ALIBABA CLOUD

阿里云

智能数据构建与管理 Dataphin

最佳实践

文档版本：20210709

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}`

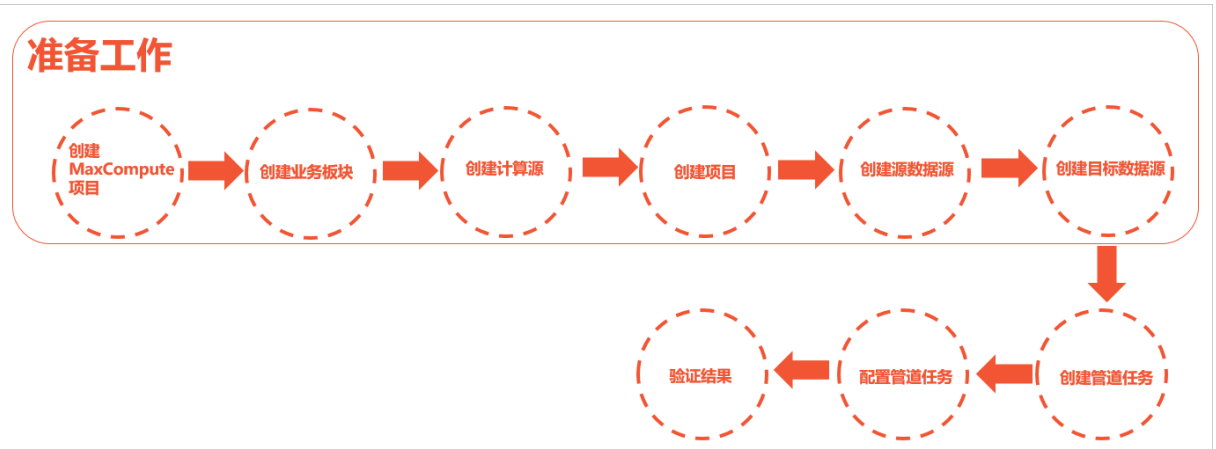
目录

1.数据集成

1.1. 一键生成目标表

在数据集成过程中，当目标数据源类型为MaxCompute时，您可以通过Dataphin提供的一键生成目标表的功能，快速创建目标表。

操作流程



主流程	描述
创建MaxCompute项目	创建Dataphin计算源的MaxCompute项目（best_practice_dev和best_practice_prod）。
创建业务板块	创建业务板块和数据域。
创建计算源	创建Dev和Prod项目的计算源。
创建项目	创建数据开发的工作空间。
创建源数据源	连接您的业务数据源至Dataphinpin平台。
创建目标数据源	创建数据集成的目标数据源（MaxCompute类型的数据源）。
创建离线单条管道	创建用于数据集成的管道任务。
一键生成目标表	一键创建目标数据源的目标表，以配置管道任务。
验证结果	验证目标表是否集成到数据。

准备工作

- 准备数据源，详情请参见准备数据源。
- 创建MaxCompute项目，详情请参见创建工作空间。

工作空间名称	选择计算引擎服务
best_practice_dev	MaxCompute

工作空间名称	选择计算引擎服务
best_practice_prod	MaxCompute

- 创建业务板块LD_best_practice和LD_best_practice_dev，创建数据域test，详情请参见新建业务板块。
- 创建Dev项目计算源（best_practice_dev），配置如下参数，详情请参见新建MaxCompute计算源。

参数	描述
计算类型	默认为MaxCompute，不支持修改。
计算源名称	输入best_practice_dev。
计算源描述	输入Dev项目的计算源。
Endpoint	默认为 http://service.cn.maxcompute.aliyun.com/api ，不支持修改。
Project Name	输入best_practice_dev。
Access ID	访问密钥中的AccessKey ID，您可以通过用户信息管理页面获取。
Access Key	访问密钥中的AccessKey Secret，您可以通过用户信息管理页面获取。

- 创建Prod项目计算源（best_practice_prod），配置如下参数，详情请参见新建MaxCompute计算源。

参数	描述
计算类型	默认为MaxCompute，不支持修改。
计算源名称	输入best_practice_prod。
计算源描述	输入Prod项目的计算源。
Endpoint	默认为 http://service.cn.maxcompute.aliyun.com/api ，不支持修改。
Project Name	输入best_practice_prod。
Access ID	访问密钥中的AccessKey ID，您可以通过用户信息管理页面获取。
Access Key	访问密钥中的AccessKey Secret，您可以通过用户信息管理页面获取。

- 创建项目best_practice和best_practice_dev，详情请参见创建Basic项目空间。

参数	描述
公用名称	输入最佳实践。
公用英文名	输入best_practice。
业务板块	选择LD_best_practice。
空间类型	选择应用层。

参数	描述
项目1：Dev	选择离线计算源为best_practice_dev。
项目2：Prod	选择离线计算源为best_practice_prod。
描述	输入简单的描述。
沙箱白名单	添加沙箱白名单： i. 单击**新建**。 ii. 在访问地址输入框中，输入RDS ID.mysql.rds.aliyuncs.com。在访问地址输入框中，输入3306。 iii. 单击图标。

- 创建源数据源**dataphin**，详情请参见新建MySQL数据源。
- 创建目标数据源，配置如下参数，详情请参见新建MaxCompute数据源。

参数	描述
数据源类型	选择**MaxCompute**。
数据源名称	输入为target。
数据源描述	输入目标数据源。
数据源配置	选择**生产+开发数据源**。
JDBC URL	输入http://service.cn.maxcompute.aliyun.com/api。
Project Name	开发环境输入为best_practice_dev，生产环境输入为best_practice_prod。
Access ID	访问密钥中的AccessKey ID，您可以通过用户信息管理页面获取。
Access Key	访问密钥中的AccessKey Secret，您可以通过用户信息管理页面获取。

步骤一：创建管道任务

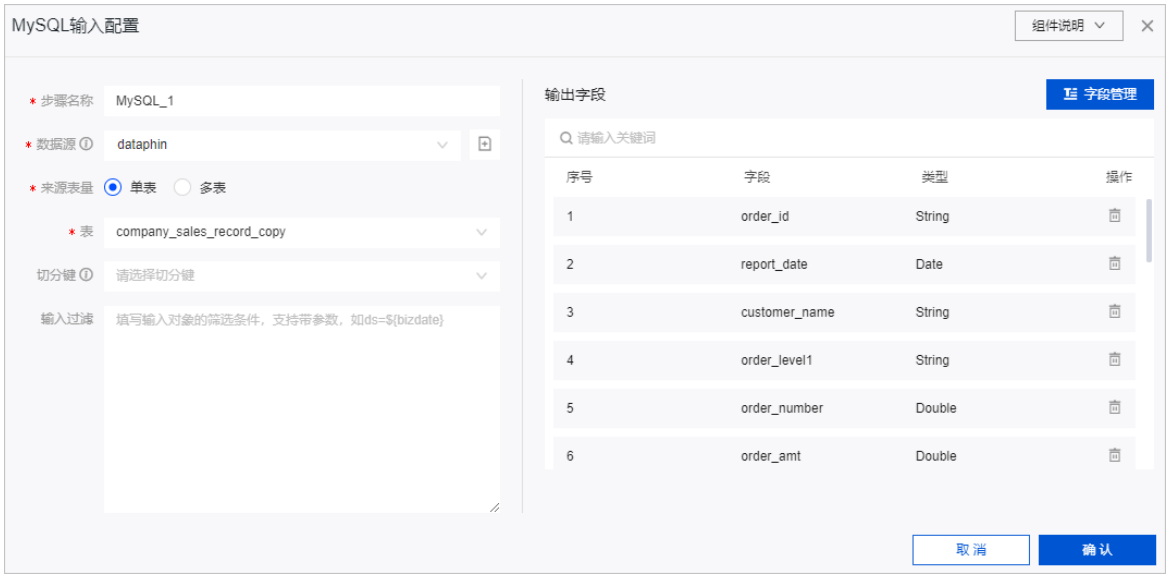
1. 登录Dataphin控制台。
2. 在Dataphin控制台页面，选择工作区地域后，单击**进入Dataphin>>**。
3. 在Dataphin首页，单击顶部菜单栏的**研发**。
4. 在数据**开发**页面，单击项目名称后的图标，单击**Dev**页签，选择**best_practice_dev**为数据开发的项目空间。
5. 在数据**开发**页面，鼠标悬停至顶部菜单栏中的**开发**上，单击**集成**。
6. 在数据**集成**页面，鼠标悬停在图标，单击**离线单条管道**。
7. 在**创建管道开发脚本**对话框，配置参数。

