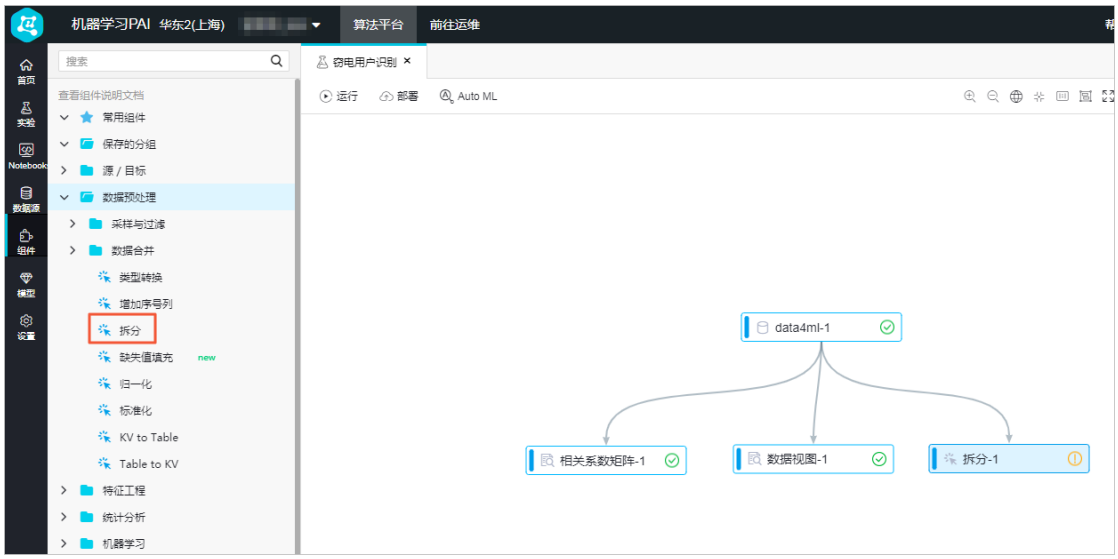


进行数据建模

完成简单的探索性分析之后，即可开始选择合适的算法模型进行数据建模。

1. 通过**拆分**组件，将数据分为训练集和测试集。

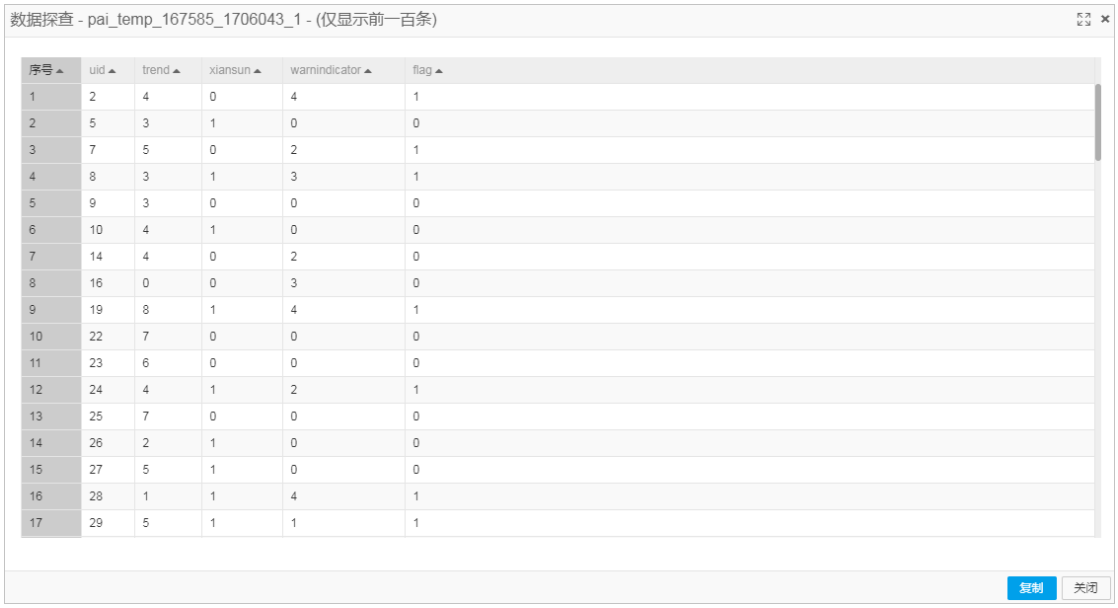
   i. 单击左侧导航栏中的**组件**，拖拽**数据预处理 > 拆分**至右侧画布。



   ii. 连线**读数据表**中ODPS源的输出和**拆分**的输入。

   iii. 右键单击**拆分**，选择**从此处开始执行**。

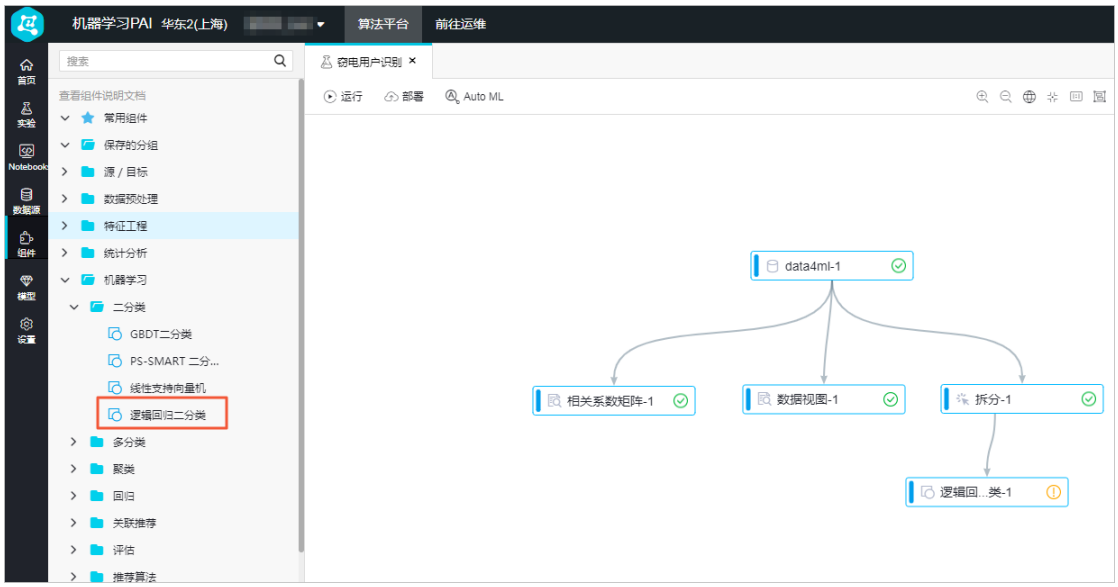
iv. 待运行完成后，右键单击**拆分**，选择**查看数据 > 查看输出桩**。



序号	uid	trend	xiansun	warnindicator	flag
1	2	4	0	4	1
2	5	3	1	0	0
3	7	5	0	2	1
4	8	3	1	3	1
5	9	3	0	0	0
6	10	4	1	0	0
7	14	4	0	2	0
8	16	0	0	3	0
9	19	8	1	4	1
10	22	7	0	0	0
11	23	6	0	0	0
12	24	4	1	2	1
13	25	7	0	0	0
14	26	2	1	0	0
15	27	5	1	0	0
16	28	1	1	4	1
17	29	5	1	1	1

