用户手册

概述

鲁班是天翼云基于开源产品合作共建的一站式大数据开发与服务平台，支持细粒度资源管控、多租户隔离、高可用，高并发，可扩展，高性能等优秀特性。通过高度融合业界前沿的各种大数据技术，为用户提供从数据交换、存储、开发、治理、可视化、服务、安全等全链路大数据解决方案。

当前鲁班中融合的上层核心开源产品是基于微众银行开源的WeDataSphere大数据平台套件，如DataSphere Studio0.9.0、Linkis0.9.4等产品联合打造而成的，具备企业级数据开发、细粒度资源管控、金融级数据安全等能力；底层是Apache Hadoop 3.2.1，Apache Spark 3.0.1等产品，支持存储EC、跨机房容灾，更高更智能的性能等优秀特性。

鲁班能够轻松处理多种数据，并进行**离线计算**、**实时计算**、**交互式查询**等，同时，可以**发布数据API**，轻松开放数据能力。利用鲁班能够在线编写数据处理程序，从而完成数据的清洗、加密、脱敏、打标、元数据抽取等多方面的数据治理任务，让您摆脱繁杂的系统运维工作，专注于数据本身。

优势

技术领先：采用最新稳定版本的大数据技术，并与国内领先的大数据团队共同维护，快速提升数据处理效率

开箱即用：产品内包含的通信、金融企业级的大数据处理模板工具，让您在自己的数据系统中快速使用我们多年沉淀的数据处理经验及方法

可视化开发：鲁班提供了开发IDE套件，支持在线编写多种语言脚本，并一键提交到大数据集群执行，返回执行状态和结果；支持页面拖拽式的工作流编辑与发布；丰富的数据可视化组件，拖拽式配置大屏或BI展示且支持UDF、函数、资源管控和智能诊断等企业级特性

场景

场景1：电信行业大数据分析 利用大数据技术实现客户离网分析，及时掌握客户离网倾向，出台客户挽留措施

场景2：旅游行业大数据分析 利用旅游大数据分析，帮助客车站，火车站，机场，旅游景点，检测人流信息，热门景点，并提供人流引流，门票售价等决策信息支撑

场景3：工业大数据分析 利用工业大数据提升制造业水平，包括产品故障诊断与预测、改进生产工艺，优化生产过程能耗、生产计划与排程

架构

数据开发的流程如下：

数据产生：业务系统每天会产生大量数据，存储在业务系统所对应的数据库中，包括结构化和非结构化数据

数据集成：您需要同步不同业务系统的数据至鲁班中，方可通过鲁班的海量数据存储与处理能力分析已有的数据。鲁班提供数据集成服务，可以支持多种数据源类型，根据预设的调度周期同步业务系统的数据。

数据开发：完成数据的同步后，可以对数据进行加工、分析与挖掘（数据开发）等处理，从而发现其价值。

数据使用：分析与处理后的结果数据，可以供业务人员使用其分析的价值，并可做数据治理。

数据展现与发布：处理成功后，可以通过报表、大屏服务展示大数据分析、处理后的成果；也可发布数据api开放数据能力。

快速体验诸葛AI大数据平台

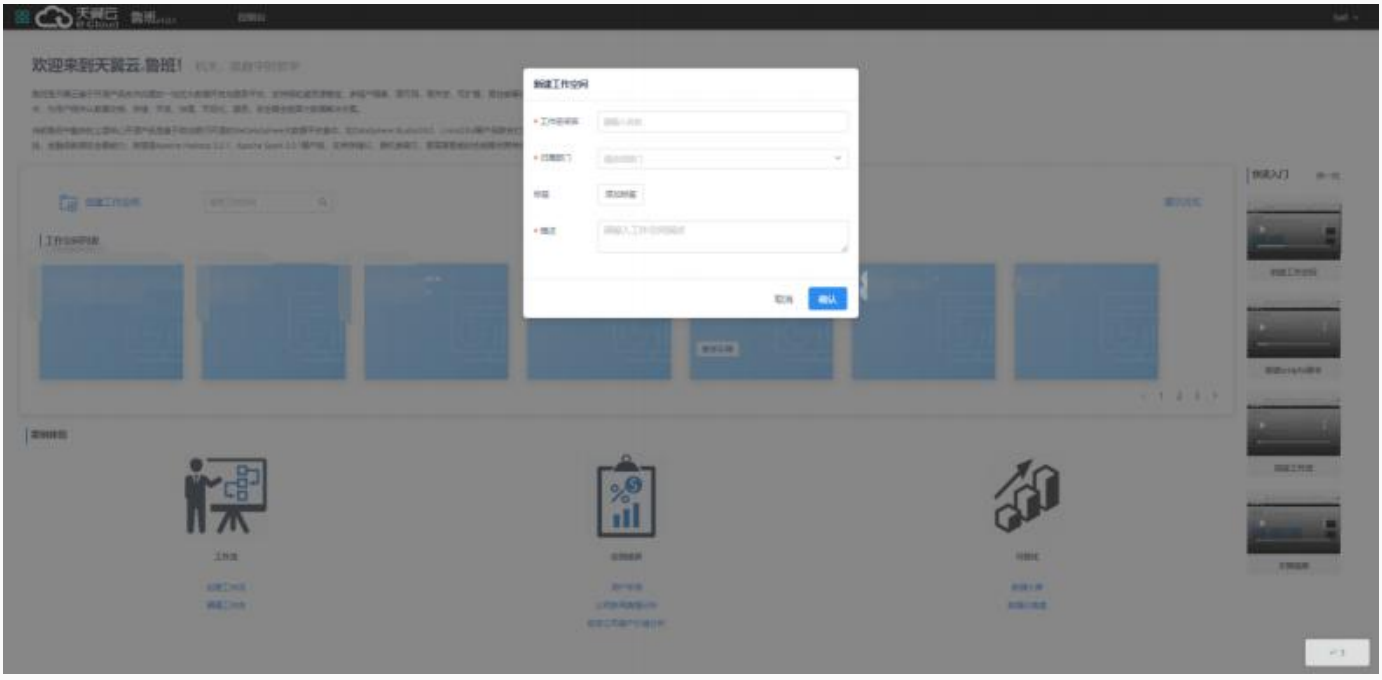
快速登录

已创建天翼云账号并开通诸葛AI大数据平台服务后，即可使用此账号登录诸葛AI大数据平台。

创建工作空间

登录后，点击创建工作空间按钮

填写必要的信息后点击确认即可创建工作空间



创建工程

进入已有的工作空间，点击 应用开发 标签页并点击进入 工作流开发

点击创建工程，填写关键信息即可创建：

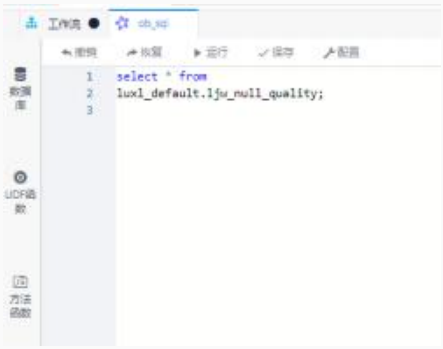
创建工作流

进入新建好的工程，点击创建工作流，填写关键信息即可创建：

工作流编辑

1. 点击左边的对应的节点拖入到右侧编辑框
2. 点击节点右边会弹出该节点的配置信息：

3. 双击节点，可以跳转对应节点系统的编辑页面：



编辑完成后保存脚本，关闭脚本标签，回到工作流页面保存工作流。

工作流运行

点击执行按钮便可开始运行

开始运行后，右击脚本点击管理台

便会弹出任务运行进度，点击左侧日志按钮便可查看日志

工作流调度

在从工作空间进入创建工程页面的工程展示区域，可以将已有工程发布到调度系统进行 运行：

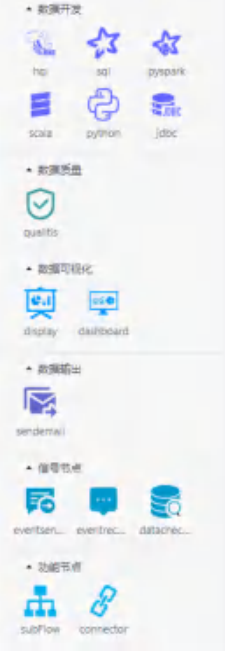
点击发布，并填写必要信息即可发布至调度系统

发布成功后会显示已发布：

此时便可进入调度系统进行执行或定时执行等设置。

工作流节点类型

在数据开发----工作流中，有多种工作流节点类型可供选择



Spark节点

spark节点分别支持sql、pyspark、scala三种方式执行spark任务，使用时只需将节点拖拽至工作台后编写代码即可。

Hive节点

hive节点支持sql方式执行hive任务，使用时只需将节点拖拽至工作台后编写hivesql代码即可。

Python节点

python节点支持执行python任务，使用时只需将节点拖拽至工作台后编写python代码即可。

JDBC节点

jdbc节点支持以jdbc方式运行sql命令，使用时只需将节点拖拽至工作台后编写sql即可，**注意需要提前在linkis console管理台配置jdbc连接信息。**

Qualitis节点

qualitis是数据质量校验节点，提供数据质量模型构建，数据质量模型执行，数据质量任务管理，异常数据发现保存以及数据质量报表生成等功能。在此节点中可以配置数据校验规则，嵌入工作流进行数据质量管控。

SendEmail节点

SendEmail节点一般作为工作流的最后一个节点，用于将工作流前面的结果信息进行发送，支持发送表格、文本、DashBoard、Display、图片等。

EventSender节点：

EventSender节点用于进行信息发送，将一段信息事件进行发送给eventReceiver。常见场景工程与工程间有依赖，工作流与工作流间有信息依赖。比如B任务依赖A任务的某些信息（A任务成功B节点才能开始执行。

EventReceiver节点：

EventReceiver节点用于接收eventSender发送过来的消息，并将接收过来的消息内容存放到工作流的上下文中，后续的节点会根据前缀去找该信息进行使用比如作为自定义变量进行使用。

DataCheck节点：

DataCheck节点用于检测数据是否ready，可以判断hive库中某个表或者分区是否存在，如果存在则进行下游的执行，在有数据依赖的任务中扮演十分重要的作用替换掉以前口头约定好的时间开始运行。

Connector节点

Connector节点的作用是为了作为节点与节点的连接，让工作流更加美观。

Subflow节点

Subflow节点是您可以在一条工作流中嵌入一条子工作流，子工作流支持发布调度，但是在实时执行时父工作流的subflow节点会跳过执行，需要跳到子工作流编辑页面进行执行。

建表并上传数据

您可以参考每个步骤的示例部分进行实际操作。

在本平台中的创建表的方式有以下两种：

- 通过Scriptis编辑运行脚本实现。
- 通过数据开发中工作流内编辑运行脚本实现

本文将为您介绍如何创建、查看表。

前提条件

请确保以下工作已经完成：

- 已开通天翼云诸葛AI大数据平台账号。
- 已创建好要使用的项目空间

创建表

1. 登录账号，进入Scriptis或工作流
2. 新建hql脚本（工作流中拖拽到右侧并双击）
3. 编辑脚本

`create table {table_name} ({columns}) partitioned by ({partition}) row format delimited fields terminated by "," stored as orc;`

建表语句示例如下

```
use database_name;
show tables;

drop table if exists ODS_T_NEWS;
--select * from ct568646781_default.ODS_T_NEWS limit 200;
--select count(1) from ct568646781_default.ODS_T_NEWS;
create table if not exists ODS_T_NEWS (
`ID` STRING comment "ID",
`TITLE` STRING comment '新闻标题',
`DATE` STRING comment '新闻日期',
`COMPNAME` STRING comment '公司名字',
`NEWSCODE` STRING comment '新闻编号码',
SENTIMENT STRING  comment '新闻评分，正面为1，中性为0，负面为-1' )
partitioned BY (dt STRING)
stored as ORC;

```

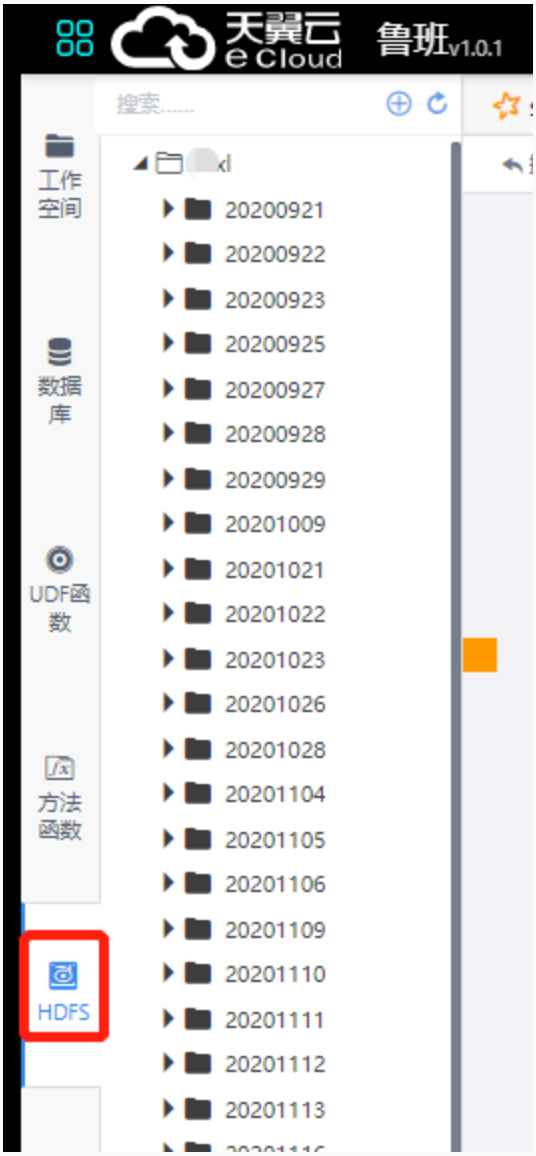
```
/**
  hive table ddl
*/

drop table if exists DWS_T_NEWS;
--select * from ct568646781_default.DWS_T_NEWS;
create  table if not exists DWS_T_NEWS (
`STATISTICDATE` STRING comment '统计日期',
`COMPNAME` STRING comment '公司名字',
`POSCOUNT` int  comment '新闻评分正面为1条数',
`NEUTRALCOUNT` int  comment '新闻评分中性为0条数',
`NEGCOUNT` int  comment '新闻评分负面为-1条数' )
stored as ORC ;

drop table if exists DWA_T_NEW;
--select * from ct568646781_default.DWA_T_NEW;
create table if not exists DWA_T_NEW (
`STATISTICDATE` STRING comment '统计日期',
`COMPNAME` STRING comment '公司名字',
`WEEK1COUNT` int  comment '新闻评分正面为-1条数' ,
`WEEK2COUNT` int  comment '新闻评分正面为-1条数' ,
`WEEK3COUNT` int  comment '新闻评分正面为-1条数' ,
`WEEK4COUNT` int  comment '新闻评分正面为-1条数' ,
`WEEK5COUNT` int  comment '新闻评分正面为-1条数' ,
`WEEK6COUNT` int  comment '新闻评分正面为-1条数' ,
`WEEK7COUNT` int  comment '新闻评分正面为-1条数' ,
`WEEK8COUNT` int  comment '新闻评分正面为-1条数' )
stored as ORC ;
```

4. 编辑好后直接运行上述建表语句即可

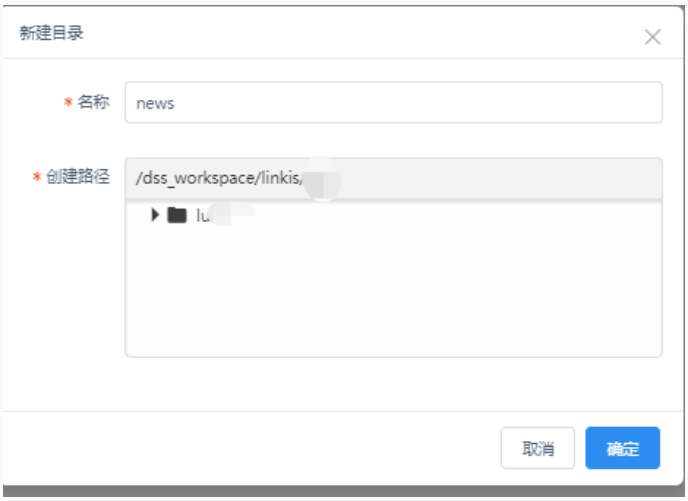
上传数据

以导入本地文件sample2020.csv至平台为例，操作如下：

1. 进入Scriptis，点击左侧HDFS标签

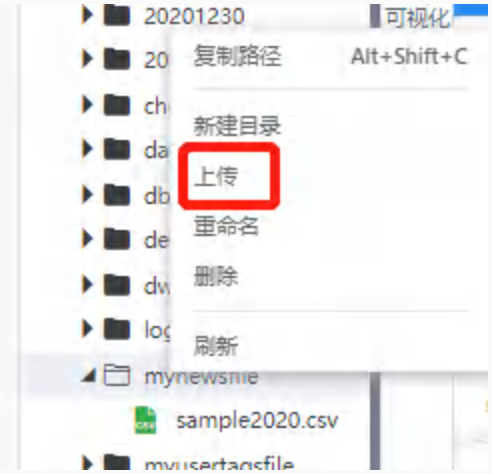


2. 新建HDFS文件目录



3. 上传文件

点击右键上传，选取本地文件后确定，即可将本地文件上传到HDFS上



4. 加载数据

```
import java.text.SimpleDateFormat
    import java.util
    import java.util.{Calendar, Random}

    import com.google.common.base.Strings
    import org.apache.spark.broadcast.Broadcast
    import org.apache.spark.sql.{SQLContext, SaveMode, SparkSession}

    //val spark: SparkSession = SparkSession.builder().getOrCreate()

    val sqlCtx: SQLContext = spark.sqlContext
    sqlCtx.setConf("hive.exec.dynamici.partition","true")
    sqlCtx.setConf("hive.exec.dynamic.partition.mode","nonstrict")


    def pre_n_week(date:String,n:Int)={
      val dateFormat = new SimpleDateFormat("yyyy-MM-dd")
      val calendar = Calendar.getInstance()
      calendar.setTime(dateFormat.parse(date))
      calendar.set(Calendar.DAY_OF_WEEK,Calendar.MONDAY)

      val monday = new SimpleDateFormat("yyyy-MM-dd").format(calendar.getTime)
```

```
23        calendar.setTime(dateFormat.parse(monday))
24        calendar.set(Calendar.DATE, -7* (n-1) )
25        val last_sun = new SimpleDateFormat("yyyy-MM-dd").format(c
alendar.getTime)
26        calendar.setTime(dateFormat.parse(monday))
27        calendar.set(Calendar.DATE,  - 7*n + 1)
28        val last_mon = new SimpleDateFormat("yyyy-MM-dd").format(c
alendar.getTime)
29        (last_mon,last_sun)
30      }
31
32      val date_str:String = "2020-06-06"
33
34      val current_date = pre_n_week(date_str,1)._2
35      val dtFormat = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss")
36      val date_arry = new util.ArrayList[String]();
37      for(i <- 0 to 300){
38        val calendar: Calendar = Calendar.getInstance()
39        calendar.setTime(new SimpleDateFormat("yyyy-MM-dd").parse(
current_date))
40        calendar.add(Calendar.DATE,-i)
41        date_arry.add(dtFormat.format(calendar.getTimeInMillis))
42      }
43
44      val schemaString = "ID,TITLE,DATE,DT,COMPNAME,NEWSCODE,SENTI
MENT"
45
46      val dtBroad: Broadcast[util.ArrayList[String]] = spark.spark
Context.broadcast(date_arry)
47      val fileDF = spark.read.format("csv").option("header", true)
.load("hdfs://ctyunns1/dss_workspace/linkis/ct568646781/dbs1124/
sample2020.csv") //此处需替换为你自己的文件地址
48
49      import spark.implicits._
50
51      val ods_t_newsDF = fileDF.na.fill("unknow")
52        .filter(x => {
53          var flag = false
54          if(!Strings.isNullOrEmpty(x(0).toString.trim)
55            && !Strings.isNullOrEmpty(x(1).toString.trim)
```

```
56            && !Strings.isNullOrEmpty(x(2).toString.trim)
57            && !Strings.isNullOrEmpty(x(3).toString.trim)
58            && !Strings.isNullOrEmpty(x(4).toString.trim)
59            && x(4).toString.trim.length >20
60            && !Strings.isNullOrEmpty(x(5).toString.trim) ) {
61            flag = true
62          }
63          flag
64        })
65        .map(x => {
66
67          val buffer = new StringBuffer()
68          for(i <- 1 to 10){
69            val x1 = x(1).toString + new Random().nextInt()
70            val index = Math.abs(new Random().nextInt(Integer.MAX_
   VALUE)) % 66
71            val x2 = dtBroad.value.get(index)
72            var x5 = x(5)
73            if(x5.toString.trim.isEmpty) {
74              x5 = -1
75            }
76            buffer.append("," + x(0).toString)
77            buffer.append("|" + x1)
78            buffer.append("|" + x2)
79            buffer.append("|" + x2.split(" ")(0))
80            buffer.append("|" + x(3).toString)
81            buffer.append("|" + x(4).toString)
82            buffer.append("|" + x5.toString + " ")
83          }
84          buffer.toString.replaceFirst(",","")
85        })
86        .flatMap(x => {
87          val str_arr = x.split(",")
88          str_arr
89        })
90         .filter(x => {
91          val arr = x.split("\\|")
92          arr.length == 7 && arr.apply(3).split("-").length == 3
93        })
94        .map(x => {
```

```
95          var str_list = x.+("  ").split("\\|")
96          (str_list(0),str_list(1),str_list(2),str_list(3),str_lis
    t(4),str_list(5),str_list(6).trim)
97        })
98        .toDF(schemaString.split(","): _*)
99
100     ods_t_newsDF.show(10)
101
102  ods_t_newsDF.write.format("hive").partitionBy("DT").mode(SaveMo
    de.Append).saveAsTable("ct568646781_default.ODS_T_NEWS")
```

查看表

当创建表成功之后，您可以通过如下命令查看表的信息。

`desc <table_name>;`

其中，table_name是查看表的名字。例如，您可执行命令

`desc ODS_T_NEWS ;`

其他表操作

删除表

`drop table if exists {table_name};`

创建分区

`ALTER TABLE table_name DROP [IF EXISTS] PARTITION partition_spec[, PARTITION partition_spec, ...] ;`

删除分区

`ALTER TABLE table_name DROP [IF EXISTS] PARTITION partition_spec[, PARTITION partition_spec, ...];`

创建一个工作流

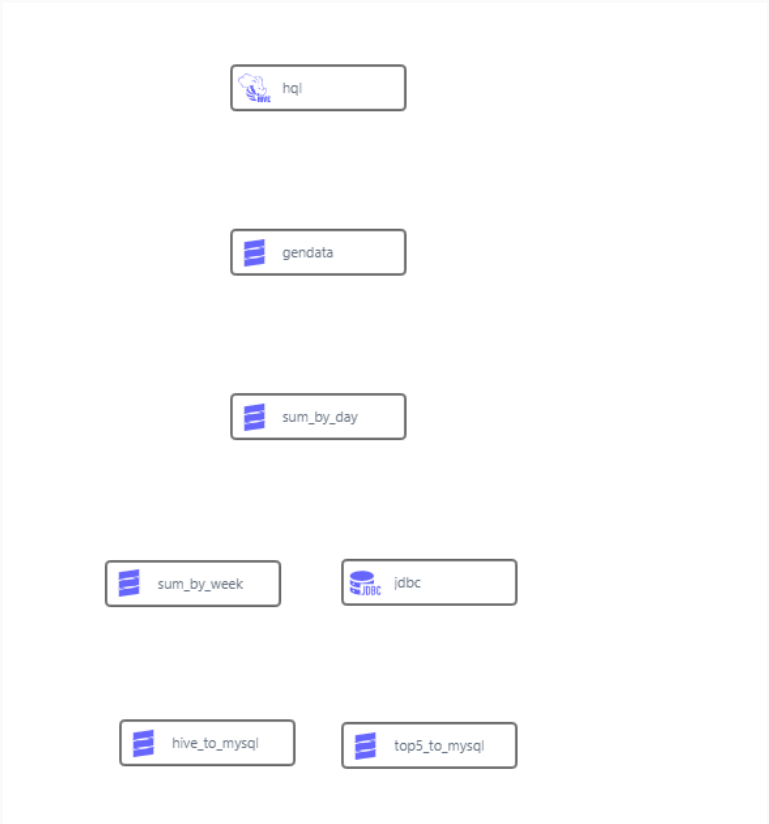
快速入门为您演示一个使用天翼云诸葛AI大数据平台对新闻舆情数据进行分析的过程。本文将依托新闻舆情分析案例为您介绍如何创建工作流、在工作流中创建节点并配置依赖关系。

1. 进入已创建好的工作空间
2. 点击应用开发，工作流开发

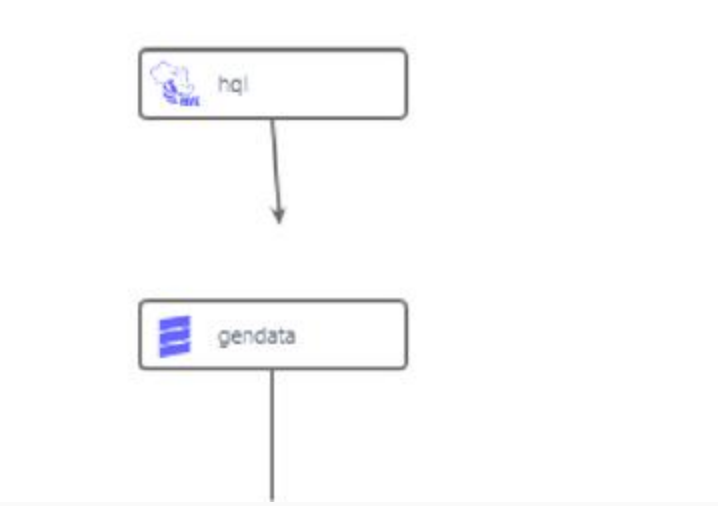
3. 创建一个工程，填写必要信息后点击确定

4. 进入创建好的工程，如新闻舆情分析并创建一个工作流

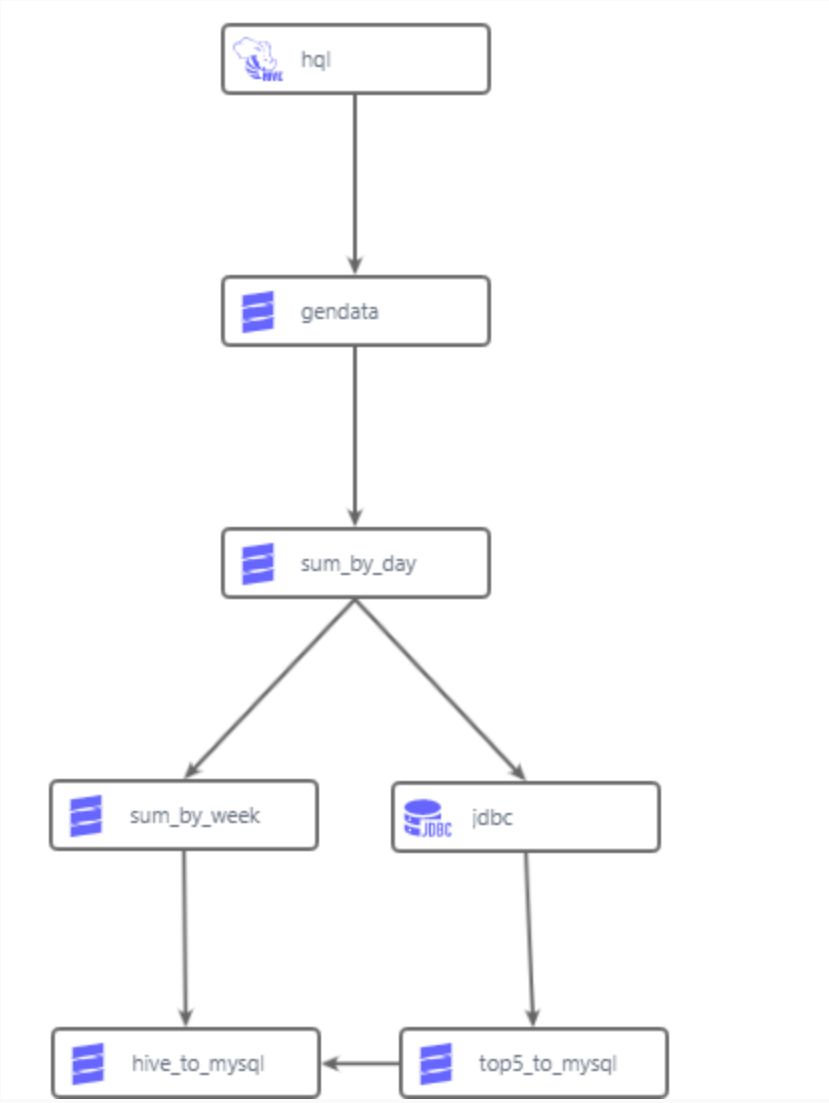
5. 将左侧需要用的节点拖拽到右侧画布中,并重命名



6. 双击节点即可打开节点编辑内容，单击节点可以在右侧给节点配置信息
7. 编写好脚本逻辑后拖拽连接工作流中任务节点 ， 确定各个节点的依赖关系



8. 连接好后如下图示



说明:

节点支持右键功能包括，删除、依赖选择、复制等基本功能，同时数据开发节点还支持脚本关联。

在实时运行工作流时，可以右键管理台来查看运行日志。

开发Spark方法函数(可选)

背景信息

在熟悉scala语法时，可在本平台快速开发应用于sparksql节点的方法函数，从而不用在本地使用Java开发UDF，省去了打包，上传等操作，更加方便快捷。

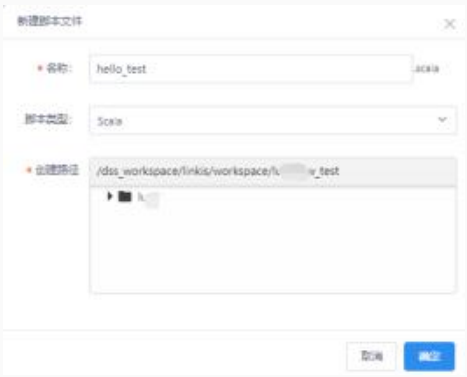
编写一个通用函数示例

1. 进入scriptis

从某个工作空间进入scriptis脚本开发界面

2. 新建一个scala节点

在工作空间下新建一个scala节点，命名为hello_test



3. 编写脚本，此处示例为给某个字符串加上‘hello’前缀

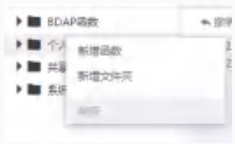
```
def helloWorld(str: String): String = "hello, " + str
```

将上面一行代码写入hello_test.scala脚本中并保存

4. 注册函数

在左侧标签中找到UDF函数，

点击个人函数目录并右键点击新增函数



在弹出的对话框中填入必要信息

说明：函数名称可自定义；函数类型选择spark，脚本路径选择之前编辑的scala文件的位置，注册格式依次填写返回值类型，参数类型，scala函数名，下面会自动生成一个预览，函数描述选填；