参数	描述
管道名称	输入test。
调度类型	**调度类型**选择**手动节点**。
描述	填写对离线单条管道的简单描述。
选择目录	默认目录为**离线管道**。

8. 单击**确定**。

步骤二：配置管道任务（一键生成目标表）

1. 在离线单条管道开发页面，单击右上方的**组件库**。
2. 单击**输入**前的>图标后，单击**MySQL**组件拖动到左侧的管道画布中。
3. 鼠标悬停至组件框内并右键单击，选择**属性配置**后，配置参数。

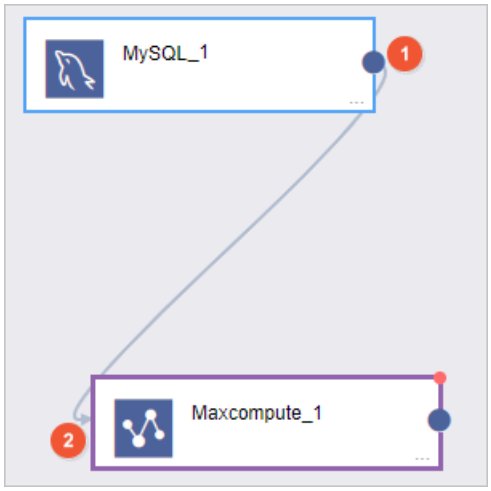


参数	描述
步骤名称	保持默认。
数据源	选择数据源（dataphin）。
来源表量	选择**单表**。
表	选择来源表company_sales_record_copy。
切分键	无需添加切分键。
输入过滤	无需添加过滤条件。
输出字段	查看输出字段。

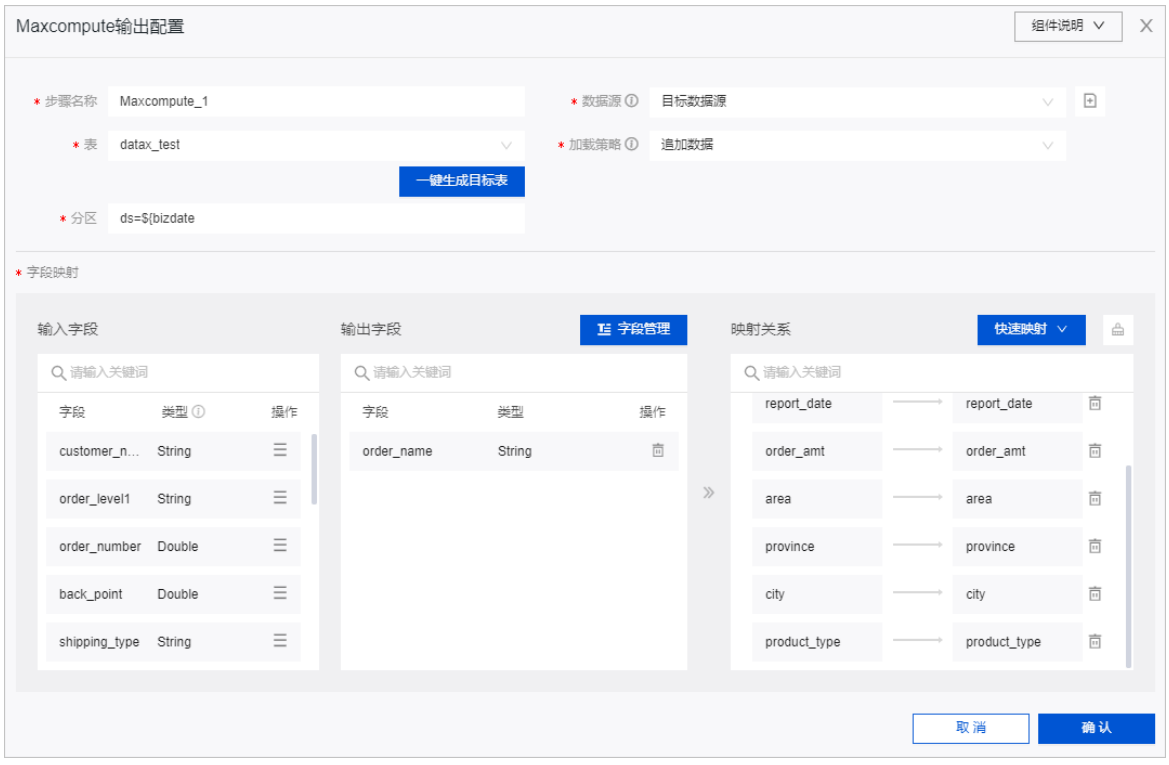
4. 单击**确认**，完成输入组件的属性配置。

5. 单击**输出**前的>图标后，单击**MaxCompute**组件拖动到左侧的管道画布中。

6. 单击下图中输入组件（**MySQL**）①处后拖动并指向输出组件（**MaxCompute**）的②处，形成有向连线。



7. 鼠标悬停至输出组件框内并右键单击，选择**属性配置**，配置参数。



参数	说明
步骤名称	保持默认。
数据源	选择目标数据源 。

参数	说明

表	创建目标表：
	i. 单击**一键生成目标表**。
	ii. 在代码输入框中，输入建表语句。
	`CREATE TABLE IF NOT EXISTS datax_test`
	`(order_id bigint comment '订单号',`
	`` `area` string comment '区域',``
	`province string comment '省份',`
	`city string comment '城市',`
	`product_type string comment '类型',`
	`order_name string comment '客户名称',`
	`report_date datetime comment '日期',`
	`order_amt double comment '销售额')`
	`PARTITIONED BY ( `ds` STRING);`
	iii. 单击**新建**。
	说明 无需选中是否在生产建表。

加载策略	选择**追加数据**。
分区	输入ds=${bizdate}。
输入字段	根据上游的输入，为您展示输入字段。
输出字段	为您展示输出字段。
快速映射	映射关系选择为**同名映射**的操作步骤： i. 单击**快速映射**后的图标。 ii. 选择**同名映射**。 iii. 在**提醒**对话框中，单击**确定**。

8. 单击**确认**，完成输出组件的属性配置。

步骤三：验证结果

1. 单击管道开发页面左上方的**预览**。
2. 在对话框中，bizdate填写为20200819，单击**确定**。

3. 单击管道开发页面左上方的**执行**。
4. 在对话框中，**bizdate**填写为20200819，单击**确定**。查看数据是否同步至MaxCompute组件。

1.2. 自定义RDBMS数据库及同步数据

为了满足不同业务场景数据集成的诉求，Dataphin支持用户自定义当前系统不支持的RDBMS数据库（关系型数据库）类型的组件，并进行数据同步。您只需要准备关系型数据库的驱动，即可自定义RDBMS数据库类型的组件。本教程以MySQL为例，为您介绍如何自定义RDBMS数据库及进行数据同步。

前提条件

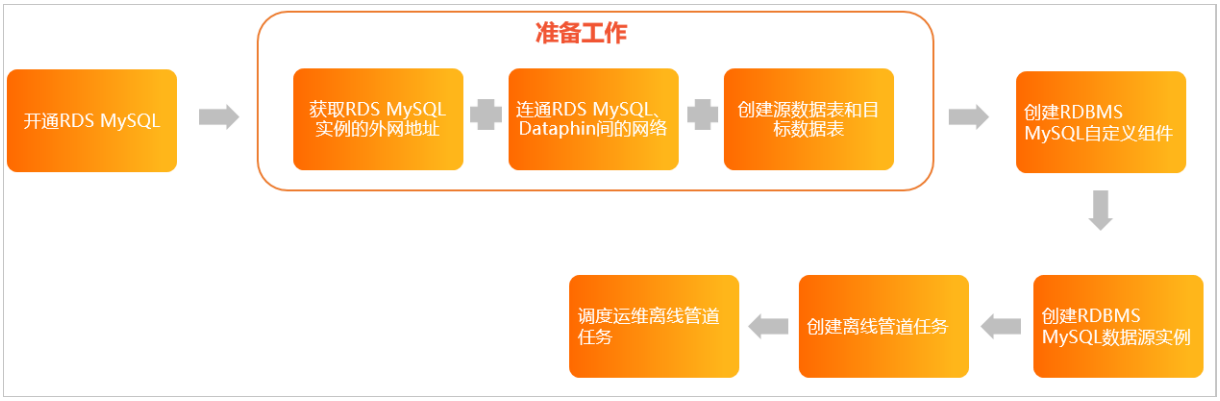
- 已开通RDS MySQL实例，且网络类型为专有网络（VPC）。如何开通RDS MySQL实例，请参见创建RDS MySQL实例。
- 已创建RDS MySQL实例的数据库和账号，创建过程中需要您记录数据库名称、用户名和密码。如何创建数据库和账号，请参见创建数据库和账号。