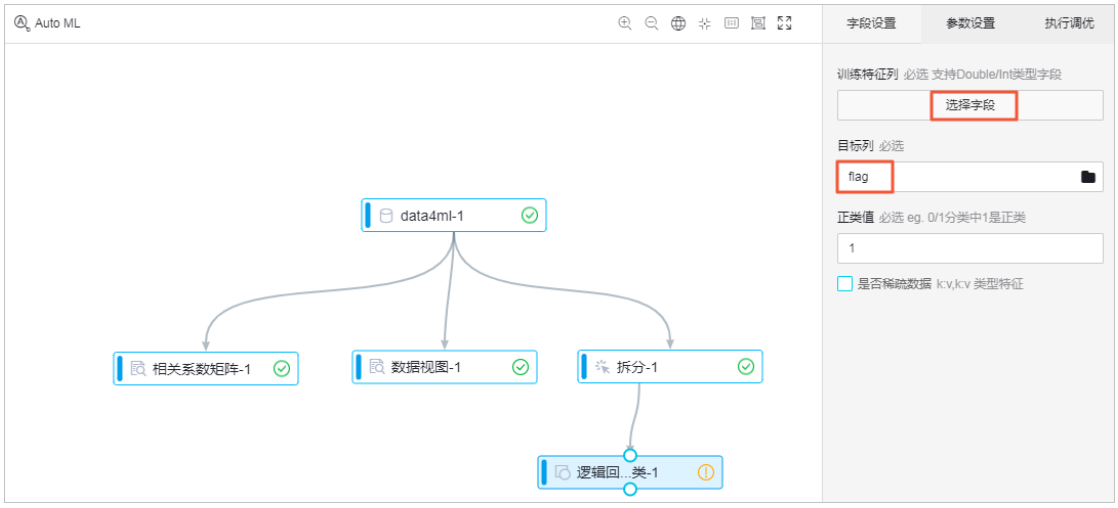
2. 通过**逻辑回归二分类**组件，对数据进行回归建模。

i. 单击左侧导航栏中的**组件**，拖拽**机器学习 > 二分类 > 逻辑回归二分类**至右侧画布。

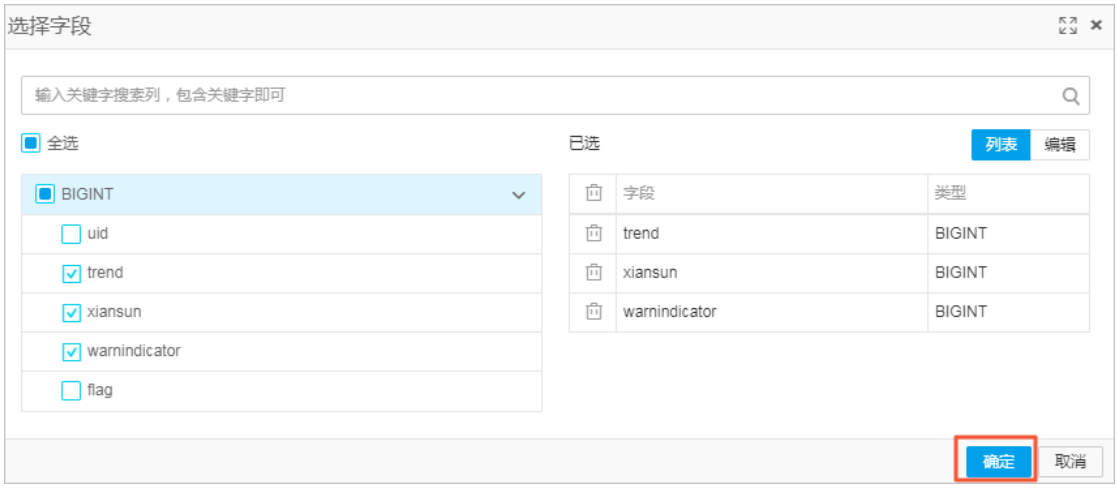


ii. 连线**拆分**中的输出表1和**逻辑回归二分类**的训练表。

iii. 双击**逻辑回归二分类**，选择右侧的**字段设置 > 选择特征列**，单击**选择字段**，并**选择目标列**为flag。

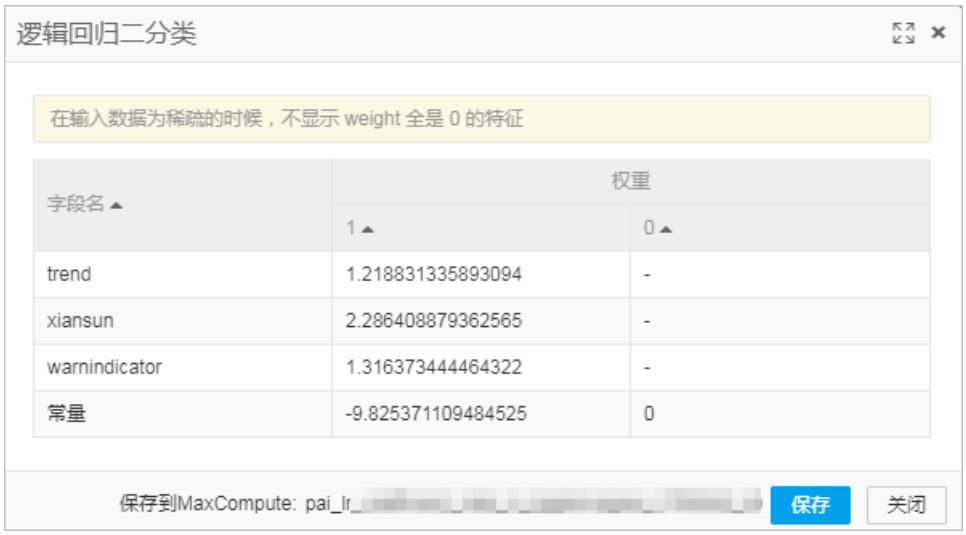


iv. 在**选择字段**对话框中，选择trend、xiansun和warnindicator3个字段，单击**确定**。