将以上信息填好后保存函数，并确保已在个人函数列表中显示并打勾。

5. 在脚本中使用

新建一个sparksql节点，编写sql

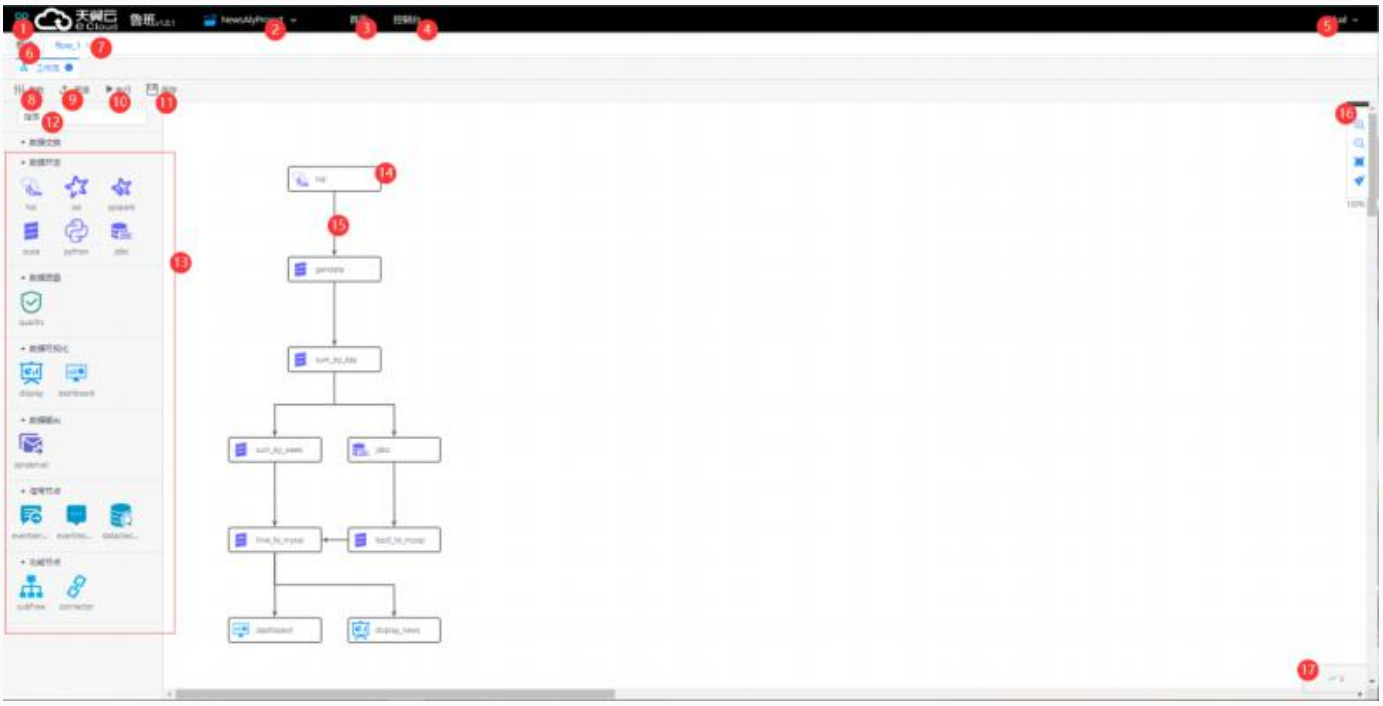
```
select hello_test("t222");
```

点击运行后

见到预期结果，运行成功。

界面功能

在进入到 **数据开发>工作流开发** 页面后，可以看到如下图示的界面：



1. 导航按钮

鼠标悬浮在此按钮上可显示导航信息

2. 工作空间显示

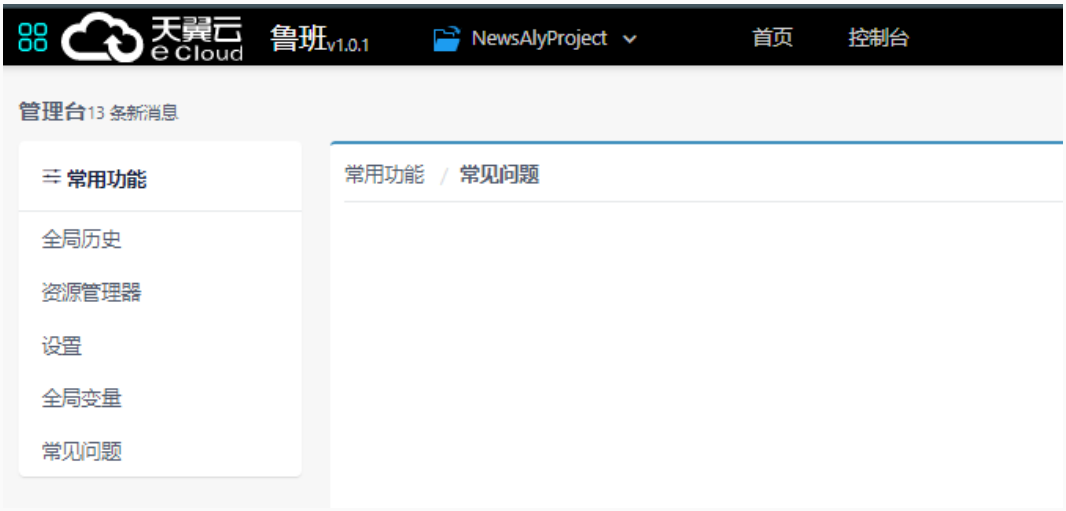
显示当前工作空间名称

3. 首页链接

点击可跳转至首页

4. 控制台按钮

点击可进入控制台



5. 用户信息按钮

6. 工程概览按钮

可进入当前工作流上一层界面

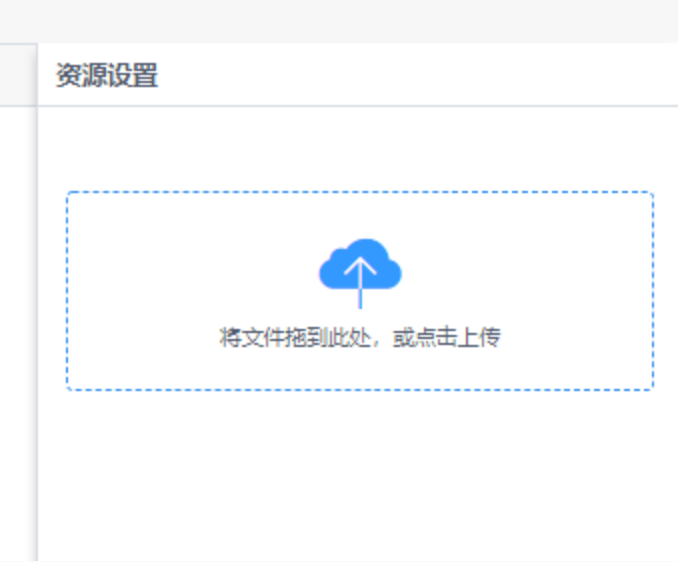
7. 工作流标签

8. 工作流参数

点击后可在右侧设置参数信息

9. 工作流资源

可点击此按钮上传工作流需要用到的资源，比如各类配置信息等



10. 工作流执行按钮

点击此按钮可实时执行工作流

11. 工作流保存按钮

点击保存当前工作流

12. 搜索框

可以快速搜索所需节点名称

13. 可供使用的工作流任务节点列表栏

此区域集中展示所有可使用的节点类型

14. 某一个工作流节点

工作流中的某个节点

15. 节点间连线

连线代表依赖关系，执行时会根据依赖关系顺序执行

16. 工具栏

由上到下依次为放大，缩小，全屏，节点排布格式化

创建工作流

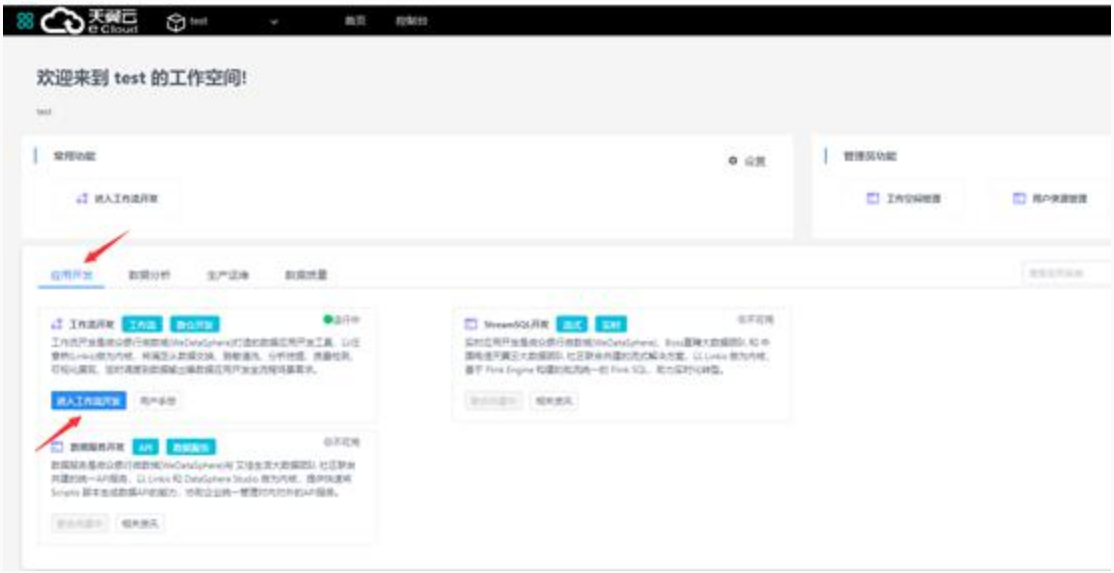
准备工作

1. 开通账号并拥有CPU，内存等资源
2. 已创建好一个工作空间

准备工作完成后即可进入以下步骤。

进入工作流开发页面

进入某一工作空间后，点击 **应用开发>进入工作流开发**，如下图。



系统跳至工作流开发首页

创建工程

点击 **创建工程**，弹出创建工程对话框，如下图，填写相应必要信息

点击确认便可在此页面看到新建的工程，进入新建的这个工程。

创建工作流

进入工程后点击 **创建工作流**

填入必要信息后点击确认即可创建工作流。

编辑工作流

在创建好工作流后便可进行工作流编辑。

工作流节点脚本编辑

脚本的编辑有两种方式：

1. 双击节点即可打开节点进入编辑。
2. 右键节点点击关联，便可将scriptis中编辑好的脚本关联到此工作流的节点上。

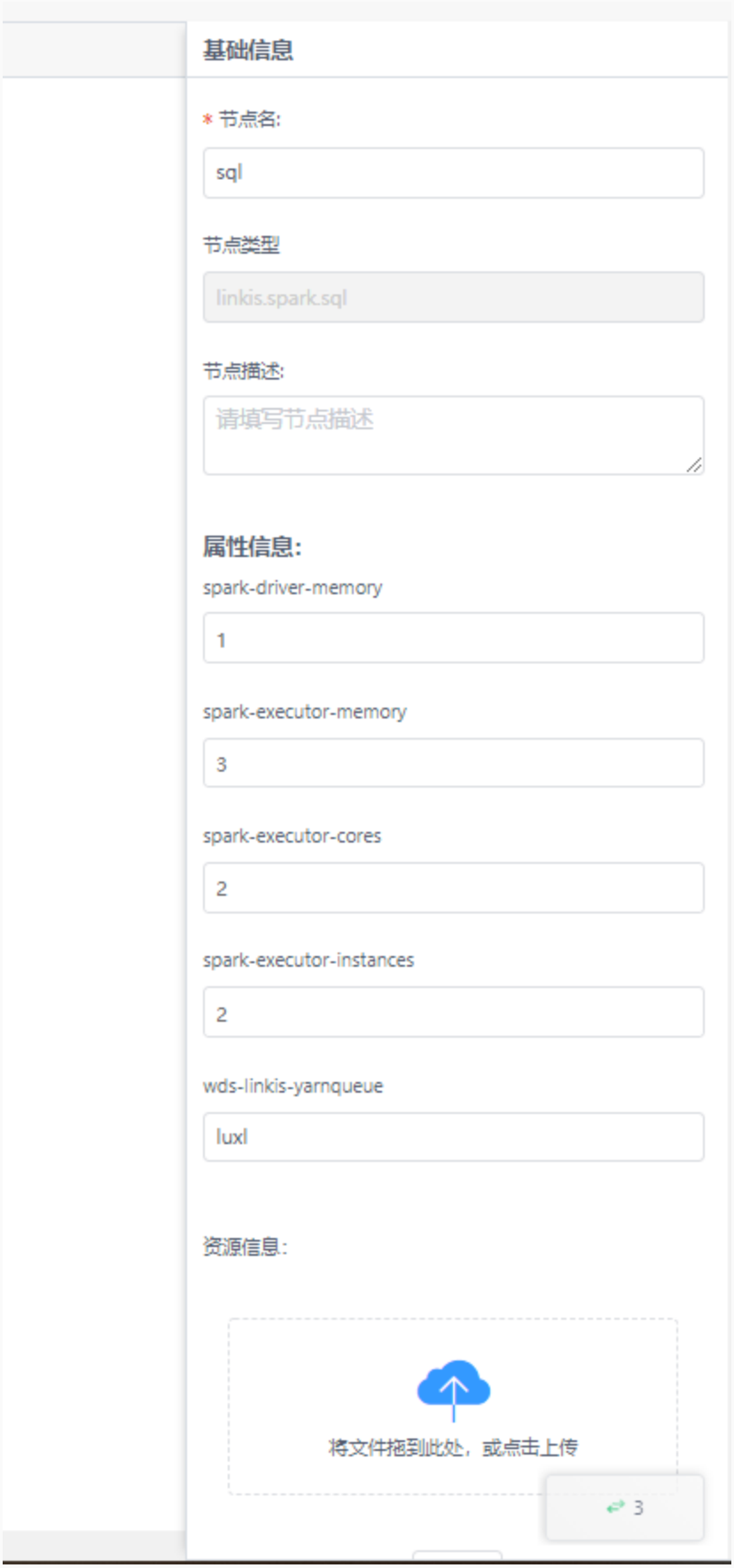
下面分别介绍两种方式：

1. 双击节点，即可打开编辑节点内容

编辑好后保存即可。

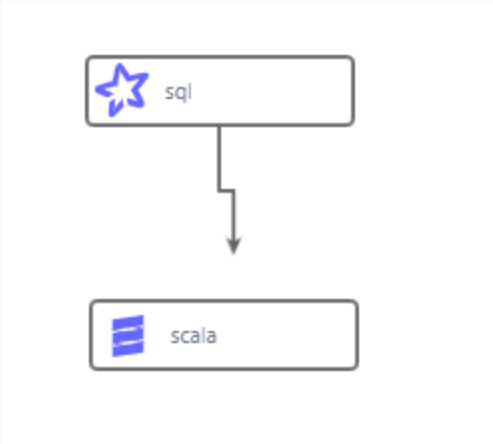
2.关联脚本的方式需要提前在scriptis中编写好脚本，在工作流节点上右键选择关联脚本，然后选择实现开发好的脚本即可。

单击工作流节点时，右侧会弹出配置信息，如提交时参数等，可以按需修改

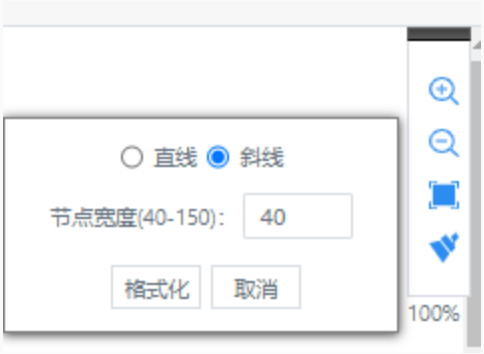

基础信息
* 节点名:
sql
节点类型
linkis.spark.sql
节点描述:
请填写节点描述
属性信息:
spark-driver-memory
1
spark-executor-memory
3
spark-executor-cores
2
spark-executor-instances
2
wds-linkis-yarnqueue
luxl
资源信息:
将文件拖到此处，或点击上传
3

工作流编排

编辑好工作流各节点后就可以对这些节点进行编排了，主要是依赖的配置，从A节点拖出箭头指向B节点，连接好后即B的执行与否依赖A节点是否成功。



将各节点连接好后可以使用右侧的编排格式化工具进行格式化，格式化时需要设定节点的图标大小，可以按需设置

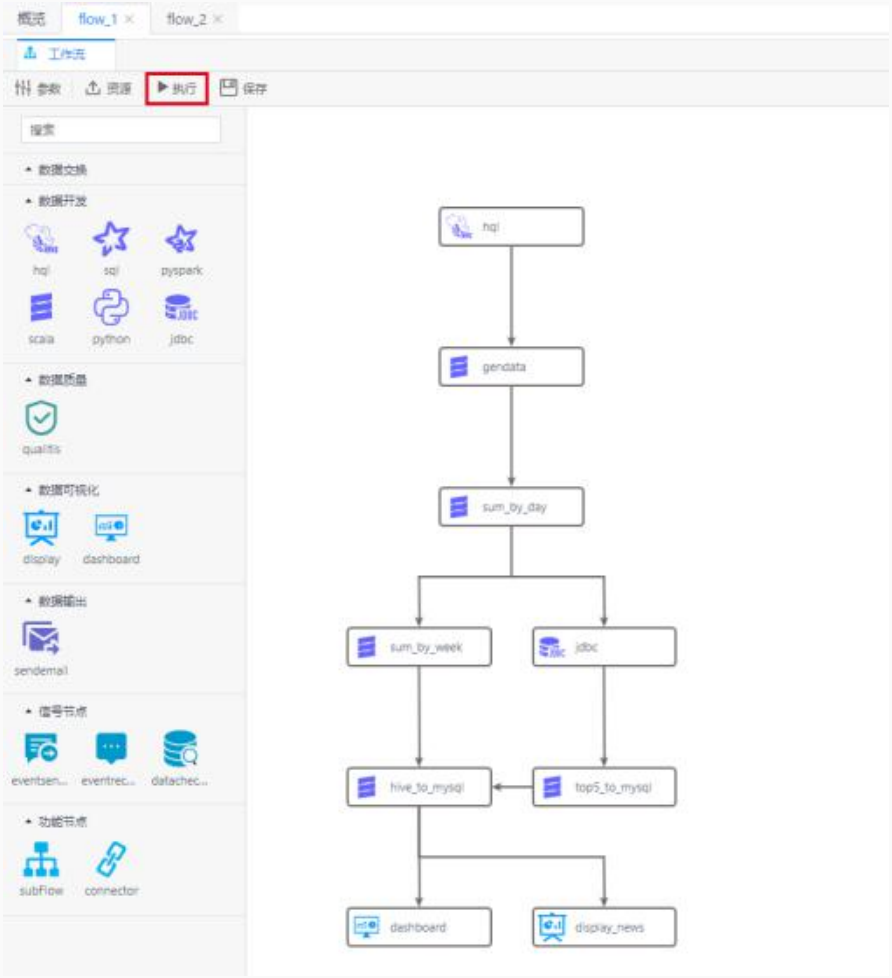


点击格式化即可自动编排节点，使整个工作流更加美观。
当然，如果对自动编排的格式不满意，也可以点击恢复按钮，恢复至之前的状态。

执行工作流

执行

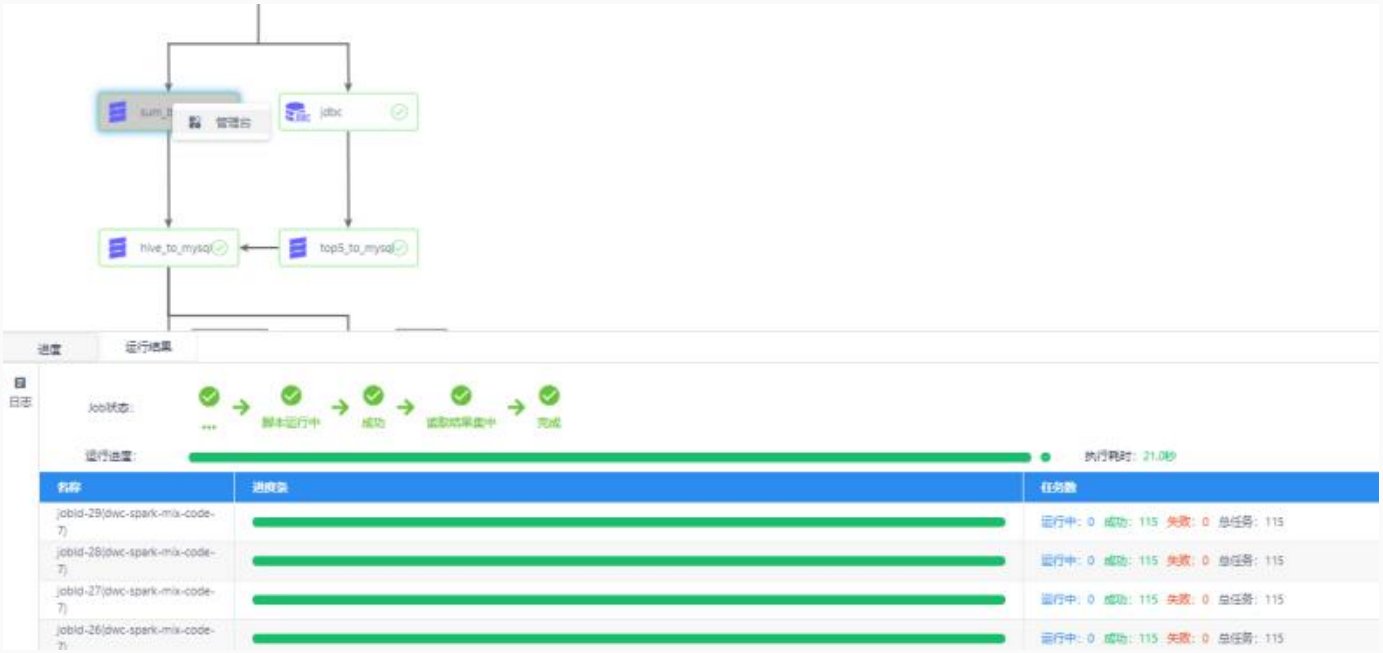
在编排好工作流并保存后，便可点击保存按钮旁边的执行按钮来进行工作流的实时运行。



除了功能节点中的subflow会跳过执行，连接节点会作为空节点运行，其他都支持实时执行。

管理台

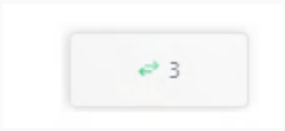
您将看到实时的工作流运行起来可以看到现在运行节点的时间，同时可以右键节点打开节点的管理台去展示该节点的进度，运行结果，运行日志等。支持任务停止等功能。



如上图示，在运行中的节点上右键，点击 **管理台** 即可查看节点运行情况，运行结果，日志等信息。

资源管理器

在运行时，可以看到右下角有一个按钮，如下图示：



这是资源管理器按钮，点击后在弹出页面可以查看任务管理器，引擎管理器，队列管理器

在这里可进行关闭引擎等操作。

发布工作流

工程支持发布调度，发布前会对工作流进行解析，以确保工作流是可以调度运行的：

发布后可以到Schedulis调度部分查看已发布的任务。

在工程的标签页上有版本信息，点击 **产看版本** 即可查看历史版本

可在弹出的对话框内进行查看，回滚等操作。

特殊工作流节点使用

eventSender节点

EventSender节点用于进行信息发送，将一段信息事件进行发送给eventReceiver。常见场景工程与工程间有依赖，工作流与工作流间有信息依赖。比如B任务依赖A任务的某些信息（A任务成功B节点才能开始执行），eventSender支持如下参数：

1. msg.type: 用来指定Job的类型，SEND用于发送消息，RECEIVE用于接收消息。
2. msg.sender: 指定消息的发送者，需使用ProjectName@WFName@JobName的格式进行定义。
3. msg.topic: 指定消息的主题，建议使用如下格式： 一级分类编码+'_'+二级分类编码+'_'+三级分类编码。
4. msg.name: 指定消息名称，由用户自定义。
5. msg.body: 指定消息的内容，没有内容发送可以为空。

**注意：msg.type默认不可变为SEND，msg.sender、msg.topic、msg.name是必填。

示例

msg.type=SEND

msg.sender=project01@flow@job01

msg.topic=bdp_tac_test

msg.name=TestDynamicReceive

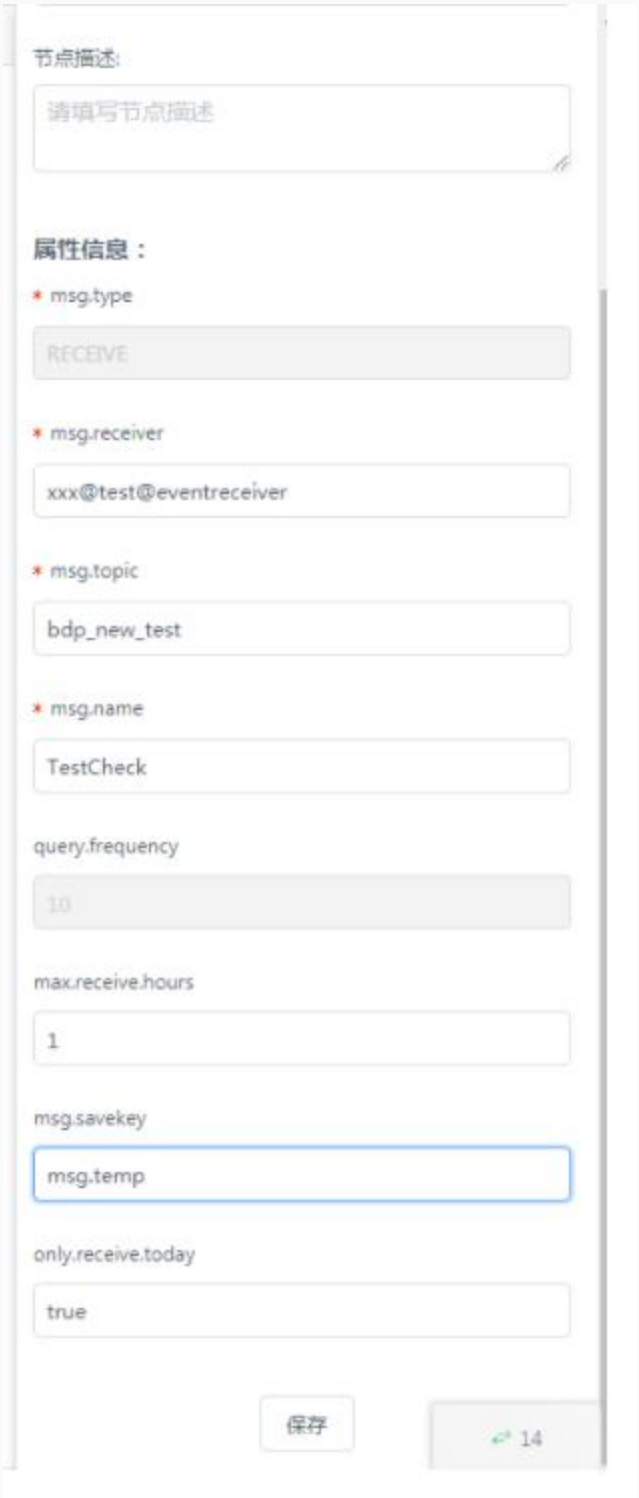
msg.body=${msg.mycontent}

eventReceiver节点

EventReceiver节点用于接收eventSender发送过来的消息，并将接收过来的消息内容存放到工作流的上下文中，后续的节点会根据前缀去找该信息进行使用比如作为自定义变量进行使用，eventReceiver支持如下参数：

1. msg.type: 用来指定Job的类型，SEND用于发送消息，RECEIVE用于接收消息。
2. msg.receiver: 指定消息的接收者，需使用projectname@jobname@rtxname的格式进行定义。
3. msg.topic: 指定消息的主题，建议使用如下格式： 一级分类编码+“”+二级分类编码+“”+三级分类编码。
4. msg.name: 指定消息名称，由用户自定义。
5. query.frequency: 由于接收消息使用的是主动轮询的方式，wait.time期间的查询次数，。
6. max.receive.hours: 最长的接收时长，以小时为单位，超过时长未接收到消息返回失败，。
7. msg.savekey: 用于保存消息内容key值，单个flow内多个接收job，需指定不同的msg.savekey保存消息内容，默认值为msg.body，后续Job可以使用该key值获取消息内容。
8. only.receive.today: 如果为true 有且只能接收job启动当天发来的消息
9. **注意：msg.type默认不可变为RECEIVE，msg.receiver、msg.topic、msg.name是必填。**

配置reveive接收对应的topic信息，并通过msg.savekey进行保存

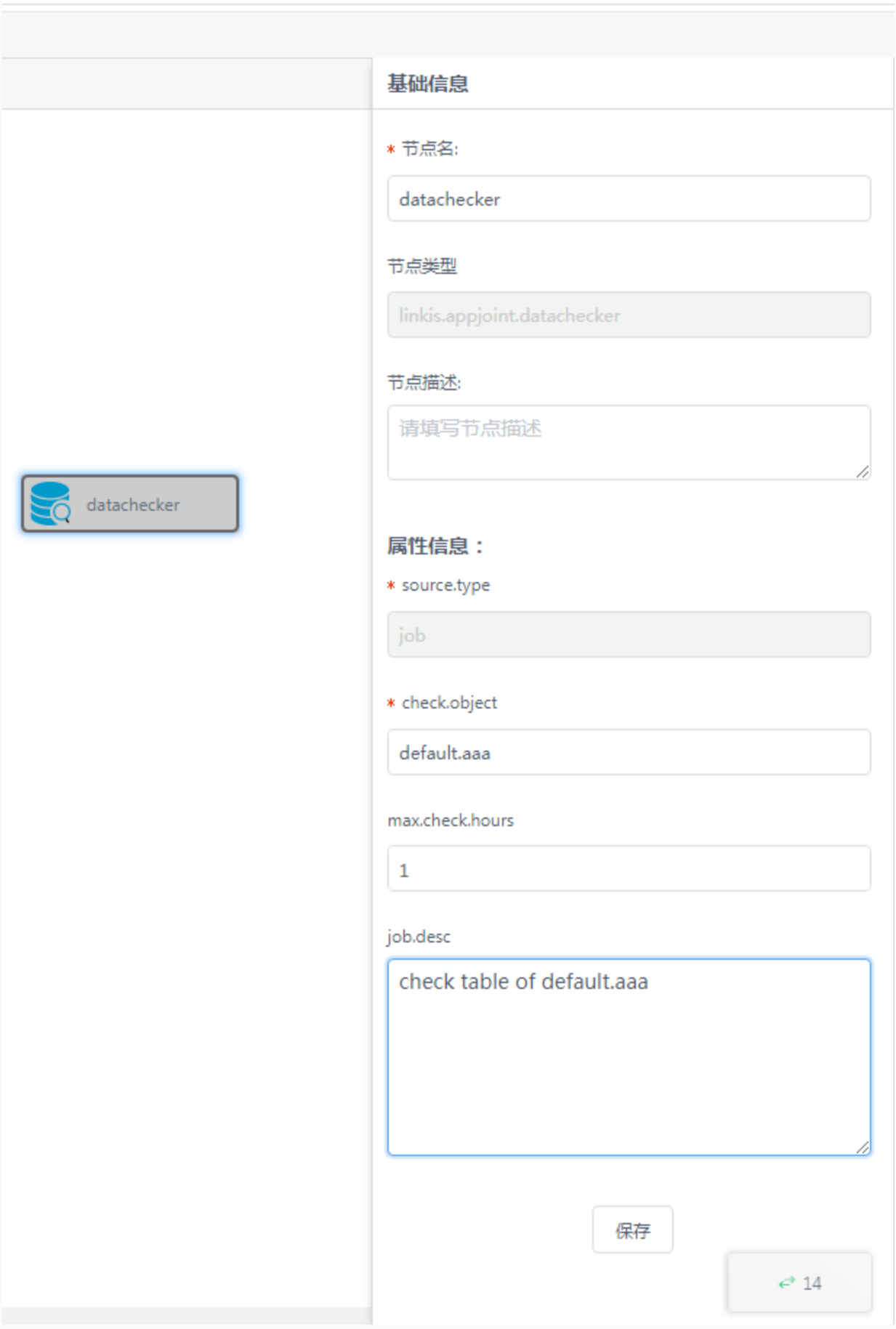


信号节点支持跨工程和工作流，这里只是示例使用。

dataCheck节点：

DataCheck节点用于检测数据是否ready，可以判断hive库中某个表或者分区是否存在，如果存在则进行下游的执行，在有数据依赖的任务中扮演十分重要的作用替换掉以前口头约定好的时间开始运行。
dataCheck支持如下参数：

1. source.type: 依赖的数据来源，job表示是由其他job产生
2. check.object: 依赖数据的名称例如：data.object.1=dbname.tablename{partitionlist}
3. max.check.hours: 描述任务的等待时间，单位是小时
4. job.desc: 追加多源信息配置。