背景信息

RDBMS数据库即关系型数据库，包括MySQL、Oracle、SQL Server、PostgreSQL、Vertica、DRDS、DB2、OceanBase、PolarDB、SAP HANA和TeraData。本教程以MySQL为例，带您体验自定义RDBMS数据库，并进行数据同步。

操作流程

自定义并应用MySQL数据库组件的流程，如下图所示。



步骤	描述
步骤一：下载自定义MySQL数据库组件的驱动	获取自定义MySQL数据库组件驱动。
步骤二：配置网络和创建数据表	在您开始自定义并应用RDBMS数据库组件前，需要配置RDS MySQL实例和Dataphin间的网络，及创建同步数据的源表和目标表。
步骤三：创建自定义组件	自定义组件的类型为**test_rdbms_mysql**。完成定义后，即可在组件库的开发模块下查询到自定义的组件。
步骤四：创建数据源实例	基于自定义的组件类型（**test_rdbms_mysql**），创建**TEST_RDBMS_MYSQL**类型的数据源实例。完成创建数据源实例后，即可将RDS MySQL实例的业务数据引入至Dataphin实例。
步骤五：创建离线管道任务	基于自定义的组件类型（**test_rdbms_mysql**）和数据源实例（**test_rdbms_mysql**），创建离线管道任务。完成离线管道任务的创建后，即可运行离线管道任务，以实现数据的集成（同步数据）。
步骤六：生产环境中运行离线管道任务	在生产环境运行离线管道任务，保障生产环境业务数据的正常产出。

步骤一：下载自定义MySQL数据库组件的驱动

请下载MySQL数据库的驱动。

步骤二：配置网络和创建数据表

- 连通RDS MySQL实例与Dataphin实例间的网络。

- 添加RDS MySQL实例的外网地址和端口至Dataphin项目空间的沙箱白名单：
  a. 获取RDS MySQL实例的外网地址、端口。

     进入数据库连接页面，获取RDS MySQL实例的外网地址和端口。

  b. 添加RDS MySQL实例的外网地址和端口至Dataphin项目空间的沙箱白名单。如何添加沙箱白名单，请参见添加沙箱白名单。
- 添加Dataphin的IP至RDS MySQL实例的白名单。如何添加Dataphin的IP至RDS MySQL实例的白名单，请参见设置IP白名单。

地域	IP白名单
华东2（上海）	100.104.228.128/26、100.104.115.192/26
华南1（深圳）	100.104.48.128/26
华北2（北京）	100.104.238.64/26
华东2（上海）、华南1（深圳）、华北2（北京）	100.104.0.0/16

- 创建同步数据的源数据表和目标数据表。

  使用命令行方式连接MySQL实例，连接后创建同步数据的源数据表和目标数据表。如何连接MySQL实例，请参见方法三：使用命令行方式连接实例。
  - 创建源数据表的代码示例如下。

```
create table xin_test_scr2
(
id string,
name string
);
insert into xin_test_scr2 values('1001','huayu1'),('1002','huayuyu2'),('1003','huayuyu3'),('1004','huayuyu4'),('1005','huayuyu5'),('1006','huayuyu6'),('1007','huayuyu7'),('1008','huayuyu8'),('1009','huayuyu9'),('1010','huayuyu10'),('1011','huayuyu11'),('1012','huayuyu12'),('1013','huayuyu13'),('1014','huayuyu14'),('1015','huayuyu15'),('1016','huayuyu16'),('1017','huayuyu17'),('1018','huayuyu18'),('1019','huayuyu19'),('1021','huayuyu21'),('10022','huayuyu22'),('1023','huayuyu23');
```

  - 创建目标数据表的代码示例如下。

```
create table xin_test_det_1
(
id string,
name string
);
```

步骤三：创建自定义组件

1. 登录Dataphin控制台。
2. 在Dataphin控制台页面，选择工作区地域后，单击**进入Dataphin>>**。
3. 进入**自定义组件**页面。
   i. 在Dataphin首页，单击**研发**。
   ii. 在数据**开发**页面，单击项目名称后的图标，选择数据开发的项目空间（**Dev**项目）。如果您当前访问的是**Dev**项目，且项目空间为您的数据开发空间，则不需要选择项目空间。
   iii. 在数据**开发**页面，将鼠标悬停在顶部菜单栏的**开发**上，单击**集成**。
   iv. 在数据**集成**页面，单击左侧导航栏的图标。
4. 在**自定义组件**页面，单击图标。
5. 在**新建自定义组件**页面，配置参数。

区域	参数	描述
	类型	选择**类型**为**RDBMS数据库**。

区域	参数	描述
基本信息	名称	填写**名称**为test_rdbms_mysql。 **注意** 因为名称定义了自定义组件的类型，所以系统不支持创建相同名称的自定义组件。
	描述	填写对自定义组件的简单描述。例如，自定义组件测试。
	选择目录	自定义组件的默认目录为**自定义组件**。
资源信息	驱动名称	填写**驱动名称**为com.mysql.jdbc.Driver。
	文件上传	上传已下载的驱动文件（mysql-connector-java-5.1.47）。

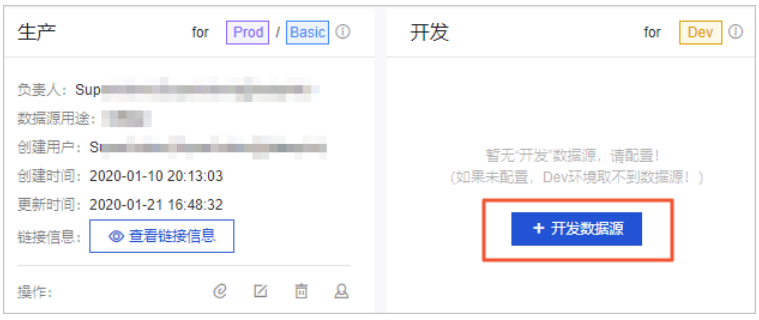
6. 单击**提交**。
7. 在**提交备注**对话框中，填写备注信息。
8. 单击**确定并提交**，完成自定义组件的提交。
9. 发布自定义组件至生产环境。
   i. 在**数据集成**页面，单击顶部菜单栏的**发布**。
   ii. 在**待发布对象列表**页面的**管道脚本**页签，
   iii. 选择**管道脚本**页签，选中test_rdbms_mysql后单击**操作**列下的图标。

   iv. 在**发布**对话框，填写发布名称或备注信息后，单击**确定**，即可将自定义组件发布至生产环境。
   v. 单击左侧导航栏的**发布记录列表**。在**发布记录列表**页面，查看自定义组件的发布状态为**发布成功**即可。

步骤四：创建数据源实例

1. 在**数据集成**页面，鼠标选停至图标后，单击**规划**。
2. 在数仓**规划**页面，单击左侧导航栏的**数据源**。
3. 在**数据源**页面，单击页面右上角的**新建数据源**。
4. 在**新建数据源**对话框，选择TEST_RDBMS_MYSQL类型的数据源。

5. 在**新建TEST_RDBMS_MYSQL数据源**对话框，配置参数。

参数		描述
数据源名称		填写**数据源名称**为test_rdbms_mysql。
数据源描述		填写数据源的简单描述。
数据源配置		配置数据源： ○ 如果开发模式是Basic模式，则选择**生产数据源**。 ○ 如果开发模式是Dev-Prod模式，则可以通过以下方式配置数据源： ■ 单击**生产+开发数据源**，配置生产环境和开发环境的数据源。 ■ 单击**生产数据源**，配置生产数据源。完成生产数据源的创建后，单击**开发数据源**，配置开发环境的数据源。  **说明** 系统支持配置**生产数据源**和**开发数据源**为相同的数据源，也可以配置为不同的数据源。
生产数据源或生产+开发数据源	**链接地址**	填写数据源的链接地址。 链接地址的格式为 `jdbc:mysql://{Public Endpoint}:3306/{DatabaseName}` ： ○ {Public Endpoint}：外网地址。 ○ {DatabaseName}：数据库名称。
	用户名	登录数据库的用户名。