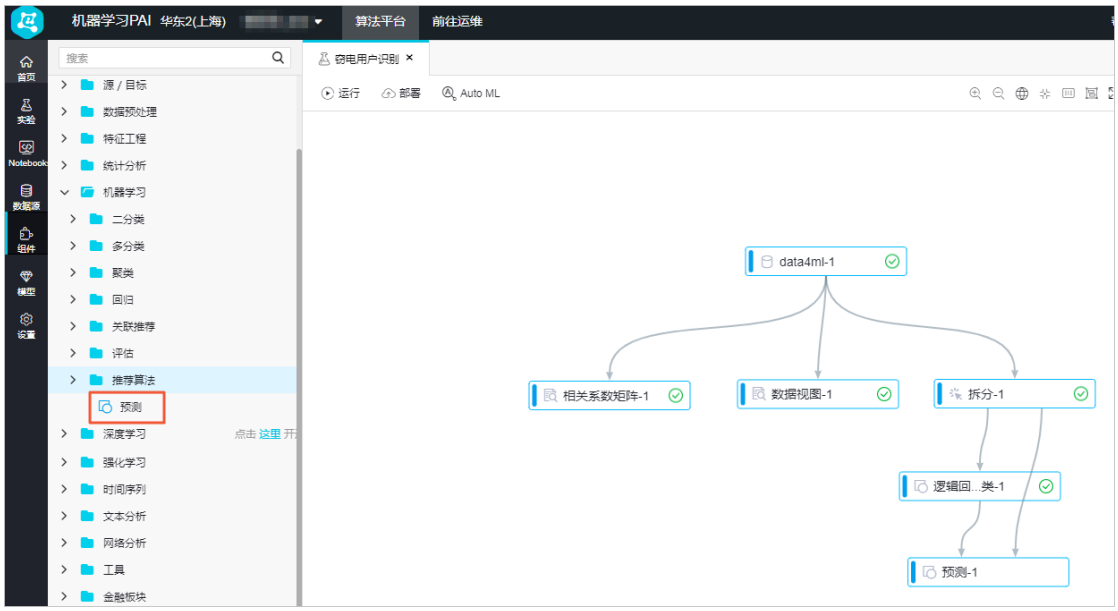
v. 右键单击 **逻辑回归二分类**，选择**从此处开始执行**。

vi. 执行完成后，选择**模型选项 > 查看模型**，即可查看数据模型。



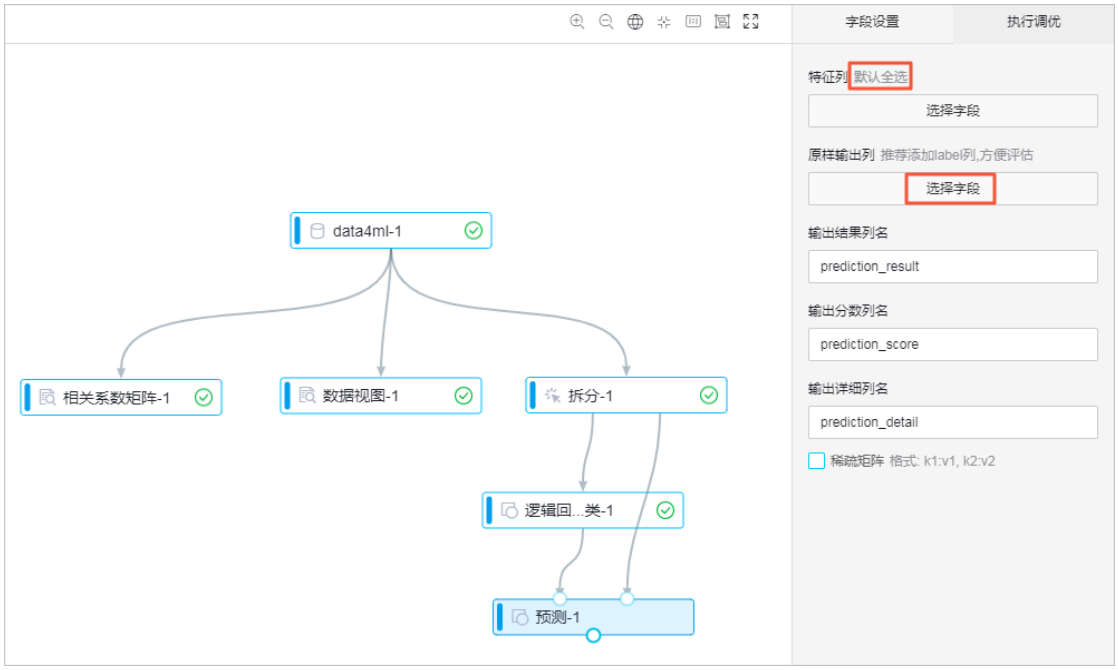
预测和评估回归模型

1. 通过**预测**组件，预测该模型在测试数据集上的效果。
   i. 单击左侧导航栏中的**组件**，拖拽**机器学习 > 预测**至右侧画布。



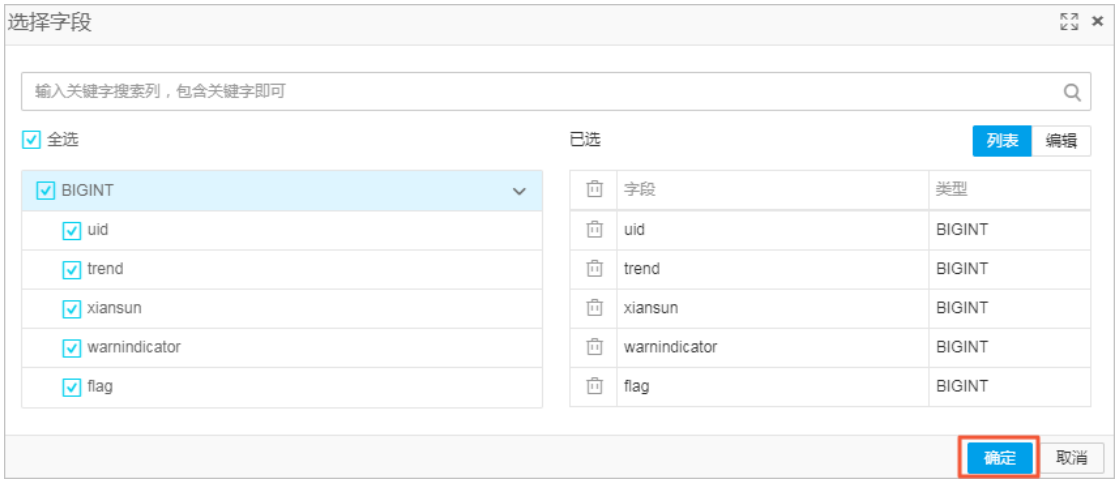
   ii. 连线**逻辑回归二分类**中的逻辑回归模型和**预测**中的模型结果输入。连线**拆分**中的输出表2和**预测**的预测数据输入。