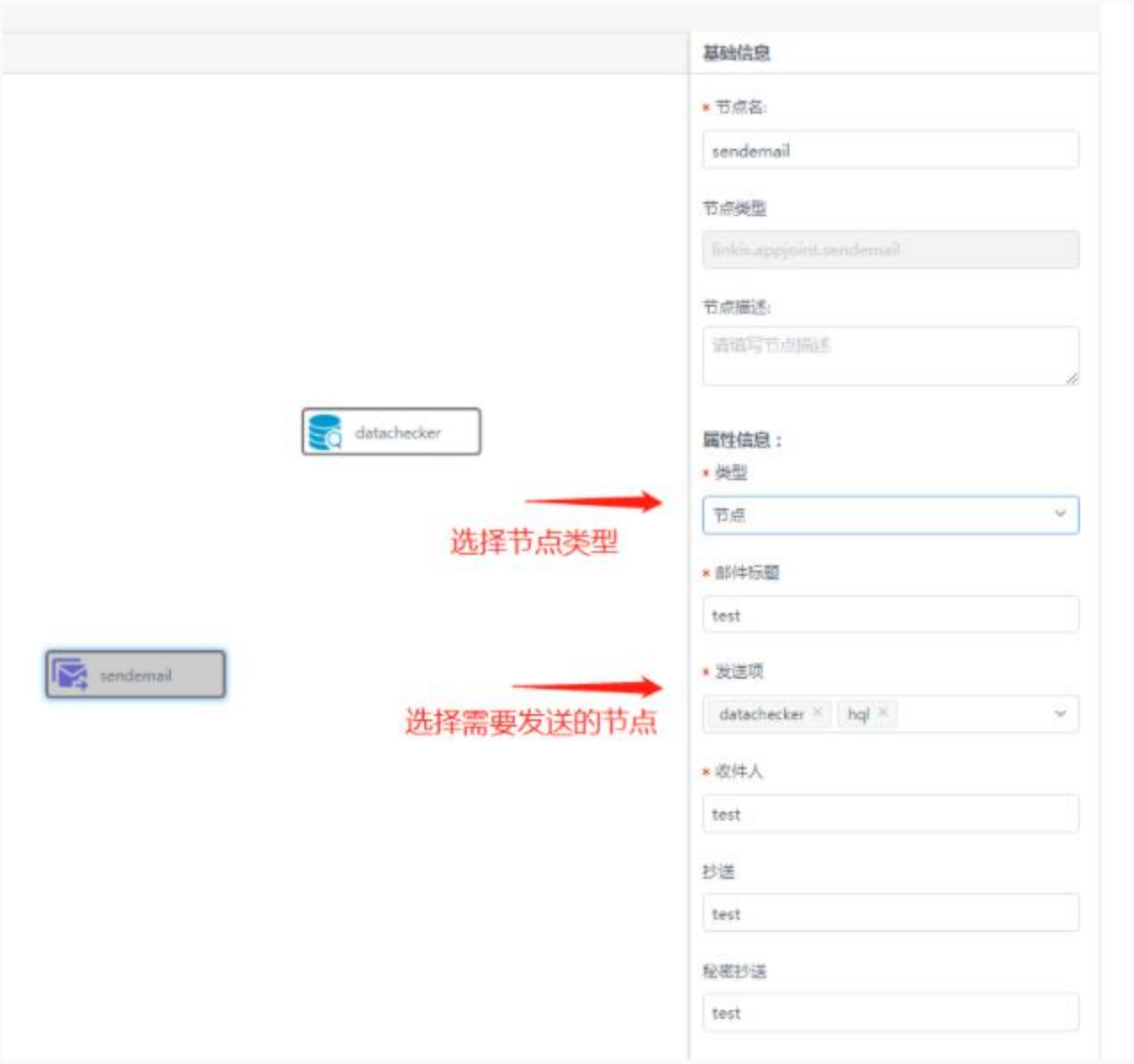


SendEmail节点:

SendEmail节点一般作为工作流的最后一个节点，用于将工作流前面的结果信息进行发送，支持发送表格、文本、DashBoard、Display、图片等，用户在使用的时候直接选择发送的对应节点就行：

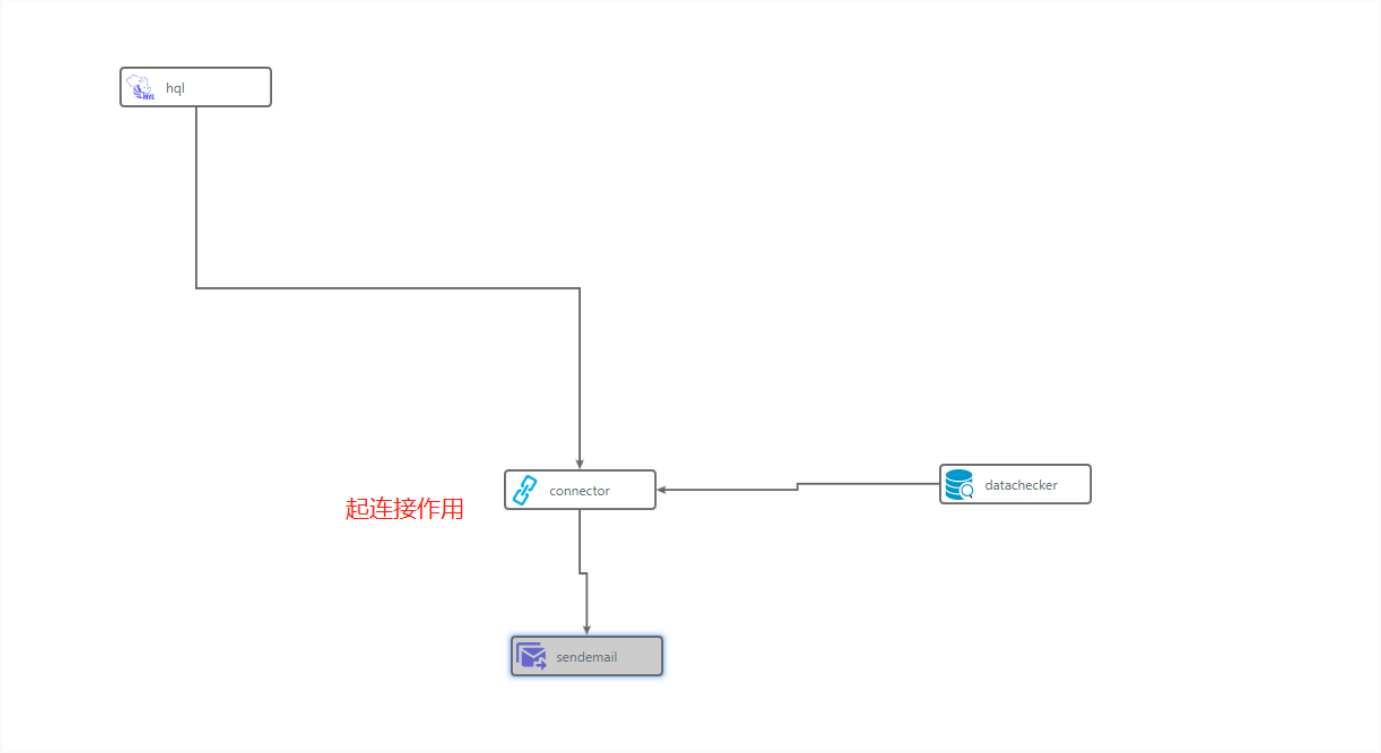
sendEmail支持如下参数：

1. 类型：支持节点、文字、文件、链接
2. 邮件标题：指定邮件表提
3. 发送项：发送的具体内容，例如：类型是节点则这里选择节点
4. 关联审批单：该邮件是否走过审批，如果未则不会进行发送
5. 其他邮件基本属性：收件人、抄送、秘密抄送



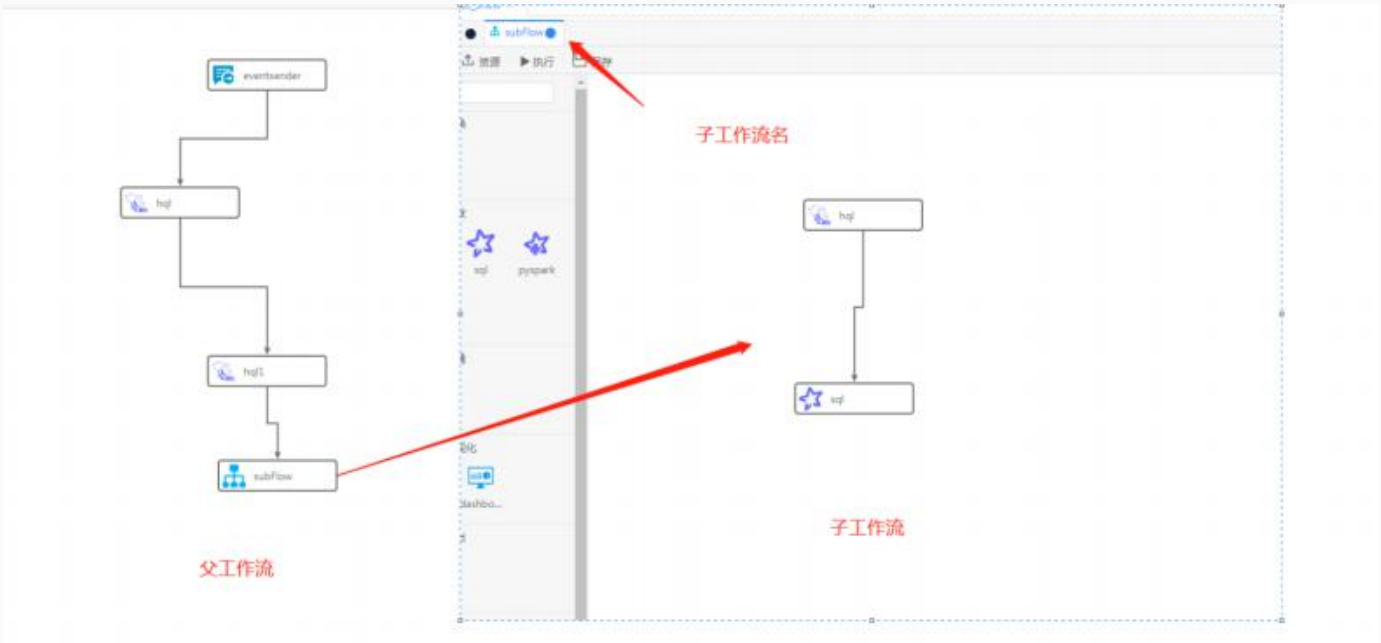
Connector节点:

Connector节点的作用是为了作为节点与节点的连接，让工作流更加好看：



Subflow节点:

Subflow节点是您可以在一条工作流中嵌入一条子工作流，子工作流支持发布调度，但是在实时执行时父工作流的subflow节点会跳过执行，需要跳到子工作流编辑页面进行执行：



功能说明

Exchangis是一个轻量级的、高扩展性的数据交换平台，支持对结构化及无结构化的异构数据源之间的数据传输，在应用层上具有数据权限管控、节点服务高可用和多租户资源隔离等业务特性，而在数据层上又具有传输架构多样化、模块插件化和组件低耦合等架构特点。

支持的数据源

目前支持以下几种数据源：

- HIVE
- HDFS
- SFTP
- ES
- MYSQL
- ORACLE

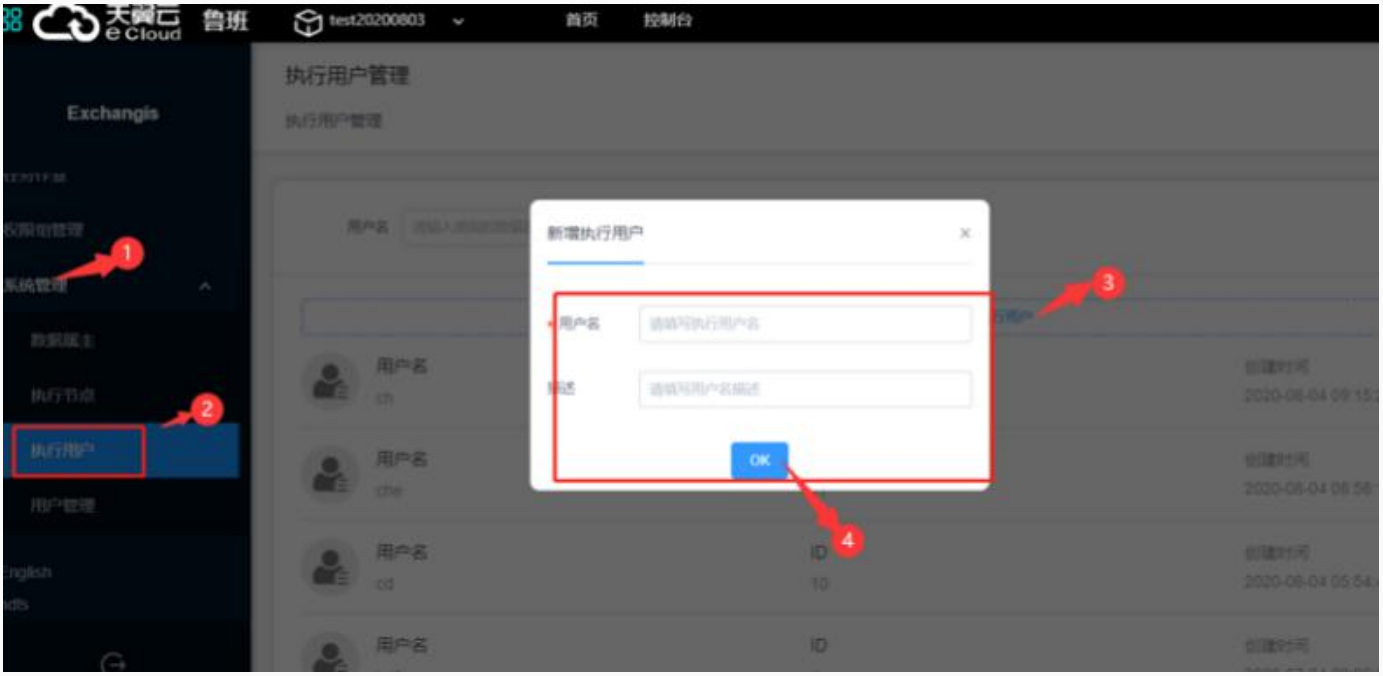
数据接入使用流程

准备工作

首先进入某个已经创建好的工作空间，点击 **数据接入** > **进入exchangis** ，便可进入数据接入模块。

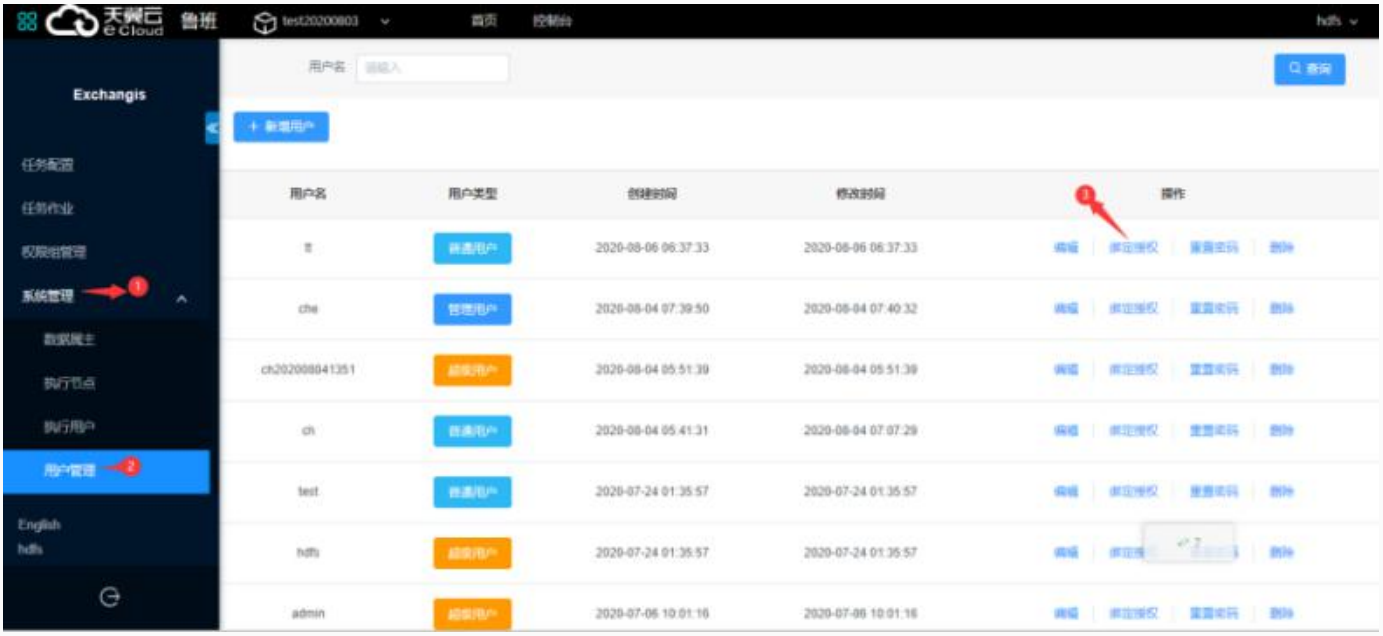
添加执行用户

点击左侧菜单栏中的**系统管理**下拉框，选择**执行用户**，进入页面。点击页面中的**新增执行用户**，在弹框中填写**用户名-描述**（非必选），点击OK。

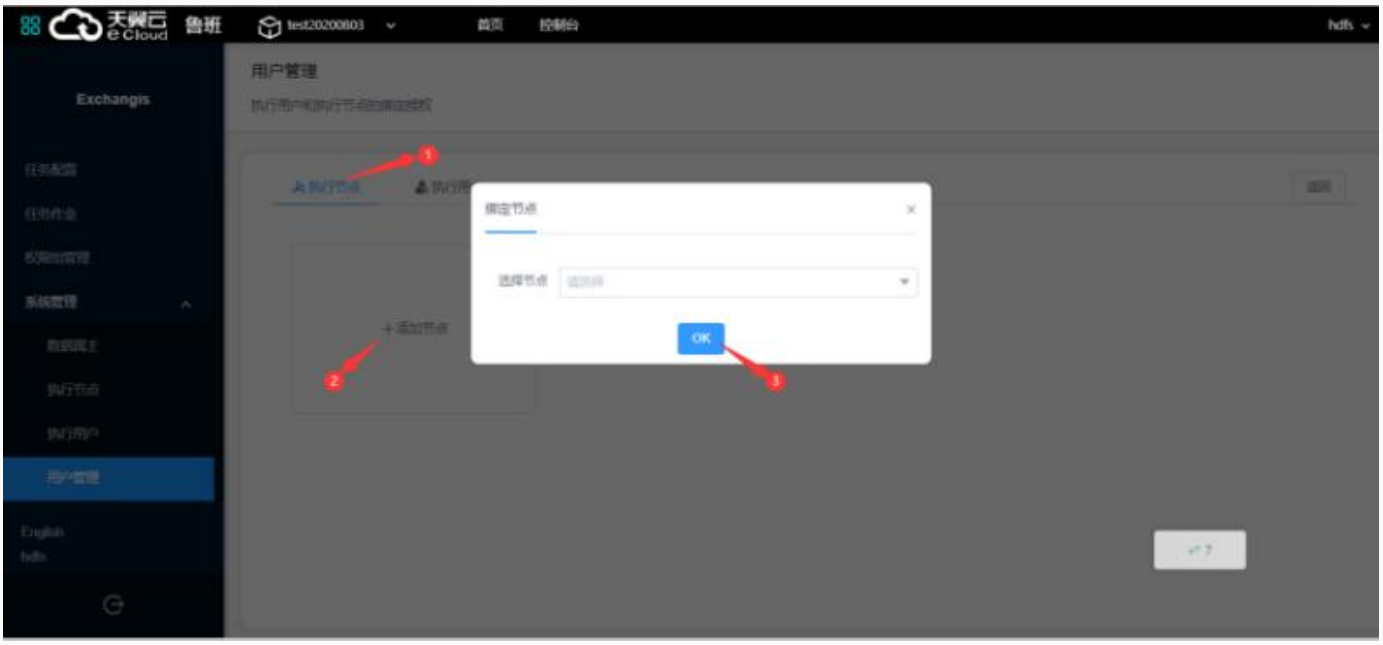


用户绑定授权

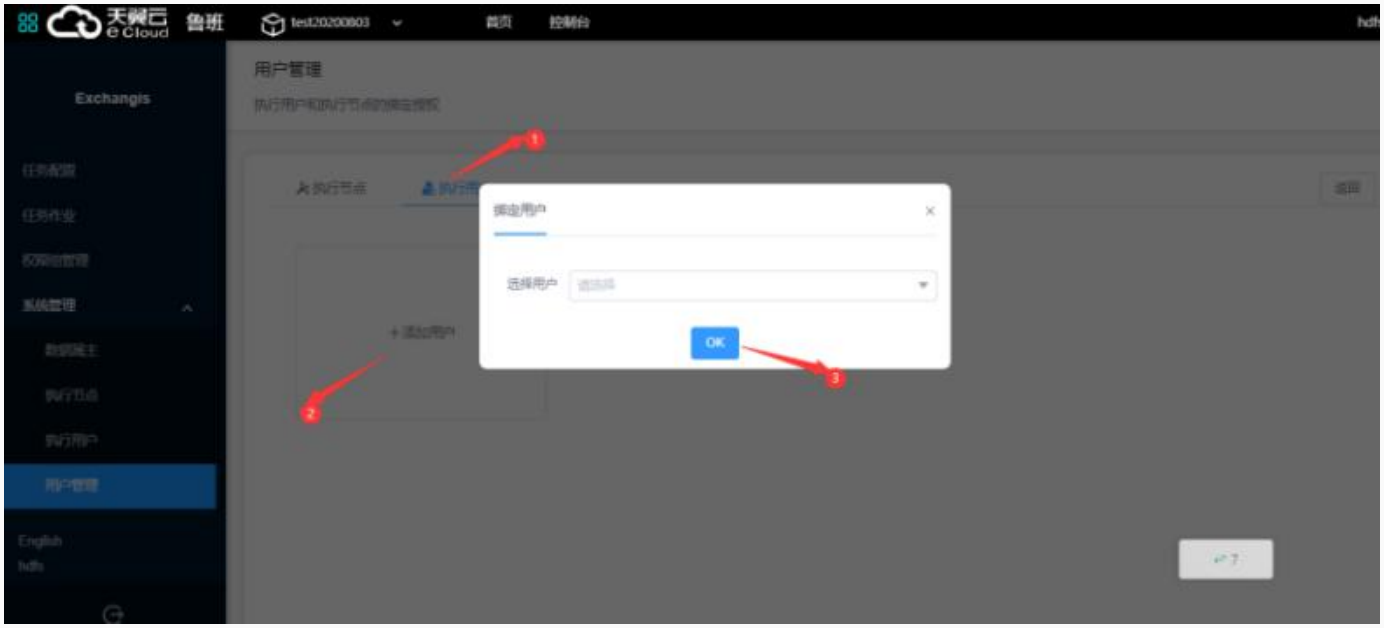
点击左侧菜单栏中的**系统管理**下拉框，选择**用户管理**进入页面，点击用户后的**绑定授权**。



在跳转页面中点击**添加节点**，在弹框中选择需要的节点，点击OK。



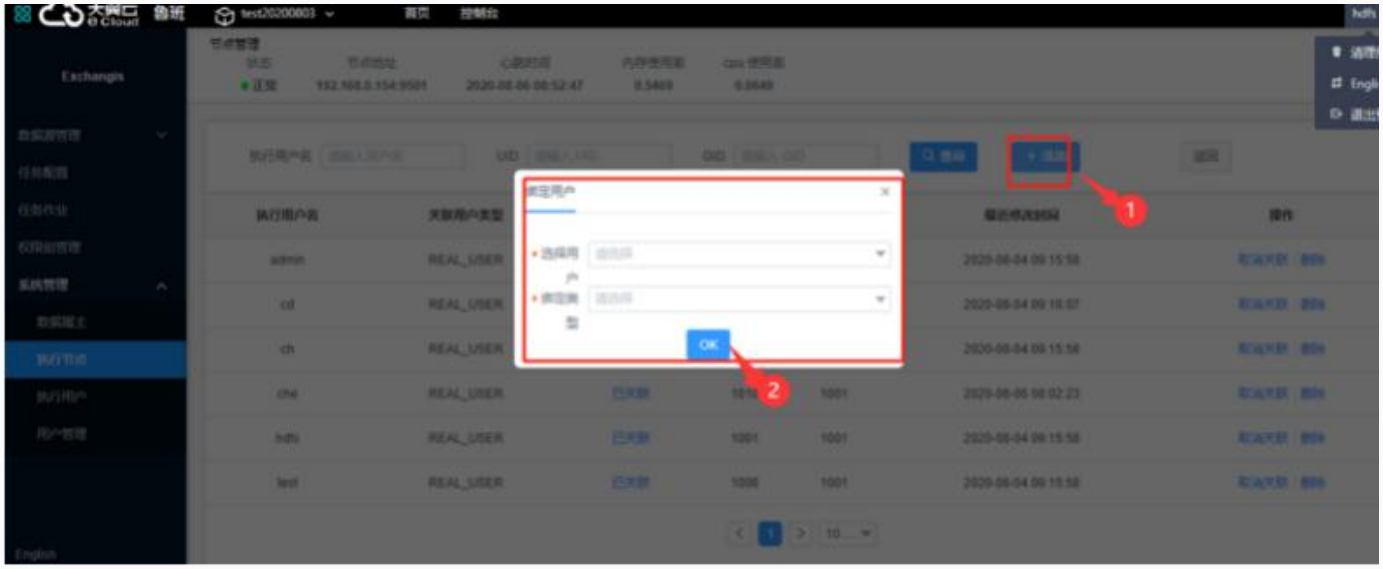
点击页面中的**执行用户**，在页面中点击**添加用户**，在弹框中选择需要的用户，点击OK。



节点关联执行用户

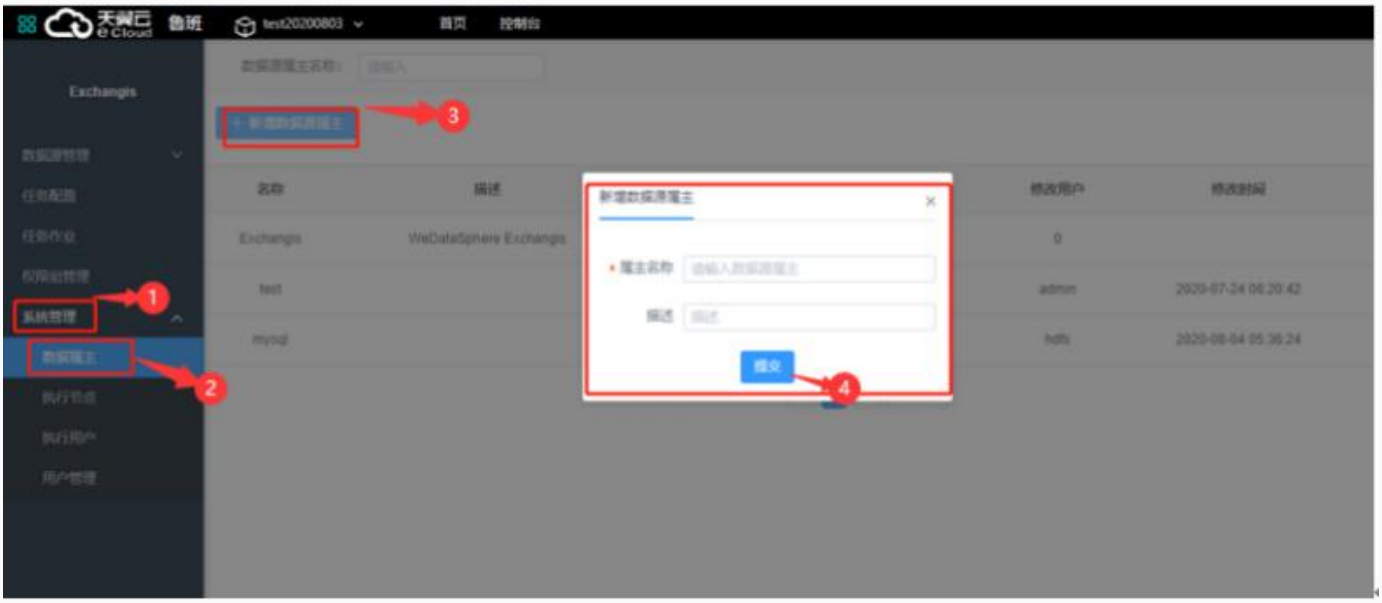
点击左侧菜单栏中的**系统管理**下拉框，选择**执行节点**进入页面，点击执行节点后的**管理节点**，进入页面。

点击页面中的**添加**按钮，在弹框选择用户以及绑定类型，点击OK。



新增数据源属主

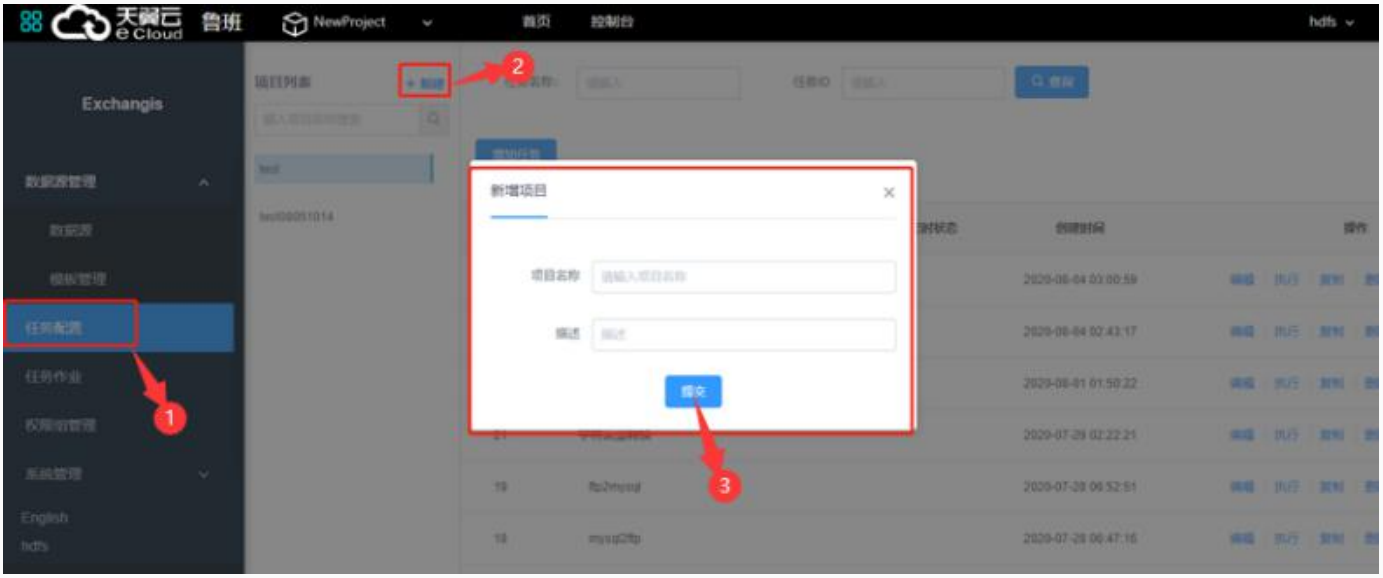
点击左侧菜单栏中的**系统管理**下拉框，选择**新增数据属主**，进入页面。



在弹框中填写**属主名称–描述**（非必选），点击提交。

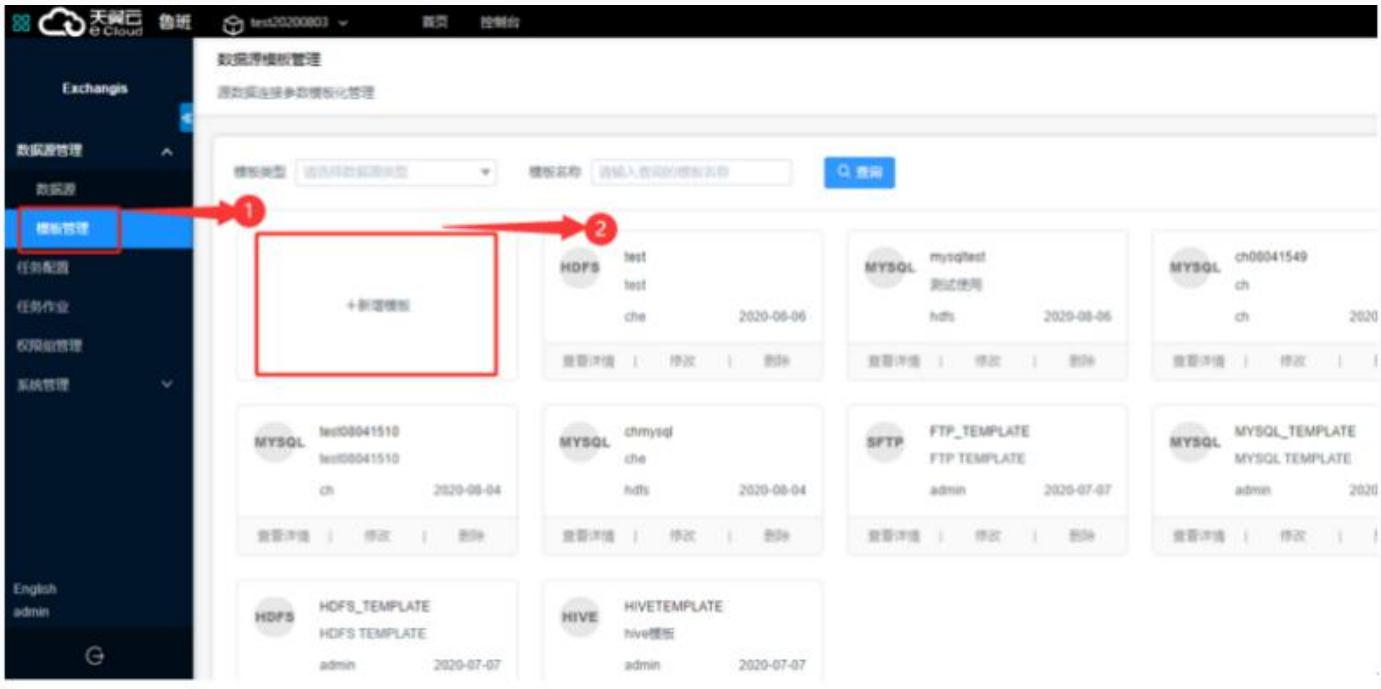
新建项目

点击左侧菜单栏下的**任务配置**，进入页面。点击页面中的**新建**，在弹框中填写**名称–描述**（非必选），点击提交。



新增数据模板

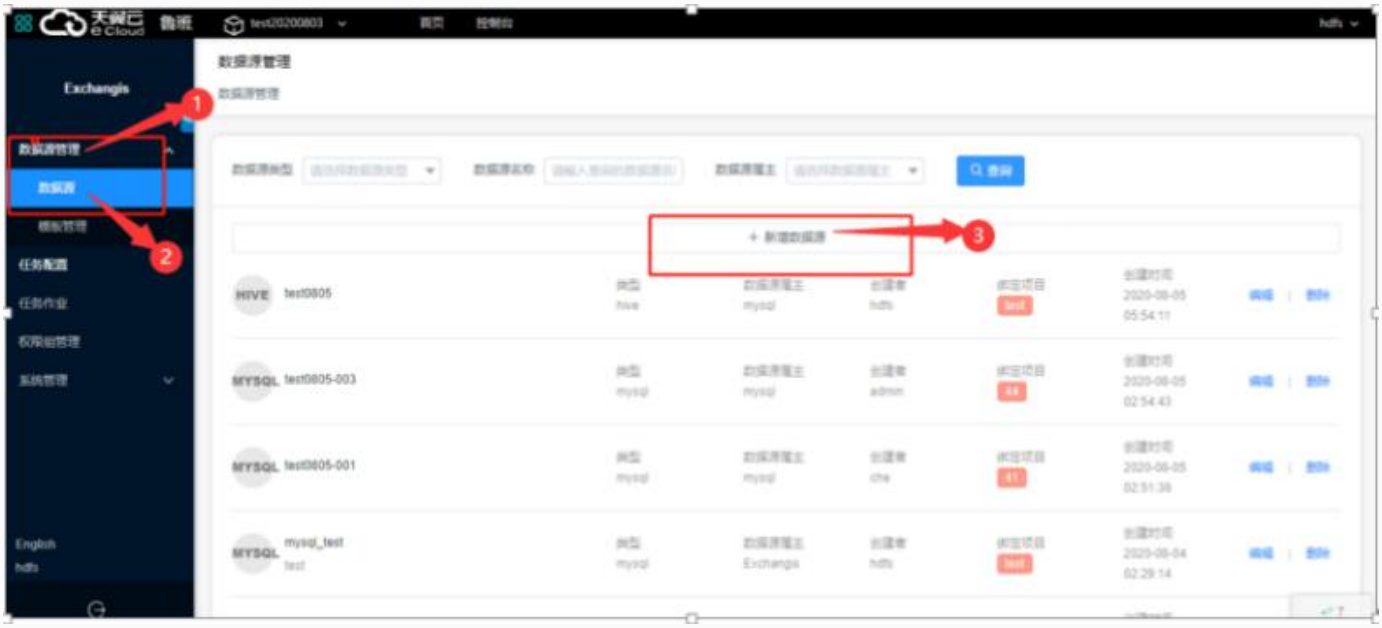
点击左侧菜单栏**数据源管理**模块下的**模板管理**进入页面，在页面中点击**新增模板**。