参数		描述
	密码	登录数据库的密码。

> 注意 自定义类型数据源不支持连接测试，请您务必保证数据源连接信息的准确性。

6. 单击**确定**。

步骤五：创建离线管道任务

1. 在**数据源**页面，鼠标选停至图标后，单击**研发**。
2. 在数据**开发**页面，鼠标选停至**开发**，单击**集成**。
3. 创建管道开发脚本。
   i. 在数据**集成**的**离线管道**页面，鼠标悬停至图标后，单击**离线单条管道**。
   ii. 在**创建管道开发脚本**对话框，配置参数。

参数	描述
管道名称	**管道名称**填写为test。
调度类型	**调度类型**选择为**手动节点**。
描述	填写简单的描述。例如，**测试自定义组件**。
选择目录	默认为**离线管道**。

iii. 单击**确定**。

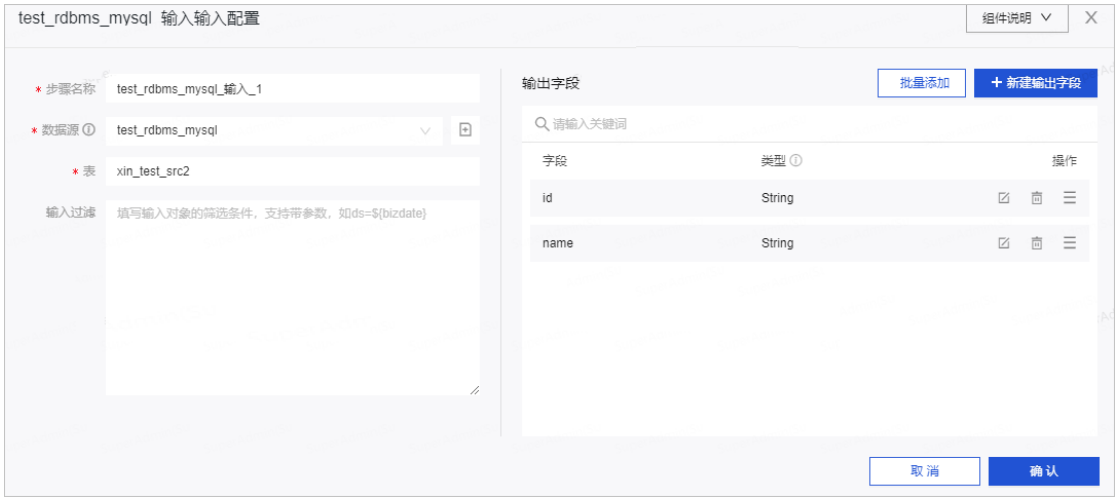
4. 开发离线管道任务。

i. 在test离线管道页面，单击页面右上角的«图标后，单击**开放**前的›图标。

ii. 拖动组件**test_rdbms_mysql_输入**和**test_rdbms_mysql_输出**至左侧的管道画布中。

iii. 鼠标选停至**test_rdbms_mysql_输入**组件框内右键单击后，单击**属性配置**。

iv. 在**test_rdbms_mysql_输入输入配置**对话框，配置输入参数。



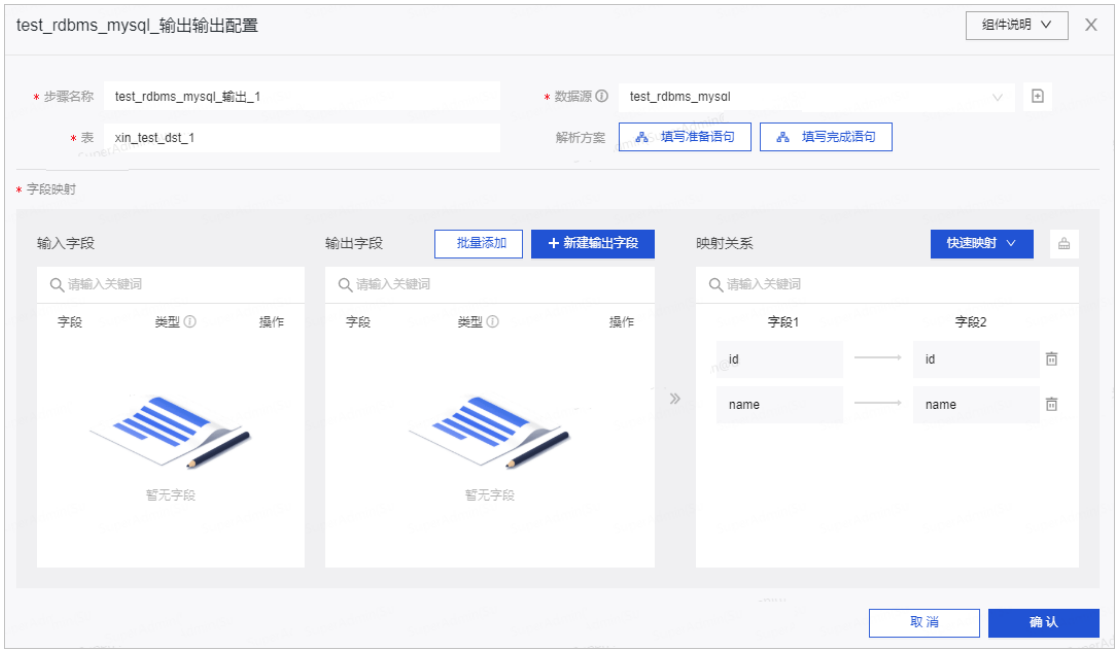
参数	描述
步骤名称	本教程中保持默认。 您也可以修改名称。步骤名称命名规则如下： ■ 只能包括字母、数字和短划线（-）。 ■ 长度为64字符以内。
数据源	选择**test_rdbms_mysql**。
表	填写来源表为**xin_test_src2**。
输入过滤	本教程中无需配置。 输入过滤即填写输入字段的过滤信息，例如 `ds=${bizdate}` 。**输入过滤**适用于以下两种场景： ■ 固定的某一部分数据。 ■ 参数过滤。
输出字段	输出字段即需要同步数据的字段。本教程中添加源表**xin_test_scr2**中的**id**和**name**字段为输入组件的输出字段： a. 在**输出字段**区域，单击**新建输出字段**。 b. 填写**输出字段**为id，**类型**选择为String。 c. 单击**新建输出字段**。 d. 填写**输出字段**为name，**类型**选择为String。

v. 单击**确认**。

vi. 单击输入组件中的①后拖动并指向输出组件中的②处，形成有向连线。

vii. 鼠标选停至**test_rdbms_mysql_输出**组件框内右键单击后，单击**属性配置**。

viii. 在**test_rdbms_mysql_输出输出配置**对话框，配置输出参数。



参数	描述
步骤名称	本教程中保持默认。 您也可以修改名称。步骤名称命名规则如下： ■ 只能包括字母、数字和短划线（-）。 ■ 长度为64字符以内。
数据源	选择**test_rdbms_mysql**。
表	填写目标表为**xin_test_dst_1**。
解析方案	本教程中无需选择。 解析方案为非必填项。选择数据输出前和输出完成的一些特殊处理方式。**解析方案**包括**填写准备语句**和**填写完成语句**： ■ **填写准备语句**：导入前执行的SQL脚本。 ■ **填写完成语句**：导入后执行的SQL脚本。
输入字段	默认展示输入组件中配置的输出字段。
输出字段	输出字段即需要同步数据的字段。本教程中添加源表**xin_test_scr2**中的**id**和**name**字段为输出组件的输出字段： a. 在**输出字段**区域，单击**新建输出字段**。 b. 填写**输出字段**为id，**类型**选择为String。 c. 单击**新建输出字段**。 d. 填写**输出字段**为name，**类型**选择为String。