   iii. 双击**预测**，进行右侧的**字段设置**。



特征列默认全选，单击**原样输出列**下的**选择字段**。

iv. 在**选择字段**对话框中，全选5个字段，单击**确定**。

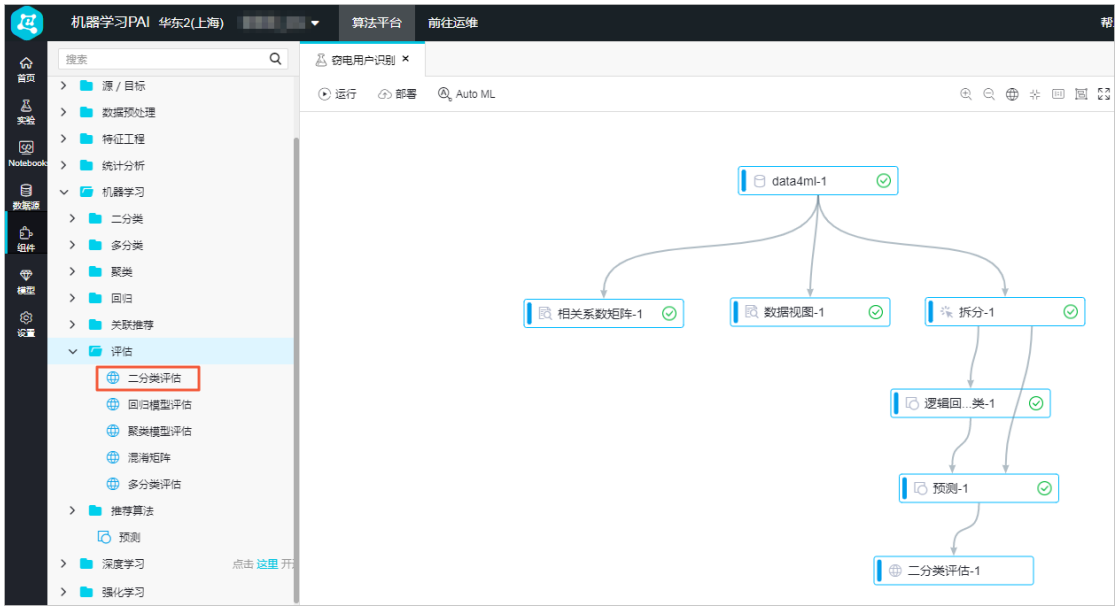


v. 右键单击**预测**，选择**从此处开始执行**。

vi. 执行完成后，选择**查看数据**。

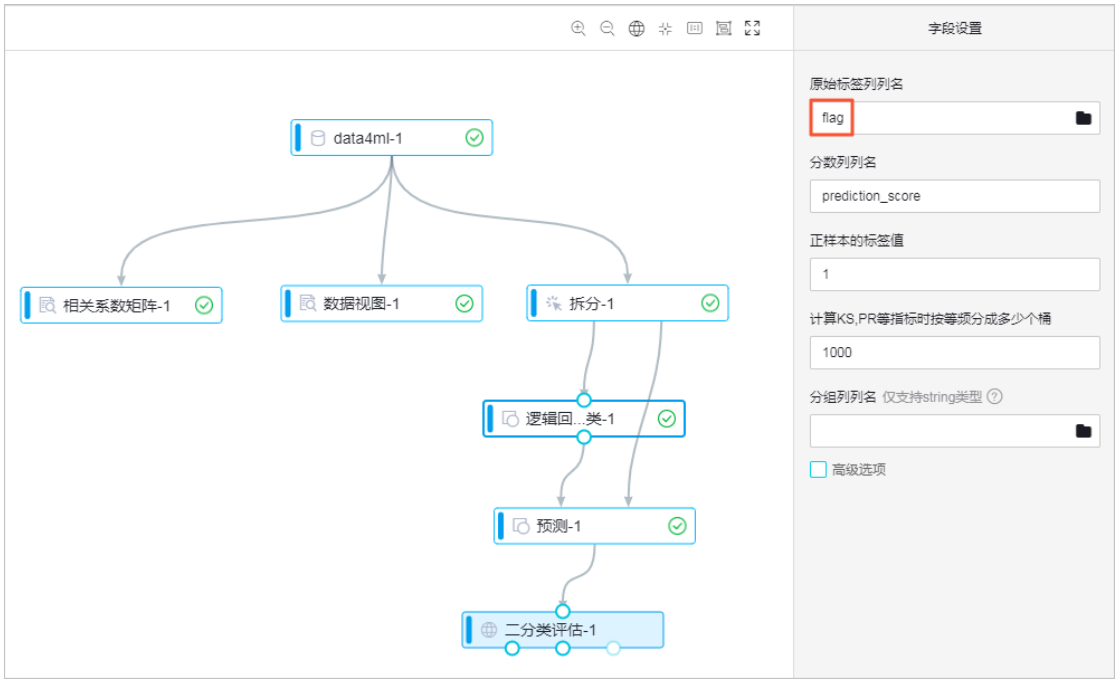
2. 通过**二分类评估**组件，获取模型效果。

i. 单击左侧导航栏中的**组件**，拖拽**机器学习 > 评估 > 二分类评估**至右侧画布。



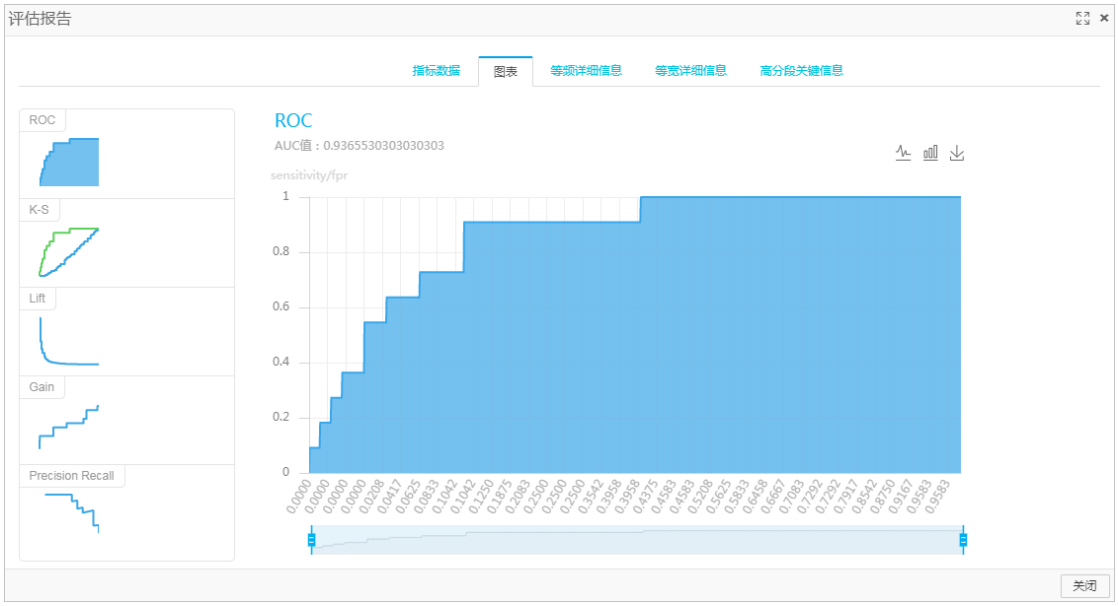
ii. 连线**预测**中的预测结果输出和**二分类评估**中的输入。

iii. 双击**二分类评估**，选择右侧的**字段设置 > 原始标签列列名**为flag。



iv. 右键单击**二分类评估**，选择**从此处开始执行**。

v. 执行完成后，选择**查看评估报告**，即可查看模型效果。



后续步骤

至此，您已通过机器学习PAI完成了用户窃电行为的识别。您还可以通过EAS在线部署，将该服务部署为可在线调用的服务，提供用户窃电行为的在线识别服务。

7.对接使用CDH

DataWorks 提供了与CDH（Cloudera's Distribution Including Apache Hadoop，以下简称CDH) 集群对接的能力，在保留继续使用CDH集群作为存储和计算引擎的前提下，您可以使用DataWorks的任务开发、调度、数据地图（元数据管理）和数据质量等一系列的数据开发和治理功能。本文为您介绍如何对接使用CDH。

前提条件

- 已部署CDH。

  支持非阿里云ECS环境部署的CDH，但需要确保部署CDH集群的ECS和阿里云网络可达。通常您可以使用高速通道、VPN等网络连通方案，来保障网络可达。

- 已开通DataWorks服务并创建好对接使用CDH的工作空间。

  > **说明** 对接使用CDH的工作空间无需绑定计算引擎，在创建工作空间时可跳过选择引擎步骤，其他步骤的操作详情可参见创建工作空间。

- 拥有一个有工作空间的管理员权限的账号，在DataWorks中新增CDH引擎配置的操作仅空间管理员可操作。为账号授权空间管理员权限的操作可参见。成员及角色管理
- 已购买并创建DataWorks的独享调度资源组。详情可参见独享资源组模式。

在DataWorks中对接使用CDH引擎时，主要配置流程为：

1. Step1：获取CDH集群配置信息
2. Step2：配置网络联通
3. Step3：在DataWorks中新增CDH集群配置

对接配置完成后，您可在DataWorks上开发CDH引擎的数据开发任务并运行，并在运行后通过DataWorks的运维中心查看任务运行情况。详情可参见使用DataWorks进行数据开发和运维监控配置。