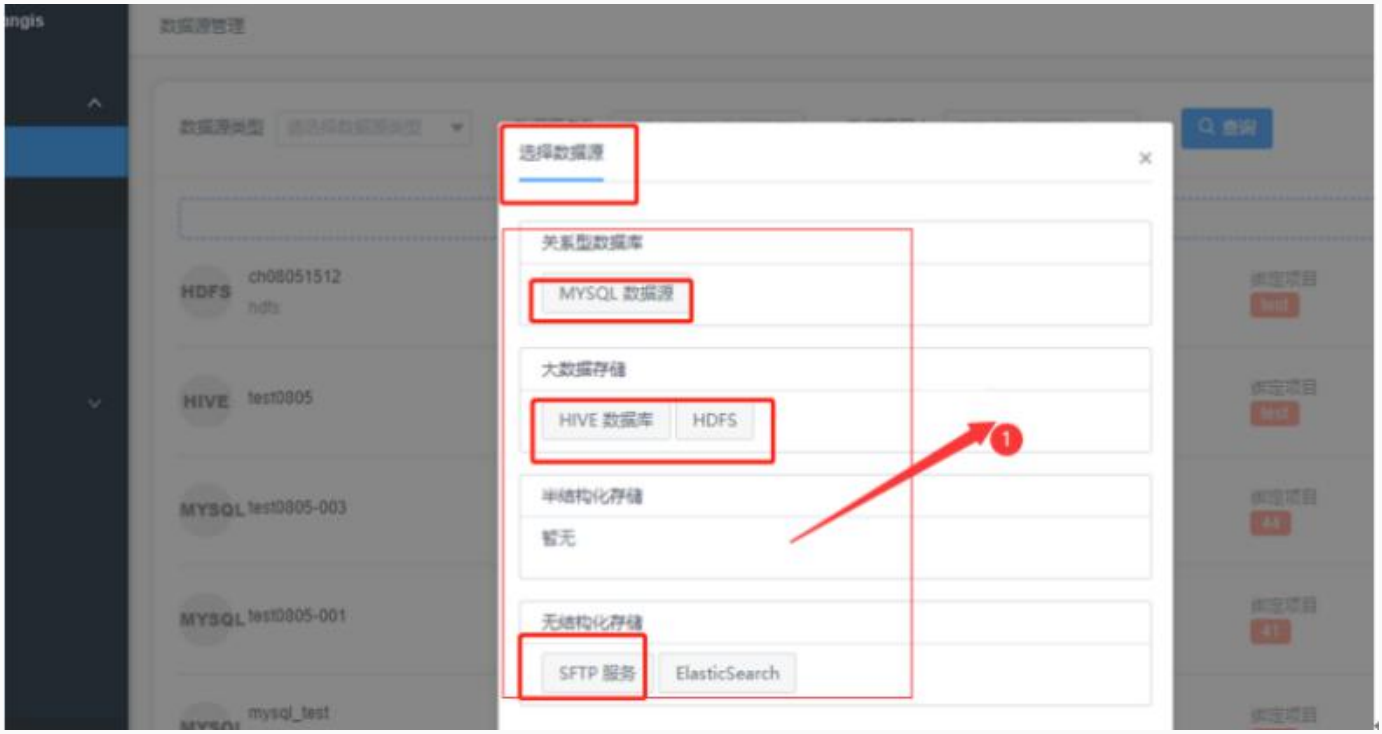
在弹框中选择数据源类型为**HIVE–填写名称–模板描述–Metastore 地址–HDFS 地址–Hadoop 配置项**（非必选）–选择**认证方式**，点击OK。

新增数据源

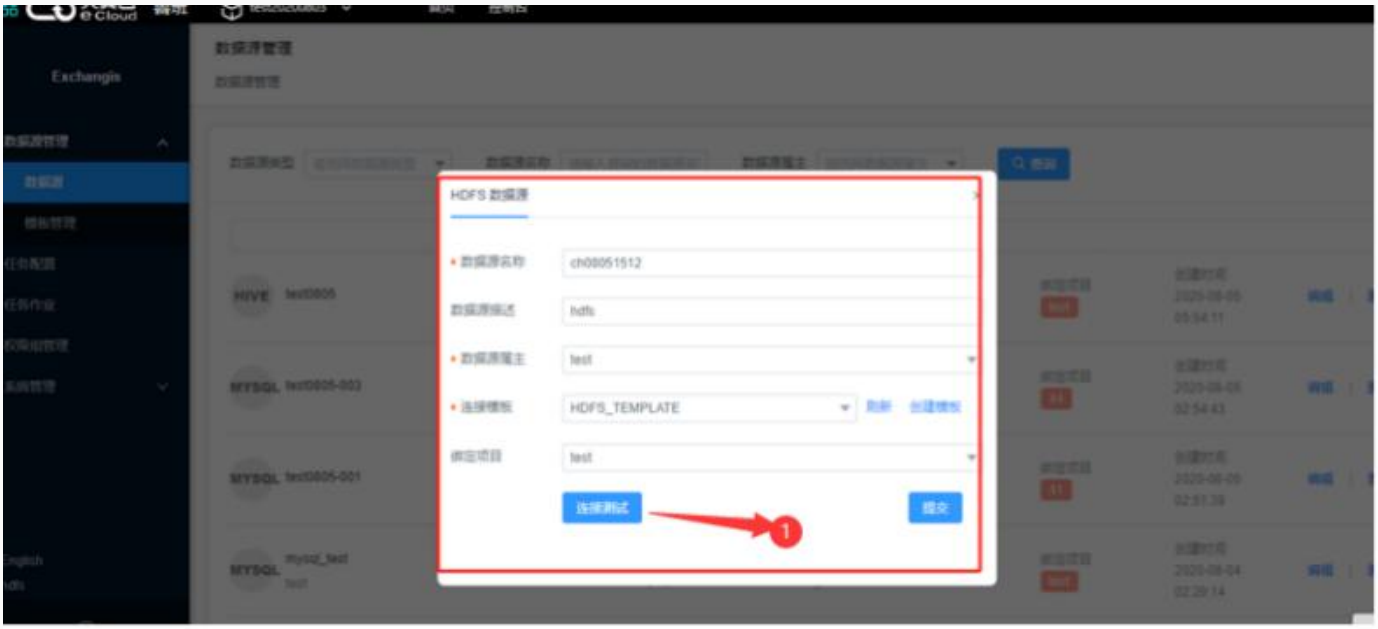
点击左侧菜单栏**数据源管理**下拉框，选择**数据源**进入页面，点击页面中**新增数据源**。



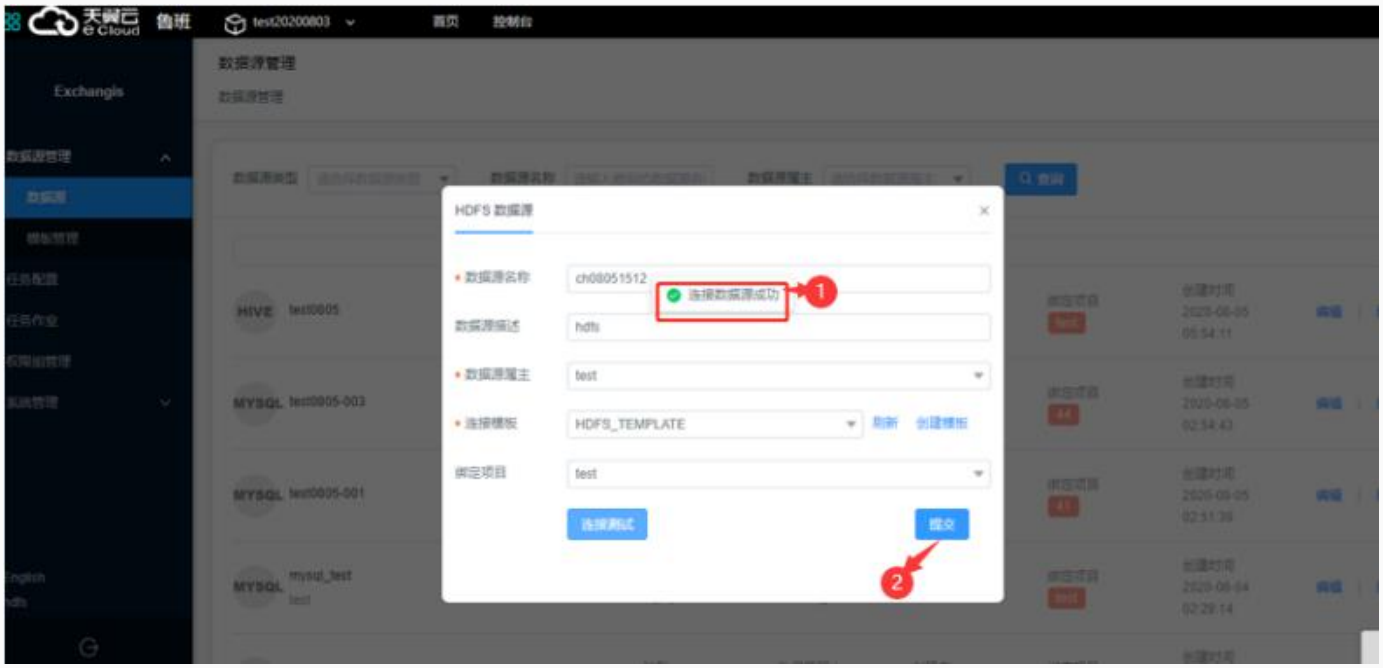
在弹框中可以选择关系数据库（MYSQL数据源），大型数据存储（HIVE数据库，HDFS）,无结构化数据存储（SFTP服务，ElasticSearch）选择需要的数据源hdfs，跳出弹框。



在弹框填写**数据源名称**–**数据源描述**（非必选）–选择已创建**数据源属主**–选择已创建**数据源模板**–绑定已创建好的项目，点击连接测试。



待页面提示联系数据源成功后，点击提交。

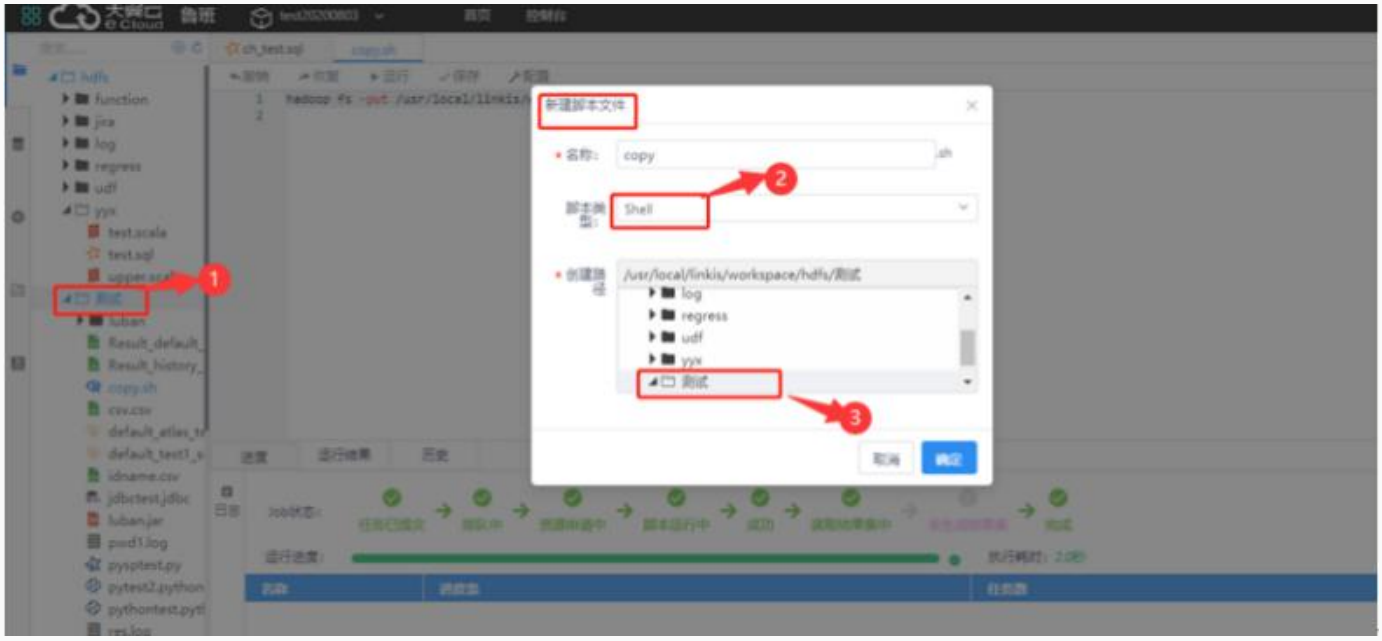


增加任务

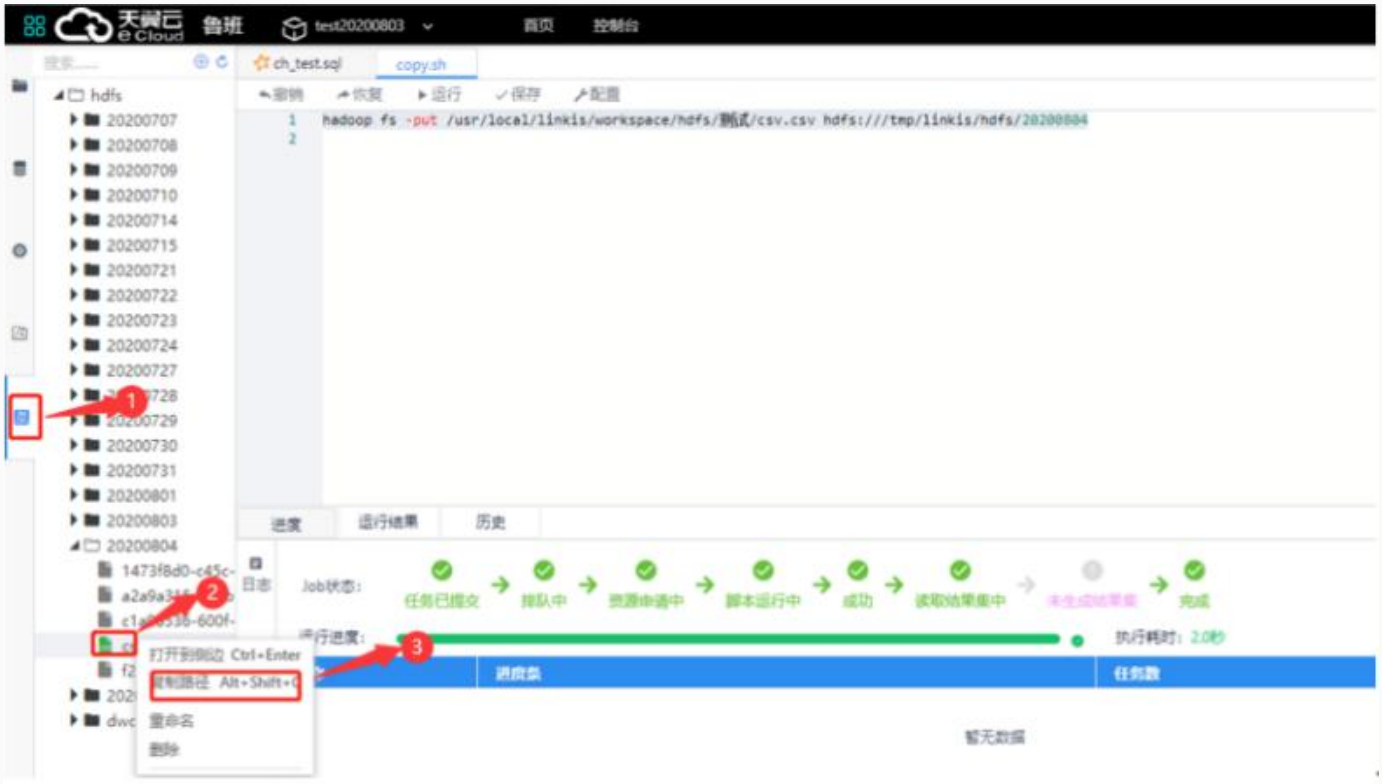
准备数据或数据源(以csv本地数据为例)。

把鼠标移动到页面左上角功能模块浏览按钮，点击数据分析中的scriptis，进入页面。

右键单击本地文件夹中，选择上传文件。



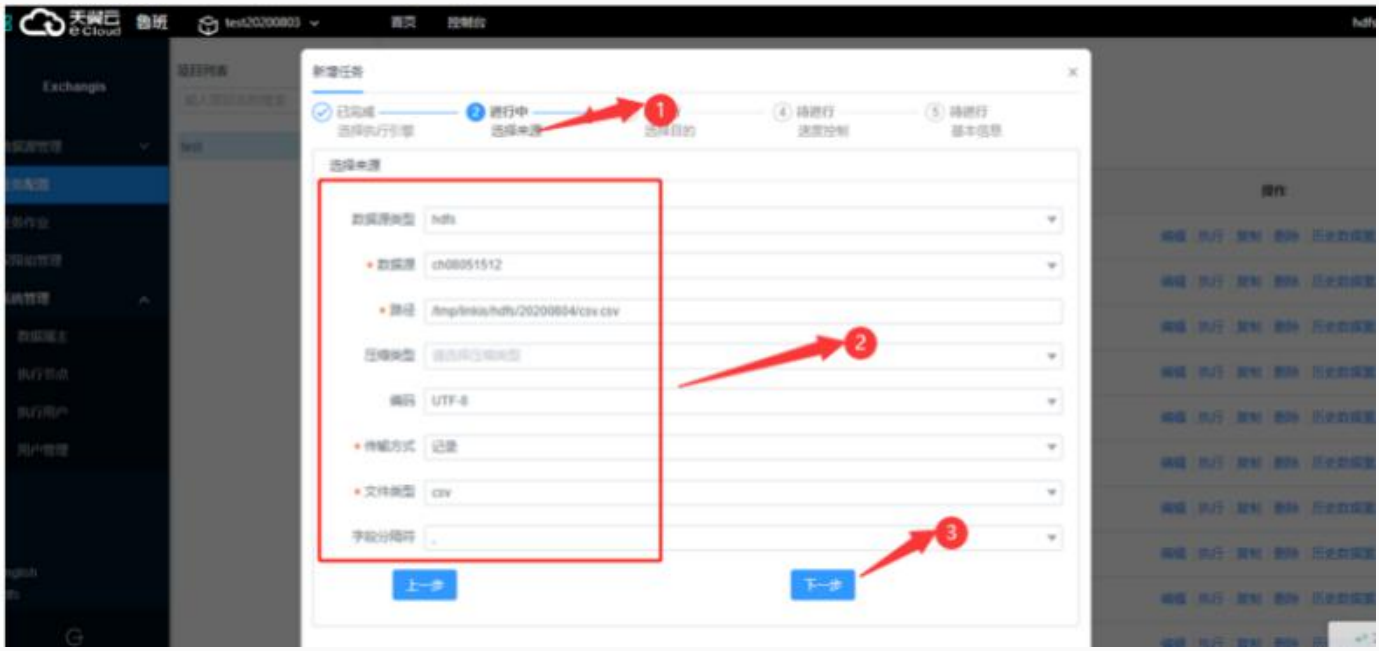
在hdfs集群中复制csv文件路径



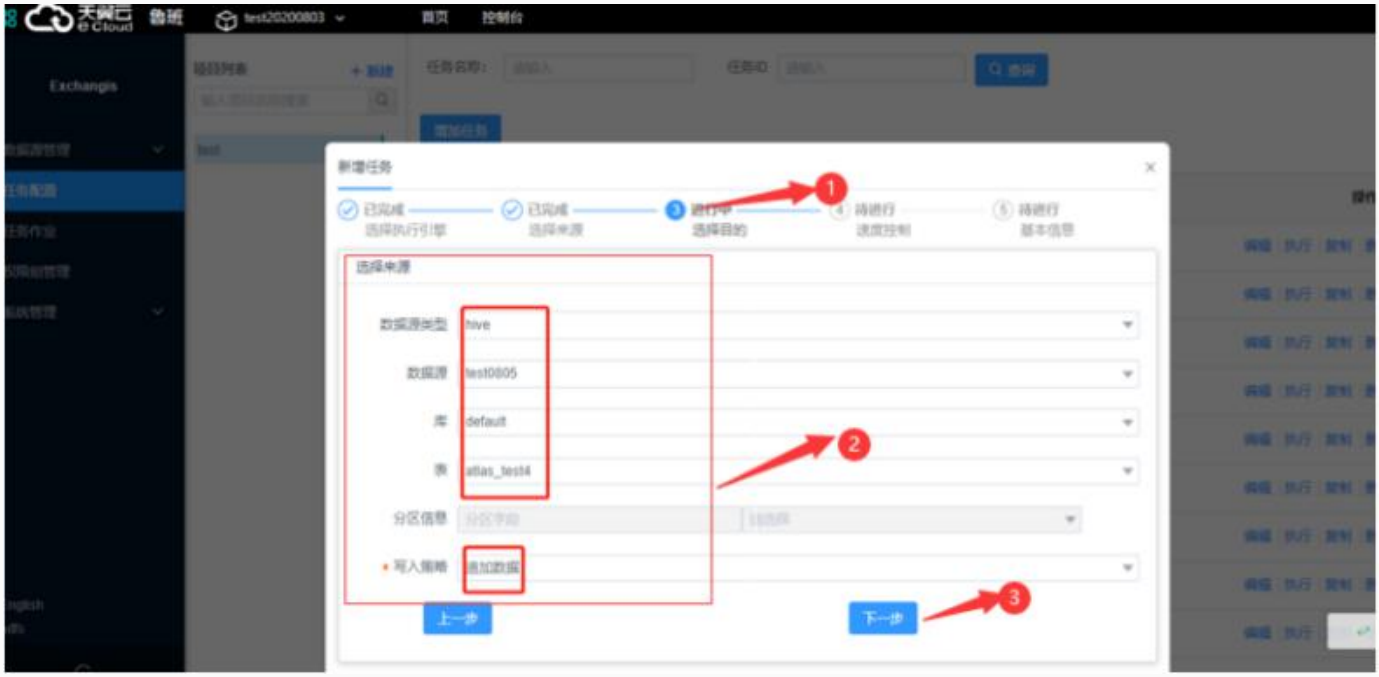
返回Exchangis页面，点击左侧菜单栏中的**任务配置**，点击已创建数据源绑定的项目，进入页面。
点击**增加任务**，在弹框选择需要执行的引擎（DATAX，SQOOP）点击下一步。

选择来源弹框中填写数据源类型为hdfs，**数据源**为已创建的数据源–**路径**为hdfs集群中复制的路径–选择**压缩类型**（非必选）–**编码**（非必选）–**传输方式**为记录–**文件类型**为csv–字段分隔符（非必选），点击下一

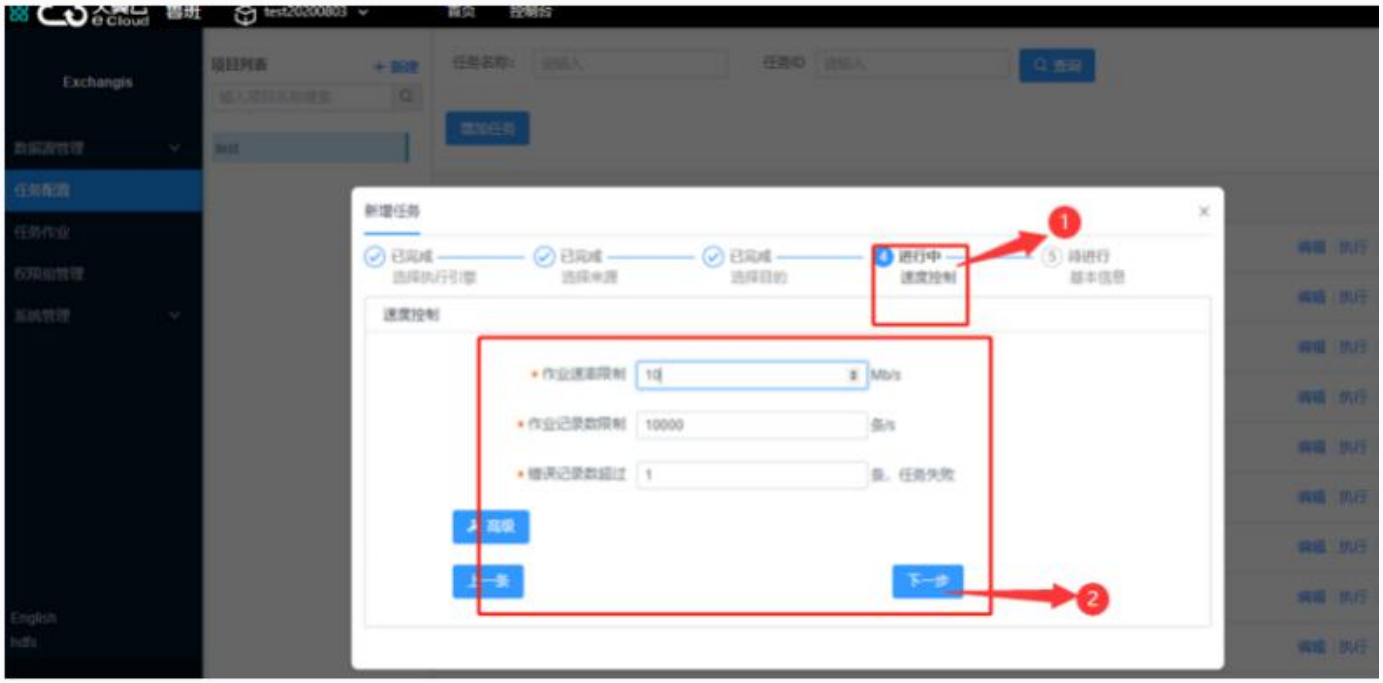
步。



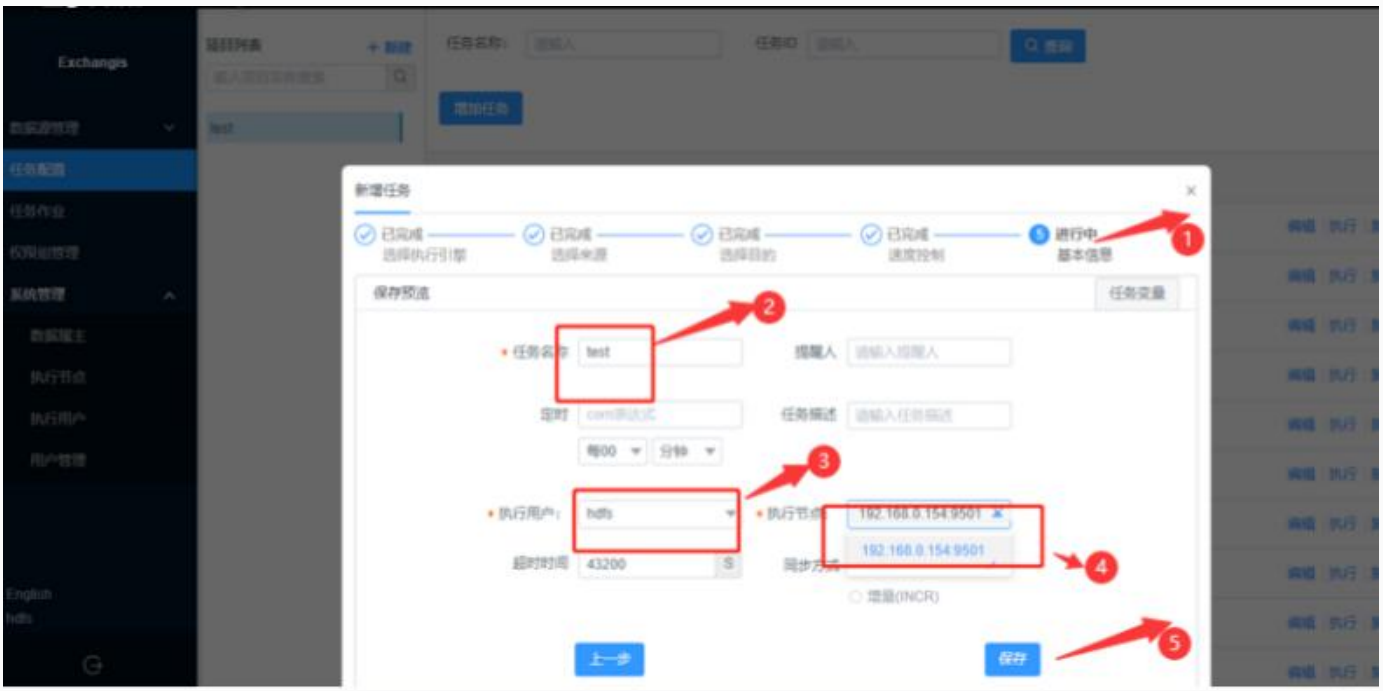
选择目的填写**数据源类型–数据源–库–表–写入策略**，点击下一步。



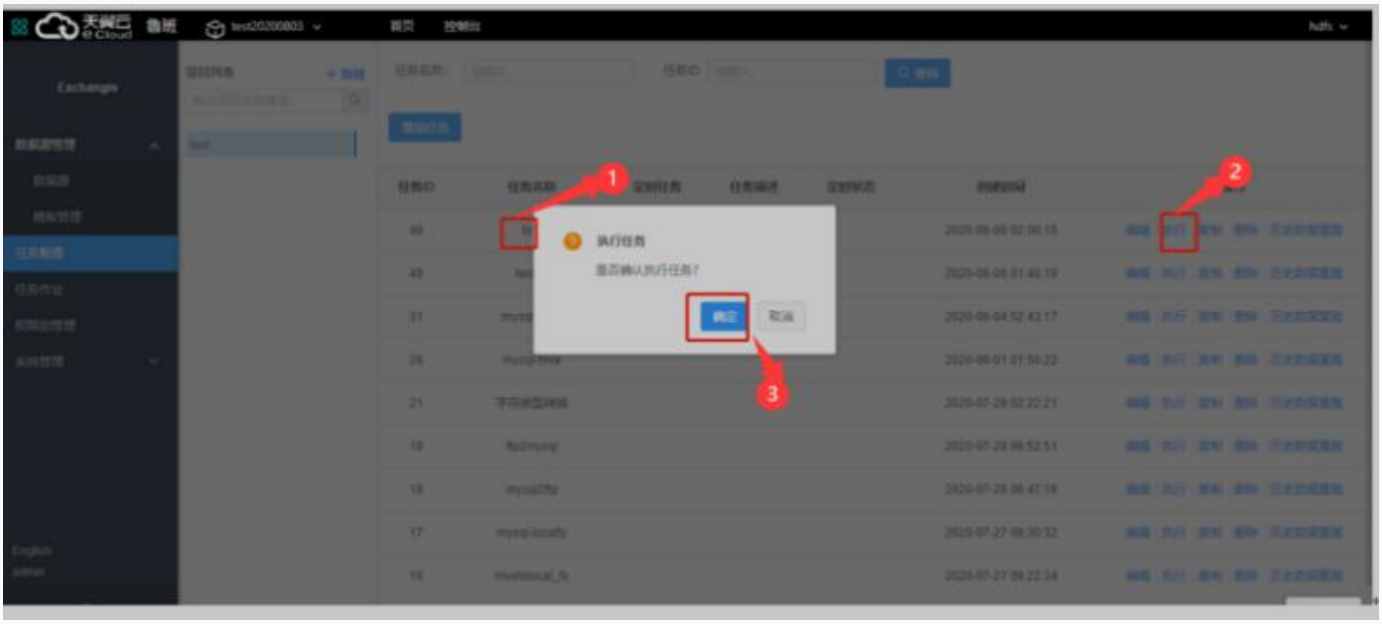
速度控制选择–**作业速率限制–作业记录数限制–错误记录数超过**限制或添加高级设置，点击下一步。