参数	描述
映射关系	单击**快速映射**后，选择**同名映射**。 映射关系指的是输入组件的输出字段和输出组件的输出字段间的映射关系。映射关系包含同名映射和同行映射： ■ 同名映射：映射字段名称相同的字段。 ■ 同行映射：映射同行的字段。同行映射后，输入字段为最终的输出字段。

ix. 单击**确认**。

5. 单击离线管道test页面上方的**执行**，执行离线管道任务，查看任务是否正常执行。执行后的结果如下，读取和写入的数据均为23，表示任务运行正常。

6. 单击页面右上方的图标，提交管道脚本。

7. 在**提交备注**对话框，填写备注信息。

8. 单击**确定并提交**。

9. 发布离线管道任务至生产环境。

i. 在**数据集成**页面，单击顶部菜单栏的**发布**。

ii. 在**待发布对象列表**页面的**管道脚本**页签，

iii. 选择**管道脚本**页签，选中test后单击**操作**列下的图标。

iv. 在**发布**对话框，填写发布名称或备注信息后，单击**确定**，即可将离线管道任务发布至生产环境。

v. 单击左侧导航栏的**发布记录列表**。在**发布记录列表**页面，查看离线管道任务的发布状态为**发布成功**即可。

步骤六：生产环境中运行离线管道任务

1. 在数据**集成**页面，单击顶部菜单栏的**运维**。
2. 在运维中心，运行离线管道任务。
   i. 单击左侧导航栏的图标。
   ii. 在**手动任务运维列表**页面，单击离线管道任务test。

   iii. 在离线管道任务test的详情页面，单击页面左上角的**运行**。
   iv. 在**运行**对话框，保持默认参数，单击**确定**。
3. 查看离线管道任务运行生成的实例运行日志。
   i. 单击左侧导航栏的图标。
   ii. 在**手动实例运维列表**页面，单击离线管道实例test。

   iii. 在离线管道实例test的详情页面，单击页面上方的**查看运行日志**。
   iv. 在**运行日志**页面，查看**读出记录总数**和**写出记录总数**。生产环境管道实例运行日志中的读取记录总数、写入记录总数，与开发环境管道任务运行结果的读取总数、写入总数保持一致（均为23）。这样离线管道任务在生产环境可以正常运行，即可保障生产环境业务数据正常产出。

```
2021-04-13 17:44:53.327 [job-1610797] INFO  DlinkTransBase -
任务启动时刻                    : 2021-04-13 17:44:51
任务结束时刻                    : 2021-04-13 17:44:53
任务总计耗时                    :                  1s
任务平均流量                    :             345B/s
记录写入速度                    :            23rec/s
读出记录总数                    :                 23
写出记录总数                    :                 23
读写失败总数                    :                  0
```

2.数据研发

2.1. 规范定义最佳实践

基于Dataphin建模理论和业务需求，明确并规范定义统计指标，以便设计出易于业务使用的数据仓库。

背景信息

规范定义是指以维度建模作为理论基础，构建总线矩阵，划分并定义数据域、业务过程、维度、原子指标、统计周期和派生指标。

在您开始使用Dataphin进行数仓模型设计前，需要完成业务调研、需求分析、构建总线矩阵（从业务数据中抽象出业务过程和维度）、明确并定义统计指标。本教程中假设已完成需求调研、业务分析和构建总线矩阵，带您体验明确并规范定义统计指标，帮助您快速理解如何基于Dataphin设计数仓模型。

基本概念

名词	描述
业务板块	业务板块定义了数据仓库的多种命名空间，是一种系统级的概念对象。当数据的业务含义存在较大差异时，您可以创建不同的业务板块，让各成员独立管理不同的业务，后续数据仓库的建设将按照业务板块进行划分。 在Dataphin中，项目可以归属至业务板块以实现规范建模功能，同一个业务板块中可能包含多个不同的项目，所以业务板块与项目的关系为1:N。
数据域	数据域即主题域，是对某个主题分析后确定的主题边界。例如，商品域、交易域、会员域等。
业务过程	业务过程即企业的业务活动事件，通常为不可拆分的事件。创建业务过程，即从顶层视角，规范业务中的事务内容的类型及唯一性。例如电商订单是一个业务过程，业务过程由下单、支付、发货和确认收货等不可拆分的事件组成，每个事件就是一个业务过程。
统计周期	统计的时间范围，也可以称为时间周期。例如最近1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，可以理解为聚合运算时的分组条件（类似于SQL中Group By的对象）。粒度是维度或维度的某些属性的组合。例如，地域（维度）和客户性别（维度属性）组合成统计粒度。 在定义粒度时，您需要充分考虑到业务和维度的关系。通常用于派生指标构建，是汇总表的唯一性识别方式。
业务限定	统计的业务范围，用于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）。
度量	事实就是度量，通常是对某个业务事件的衡量，通常为数字，如某笔订单的金额。 **注意** 请注意区分度量和原子指标。任何数据仓库都有维度和度量重要概念，但指标是业务分析中的概念。

名词	描述
维度	维度是分析业务的角度，是对应业务流程中的业务对象。例如客户、商品、部门等都可以作为分析业务的角度。
派生指标	派生指标是分析业务的指标。由原子指标、统计周期、统计粒度和业务限定组成。例如原子指标为支付金额，最近1天海外买家支付金额则为派生指标（最近1天为时间周期，海外为业务限定、买家为维度）。
原子指标	原子指标定义了业务分析的度量和统计方法（类似于SQL中Select后的聚合表达式，例如Sum）。

案例

A电商公司，销售某品牌多种零食。

商品种类	单价
干果类	10元/500g
膨化类	8元/袋
饮品类	15元/瓶

买家和卖家可以通过电商平台进行交易。

订单ID	买家ID	买家姓名	商品种类	支付金额	支付方式
29296	1001	张三	干果类	34	花呗
29297			干果类	67	花呗
29298	1003	李四	膨化类	56	支付宝

本案例中，明确及规范定义指标如下。

定义指标	业务数据
业务板块	电商业务
数据域	交易域
维度	商品种类
业务过程	下单购买

定义指标	业务数据
业务限定	商品种类为干果类
时间周期	最近1天
原子指标	销售总额
派生指标	最近1天干果类商品销售总额

2.2. DataX同步数据

DataX是异构数据源离线同步的工具，支持多种异构数据源之间高效的数据同步。Dataphin系统内嵌了DataX组件，支持通过构建Shell任务调用DataX，实现数据同步。本教程以RDS MySQL数据库为例，为您介绍基于Dataphin如何调用DataX同步数据。

前提条件

- 已开通RDS MySQL实例，且RDS MySQL实例的网络类型为专有网络。如何开通RDS MySQL实例，请参见创建RDS MySQL实例。
- 已创建RDS MySQL实例的数据库和账号，创建过程中需要您记录数据库名称、用户名和密码。如何创建数据库和账号，请参见创建数据库和账号。

背景信息

Dataphin系统内嵌了DataX组件，在Dataphin中创建和运行DataX任务（Shell任务）即可将DataX调用起来，以实现数据同步。

DataX支持同步数据的数据源包括MySQL、Oracle、SQL Server、PostgreSQL、HDFS、Hive、HBase等。DataX的更多信息，请参见DataX。

使用限制

Shell任务不支持通过内网地址访问RDS MySQL实例。

操作流程

功能	描述
步骤一：连通RDS MySQL实例与Dataphin间的网络	在您开始同步数据前，首先需要连通RDS MySQL实例和Dataphin间的网络。
步骤二：创建数据同步的源表和目标表	登录至RDS MySQL实例，创建本教程中用于数据同步的源表和目标表。
步骤三：下载并配置DataX任务的代码模板	下载并配置DataX任务的代码模板后，保存为**datax.json**。
步骤四：上传datax.json文件至Dataphin	上传DataX任务代码文件至Dataphin平台后，DataX任务即可调用。
步骤五：创建DataX任务	在开发环境创建并运行同步数据的DataX任务。

功能	描述
步骤六：运行生产环境中的DataX任务	在生产环境运行DataX任务，保障生产环境业务数据的正常产出。

步骤一：连通RDS MySQL实例与Dataphin间的网络

1. 申请RDS MySQL实例的外网地址。如何申请外网地址，请参见申请或释放外网地址。
2. 在数据库连接页面，获取RDS MySQL实例的外网地址和端口。

3. 添加RDS MySQL实例的外网地址和端口至Dataphin项目空间的沙箱白名单。如何添加沙箱白名单，请参见添加沙箱白名单。
4. 添加 `0.0.0.0/0` 至RDS MySQL实例的白名单。如何添加白名单，请参见设置IP白名单。