同时您可使用DataWorks的数据质量、数据地图功能，进行数据和任务管理。详情可参见数据质量规则配置和数据地图配置。

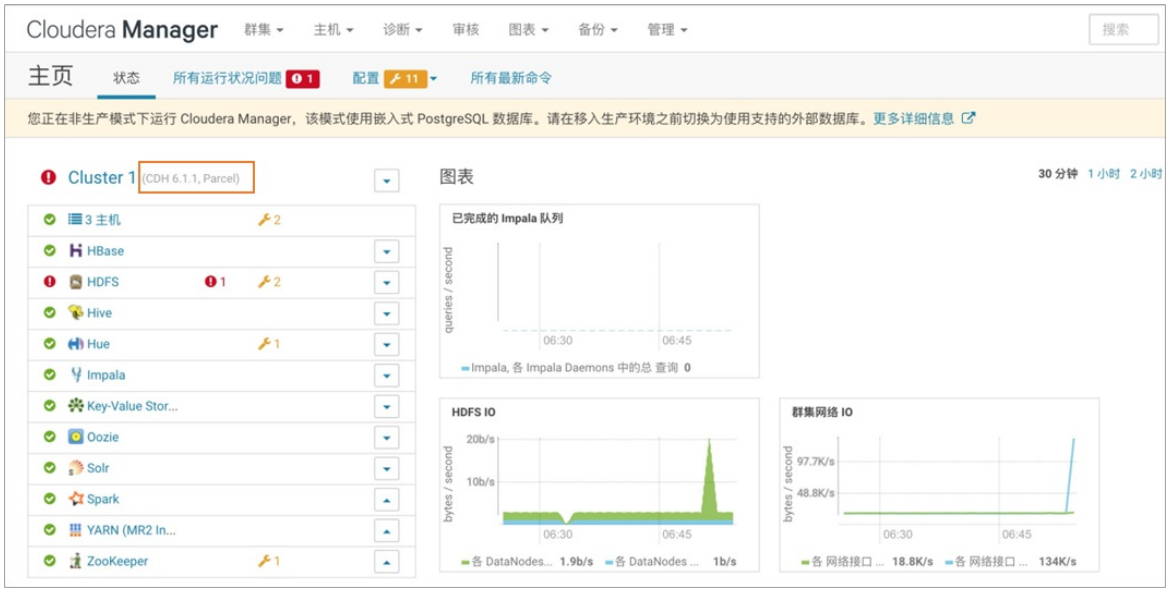
使用限制

- 在DataWorks中使用CDH相关功能，必须使用DataWorks的独享调度资源组。
- 您需要先保障CDH集群和独享调度资源组的网络可达后再进行后续的相关操作。
- 目前DataWorks支持的CDH版本有：cdh6.1.1、cdh5.16.2、cdh6.2.1和cdh6.3.2。

Step1：获取CDH集群配置信息

1. 获取CDH版本信息，用于后续DataWorks中新增CDH引擎配置。

   登录Cloudera Manager，在主界面集群名称旁可查看当前部署的CDH集群版本，如下图所示。



2. 获取Host地址与组件地址信息，用于后续DataWorks中新增CDH引擎配置。
   - 方式一：使用DataWorks JAR包工具获取。
     a. 登录Cloudera Manager，下载工具JAR包。

     ```
     wget https://dataworks-public-tools.oss-cn-shanghai.aliyuncs.com/dw-tools.jar
     ```

     b. 运行工具JAR包。

     ```
     export PATH=$PATH:/usr/java/jdk1.8.0_181-cloudera/bin
     java -jar dw-tools.jar <user> <password>
     ```

     其中 `<user>` 和 `<password>` 分别是Cloudera Manager的用户名和密码。

     c. 在运行结果中查看并记录CDH的Host地址和组件地址信息。

     ```
     [root@cdh-header-1-cn-shanghai ~]# wget https://dataworks-public-tools.oss-cn-shanghai.aliyuncs.com/dw-tools.jar
     --2021-01-08 18:52:55--  https://dataworks-public-tools.oss-cn-shanghai.aliyuncs.com/dw-tools.jar
     Resolving dataworks-public-tools.oss-cn-shanghai.aliyuncs.com (dataworks-public-tools.oss-cn-shanghai.aliyuncs.com)... 106.14.228.176
     Connecting to dataworks-public-tools.oss-cn-shanghai.aliyuncs.com (dataworks-public-tools.oss-cn-shanghai.aliyuncs.com)|106.14.228.176|:443...
     connected.
     HTTP request sent, awaiting response... 200 OK
     Length: 6743456 (6.4M) [application/java-archive]
     Saving to: 'dw-tools.jar.1'

     100%[=====================================================================================================>] 6,743,456   36.0MB/s   in 0.2s

     2021-01-08 18:52:55 (36.0 MB/s) - 'dw-tools.jar.1' saved [6743456/6743456]

     [root@cdh-header-1-cn-shanghai ~]# export PATH=$PATH:/usr/java/jdk1.8.0_181-cloudera/bin
     [root@cdh-header-1-cn-shanghai ~]# java -jar dw-tools.jar admin admin
     Hosts:
     192.168.22.217  cdh-header-1-cn-shanghai
     192.168.22.219  cdh-worker-2-cn-shanghai
     192.168.22.218  cdh-worker-1-cn-shanghai

     Urls:
     HiveServer2:    jdbc:hive2://cdh-header-1-cn-shanghai:10000
     Hive Metastore: thrift://cdh-header-1-cn-shanghai:9083
     YARN ResourceManager:   http://cdh-header-1-cn-shanghai:8032
     Impala Daemon:  jdbc:impala://cdh-worker-1-cn-shanghai:21050
     ```

   - 方式二：在Cloudera Manager页面手动查看。

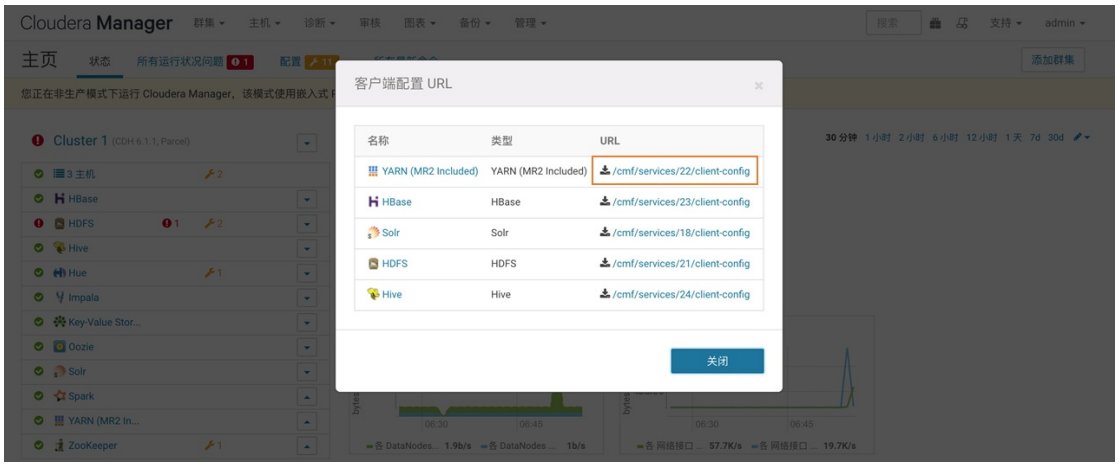
登录Cloudera Manager，在主机（Hosts）下拉菜单中选择角色（Roles），根据关键字和图标识别出需要配置的服务，然后看左侧对应的主机（Host），按照格式补全要填写的地址。默认端口号可以参考方法一的输出结果样例。

其中

- HS2：HiveServer2
- HMS：Hive Metastore
- ID：Impala Daemon
- RM：YARN ResourceManager

3. 获取配置文件，用于后续上传至DataWorks。
   i. 登录Cloudera Manager。
   ii. 在**状态**页面，单击集群的下拉菜单中的**查看客户端配置 URL**。