基本信息填写任务名称–提醒人（非必选）–定时（非必选）–任务描述（非必选）–执行用户–执行节点–超时时间（非必选）–同步方式（非必选），点击保存。



待任务新增成功后，点击任务后的执行按钮，在弹框中点击确认，页面提示任务开始执行。



任务状态为成功后，返回scriptis页面，在数据库中找到库名default，点击进入找到student表，右键点击选择查询表，在表中可以看到上传的数据。


default_student_...
copy.sh
运行
保存
进度
运行结果
历史
id
name
sex
address
age
30
小程
男
安徽
29
10
C◆◆
◆◆
◆◆◆◆
29
2
晓明
男
安徽
20

功能说明

Scriptis是一个高效的在线开发IDE，提供语法高亮、代码自动补全、智能纠错和语法错误提示等功能，并支持在线开发、在线调试、多人协同编辑、一键发布UDF资源和函数至鲁班。

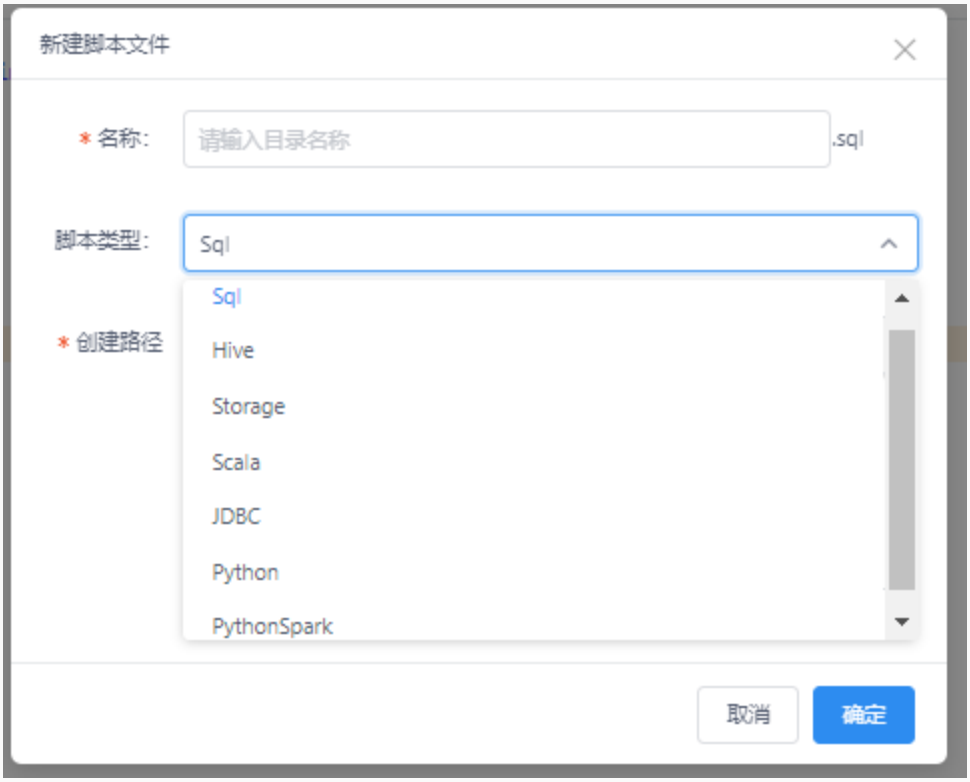
页面布局

页面布局如图。左侧分别为“文件系统”、“数据仓库”、“UDF函数”、“方法函数”和“HDFS”标签，可逐一打开查看，也可通过编辑脚本的方式进行对相关资源的操作。

编写和运行脚本

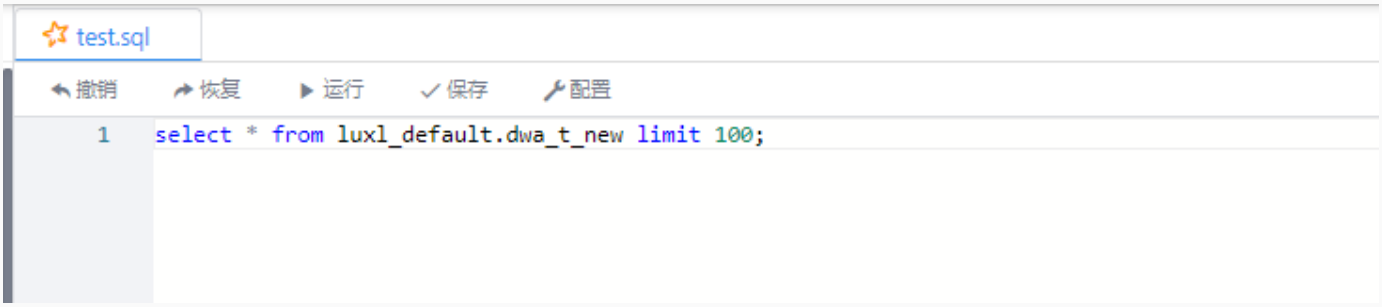
右键点击根目录，选择 **新建目录**，创建新目录，亦可在已有的目录下创建多级子目录

选择某目录并右键，选择 **新建脚本** 可创建shell、python、sql等多种语言脚本



此处以创建sparksql脚本为例。

创建脚本文件对话框中的脚本类型选择sql，创建完成后，自动打开该脚本编辑页面，并输入脚本内容。



点击运行按钮即可运行脚本，运行进度，日志，结果集等将会在下方弹出

用户可点击 **配置** 按钮自定义参数，并可在脚本代码中直接将对应的值替换成${参数名}形式，亦可成功执行。

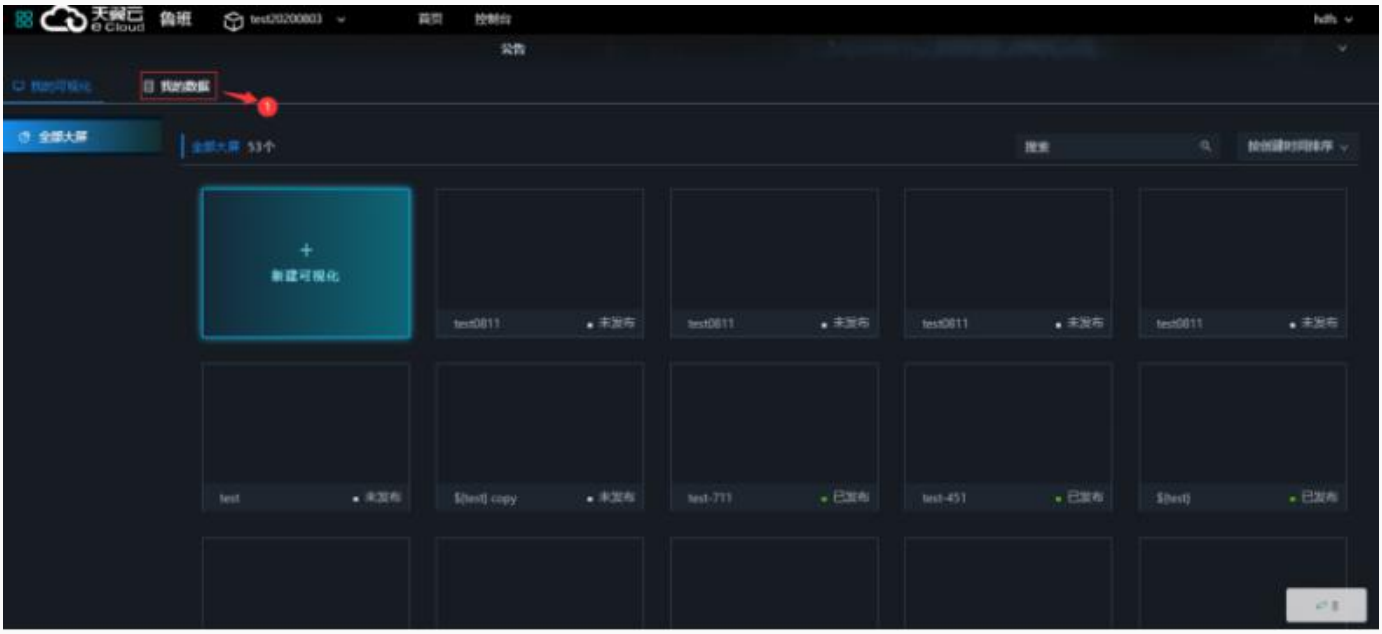
功能说明

天翼视屏用于数据可视化展示，提供丰富的组件和模板，满足您会议展览、业务监控、风险预警、地理信息分析等多种业务的展示需求。

创建数据源

进入需要的工作空间，点击 **数据分析** > **天翼视屏**

在可视化首页，点击页面中 **我的数据**。



在页面中点击添加数据按钮，在弹框中根据需求可以填写数据类型，再根据不同类型填写不同信息。

说明:

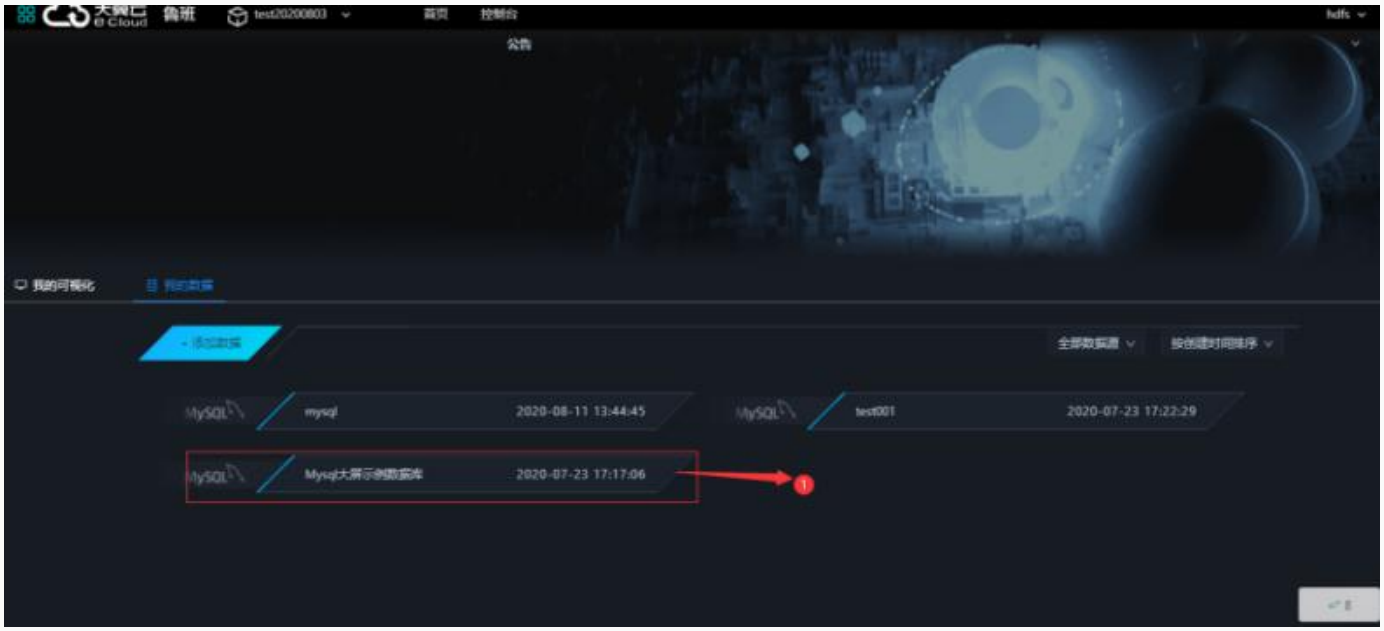
请确保数据库可以被公网访问

请确保数据库没有被防火墙禁止

请确保数据库域名能够被解析

请确保数据库已经启动

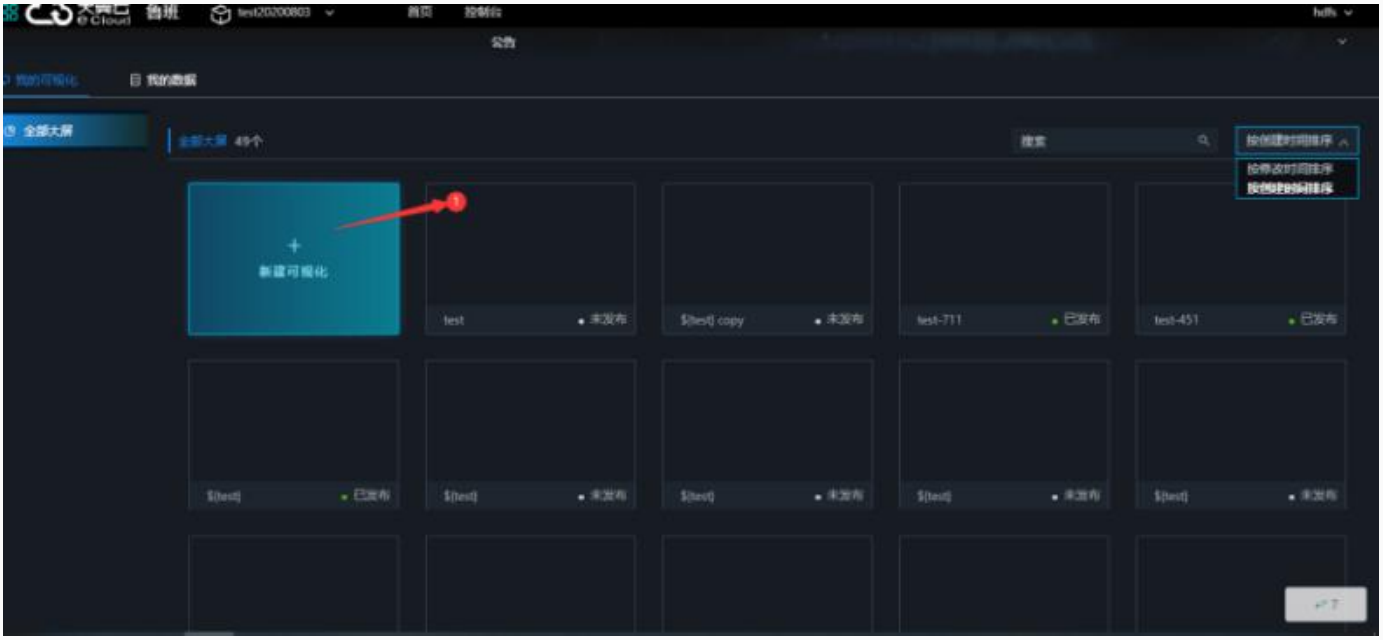
创建完成，并且测试通过后，确认数据源中已有刚刚创建的数据源。



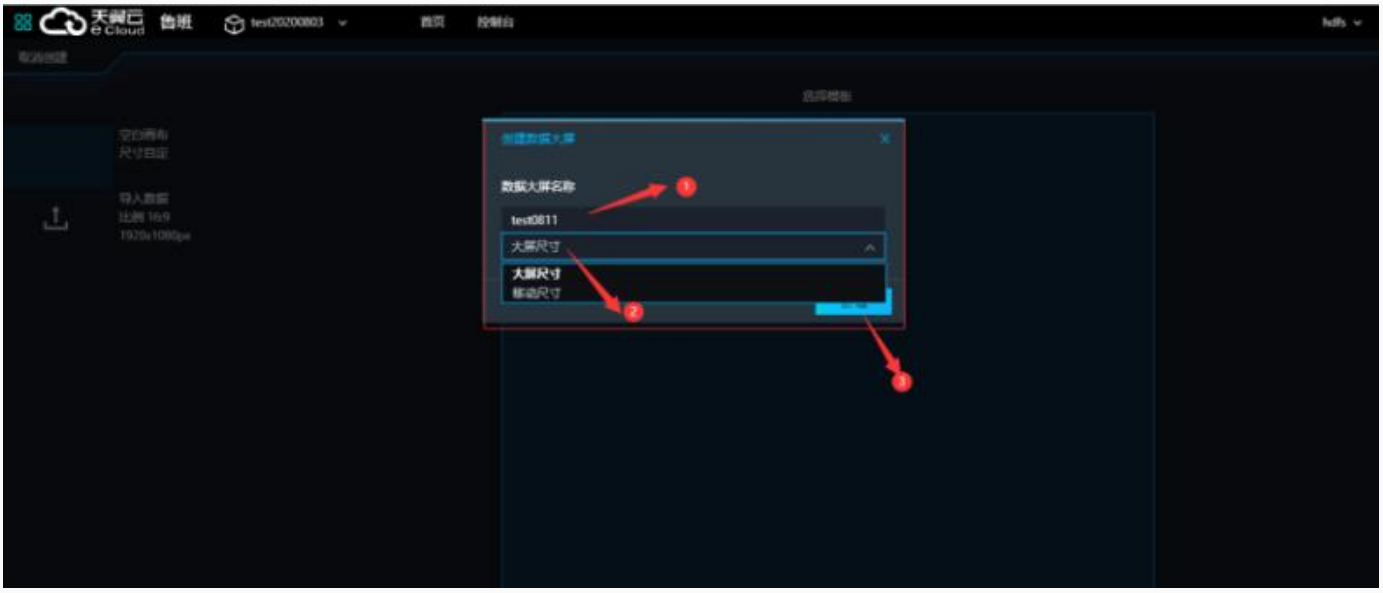
创建画布

进入页面

进入天翼视屏后点击页面 **新建可视化**



进入页面点击 **创建**，在弹框中填写大屏名称以及选择大屏尺寸下拉框，点击 **创建**。



创建成功后跳转页面，

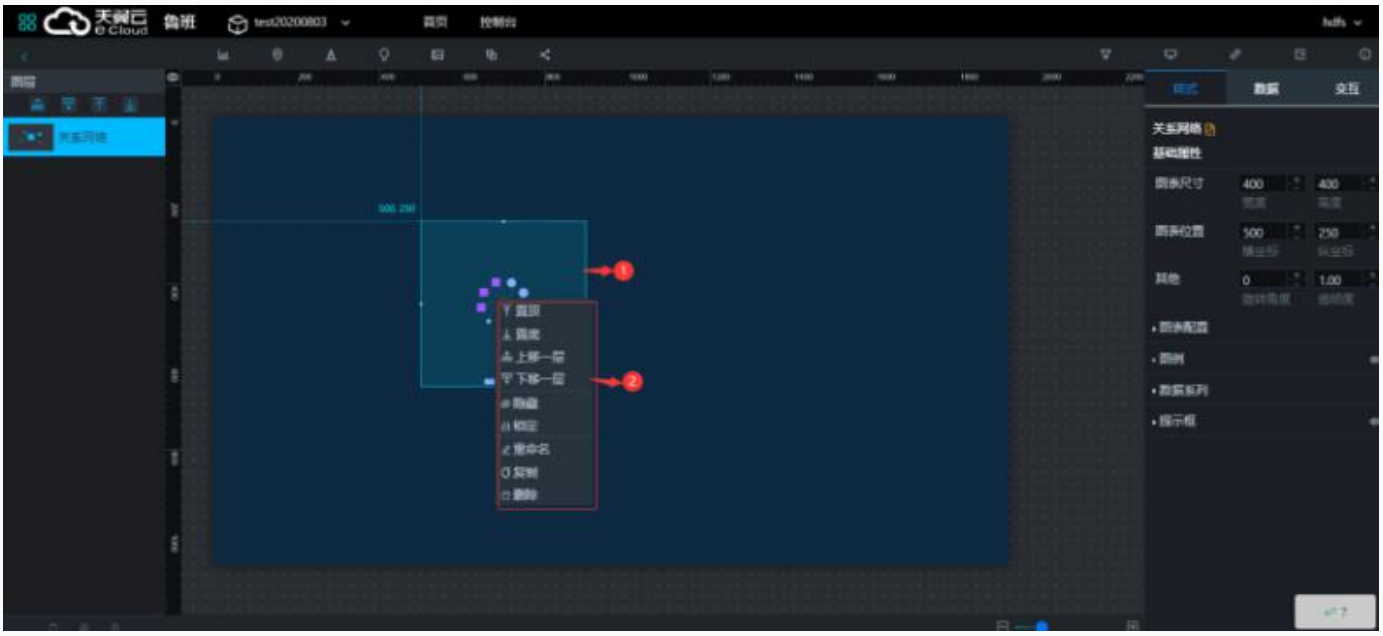
页面介绍

在页面中可以1设计屏幕大小，2背景颜色，3背景图，4页面缩方式，5查看缩略图。

添加可视化组件

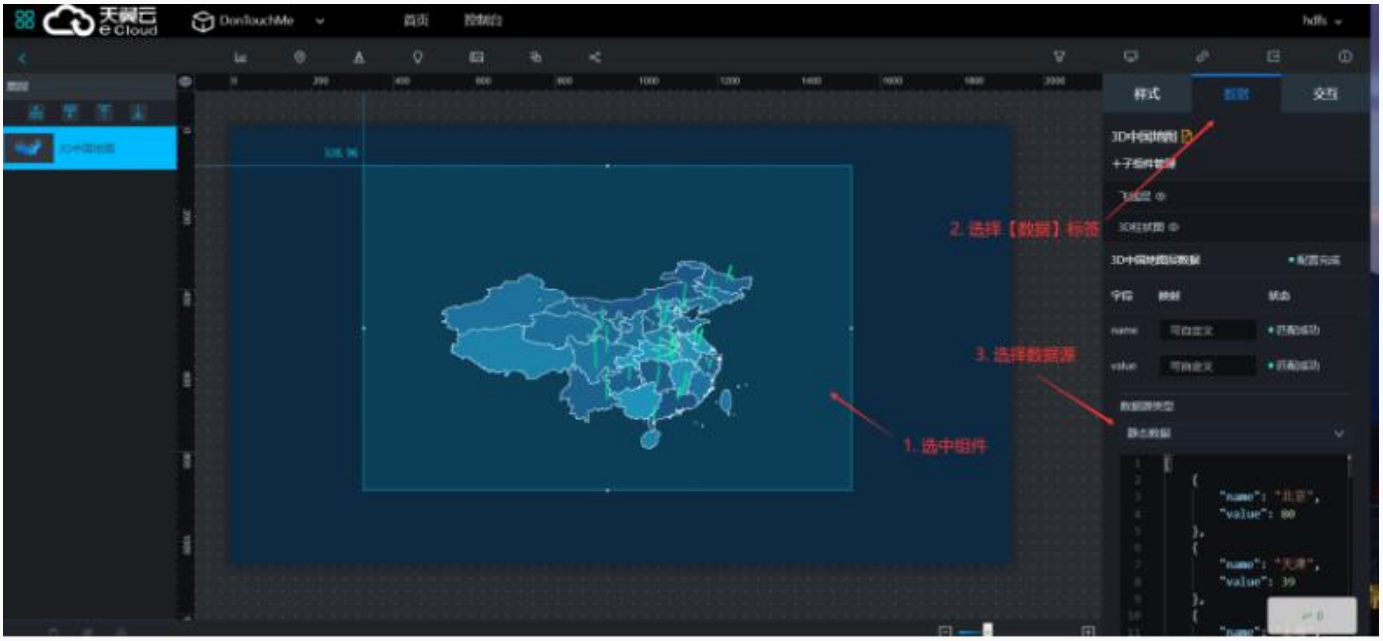
选择组件区的组件，点击后，图表会出现在画布区，可用的组件包括各种统计图标、地图、文字、轮播等

调整图表，右侧配置部分可对大小位置进行调整，右键点击图片可以对其图片进行置顶，置底，上移一层，下移一层等操作。直到视效满意



关联数据源

1. 选中画布中的组件
2. 点击功能区的 **数据** 选项卡
3. 数据源可选择静态数据、接口（支持http和https2.0）以及预配置的上述多种数据源。



数据源支持以下类型：

以mysql数据源为例使用方式如下：

预览检查并发布

预览检查后，点击发布，开启发布开关，即可通过链接进行分享，并进行权限控制

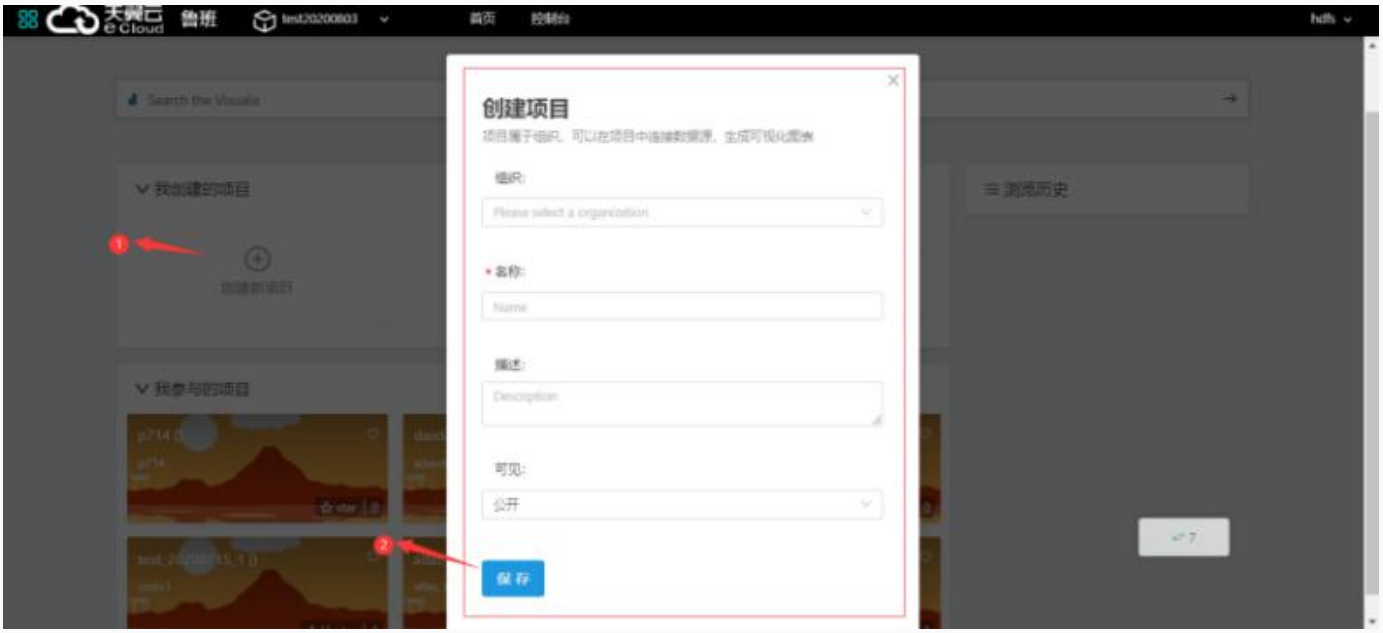
功能说明

Visualis支持拖拽式报表定义、图表联动、钻取、全局筛选、多维分析、实时查询等数据开发探索的分析模式，并做了水印、数据质量校验等金融级增强。

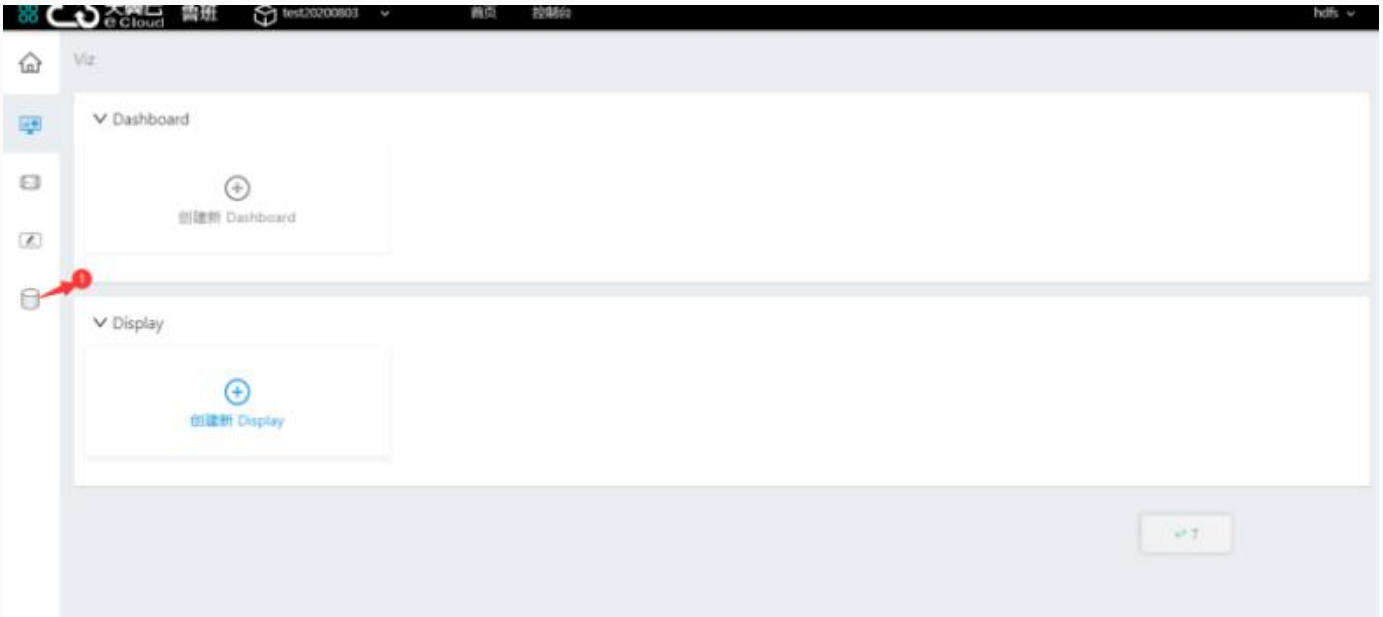
创建一个dashboard

创建数据源

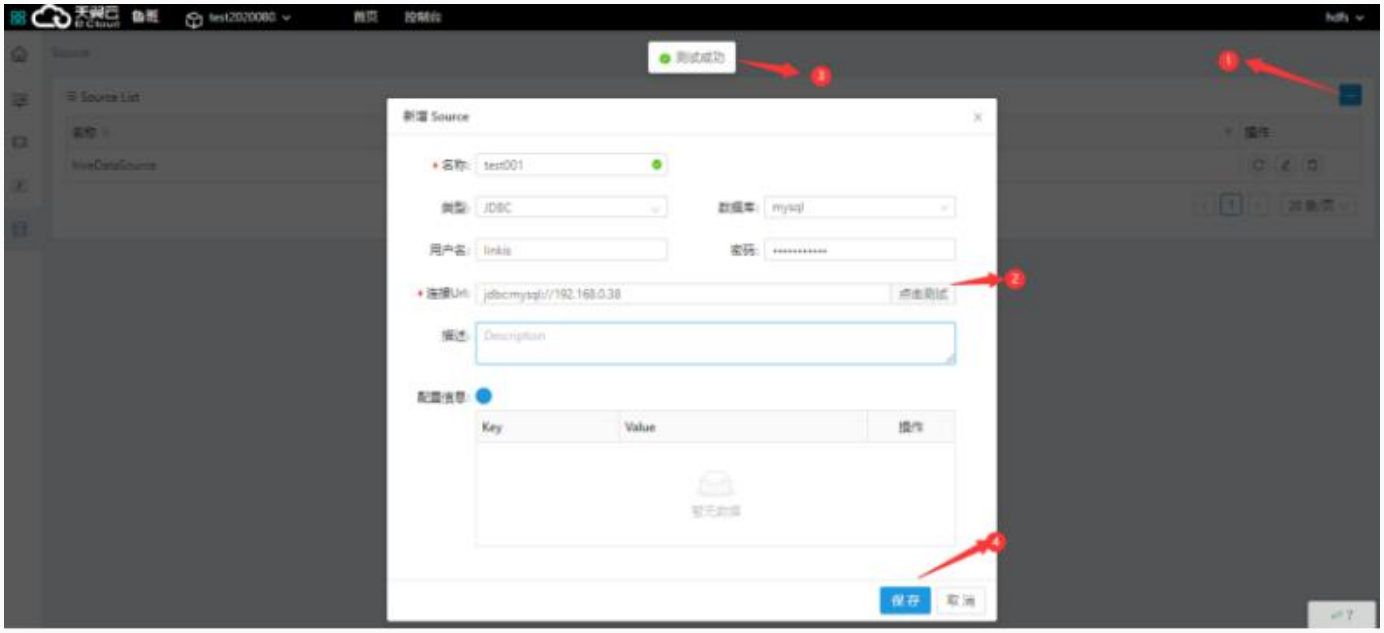
在任意工作空间下，点击**数据分析 > 进入Visualis**，点击创建新项目