> **注意** 完成数据同步后，请立即删除 `0.0.0.0/0` 。

步骤二：创建数据同步的源表和目标表

使用命令行方式连接RDS MySQL实例，请参见方法三：使用命令行方式连接实例。创建同步数据的源数据表和目标数据表：

1. 创建源数据表的代码示例如下。

```
create table datax_test1( area varchar(255),province varchar(255) );
insert into datax_test1 values('华北','山东省'),('华南','河南省');
```

2. 创建目标数据表的代码示例如下。

```
create table datax_test2( area varchar(255),province varchar(255) );
```

步骤三：下载并配置DataX任务的代码模板

1. 下载DataX任务的代码模板。
2. 配置代码模板中的如下参数后，保存为datax.json至本地。

参数	描述
{username}	配置为已创建RDS MySQL实例的用户名，即登录数据库的用户名。
{password}	登录数据库的密码。本教程中配置为前提条件已创建RDS MySQL实例的密码，即登录数据库的密码。
{{Public Endpoint}:}	链接地址。本教程配置为已获取的RDS MySQL实例的外网地址。
{DatabaseName}	数据库名称。本教程配置为前提条件中已创建的RDS MySQL实例的数据库名称。

参数	描述
{table_name1}	源数据表的表名。本教程配置为datax_test1。
{table_name2}	目标数据表的表名。本教程配置为datax_test2。
{columnname1}	数据同步的字段。本教程配置为area。
{columnname2}	数据同步的字段。本教程配置为province。

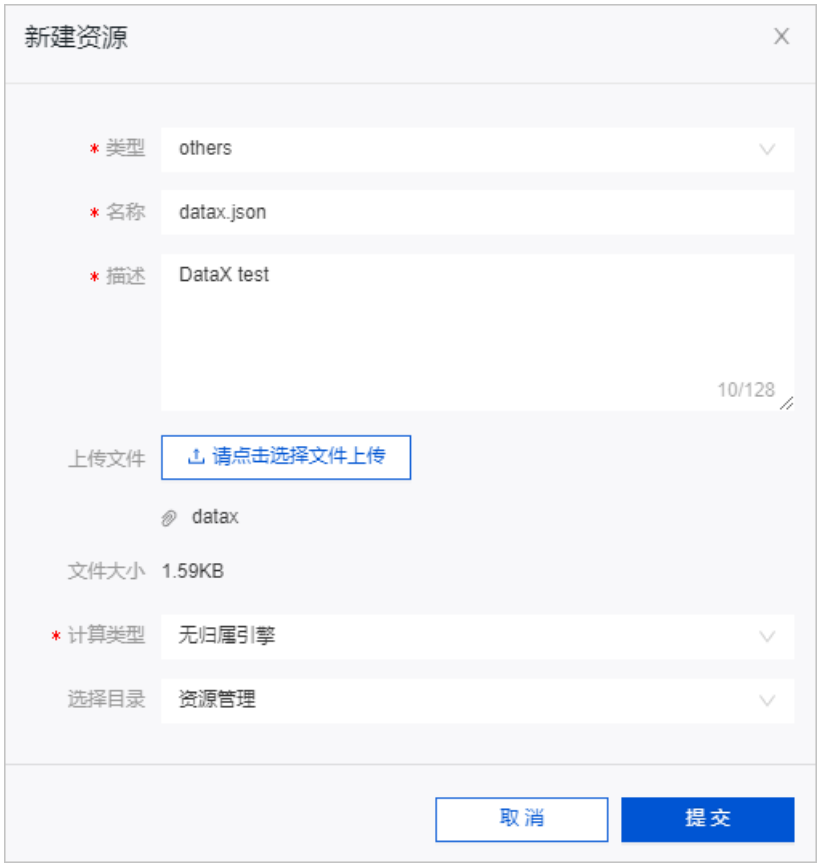
代码模板的代码如下。

```
{
  "job": {
    "content": [
      {
        "reader": {
          "name": "mysqlreader",
          "parameter": {
            "column": [
              "{columnname1}",
              "{columnname2}"
            ],
            "connection": [
              {
                "jdbcUrl":
["jdbc:mysql://{Public Endpoint}:3306/{DatabaseName}"],
                "table": ["{table_name1}"]
              }
            ],
            "password": "{password}",
            "username": "{username}"
          }
        },
        "writer": {
          "name": "mysqlwriter",
          "parameter": {
            "column": [
            "{columnname1}",
            "{columnname2}"
            ],
            "connection": [
              {
                "jdbcUrl":
"jdbc:mysql://{Public Endpoint}:3306/{DatabaseName}",
                "table": ["{table_name2}"]
              }
            ],
            "password": "{password}",
            "username": "{username}"
          }
        }
      }
    ],
    "setting": {
      "speed": {
        "channel": "1"
      }
    }
  }
}
```

步骤四：上传datax.json文件至Dataphin

1. 登录Dataphin控制台。

2. 在DataphinLink控制台页面，选择工作区地域后，单击**进入DataphinLink>>**。
3. 在DataphinLink首页，单击**研发**。
4. 在数据**开发**页面，单击**数据处理**。
5. 在**数据处理**页面的左侧导航栏，单击**资源管理**图标。
6. 在**资源管理**页面，单击**资源管理**后的图标。
7. 在**新建资源**对话框中，配置参数后，单击**提交**。



参数	描述
类型	本教程中需要上传的文件格式为JSON，则类型选择**others**。 系统支持选择的类型包括**file**、**jar**、**python**和**others**，适用场景说明如下： ○ 上传的文件格式为XLS、DOC、TXT、CSV，则类型选择为**file**。 ○ 上传的文件格式为JAR，则类型选择为**jar**。 ○ 上传的文件格式为PY，则类型选择为**python**。 ○ 上传的文件格式非XLS、DOC、TXT、CSV、JAR、PY，则类型选择为**others**。

参数	描述
名称	本教程中的名称为**datax.json**。 本教程名称的命名规则如下： ○ 名称必须以.json结尾。 ○ 字母、数字、下划线（_）或半角句号（.）组合组成。 ○ 不能以数字开头。
描述	填写资源的描述，例如DataX test。
上传文件	选择步骤三：下载并配置DataX任务的代码模板中保存至本地的**datax.json**文件。
计算类型	本教程中上传的资源（**datax.json**）用于DataX代码任务中引用，因此选择**无归属引擎**。 计算类型用于定义资源文件是否需要上传至计算引擎的存储层。Dataphin系统支持的计算类型包括**MaxCompute**、**Flink**和**无归属引擎**，适用场景说明如下： ○ 自定义MaxCompute类型的函数时，**计算类型**选择为**MaxCompute**。 ○ 自定义Flink类型的函数时，**计算类型**选择为**Flink**。 ○ 代码任务引用的资源文件，**计算类型**选择为**无归属引擎**。
选择目录	默认为**资源管理**。

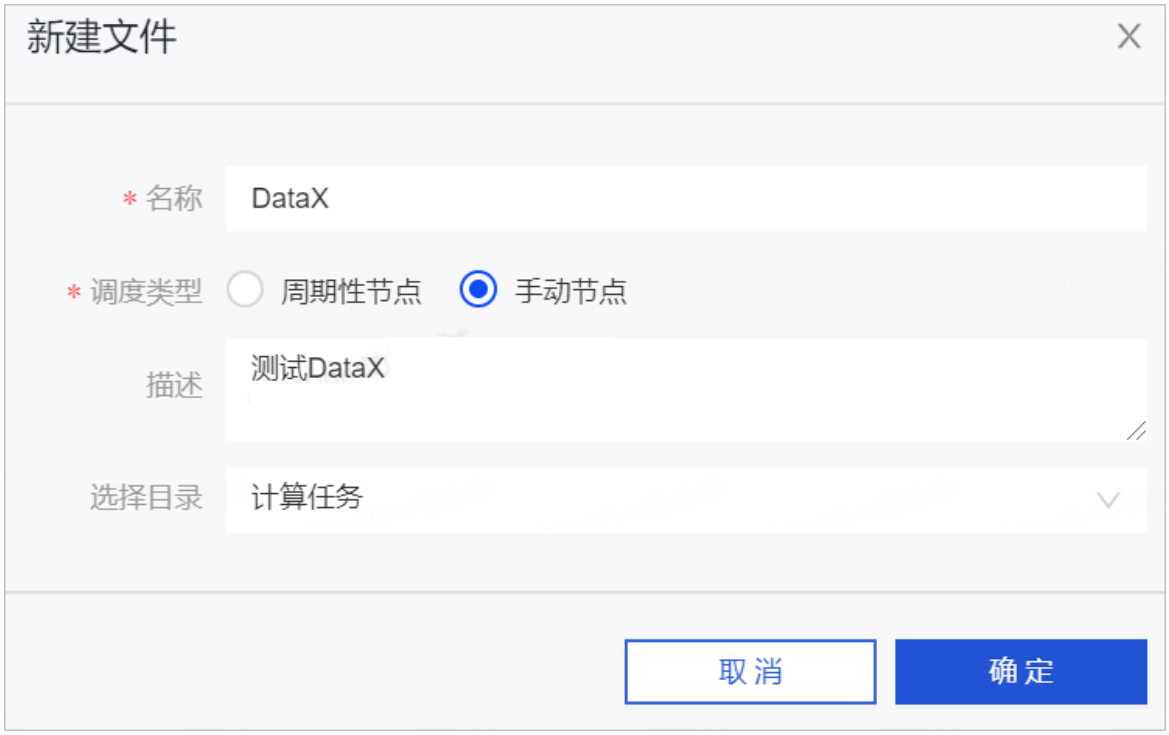
8. 在**提交备注**对话框，填写备注信息后，单击**确定并提交**。
9. 发布资源文件至生产环境。
   i. 在**数据开发**页面，单击顶部菜单栏的**发布**。
   ii. 在**待发布对象列表**页面，单击**数据处理**页签。
   iii. 在**数据处理**页签，单击**datax.json**资源的**操作**列下的图标。