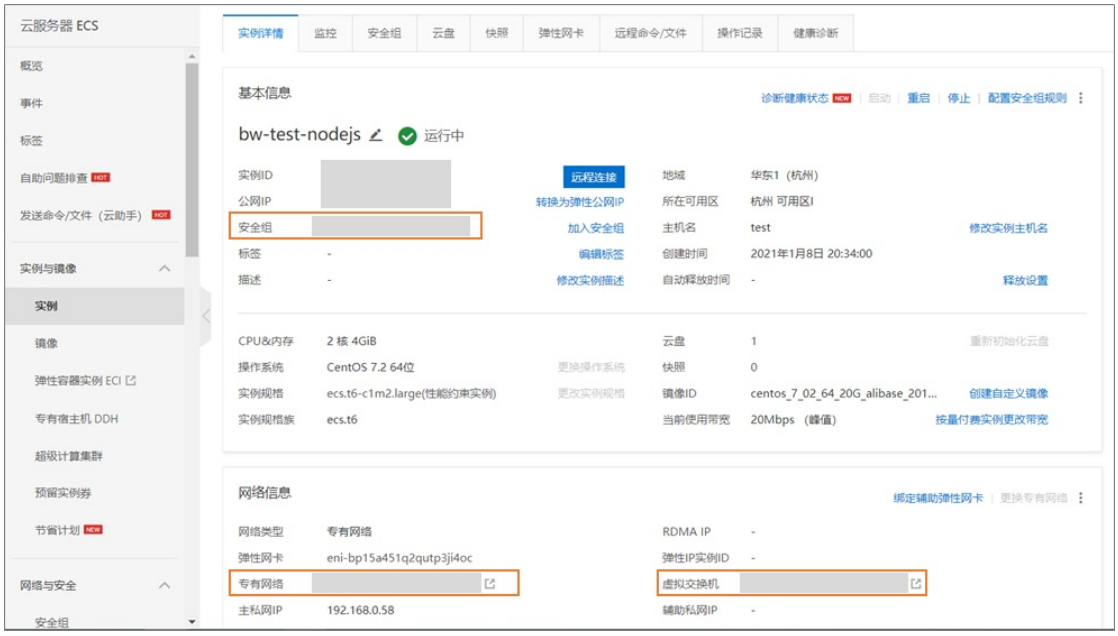
iii. 在对话框中下载YARN的配置包。



4. 获取CDH集群的网络信息，用于后续与DataWorks的独享调度资源组网络联通配置。

i. 登录部署CDH集群的ECS控制台。

ii. 在实例列表中找到部署CDH集群的ECS实例，在**实例详情**中查看并记录**安全组**、**专有网络**、**虚拟交换机**信息。



Step2：配置网络联通

DataWorks的独享调度资源组购买创建完成后，默认与其他云产品网络不可达，在对接使用CDH时，您需获取部署CDH集群的网络信息，将独享调度资源组绑定至CDH集群所在的VPC网络中，保障CDH集群与独享调度资源组的网络联通。

1. 进入独享资源组网络配置页面。

i. 登录DataWorks控制台。

ii. 在左侧导航栏，单击**资源组列表**，默认进入**独享资源组**页签。

iii. 单击已购买的独享调度资源组后的**网络设置**。

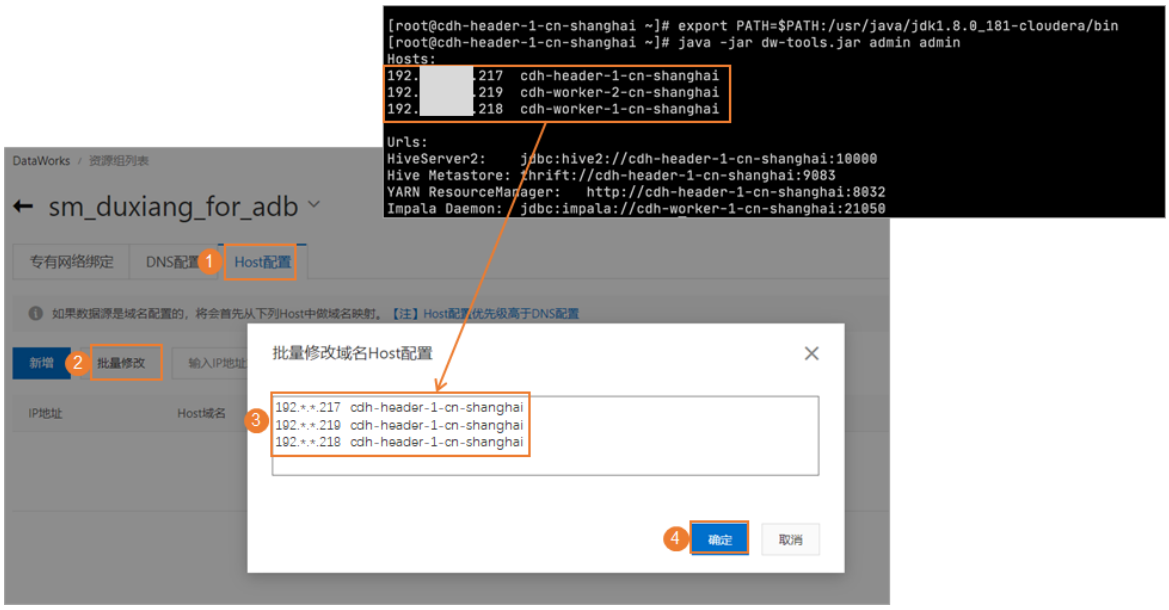
2. 绑定VPC。

在**专有网络绑定**页签，单击**新增绑定**，在配置页面选择上述步骤记录的CDH集群所在VPC、交换机、安

全组。

3. 配置Host。

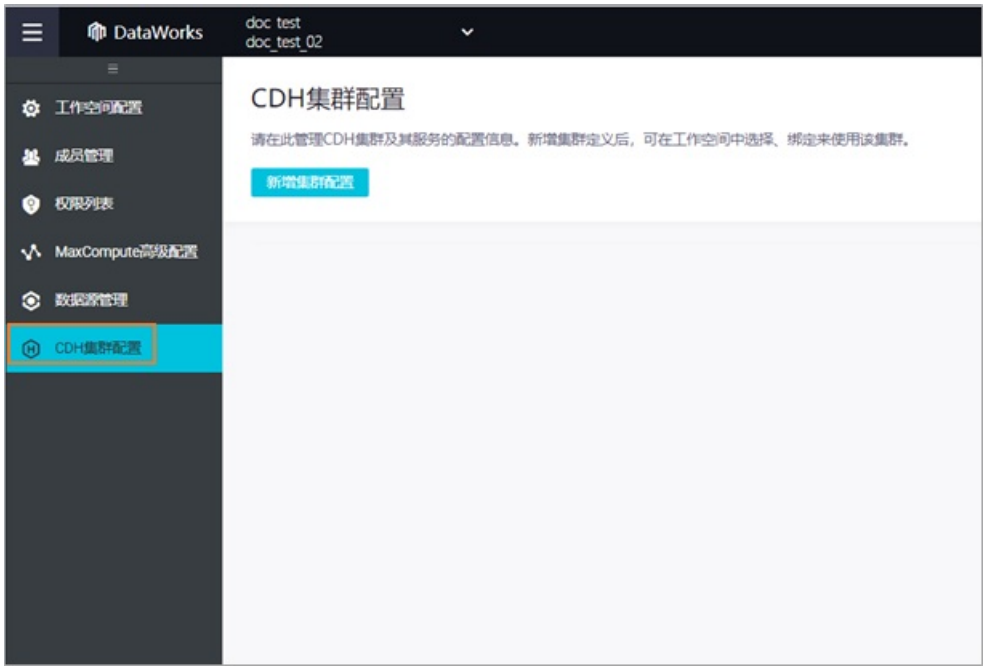
   在**Host配置**页签，单击**批量修改**，在对话框中配置为上述步骤中记录的Host地址信息。