在弹框中选择**组织**(非必选)–**名称**–**描述**（非必选）–选择**可见**，点击**保存**。
点击已创建项目进入页面，点击source图标。

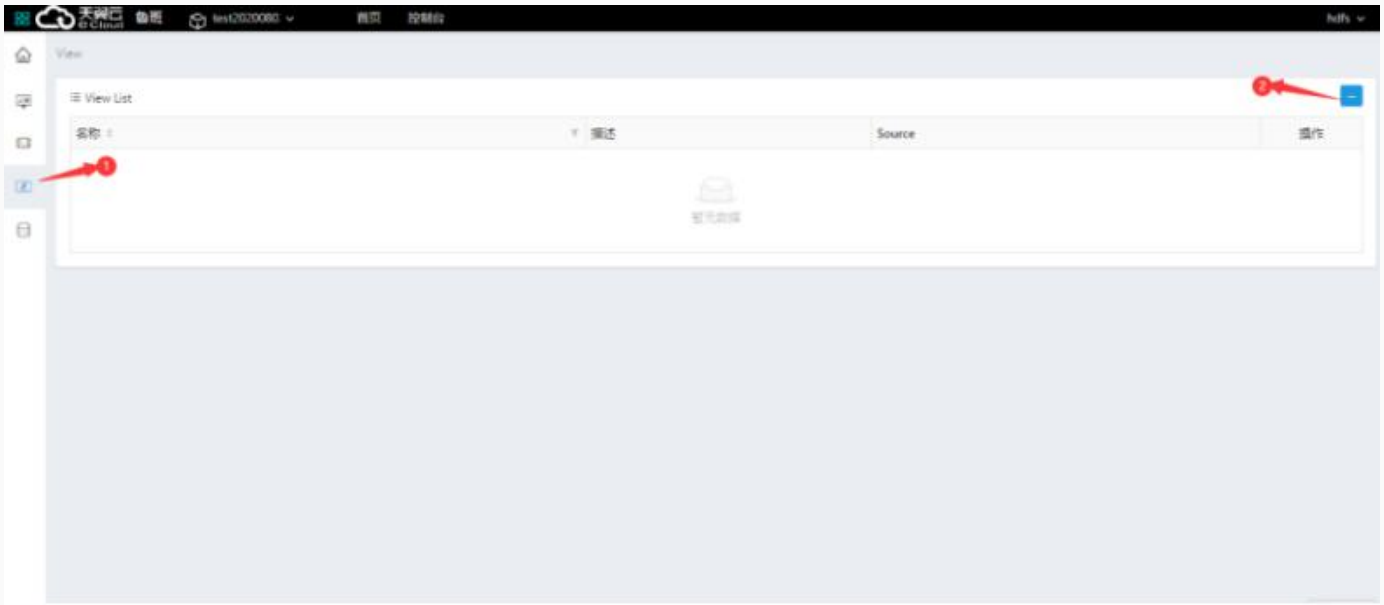


点击页面**新增**按钮，填写相应信息**点击测试**，右上角提示测试成功后，点击**保存**。

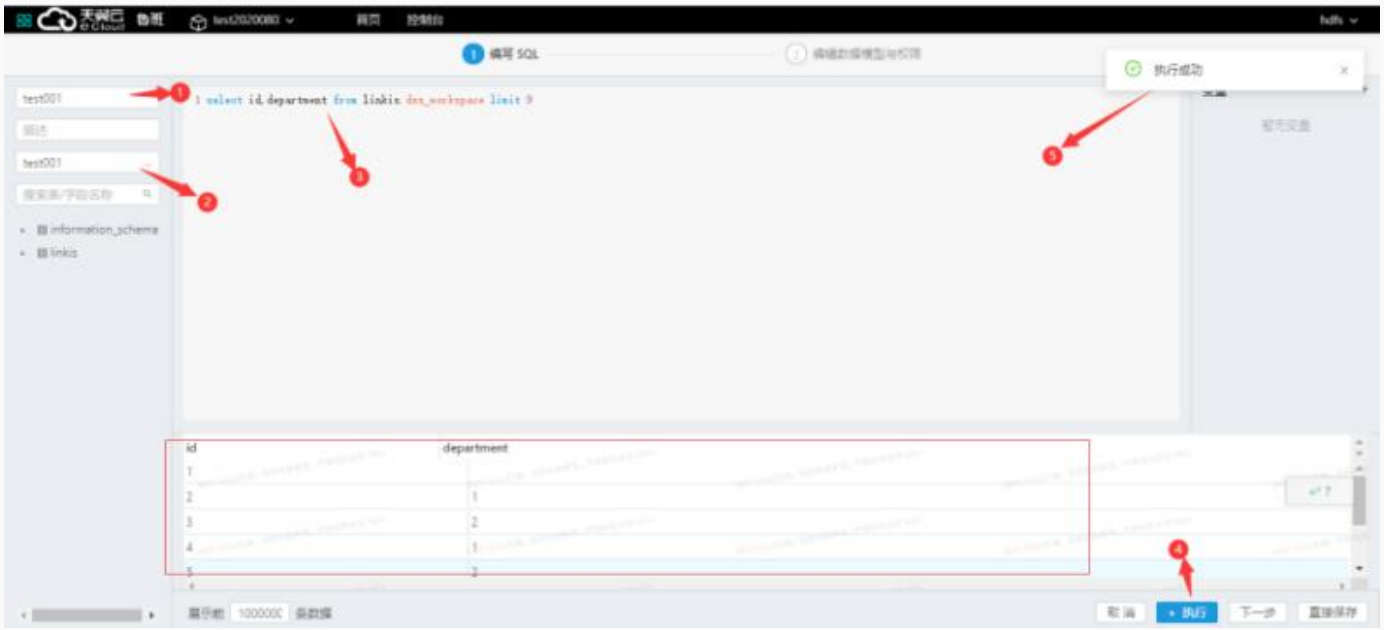


创建视图

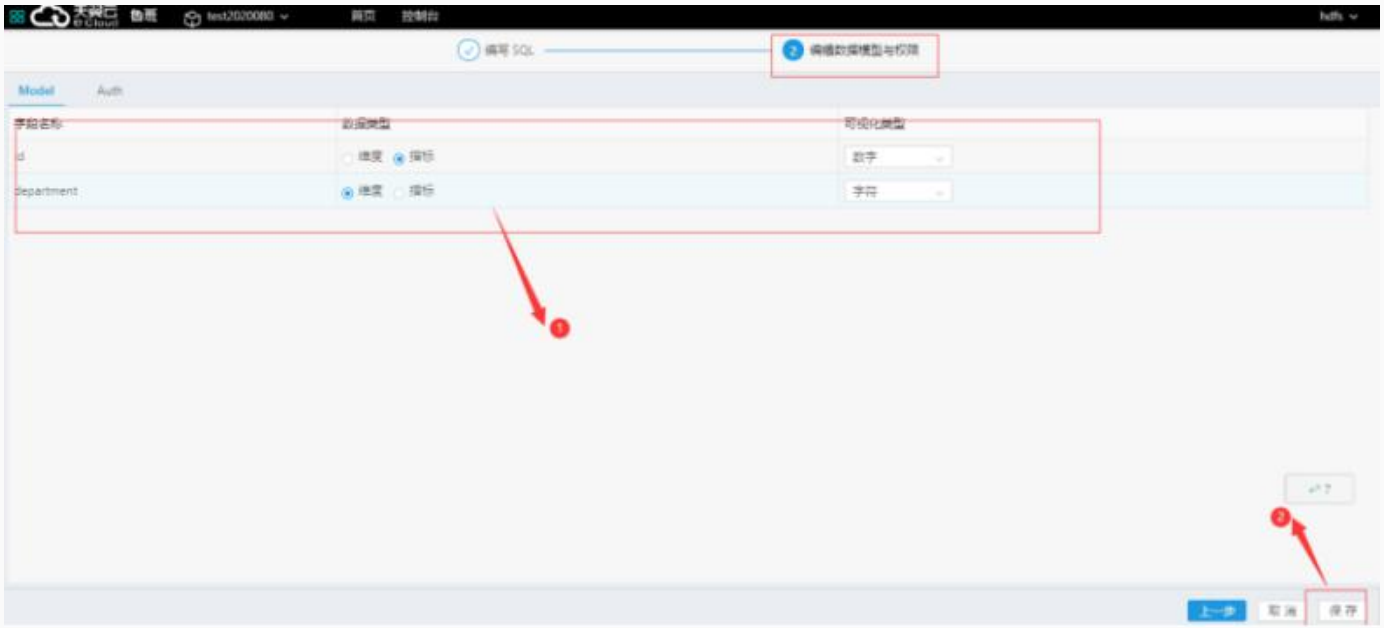
点击View图标，在页面点击**新增**按钮。



跳转页面，在页面中填写名称，以及选择source，填写SQL语句，点击**执行**

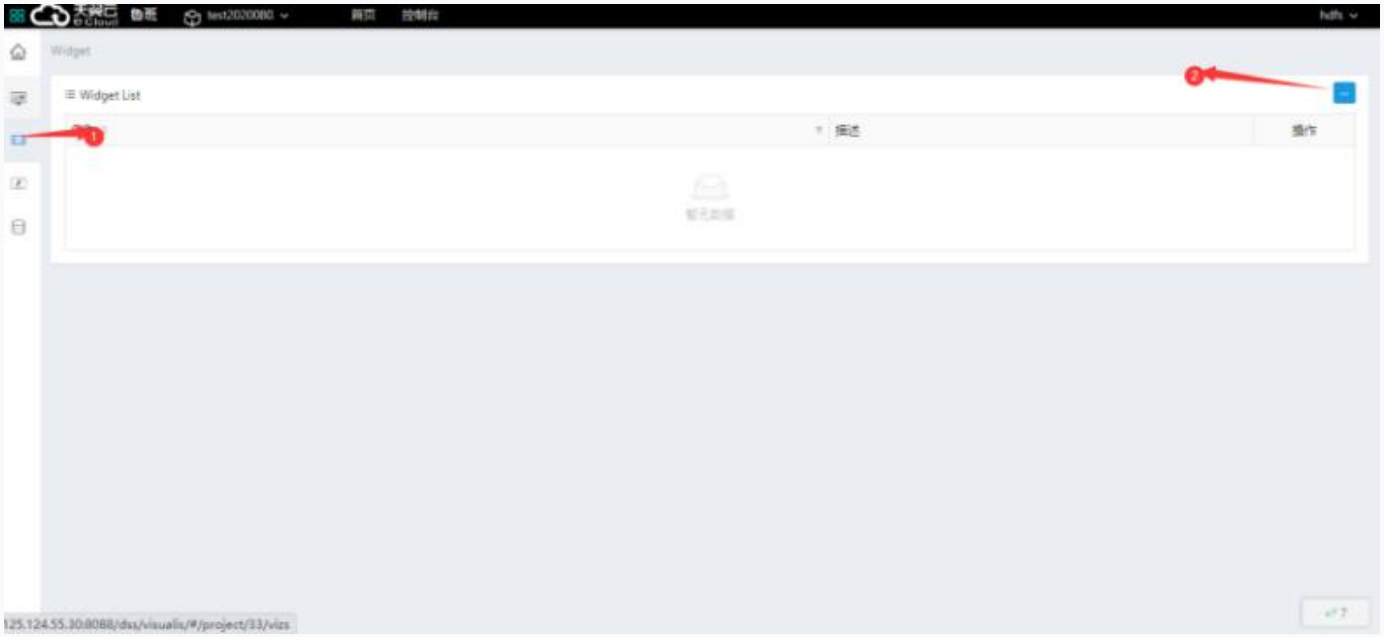


点击**下一步**编写数据模型与权限，点击**保存**。



创建视图组件

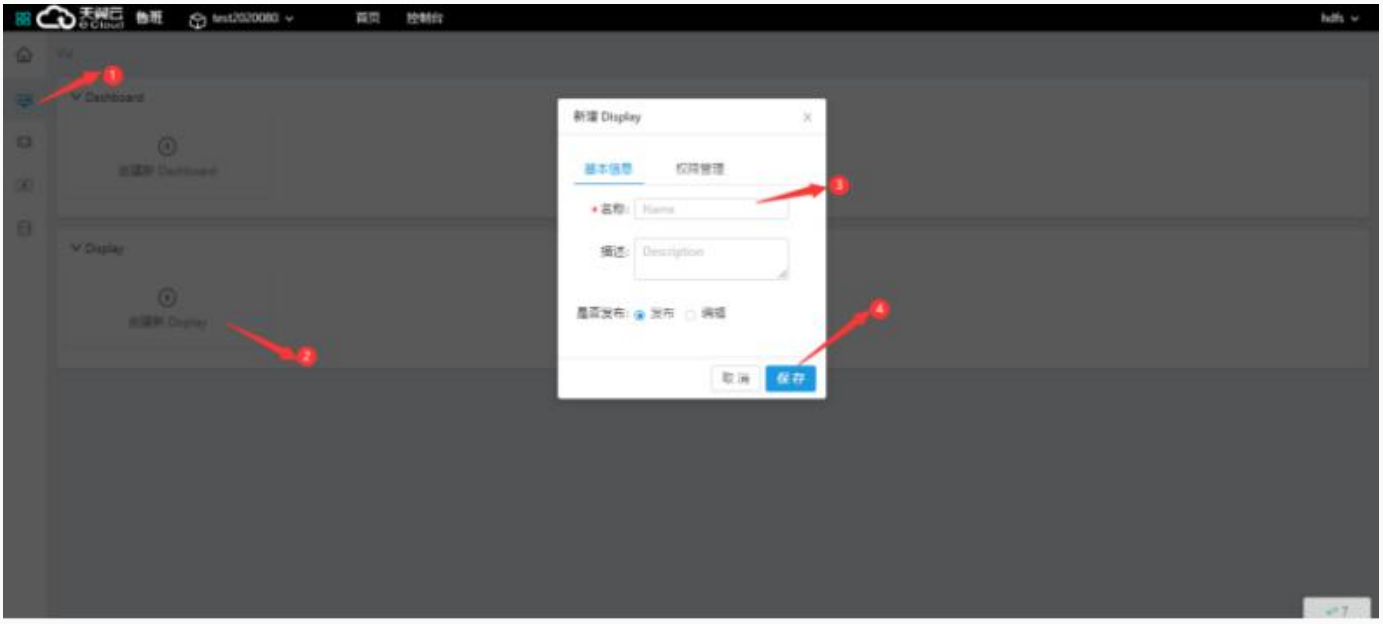
点击widget图标进入页面，点击**新增**widget按钮。



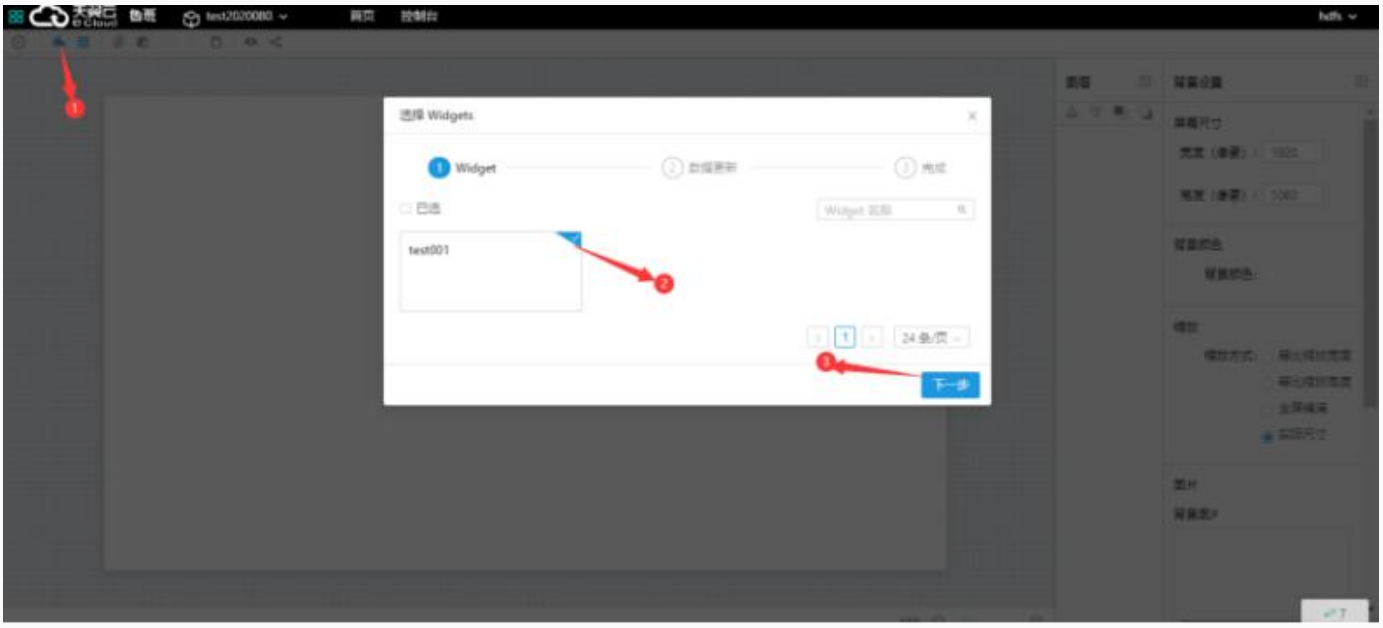
跳转页面填写widget名称，选择View，**拖入维度指标**点击**执行**成功后，点击**保存**按钮。

创建显示页面

点击Viz图标进入页面，点击创建Display在弹框中填写**名称**–**描**述（非必选）–选择**是否发布**，点击**保存**。



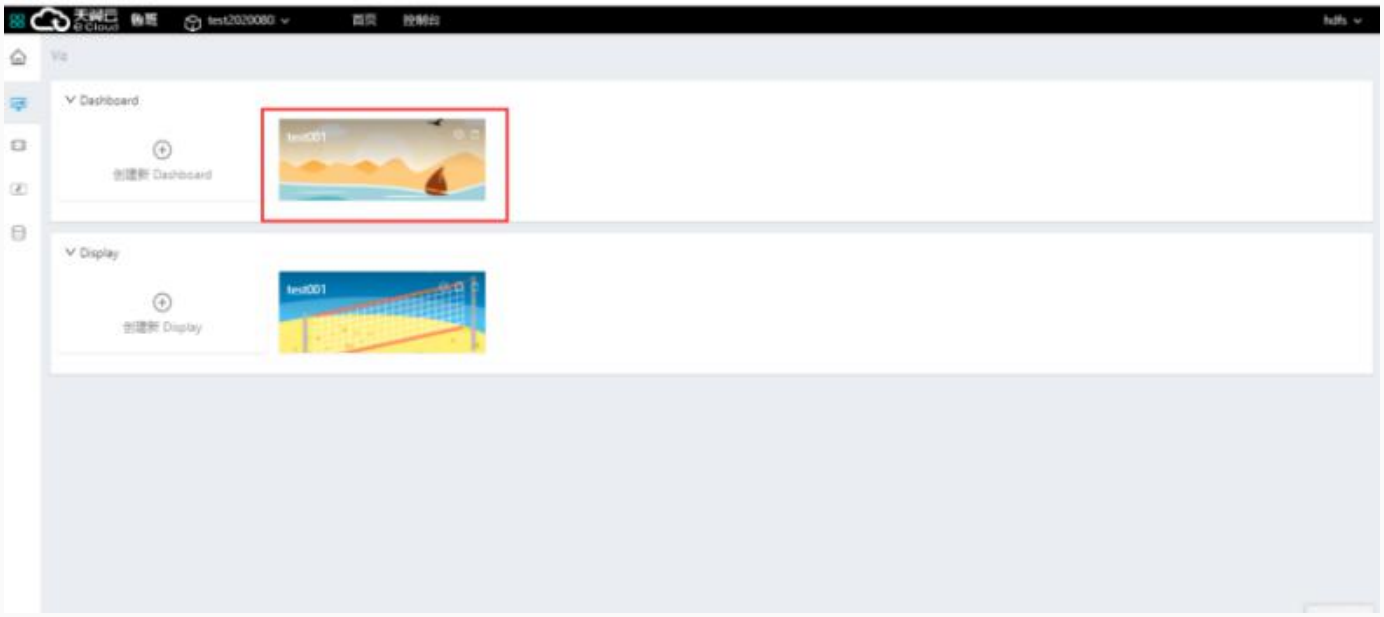
点击已创建的Display进入页面，点击页面中的图表图标，在弹框中勾择已创建的widget



创建Dashboard（可选步骤）

弹框中填写**名称**–**描述**（非必选）–选择**是否发布**，点击**保存**。


Dashboard
创建新 Dashboard
Display
创建新 Display
test001
新建 Portal
基本信息
权限管理
名称:
test001
描述:
Description
是否发布:
发布
编辑
取消
保存


Dashboard
创建新 Dashboard
test001
Display
创建新 Display
test001

功能说明

Qualitis是一套金融级、一站式的数据质量管理平台，提供了数据质量模型定义，数据质量结果可视化、可监控等功能，并用一整套统一的流程来定义和检测数据集的质量并及时报告问题。

创建项目

进入某个已有的项目 点击 **数据质量** > **进入Qualitis**

点击左侧栏的**规则配置**。在其下的二级菜单中，点击**我的项目**，进入项目模块。

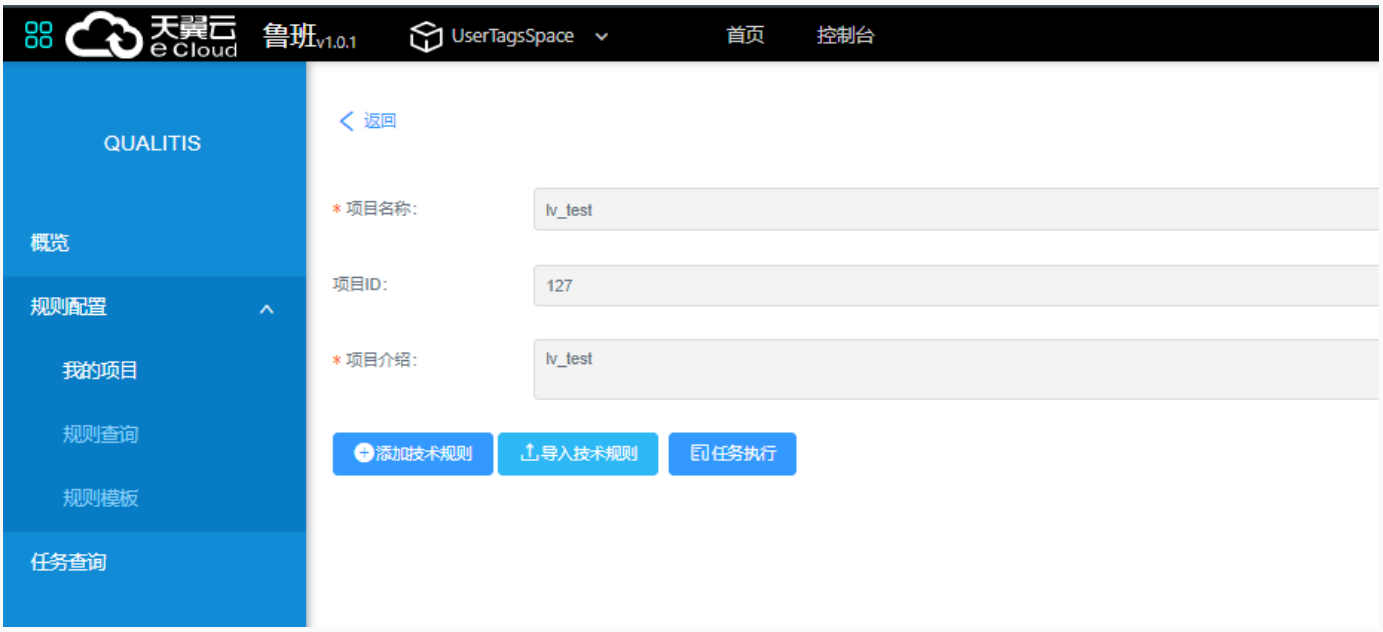
点击页面左上方的新增项目，弹出新增项目界面。

填入 **项目名称 项目介绍**

点击确定即可。

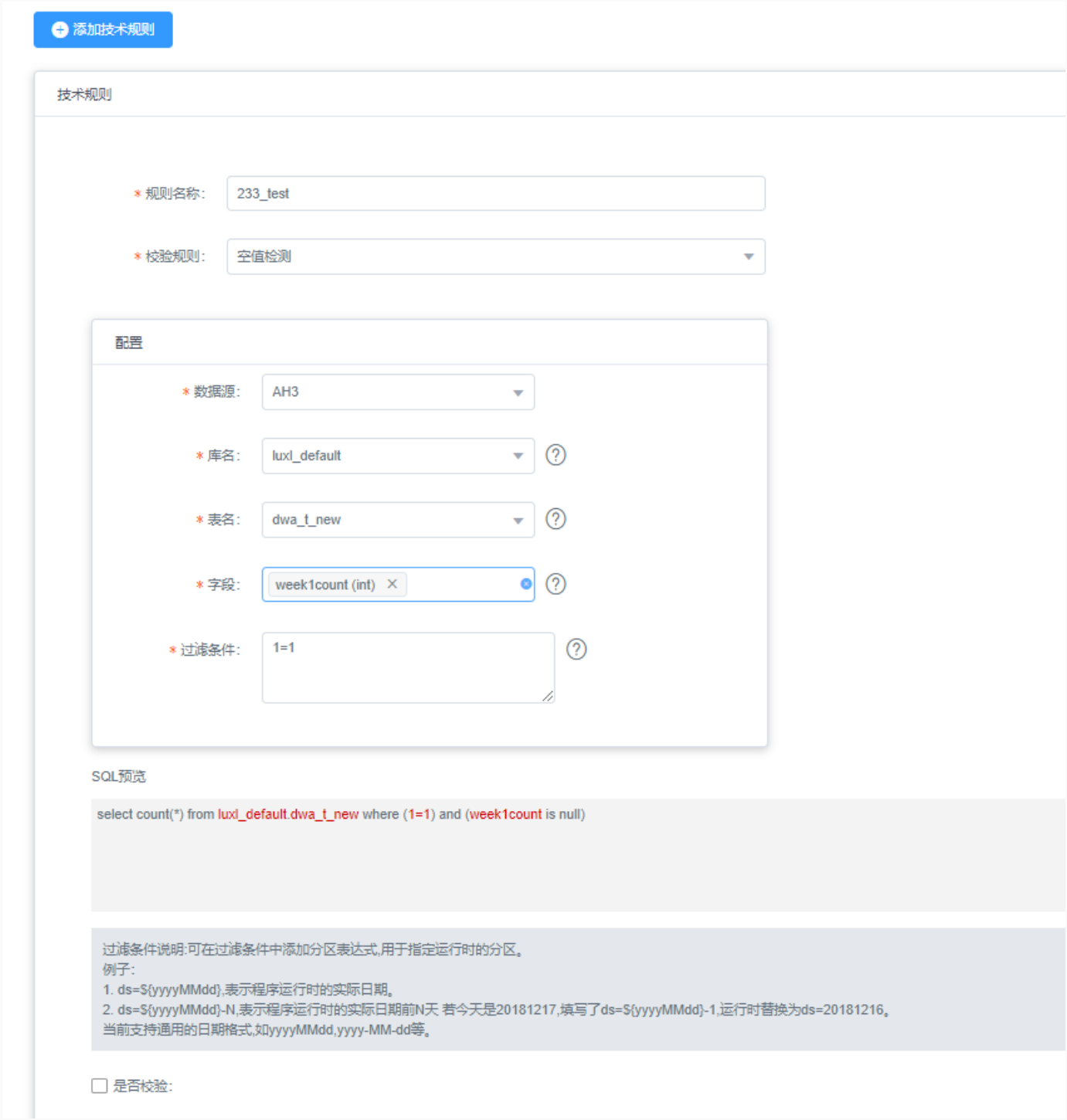
创建规则

保存项目之后，左下角可以**添加技术规则**。可以新建单表规则，自定义规则以及跨表规则。



单表规则创建示例

点击**添加技术规则**，进入**新增单表技术规则**页面。



规则名称: 技术规则的名称，不可重复。

校验规则: 系统的校验模版，选择不同的校验模版，可以进行不同的校验。

配置: 选择模版中真正校验的数据源，选择数据源，会替换掉校验模版中的占位符，真正执行的SQL语句可以在SQL预览中查看。其中过滤条件可以填写系统提供的表达式，该表达式会在任务真正执行的时候，替换成实际的日期时间进行执行。

提供的表达式如下：

表达式	当天日期	替换值

表达式	当天日期	替换值
${yyyyMMdd}[-N(可选)]	2018年12月17号	20181217(前N天)
${yyyy-MM-dd}[-N(可选)]	2018年12月17号	2018-12-17(前N天)
${yyyyMMddHH}[-N(可选)]	2018年12月17号15点	2018121715(前N天)只减天数
${yyyy/MM/dd}[-N(可选)]	2018年12月17号	2018/12/17(前N天)

是否校验

如果不进行校验，不会监控任务的输出结果。

如果选择校验，并在其中选择监控的输出结果，并设定校验的阈值，当结果超出校验阈值的时候，任务就是不通过校验的状态。

比较运行结果和阈值的方式有以下四种：(假设设定的阈值为x，本次任务的运行结果为r)

1.月波动：将任务的运行结果和本条技术规则本月的运行结果的平均值y进行比较，如果(1-x)*y<=r<=(1+x)*y，任务通过校验，否则任务不通过校验。

2.周波动：和月波动同理，计算的平均值是本周的平均值。

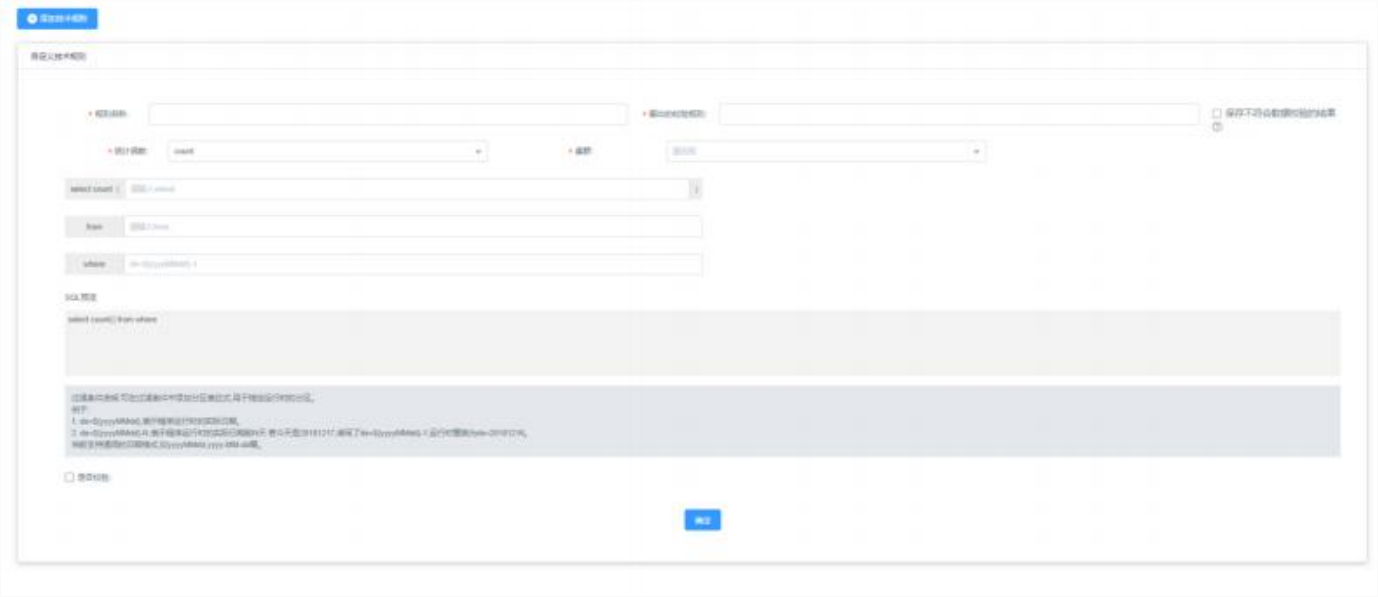
3.日波动：和月波动周波动同理，计算的平均值是本日的平均值。

4.固定值：和一个固定值进行比较，比较的方式有等于，大于等等，如果比较选择比较方式是等于，那么如果r=x，那么任务不通过校验。

如上图所示，上图创建了一个监控字段不为空的的技术规则。在SQL预览中可以看到实际执行的SQL语句。 质量校验那里，会监控字段为空的数目，如果字段为空的数目不为0，那么会不通过校验。