   iv. 在**发布**对话框，填写发布名称或备注信息后，单击**确定**，即可将资源文件发布至生产环境。

v. 单击左侧导航栏的**发布记录列表**。在**发布记录列表**页面，查看资源文件的发布状态为**发布成功**即可。

步骤五：创建DataX任务

1. 请参见步骤四：上传datax.json文件至Dataphin，进入数据**开发**页面。
2. 在数据**开发**页面，单击左侧区域的**数据处理**。
3. 在**数据处理**页面，单击左侧导航栏**计算任务**图标。
4. 在**计算任务**页面，单击**计算任务**后的图标，选择**通用脚本 > SHELL**。
5. 在**新建文件**对话框，配置参数后，单击**确定**。



参数	描述
名称	本教程中的名称为DataX。

参数	描述
调度类型	本教程中选择任务的调度类型为**手动节点**。 调度类型用于定义任务发布至生产环境的调度方式。Datpahin系统支持的调度类型包括**周期性节点**和**手动节点**，适用场景说明如下： ○ 任务需要参与系统的周期性调度，且需要依赖上游节点，则**调度类型**选择为**周期性节点**。 ○ 任务不需要参与系统的周期性调度，且需要依赖上游节点，则**调度类型**选择为**手动节点**。该类型的任务在生产环境的运行需要您手动触发。
描述	填写对任务的简单描述，例如测试DataX。
选择目录	默认为**计算任务**。

6. 在代码编写页面，编写并运行DataX任务的代码。代码如下。

```
@resource_reference{"datax.json"}
python $DATAX_HOME/bin/datax.py datax.json #Dataphin系统已内置DataX的安装目录为DATAX_HOME/bin/datax.py。
```

其中，resource_reference{}用于调用已上传的**datax.json**资源文件。

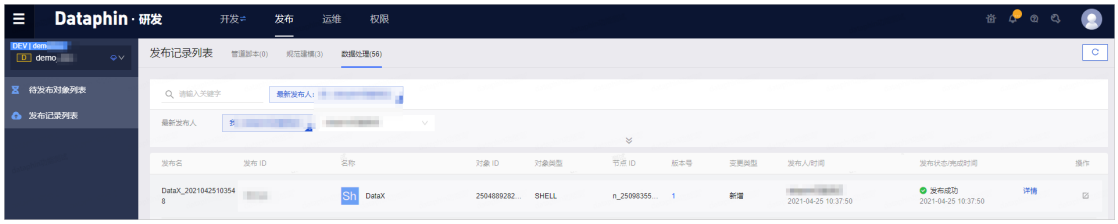
7. 单击页面右上角的**执行**，即可运行DataX任务。运行结果显示的读写失败总数为0时，表示DataX任务同步数据成功。

```
2021-04-25 10:26:56.500 [job-0] INFO  JobContainer -
任务启动时刻                    : 2021-04-25 10:26:44
任务结束时刻                    : 2021-04-25 10:26:56
任务总计耗时                    :                 12s
任务平均流量                    :               1B/s
记录写入速度                    :              0rec/s
读出记录总数                    :                   2
读写失败总数                    :                   0
```

8. 发布DataX任务至生产环境。
   i. 在**计算任务**页面，单击顶部菜单栏的**发布**。
   ii. 在**待发布对象列表**页面，单击**数据处理**页签。
   iii. 在**数据处理**页签，单击DataX任务的**操作**列下的图标。

   iv. 在**发布**对话框，填写发布名称或备注信息后，单击**确定**，即可将DataX任务发布至生产环境。

v. 单击左侧导航栏的**发布记录列表**。在**发布记录列表**页面，查看DataX任务的发布状态为**发布成功**即可。

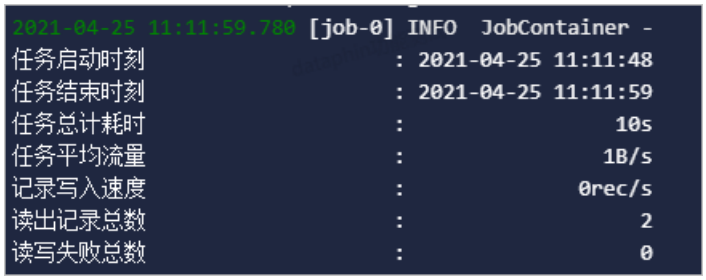


步骤六：运行生产环境中的DataX任务

1. 请参见步骤四：上传datax.json文件至Dataphin，进入数据**开发**页面。
2. 在数据**开发**页面，单击顶部菜单栏的**运维**。
3. 在**运维**中心，单击项目名称后的Prod Dev图标，切换至生产环境（Prod Dev）。
4. 在**运维**中心，运行DataX任务。
   i. 单击左侧导航栏的图标。
   ii. 在**手动任务运维列表**页面，单击DataX任务。
   iii. 在DataX任务的详情页面，单击页面左上角的**运行**。

   iv. 在**运行**对话框，保持默认参数，单击**确定**。
5. 查看DataX任务运行生成的实例运行日志。
   i. 单击左侧导航栏的图标。
   ii. 在**手动实例运维列表**页面，单击DataX任务。
   iii. 在DataX任务的详情页面，单击页面上方的**查看运行日志**。

iv. 在**运行日志**页面，查看**读写失败总数**。运行日志显示的读写失败总数为0时，表示DataX任务在生产环境同步数据成功，即可保障生产环境业务数据正常产出。



2.3. 使用Python读文件

Dataphin仅支持开发基于Python的脚本，不支持开发依赖第三方组件的脚本。开发基于第三方组件的脚本，需要通过pip install下载第三方组件。本文为您介绍基于Dataphin如何通过构建Shell任务调用Python读取第三方文件。

前提条件

- 添加访问地址mirrors.aliyun.com和端口*至项目空间的沙箱白名单，详情请参见设置IP白名单。
- 已准备Python支持读取的文件，例如TXT、CSV、XLS、XLSX或PDF等格式文件。

步骤一：上传文件

1. 登录Dataphin控制台。
2. 在Dataphin控制台页面，选择工作区地域后，单击**进入Dataphin>>**。
3. 进入**资源管理**页面。
   i. 在Dataphin首页，单击**研发**。
   ii. 在数据**开发**页面，单击**数据处理**。
   iii. 在左侧导航栏，单击**资源管理**图标。
4. 在**资源管理**页面，单击**资源管理**后的图标。
5. 在**新建资源**对话框中，配置参数。

参数	描述
类型	选择**others**。
名称	上传文件的名称需要以文件类型结尾。例如test.xlsx。
描述	填写资源的描述。
上传文件	选择本地的文件，例如test.xlsx。
计算类型	选择**无归属引擎**。 **注意** 文件资源存储至Dataphin系统，因此仅支持选择**无归属引擎**。
选择目录	默认为**资源管理**。