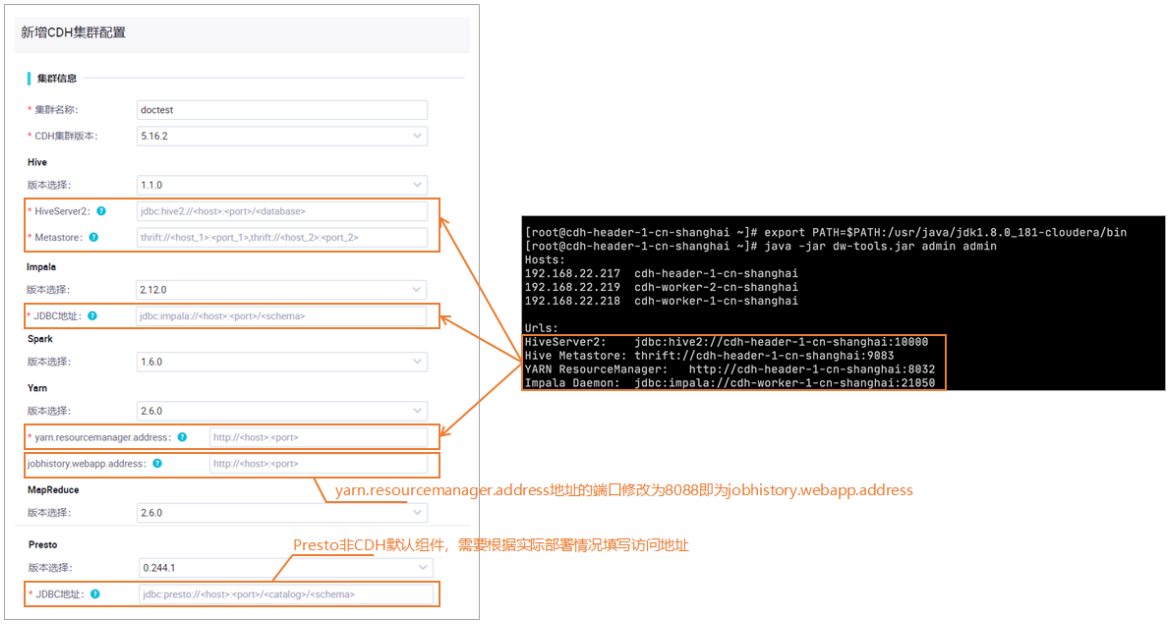
Step3：在DataWorks中新增CDH集群配置

只有工作空间管理员才能进行新增CDH集群配置操作，操作时请使用拥有空间管理员权限的账号。

1. 进入项目空间管理页面。
   i.
   ii. 在左侧导航栏，单击**工作空间列表**。
   iii. 在对应工作空间的操作列单击**工作空间配置**。
   iv. 在右侧配置页面中单击**更多设置**。
2. 在项目空间管理页面，单击**CDH 集群配置**。



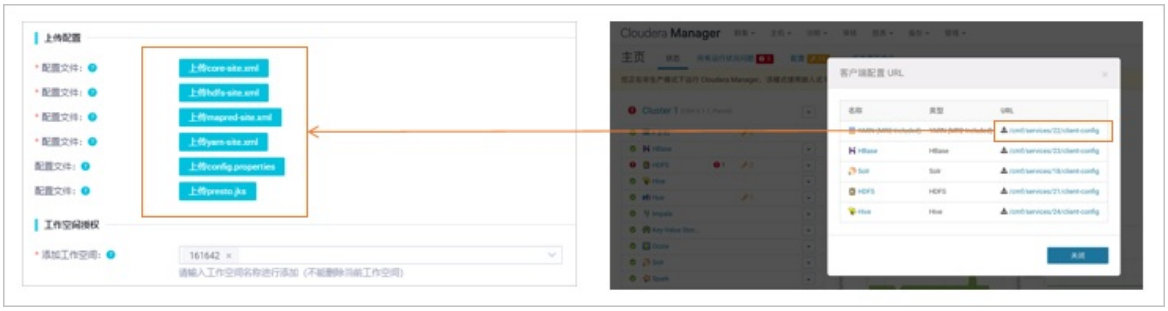
3. 在**CDH集群配置**页面单击**立即新增**，在**新增CDH集群配置**对话框中，填写上述步骤Step2：配置网络联通中记录的组件地址信息。



其中：

- 集群名称：可自定义集群名称。
- 版本信息：根据实际情况选择对应的CDH和组件版本。
- 地址信息：根据上述步骤中记录的地址信息填写。其中：
  - Yarn的**jobhistory.webapp.address**信息：**yarn.resourcemanager.address**地址的端口修改为8088即为**jobhistory.webapp.address**。
  - Presto的**JDBC地址**：Presto非CDH默认组件，需要根据实际部署情况填写访问地址。

4. 上传配置文件并授权给其他工作空间。



5. 配置访问身份的映射关系。

   如果您希望在运行任务时，对不同云账号在CDH集群内可访问的数据进行数据权限隔离，则可开启**Kerberos账号（principal）认证**，并配置**云账号**与**Kerberos账号**的权限映射关系。

   > **说明** **Kerberos账号**为CDH集群的访问账号。CDH集群通过Sentry或Ranger组件为**Kerberos账号**进行不同权限的配置，实现数据权限隔离。与**Kerberos账号**存在映射关系的**云账号**拥有相同的CDH集群数据访问权限。请填写格式为 `实例名@领域名` 的Kerberos账号（principal），例如，cdn_test@HADOOP.COM。

6. 单击**确定**，完成新增CDH集群配置。

   完成新增CDH集群配置后，已授权的工作空间中可新增此CDH引擎，用于后续编辑并运行数据开发等任务。

Step4：在DataWorks中新增CDH引擎

1. 在项目空间管理页面，单击**工作空间配置**。
2. 在**计算引擎信息**区域的CDH页签单击**增加实例**，在弹窗中配置实例信息。

   新增引擎实例时，可选择使用**快捷模式**或**安全模式**访问模式，**安全模式**可以实现不同云账号运行任务时的数据权限隔离。不同访问模式的配置界面如下：

   - **快捷模式**的实例信息配置。

- **安全模式**的实例信息配置。

i. 填写**实例显示名称**。

ii. 选择**访问模式**

- **快捷模式**

  该访问模式使用便捷，多个云账号对应一个集群账号，多个账号均可访问同一个集群账号内的数据，无法实现不同云账号运行任务时的数据权限隔离。

- **安全模式**

  该访问模式允许您配置云账号与CDH集群账号的身份映射关系，实现不同云账号运行任务时的数据权限隔离。

iii. 选择上述新增的CDH集群配置。

如果上一步**访问模式**选择**快捷模式**，则此处选择未开启Kerberos认证的CDH集群。如果**访问模式**选择**安全模式**，则此处需要选择已开启Kerberos认证的CDH集群。您可以进入工作空间配置查看CDH集群是否开启Kerberos认证。

iv. 设置访问集群的认证信息。

- **快捷模式**

  当前仅支持指定特定账号，建议使用admin或hadoop账号。该账号仅用于下发任务。

- **安全模式**

  您可以根据需求选择**调度访问身份**。该身份用于在任务提交调度后自动调度运行任务，并且需要配置云账号与CDH集群账号的身份映射，详情请参见配置访问身份映射。

  > **说明** 在DataStudio页面，运行任务所使用的身份均为当前已登录云账号映射的集群访问身份。因此，除了需要为调度访问身份配置身份映射外，建议为项目空间开发成员也配置身份映射，避免页面运行任务失败。

v. 选择已经购买好的独享调度资源组。

vi. 单击**测试连通性**。

如果网络连通测试失败，可能是因为独享调度资源组没有绑定CDH集群所在的专有网络，或者独享调度资源组没有设置Host，请参见Step2：配置网络联通检查独享调度资源组的网络配置。

3. 单击**确定**，创建计算引擎实例。

此步骤会触发独享调度资源组的初始化（安装访问CDH集群的客户端以及上传配置文件），您需要等待**独享资源组初始化状态** 从**准备中**变成**完成**，CDH引擎实例才创建完成。

4. 在创建的CDH引擎实例页面单击**测试服务连通性**，DataWorks会运行测试任务测试客户端和配置文件是否正确安装。

如果测试结果显示失败，您可以查看日志并提交工单联系DataWorks技术支持。

使用DataWorks进行数据开发

完成新增CDH引擎后，您就可以在DataStudio（数据开发）中创建Hive、Spark、MapReduce、Impala或者Presto任务节点，直接运行任务或者设置周期调度运行任务。以下以创建并运行一个Hive任务为例，为您介绍在DataWorks中如何进行CDH引擎的数据开发和运行。