自定义规则创建示例

点击**新增自定义技术规则**，进入**新增自定义技术规则**页面。



规则名称：技术规则的名称，不可重复。

输出的校验规则：校验的列的别名

统计函数：自定义SQL的统计函数

集群：选择提交任务的集群

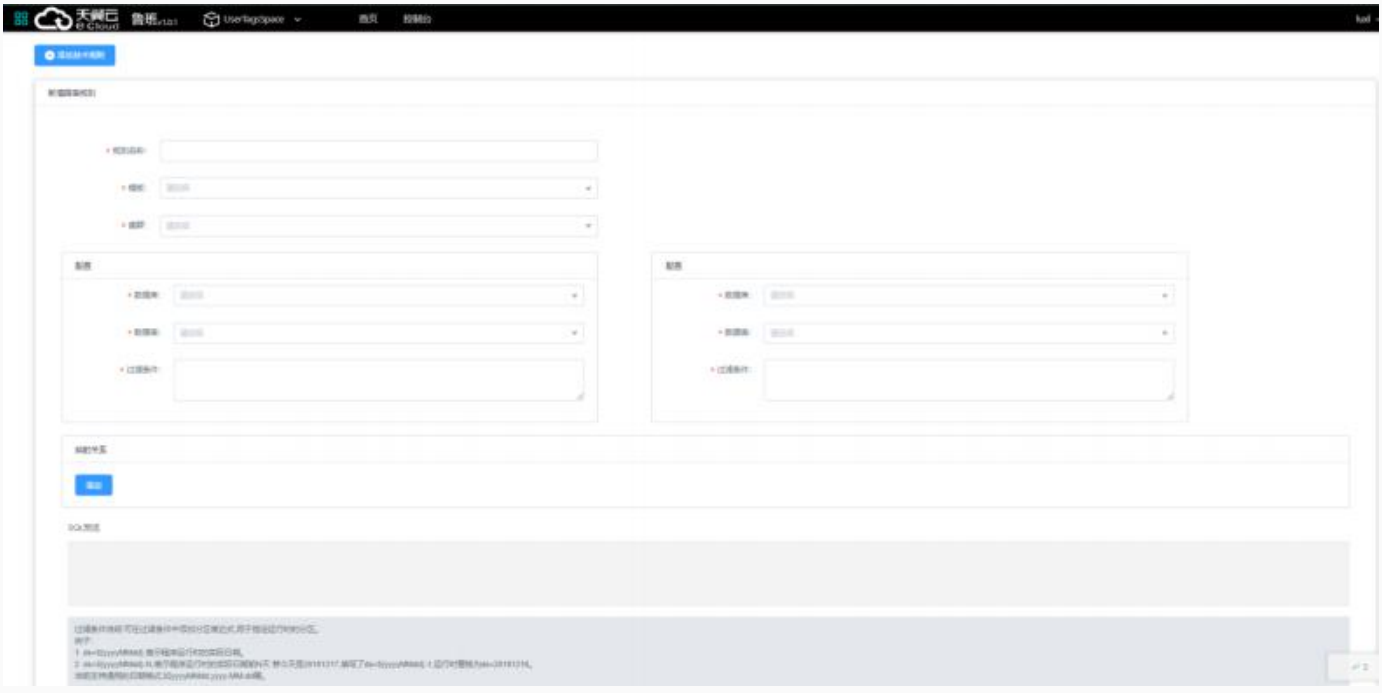
保存不符合数据校验的结果：如果勾选，则会将没通过校验的数据提取出来并保存，否则不会保存；

将要执行的SQL语句会在预览中显示；

自定义技术规则会生成自定义的规则模板。

跨表规则创建示例

点击**新增跨表技术规则**，进入**新增跨表技术规则**页面。



跨表技术规则能选择同一个集群中两张表，并对两张表进行数据校验。
跨表技术规则目前提供以下两种默认模版，：

- 准确性校验 准确性校验可以比较两个表之间，所选字段的数据记录的准确性差异性。
- 通用校验 通过通用校验模版，可自定义跨表校验SQL。

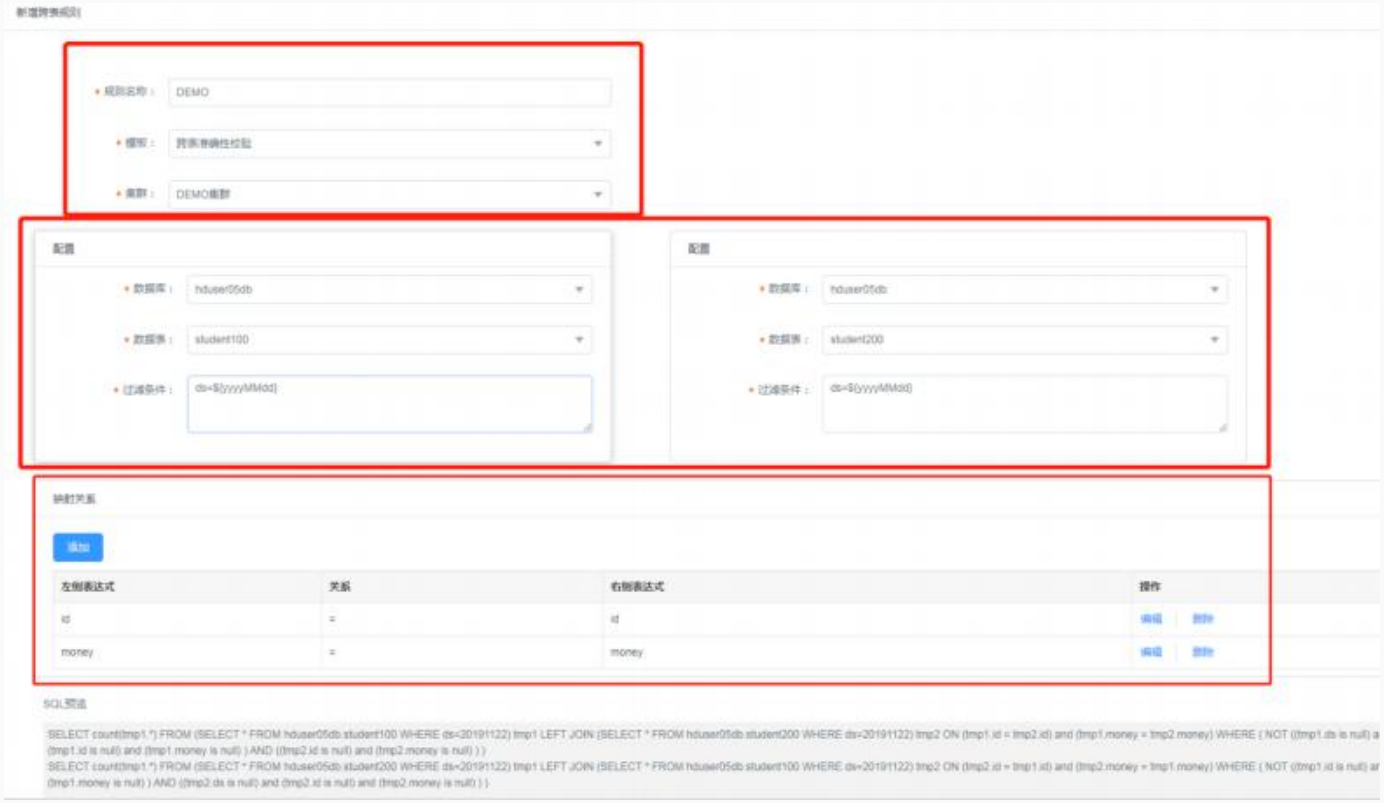
以准确性校验为例:
有表A和表B，数据分别如下：
A表

ColumnA1	ColumnA2
a	1
b	2

B表

ColumnB1	ColumnB2
a	1
b	3

如果对比ColumnA1和ColumnB1中的数据，则表A和表B完全一致。 如果对比ColumnA1和ColumnB1，以及ColumnA2和ColumnB2中的数据，则表A和表B中的数据有一条不一致。
配置方法如下所示：


DEMO
DEMO集群
配置
hduser05db
student100
ds=${yyyyMMdd}
配置
hduser05db
student200
ds=${yyyyMMdd}
映射关系
添加
左侧表达式
关系
右侧表达式
操作
id
=
id
编辑
删除
money
=
money
编辑
删除
SQL预览

规则模板配置介绍

以下例子均以Cluster0002集群，allenzhou_ind库，test_table表为例。假设该表结构如下：

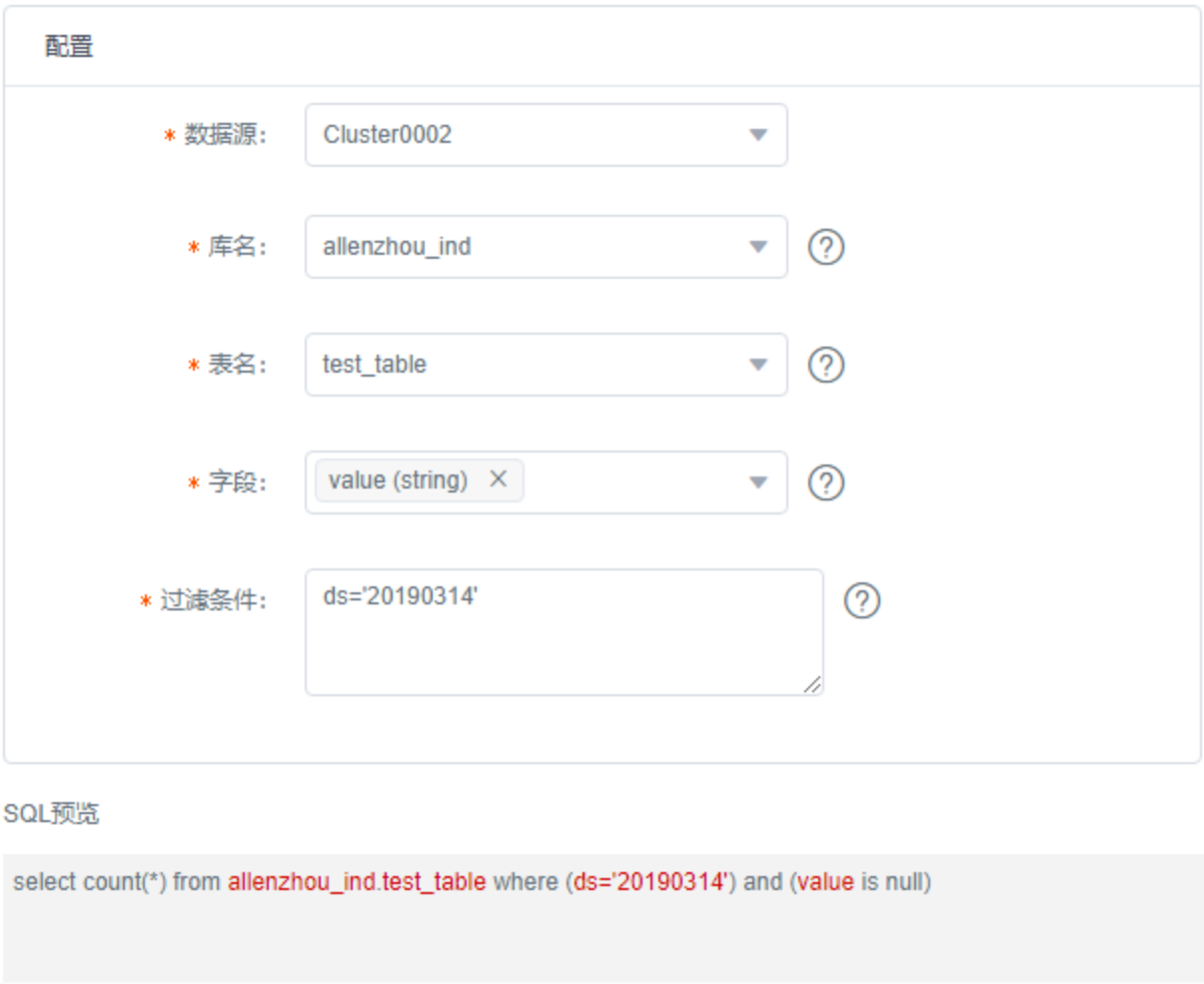
字段名称	字段类型
key	int
value	string
ds	string(一级分区)
day	string

空值检测

语义： 指定一个表中的某一个字段，检测出该字段为空的记录条数。

举例： 需求：希望找出Cluster0002集群中，allenzhou_ind库中，test_table表，在ds='20190314'分区下，value字段为空的记录条数。

配置： 首先选择空值检测模版。 选择希望检测的集群，库名，表名，字段，并填入分区过滤条件，如下图所示：



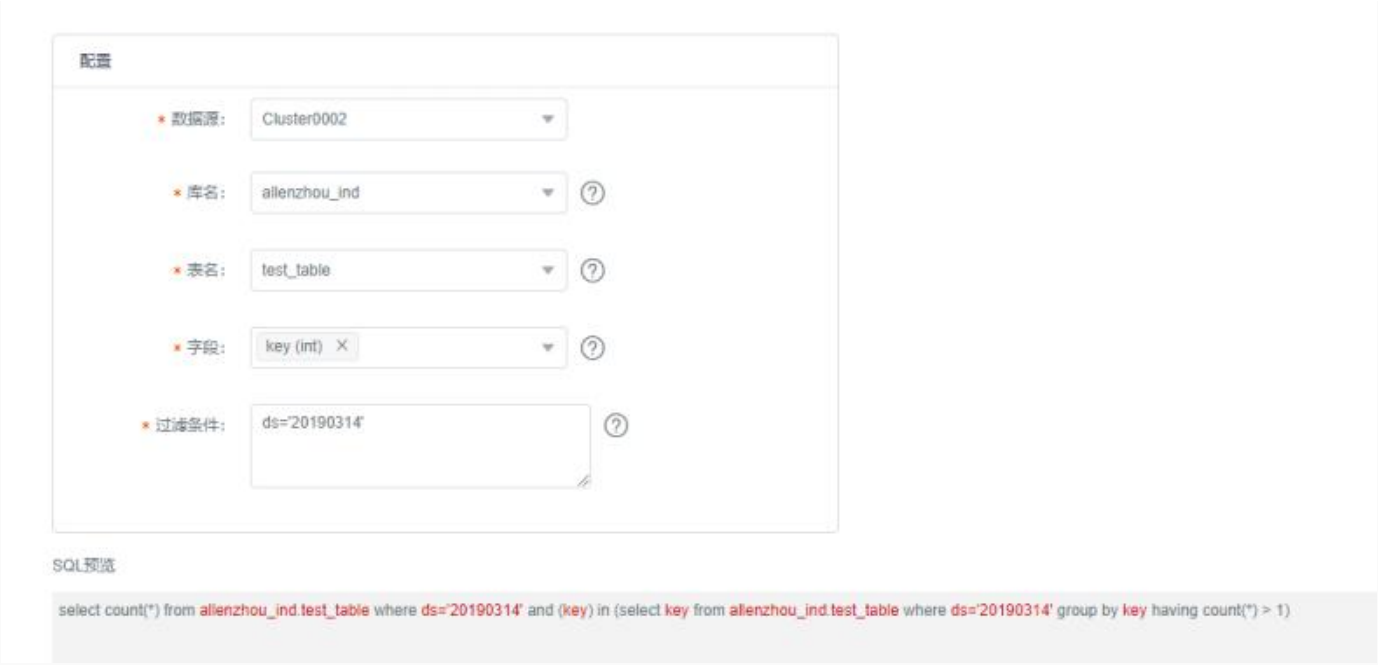
可以在SQL预览中查看执行的SQL是否和预期一致。

主键检测

语义： 指定一个表中的多个字段，检测这些字段的组合在该表中是否具有唯一性。

举例： 需求：希望找出Cluster0002集群中，allenzhou_ind库中，test_table表，在ds='20190314'分区下，key字段是否具有唯一性。

配置： 首先选择主键检测模版。 选择希望检测的集群，库名，表名，字段，并填入分区过滤条件，如下图所示：



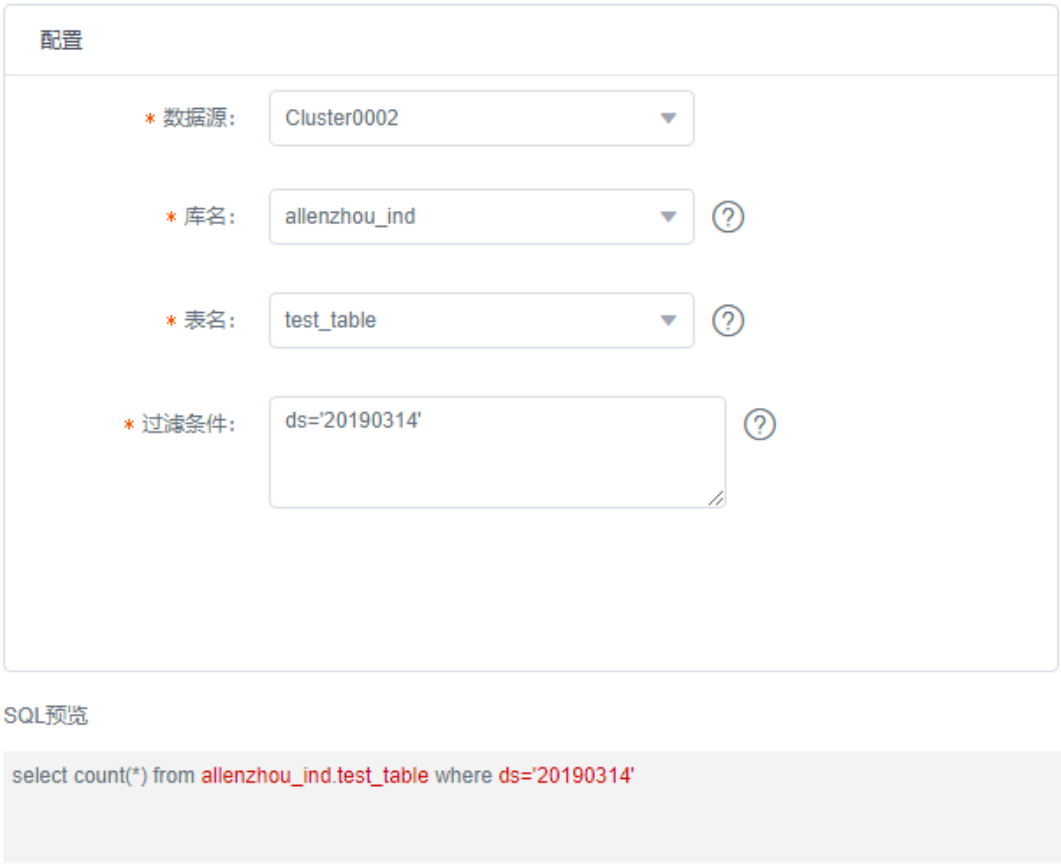
可以在SQL预览中查看执行的SQL是否和预期一致。

表行数检测

语义： 指定一个表，检测该表的行数是否达到预期。

举例： 需求：希望找出Cluster0002集群中，allenzhou_ind库中，test_table表，在ds='20190314'分区下，记录条数为多少。

配置： 首先选择表行数检测模版。 选择希望检测的集群，库名，表名，字段，并填入分区过滤条件，如下图所示：



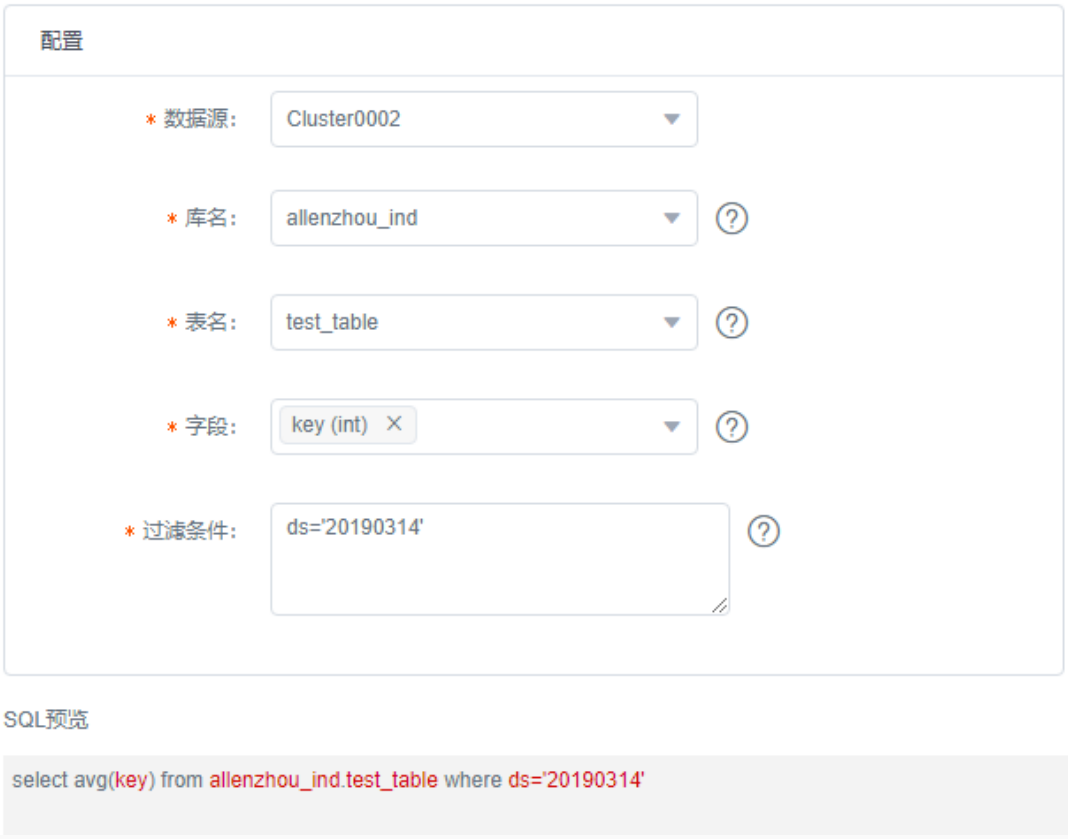
可以在SQL预览中查看执行的SQL是否和预期一致。

平均值检测

语义： 指定一个表中一个字段，检测该字段的平均值是否达到预期。

举例： 需求：希望找出Cluster0002集群中，allenzhou_ind库中，test_table表，在ds='20190314'分区下，key字段的平均值为多少。

配置： 首先选择平均值检测模版。 选择希望检测的集群，库名，表名，字段，并填入分区过滤条件，如下图所示：



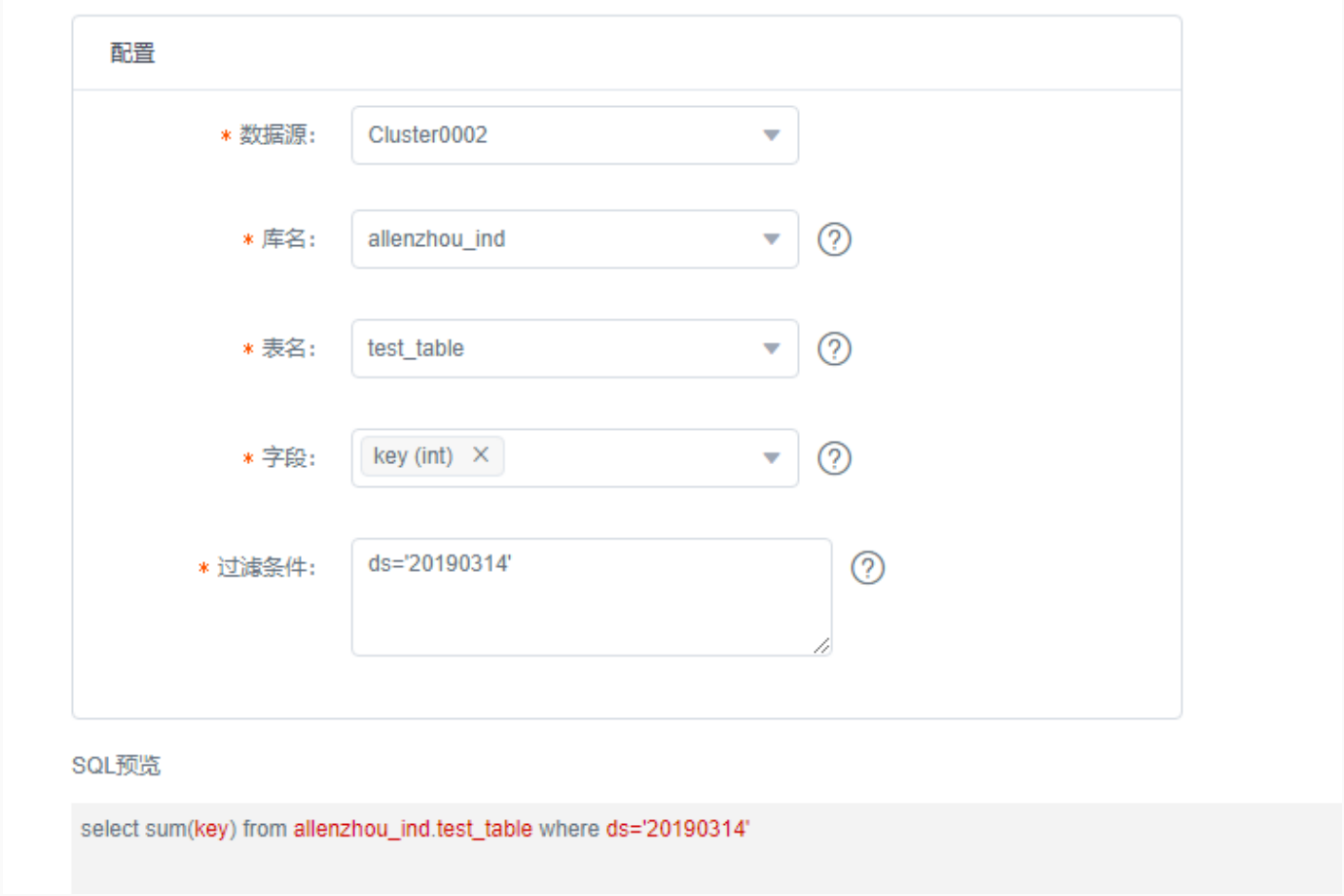
可以在SQL预览中查看执行的SQL是否和预期一致。

总和检测

语义： 指定一个表中一个字段，检测该字段的总和是否达到预期。

举例： 需求：希望找出Cluster0002集群中，allenzhou_ind库中，test_table表，在ds='20190314'分区下，key字段的总和值为多少。

配置： 首先选择总和检测模版。 选择希望检测的集群，库名，表名，字段，并填入分区过滤条件，如下图所示：



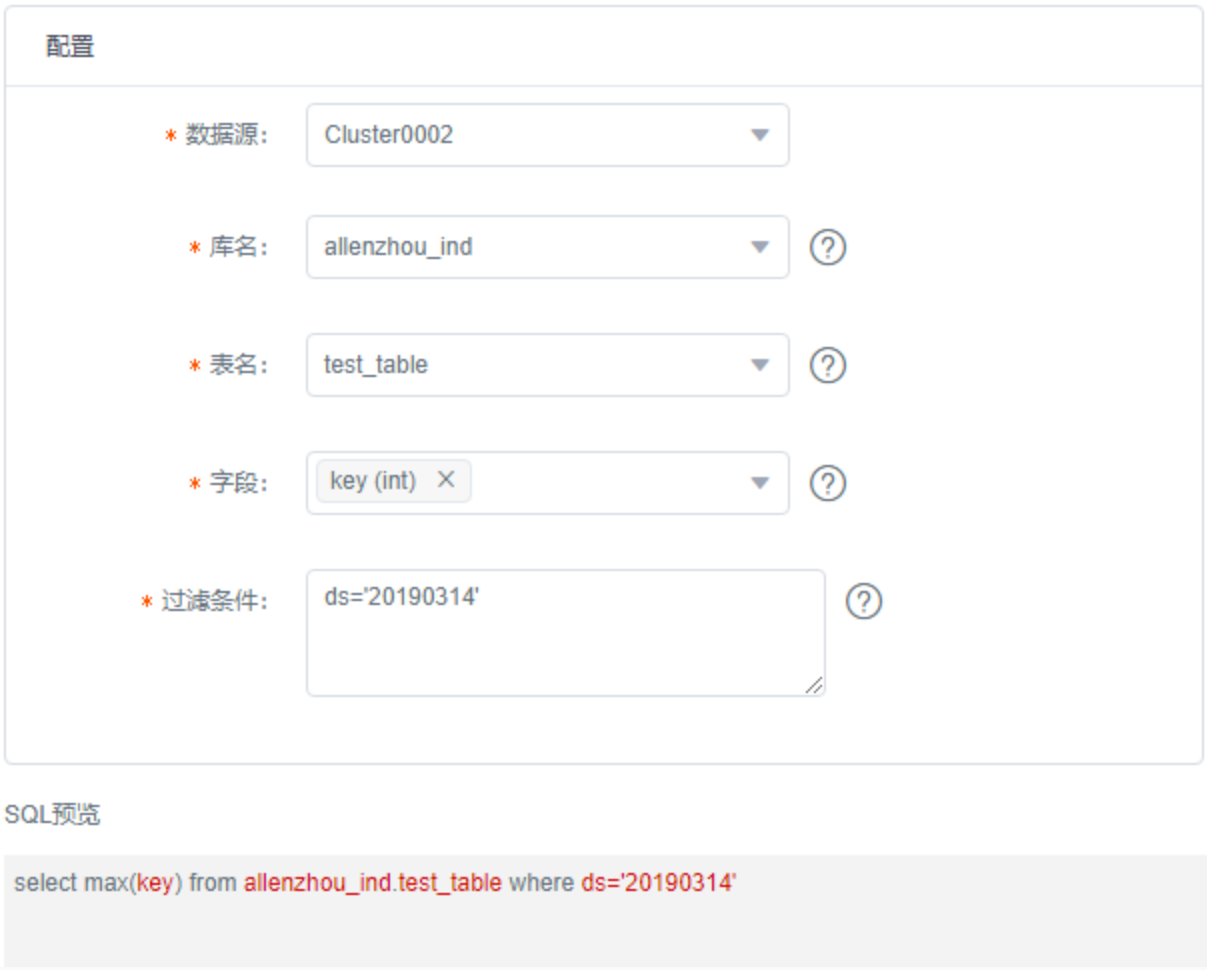
可以在SQL预览中查看执行的SQL是否和预期一致。

最大值检测

语义： 指定一个表中一个字段，检测该字段的最大值是否达到预期。

举例： 需求：希望找出Cluster0002集群中，allenzhou_ind库中，test_table表，在ds='20190314'分区下，key字段的最大值为多少。