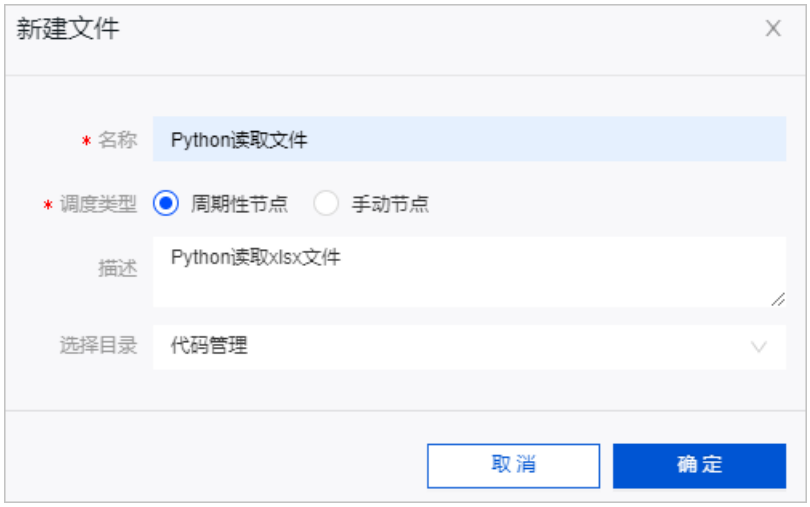
6. 单击**提交**，完成资源的提交。
7. 在**提交备注**对话框，填写备注信息。
8. 单击**确定并提交**。

步骤二：创建Shell任务

1. 在**数据处理**页签，单击左侧导航栏**计算任务**图标。
2. 在**计算任务**页面，单击**计算任务**后的图标，选择**通用脚本 > SHELL**。

3. 编写DataX任务代码。

   i. 在**新建文件**对话框，配置参数。



参数	描述
名称	填写计算任务的名称，例如Python读取文件。
调度类型	选择任务的调度类型为**周期性节点**。
描述	填写对任务的简单描述。
选择目录	系统自动选择为**代码管理**。

   ii. 单击**确定**。

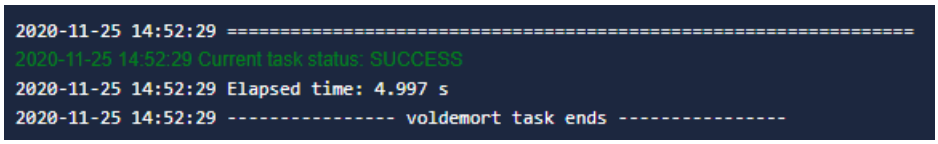
步骤三：编写并运行Shell任务代码

1. 在代码编写页面，编写代码。

```
#在Dataphin的Linux服务器上新建目录。
mkdir -p /tmp/chars/ && \
#指定目录/tmp/chars/为python源。
pip install -i https://mirrors.aliyun.com/pypi/simple/ \
--target=/tmp/chars/ \
openpyxl
#指定的python源写入至openfile.py。
cat >openfile.py <<EOF
@resource_reference{"test.xlsx"}
# -*- coding:utf-*-
import os
import sys
sys.path.append('/tmp/chars/')
import openpyxl
print '========= python execute ok =========='
print("start===============")
args = sys.argv
# 打开excel文件，获取sheet名
wb = openpyxl.load_workbook(args[1])
#  wb.get_sheet_names 这个方法已过时 会有一个警告
print(wb.worksheets[0])
EOF
#python中调用文件。
python openfile.py test.xlsx
```

其中，test.xlsx参数需要替换为您已上传的文件。

2. 单击页面右上角的**执行**，即可运行任务代码。运行结果的状态为SUCCESS，表示读取文件成功。



2.4. Java UDF最佳实践

为了满足复杂的数据开发场景，Dataphin智能研发版支持自定义Java UDF函数。本教程以Java自带函数（toLowerCase）为例，为您介绍如何基于Dataphin自定义Java UDF函数。

前提条件

下载JAR包。

背景信息

本教程基于下载的JAR包自定义的Java UDF函数，实现大写字母转换为小写字母。您也可以编写Java UDF代码，以实现更多的功能，请参见IntelliJ IDEA Java UDF开发最佳实践。

本教程中的JAR包的代码如下。

```
package org.alidata.odps.udf.examples;
import com.aliyun.odps.udf.UDF;
public final class javaudf extends UDF {
 public String evaluate(String s) {
  if (s == null) {
    return null;
  }
    return s.toLowerCase();
 }
}
```

其中，

- JAR包路径为org.alidata.odps.udf.examples。
- class文件名为javaudf。

步骤一：上传JAR包

1. 登录Dataphin控制台。
2. 在Dataphin控制台页面，选择工作区地域后，单击**进入Dataphin>>**。
3. 进入**资源管理**页面。
   i. 在Dataphin首页，单击**研发**。
   ii. 在数据**开发**页面，单击**数据处理**。
   iii. 在左侧导航栏，单击**资源管理**图标。
4. 在**资源管理**页面，单击**资源管理**后的图标。
5. 在**新建资源**对话框中，配置参数。

参数	描述
类型	选择**jar**。
名称	上传文件的名称需要以文件类型结尾。例如javaudf.jar。
描述	填写资源的描述。
上传文件	选择本地JAR文件，例如javaudf.jar。
计算类型	选择**MaxCompute**。
选择目录	选择用于存放JAR包的目录。系统默认为**资源管理**，保持默认即可。

6. 单击**提交**，完成资源的提交。
7. 在**提交备注**对话框，填写备注信息。
8. 单击**确定并提交**。
9. （可选）发布资源至生产环境。
   - 如果您的开发模式是Dev-Prod模式，则需要发布资源至生产环境，详情请参见管理发布任务。
   - 如果您的开发模式是Basic模式，则提交成功的资源，即可进入生产环境。

步骤二：创建MAXC函数

1. 在**数据处理**页签，单击左侧导航栏的**函数管理**图标。
2. 单击**函数管理**后的图标，选择**MAXC函数**。

3. 在**新建函数**对话框，配置参数。

参数	描述
名称	填写函数的名称，例如java
选择资源	选择已上传的资源**javaudf.jar**。
类名	类名的格式为JAR包路径.class文件名。填写org.alidata.odps.udf.examples.javaudf。
类型	函数的类型。选择**字符串**。
命令格式	定义引用函数的格式。填写to_char(string i)。
使用文档	填写函数的使用描述，例如javaudf。
选择目录	默认为MAXC函数-用户自定义函数，保持默认。

4. 单击**提交**，完成资源的提交。
5. 在**提交备注**对话框，填写备注信息。
6. 单击**确定并提交**。
7. （可选）发布函数至生产环境。
   - 如果您的开发模式是Dev-Prod模式，则需要发布函数至生产环境，详情请参见管理发布任务。
   - 如果您的开发模式是Basic模式，则提交成功的函数，即可进入生产环境。

步骤三：新建SQL任务

1. 在**数据处理**页签，单击左侧导航栏**计算任务**图标。

2. 在**计算任务**页面，单击**计算任务**后的图标，选择**MAXC任务 > MAX_COMPUTE_SQL**。

3. 在**新建文件**对话框，配置参数。

参数	描述
名称	填写计算任务的名称，例如javaudf。
调度类型	选择任务的调度类型为**周期性节点**。
描述	填写对任务的简单描述。
选择目录	系统自动选择为**计算任务**。

4. 单击**确定**。

步骤四：使用Java UDF函数

1. 在SQL任务的代码编写页面，编写代码，例如 `select java('ABCGDfagHH');` 。
2. 单击页面右上方的**执行**，查看运行结果。

（可选）

（可选）调度运维

如果需要定期的运行SQL任务，则需要配置SQL任务的调度参数并发布至生产环境，参与生产环境的调度。

1. 在代码编写页面，单击页面上方的**调度配置**，配置调度参数，详情请参见调度配置。
2. 保存、提交和发布SQL任务。
   i. 单击页面右上方的图标，保存代码。
   ii. 单击页面右上方的图标，提交代码。
   iii. 在**提交备注**对话框，填写备注信息。
   iv. 单击**确定并提交**。
   v. （可选）发布SQL任务至生产环境。
      - 如果开发模式是Dev-Prod模式，则需要发布SQL任务至生产环境，详情请参见管理发布任务。
      - 如果开发模式是Basic模式，则提交成功的SQL任务，即可进入生产环境。