1. 进入DataStudio页面。

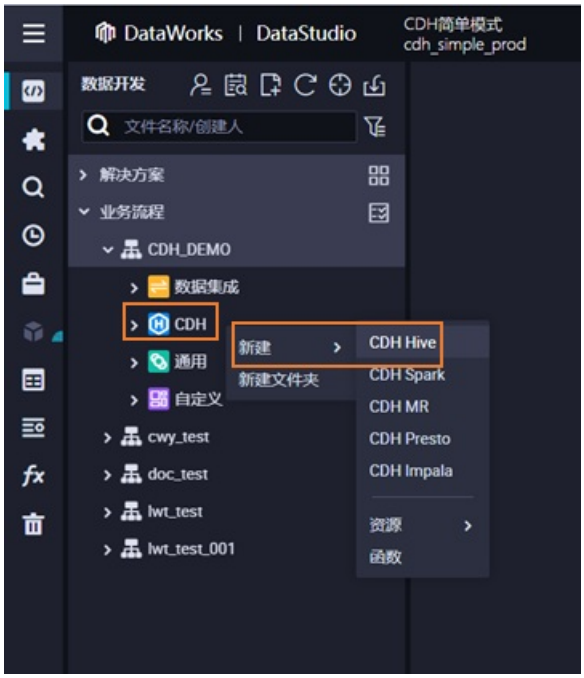
   i.

   ii. 在左侧导航栏，单击**工作空间列表**。

   iii. 在对应工作空间的操作列单击**进入数据开发**。

2. 创建业务流程，根据界面提示填写业务流程信息。

3. 单击创建好的业务流程，在**CDH引擎**文件夹上右键选择**新建 > CDH Hive**。



4. 在右侧代码编辑框中编写Hive SQL，完成代码编辑后单击顶部运行图标，选择调度资源组并确认，运行完毕后可以查看Hive SQL的运行结果。
5. 如果想要设置任务周期调度，单击右侧的**调度配置**，在弹窗中设置时间属性、资源属性和调度依赖，完成后单击提交任务，提交成功后任务就可以按照配置周期调度运行，调度配置详情可参见 配置基础属性。
6. 在运维中心中可以查看提交的周期任务，在周期实例中查看任务周期调度的运行情况。详细可参见查看周期任务。

运维监控配置

CDH引擎的任务支持使用DataWorks运维中心的智能监控功能，通过自定义报警规则、配置任务告警，根据设置的报警规则自动触发任务运行异常报警。自定义报警规则操作可参见自定义规则，配置任务告警操作可参见基线管理。

数据质量规则配置

在DataWorks上使用CDH引擎时，可使用DataWorks的数据质量服务进行数据查、对比、质量监控、SQL扫描和智能报警等功能，数据质量服务的详细操作可参见数据质量概述。

数据地图配置

在DataWorks上使用CDH引擎时，可使用DataWorks的数据地图服务采集CDH集群中Hive数据库、表、字段、分区元数据，便于实现全局数据检索、元数据详情查看、数据预览、数据血缘和数据类目管理等功能。

> **说明** 当前仅支持Hive数据库。

DataWorks上数据地图功能的详细介绍与配置指导可参见概述。

如果要您希望可以实时感知CDH集群中Hive元数据的变更，或者要在数据地图中查看血缘和元数据变更记录，需要将DataWorks的Hive Hook 嵌入到目标集群，并通过阿里云日志服务采集Hive Hook产生的日志。

配置Hive Hook后，元数据变更消息会被记录到HS2和HMS服务器的日志文件*/tmp/hive/hook.event.*.log*中，使用阿里云日志服务采集后供DataWorks读取，下载DataWorks小工具 `dw-tools.jar` ，在同一目录下创建 `config.json` 文件并补全配置项的值，最后执行工具一键创建日志采集。

配置Hive Hook和采集Hive Hook日志的操作步骤如下。

1. 配置Hive Hook。
   i. 登录HS2和HMS服务器，并进入*/var/lib/hive*目录，下载DataWorks Hive Hook。

```
# CDH 6.x 版本下载 dataworks-hive-hook-2.1.1.jar
wget https://dataworks-public-tools.oss-cn-shanghai.aliyuncs.com/dataworks-hive-hook-2.1.1.jar
# CDH 5.x 版本下载 dataworks-hive-hook-1.1.0-cdh5.16.2.jar
wget https://dataworks-public-tools.oss-cn-shanghai.aliyuncs.com/dataworks-hive-hook-1.1.0-cdh5.16.2.jar
```

   ii. 登录Cloudera Manager首页后，进入**Hive > 配置** ，把Hive辅助JAR目录配置项设置为 `/var/lib/hive` 。
   iii. 在**Hive > 配置** 中，将 `hive-site.xml` 的Hive服务高级配置代码段（安全阀） 配置项添加以下内容。

```
<property>
  <name>hive.exec.post.hooks</name>
  <value>com.cloudera.navigator.audit.hive.HiveExecHookContext,org.apache.hadoop.hive.ql.hooks.LineageLogger,com.aliyun.dataworks.meta.hive.hook.LineageLoggerHook</value>
</property>
```

   iv. 在**Hive > 配置** 中，将 `hive-site.xml` 的Hive Metastore Server高级配置代码段（安全阀） 配置项添加以下内容。

```
<property>
  <name>hive.metastore.event.listeners</name>
  <value>com.aliyun.dataworks.meta.hive.listener.MetaStoreListener</value>
</property>
<property>
  <name>hive.metastore.pre.event.listeners</name>
  <value>com.aliyun.dataworks.meta.hive.listener.MetaStorePreAuditListener</value>
</property>
```

   v. 配置完成后，根据Cloudera Manager的提示部署客户端配置，然后重启Hive服务。

> **说明** 如果重启失败，保留日志用于排查问题，为防止影响正常作业可以先去掉上面两个步骤添加的配置再重启恢复Hive服务。如果添加配置后重启成功，查看服务器*/tmp/hive/*下是否产生名称以 `hook.event` 开头的日志文件，例如 `hook.event.1608728145871.log` 。

2. 采集Hive Hook日志。
   i. 登录Cloudera Manager，下载工具JAR包。

```
wget https://dataworks-public-tools.oss-cn-shanghai.aliyuncs.com/dw-tools.jar
```

ii. 在小工具所在目录创建 `config.json` ，根据以下文件内容要求修改并保存文件。

```
// config.json
{
    "accessId": "<accessId>",
    "accessKey": "<accessKey>",
    "endpoint": "cn-shanghai-intranet.log.aliyuncs.com",
    "project": "onefall-test-pre",
    "clusterId": "1234",
    "ipList": "192.168.0.1,192.168.0.2,192.168.0.3"
}
```

其中：

- **accessId**：阿里云账号的AccessKey ID。
- **accessKey**：阿里云账号的AccessKey Secret。
- **endpoint**：填写为日志服务project的访问域名中的私网域名，详细可参见服务入口。
- **project**：填写为使用的阿里云日志服务的project名称，您可参见管理Project获取日志服务的project名称。
- **clusterId**：填写为DataWorks生成的CDH集群ID，可以提交工单获取此ID。
- **ipList**：填写为HS2和HMS的所有服务器的IP列表（即部署了DataWorks Hive Hook的所有服务器IP），多个IP使用英文逗号（,）分隔。

iii. 运行配置文件。

```
java -cp dw-tools.jar com.aliyun.dataworks.tools.CreateLogConfig config.json
```

iv. 安装客户端。

```
wget http://logtail-release-cn-shanghai.oss-cn-shanghai.aliyuncs.com/linux64/logtail.sh -O logtail.sh; chmod 755 logtail.sh; ./logtail.sh install cn-shanghai
```

其中**cn-shanghai**改为日志服务对应的Region。

3. 完成上述步骤后，在阿里云日志服务的指定project下会生成名为hive-event日志库、名为hive-event-config的logtail配置以及名为hive-servers的机器组。您可以查看并记录阿里云账号ID、日志服务的endPoint和Project信息，将这些信息通过提交工单提供给DataWorks技术人员，由技术人员进行后续的配置。