配置： 首先选择最大值检测模版。 选择希望检测的集群，库名，表名，字段，并填入分区过滤条件，如下图所示：



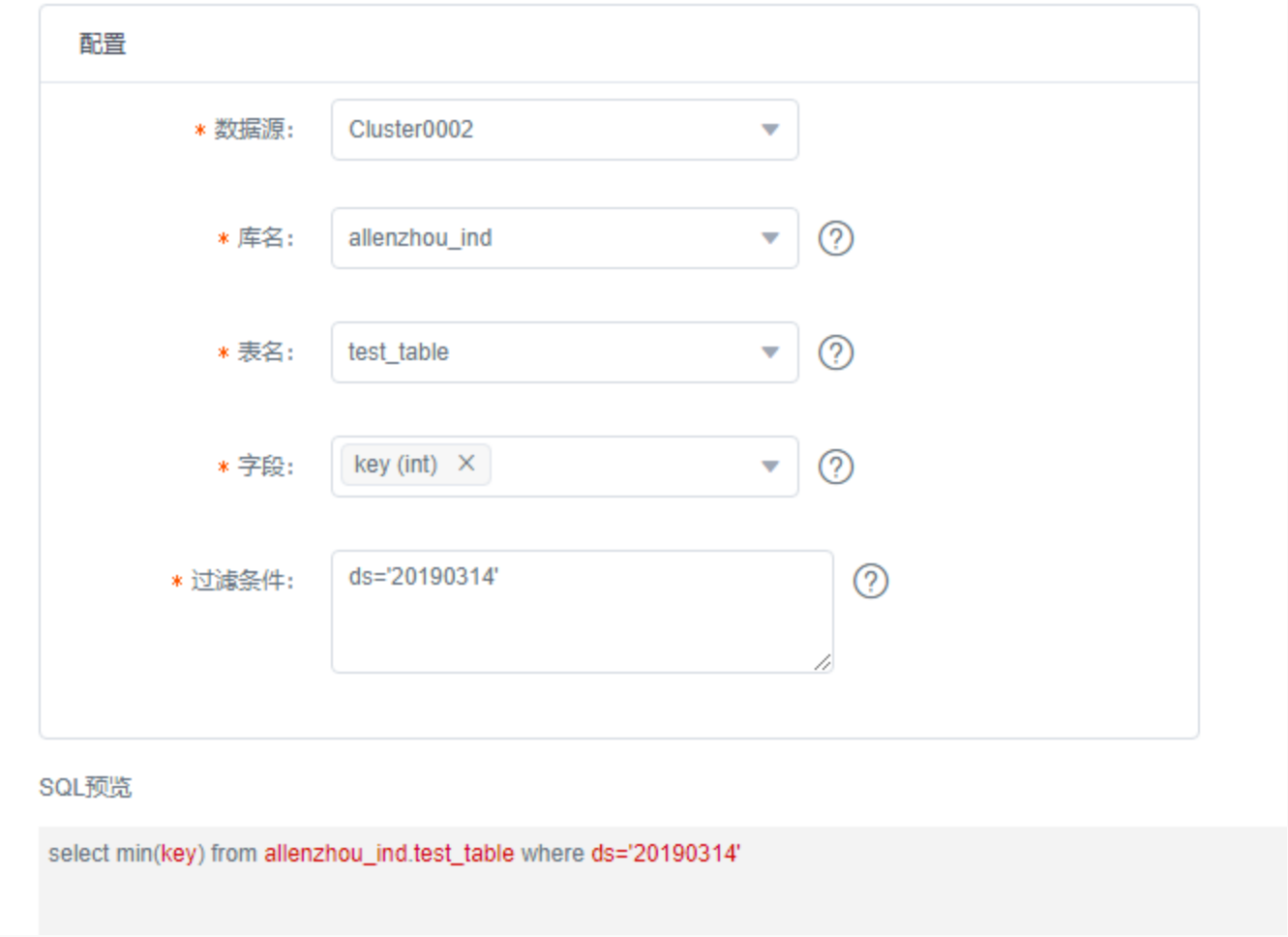
可以在SQL预览中查看执行的SQL是否和预期一致。

最小值检测

语义： 指定一个表中一个字段，检测该字段的最小值是否达到预期。

举例： 需求：希望找出Cluster0002集群中，allenzhou_ind库中，test_table表，在ds='20190314'分区下，key字段的最小值为多少。

配置： 首先选择最小值检测模版。 选择希望检测的集群，库名，表名，字段，并填入分区过滤条件，如下图所示：



可以在SQL预览中查看执行的SQL是否和预期一致。

正则表达式检测

语义： 指定一个表中一个字段，找出该字段不满足给定正则表达式的记录条数。

举例： 需求：希望找出Cluster0002集群中，allenzhou_ind库中，test_table表，在ds='20190314'分区下，value字段不满足正则表达式'[0-9][a-z][A-Z]'的记录数。

配置： 首先选择正则表达式检测模版。 选择希望检测的集群，库名，表名，字段，并填入分区过滤条件和正则表达式。如下图所示：

可以在SQL预览中查看执行的SQL是否和预期一致。

日期格式检测

语义： 指定一个表中一个字段，找出该字段不满足选中日期格式的字段。

举例： 需求：希望找出Cluster0002集群中，allenzhou_ind库中，test_table表，在ds='20190314'分区下，day字段不满足日期格式yyyyMMdd的记录数。

配置： 首先选择日期格式检测模版。 选择希望检测的集群，库名，表名，字段，并填入分区过滤条件，选中日期格式。如下图所示：

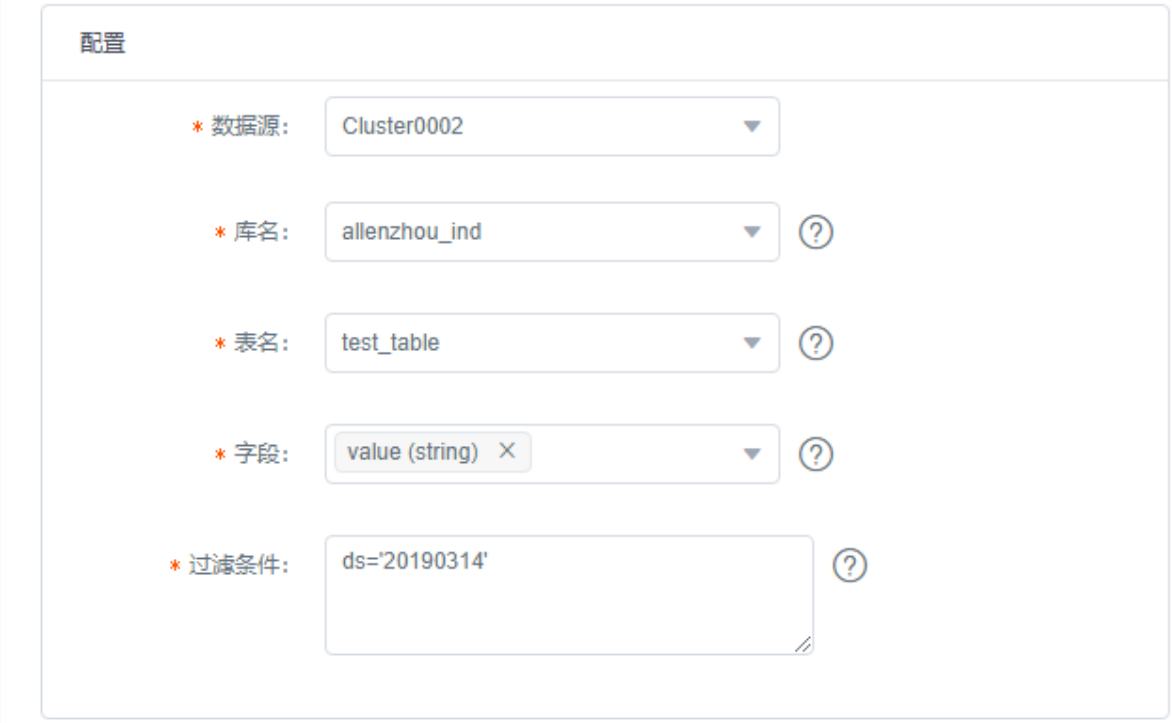
可以在SQL预览中查看执行的SQL是否和预期一致。

字段数值类型检测

语义： 指定一个表中一个字段，找出该字段不满足数值类型的字段(是不是都是数字)。

举例： 需求：希望找出Cluster0002集群中，allenzhou_ind库中，test_table表，在ds='20190314'分区下，value字段不满足数值类型的记录数。

配置： 首先选择字段数值类型检测模版。 选择希望检测的集群，库名，表名，字段，并填入分区过滤条件。如下图所示：



可以在SQL预览中查看执行的SQL是否和预期一致。

枚举值检测

语义： 指定一个表中一个字段，找出该字段不在所给枚举值中的记录条数。

举例： 需求：希望找出Cluster0002集群中，allenzhou_ind库中，test_table表，在ds='20190314'分区下，value字段不在“1，2，3，4”中的记录数。

配置： 首先选择枚举值检测模版。 选择希望检测的集群，库名，表名，字段，并填入分区过滤条件和枚举值(逗号分隔)。如下图所示：

可以在SQL预览中查看执行的SQL是否和预期一致。

数值范围检测

语义： 指定一个表中一个字段，找出该字段不在所给数值范围中的记录条数。

举例： 需求：希望找出Cluster0002集群中，allenzhou_ind库中，test_table表，在ds='20190314'分区下，key字段不在[0-5]的记录数。

配置： 首先选择数值范围检测模版。 选择希望检测的集群，库名，表名，并填入分区过滤条件和数值范围。如下图所示：

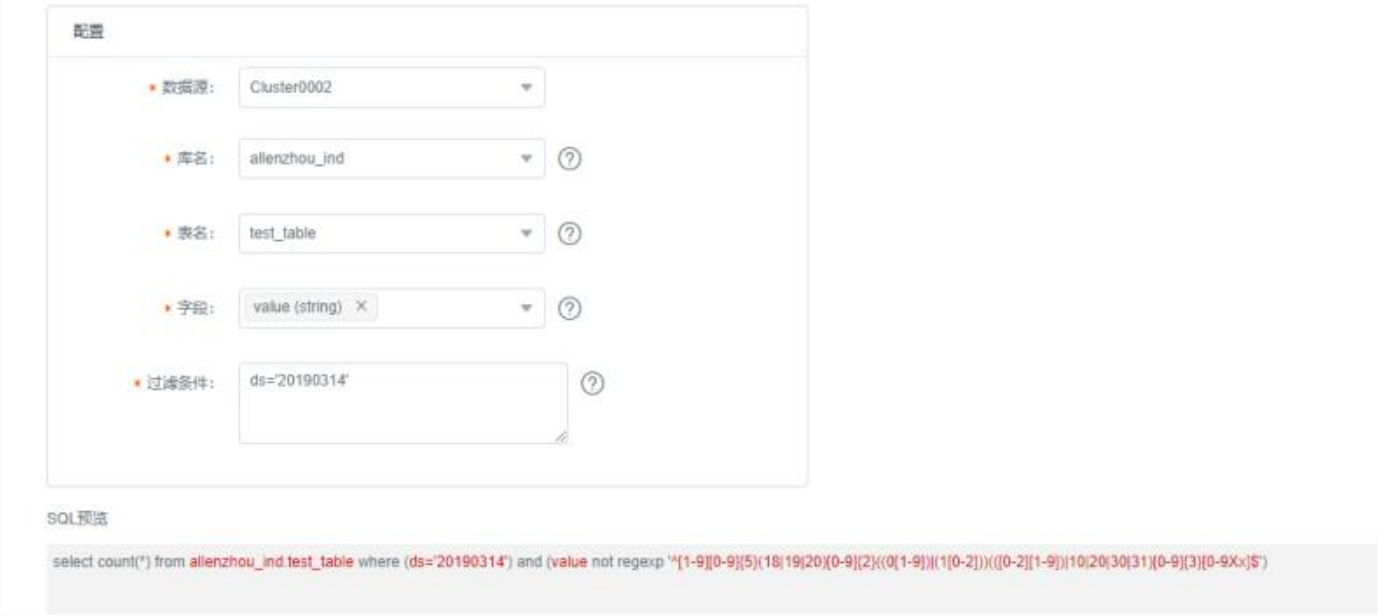
可以在SQL预览中查看执行的SQL是否和预期一致。

身份证校验

语义： 指定一个表中一个字段，找出该字段不符合身份证格式的记录条数(不能查询是否是真正的身份证，只能校验格式)。

举例： 需求：希望找出Cluster0002集群中，allenzhou_ind库中，test_table表，在ds='20190314'分区下，value字段不符合身份证格式记录数。

配置： 首先选择身份证检测模版。 选择希望检测的集群，库名，表名，字段，并填入分区过滤条件。如下图所示：



可以在SQL预览中查看执行的SQL是否和预期一致。

逻辑类校验

语义： 指定一个表，前置条件和后置条件，找到该表中满足该前置条件，但不满足后置条件的记录条数。

举例： 需求：希望找出Cluster0002集群中，allenzhou_ind库中，test_table表，在ds='20190314'分区下，若key<10(前置条件), 找到不满足value>100(后置条件)的记录条数。
配置： 首先选择逻辑类检测模版。 选择希望检测的集群，库名，表名，字段，并填入分区过滤条件，前置条件和后置条件。如下图所示：

可以在SQL预览中查看执行的SQL是否和预期一致。

空字符串检测

语义： 指定一个表中一个字段，找出该字段为空字符串的记录条数。
举例： 需求：希望找出Cluster0002集群中，allenzhou_ind库中，test_table表，在ds='20190314'分区下，value字段为空字符串的记录数。
配置： 首先选择空字符串检测模版。 选择希望检测的集群，库名，表名，字段，并填入分区过滤条件。如下图所示：

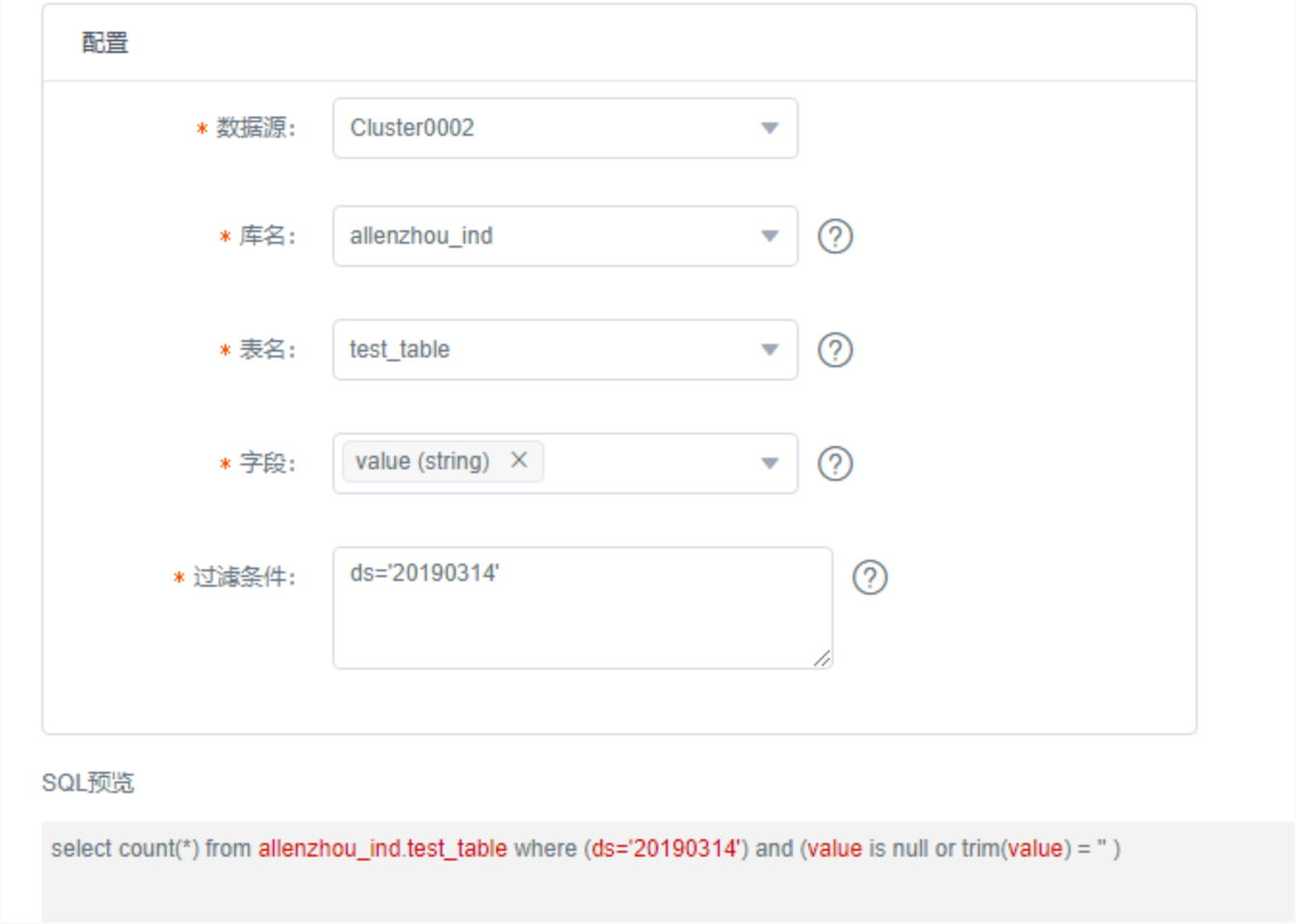
可以在SQL预览中查看执行的SQL是否和预期一致。

空值或空字符串检测

语义： 指定一个表中一个字段，找出该字段为空值或空字符串的记录条数。

举例： 需求：希望找出Cluster0002集群中，allenzhou_ind库中，test_table表，在ds='20190314'分区下，value字段为空值或空字符串的记录数。

配置： 首先选择空值或空字符串检测模版。 选择希望检测的集群，库名，表名，字段，并填入分区过滤条件。如下图所示：

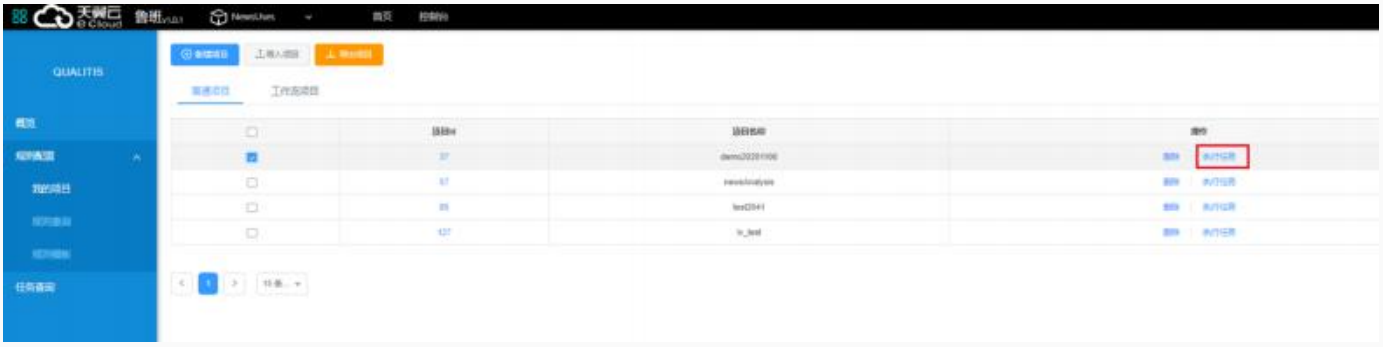


最后在SQL预览中查看执行的SQL是否和预期一致。

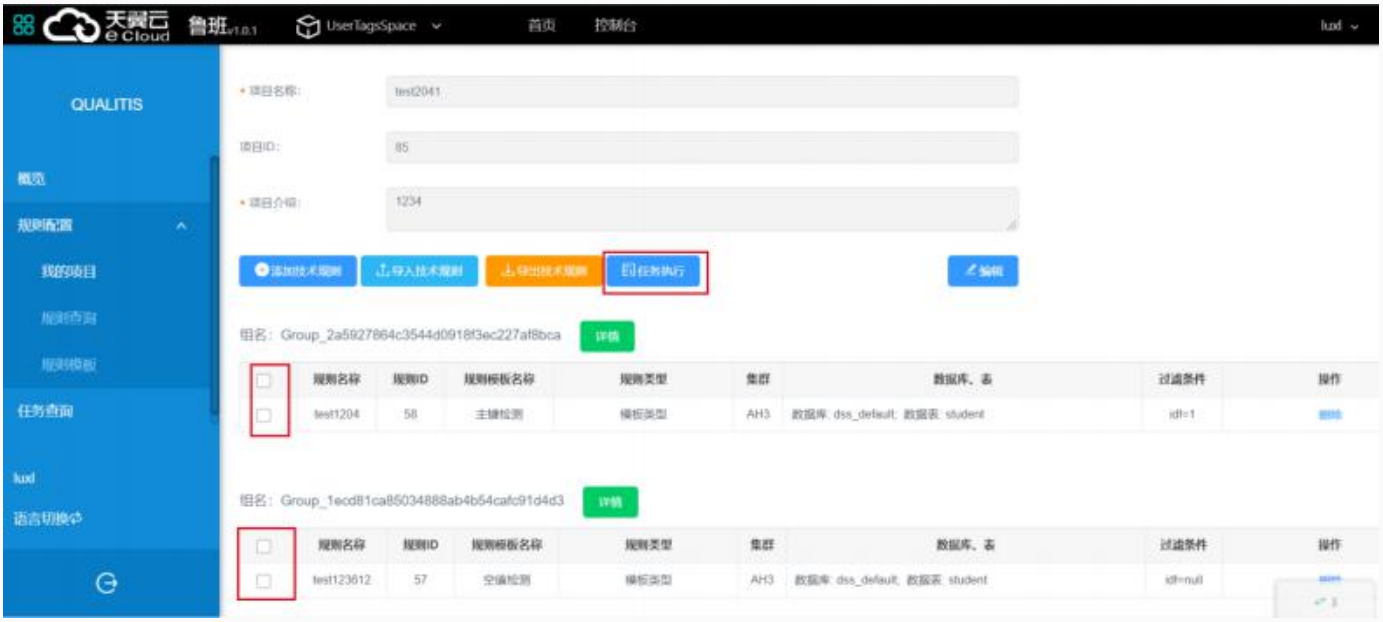
任务执行

任务可以从两个维度执行，项目维度和规则维度。

项目维度进行任务执行，会将项目下所有规则都提交执行。执行方式如下图所示：

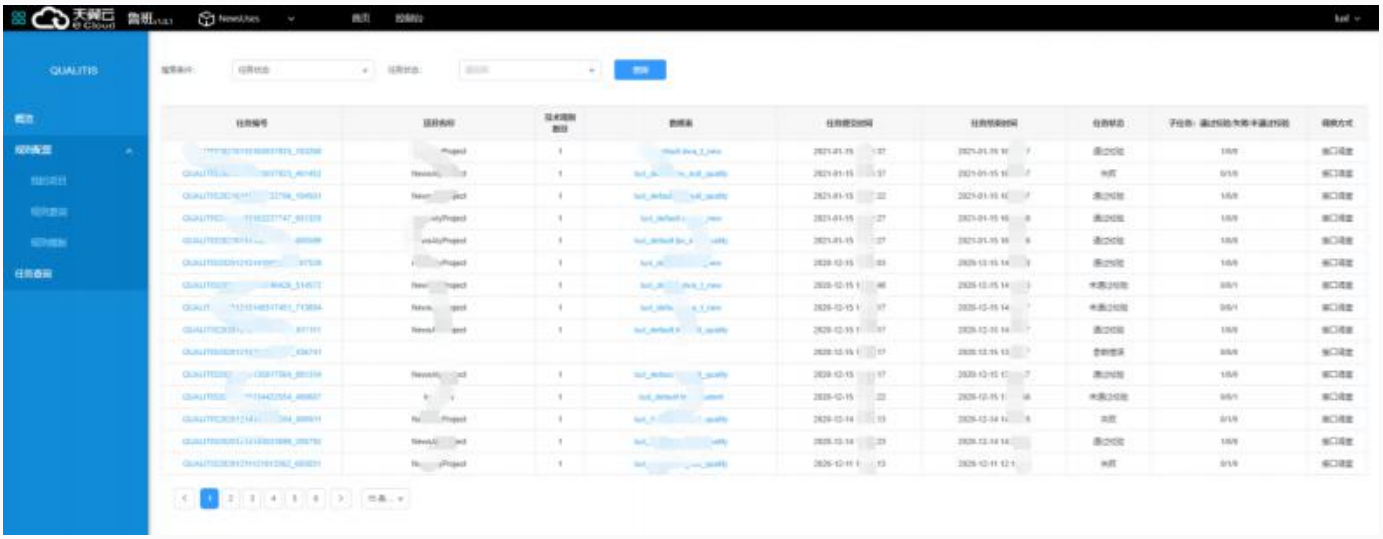


规则维度则是勾选任务执行。执行方式如下图所示：



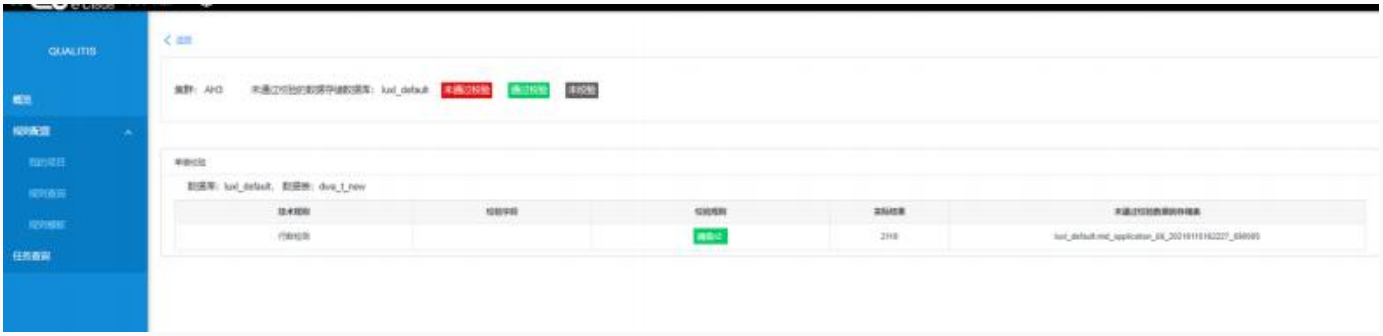
任务查看

点击左侧菜单栏中**任务查询**，即可进入任务界面。



提交的任务可以在任务界面中查找到，点击**某一任务**即可进入任务详情界面。

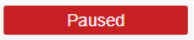
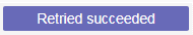
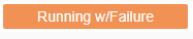
任务的校验结果可以点击**任务状态**进行查看。

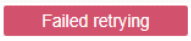
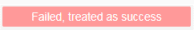
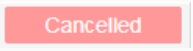


功能说明

Schedulis 是一个批量任务调度系统，其中的任务指的是大数据领域数据开发过程中的 ETL 任务，包括常规的 HadoopMR、 Hive、Spark、 Python大数据任务，还包括特色的 DataChecker（数据到达检查任务） 和 EventChecker（工作流之间的依赖任务） 的任务类型。
Schedulis 通过灵活的配置可以自由的组织工作流，可以支持工作流立即执行、定时调度、循环执行、同一工作流的并发执行和工作流的条件执行等多种任务调度方式。

作业流和任务的运行状态颜色说明

状态	状态说明	颜色示例	任务	子作业流	作业流
READY	准备状态1，默认的状态		○	○	○
PREPARING	准备状态2，表示作业流成功提交之后还没有执行的状态		×	×	○
RUNNING	表示任务在运行中的状态		○	○	○
PAUSED	表示暂停运行的状态		×	×	○
RETRIED_SUCCESS	表示因为执行失败后重试才执行成功的状态		○	×	×
SUCCEEDED	表示执行成功的状态		○	○	○
KILLING	表示任务被杀死中的状态		○	○	○
KILLED	表示任务已经杀死的状态		○	○	○
FAILED	表示任务执行失败的状态		○	○	○
FAILED_WAITING	表示任务已经执行失败在等待人工处理中的状态		○	×	×
FAILED_FINISHING	表示任务将要失败的状态		×	○	○

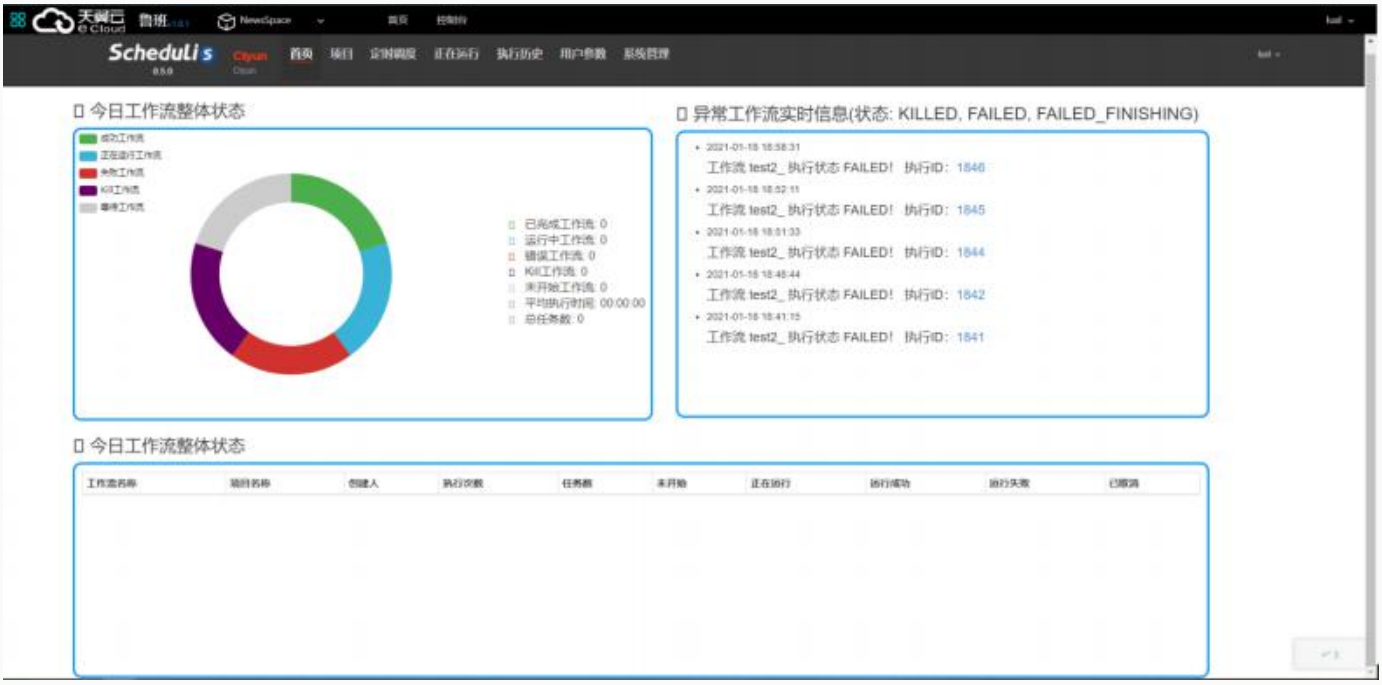
状态	状态说明	颜色示例	任务	子作业流	作业流
FAILED_RETRYING	表示任务已经失败准备重试的状态		○	–	–
SKIPPED	表示任务跳过执行的状态		○	○	–
FAILED_SKIPPED	表示任务执行失败后才会跳过执行的状态		○	–	–
DISABLED	表示任务关闭执行的状态（最终任务会由disabled变成skipped状态）		○	○	○
QUEUED	表示当前没有资源执行任务，任务在队列中等候分配资源的状态		○	×	×
FAILED_SUCCEEDED	表示任务是执行失败的，但是可以使用配置允许它是成功的状态		○	○	○
CANCELLED	表示任务取消执行的状态		○	○	×

注：○表示拥有此状态，×表示没有此状态

页面功能简介

说明：请在 Chrome 浏览器中操作使用 Schedulis 系统。

首页



进入某个工作空间，点击 **生产运维 > 进入Schedulis** 即可进入首页，首页包含三块信息：今日工作流整体状态（统计图表）、今日工作流整体状态（详情表格）和异常工作流实时信息。

统计图表支持动态式与用户交互：用户将鼠标光标放到其中一块工作流状态区域，该统计图表可以动态地为用户显示相应工作流状态的数目和占所有工作流的百分比。

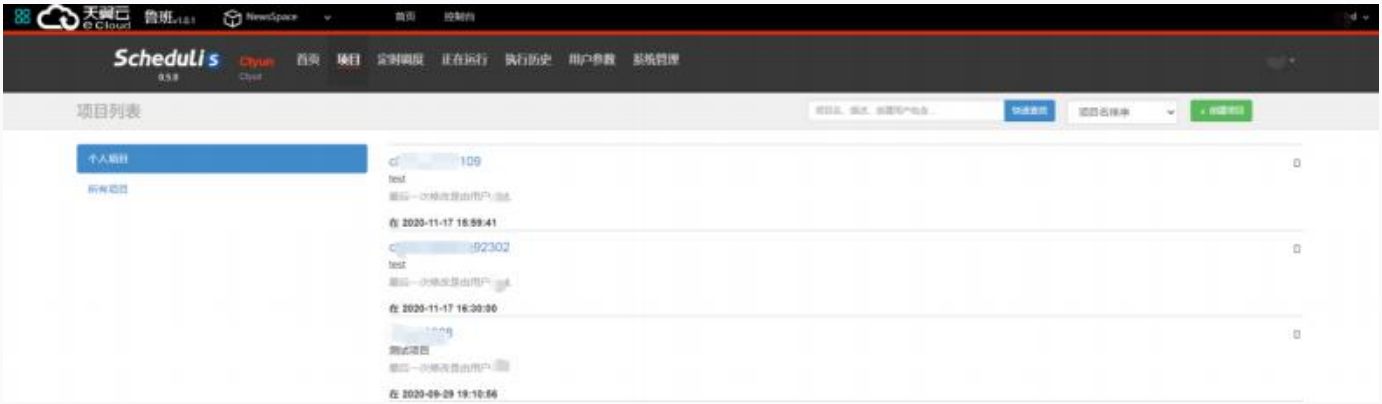
详情表格为用户列出了工作流的今日状态基本信息，譬如：运行成功数目和运行失败数目等。

异常工作流实时信息为用户列出了一些工作流的非手动异常信息，包括：KILLED，FAILED 和 FAILED_FINISHING

项目页

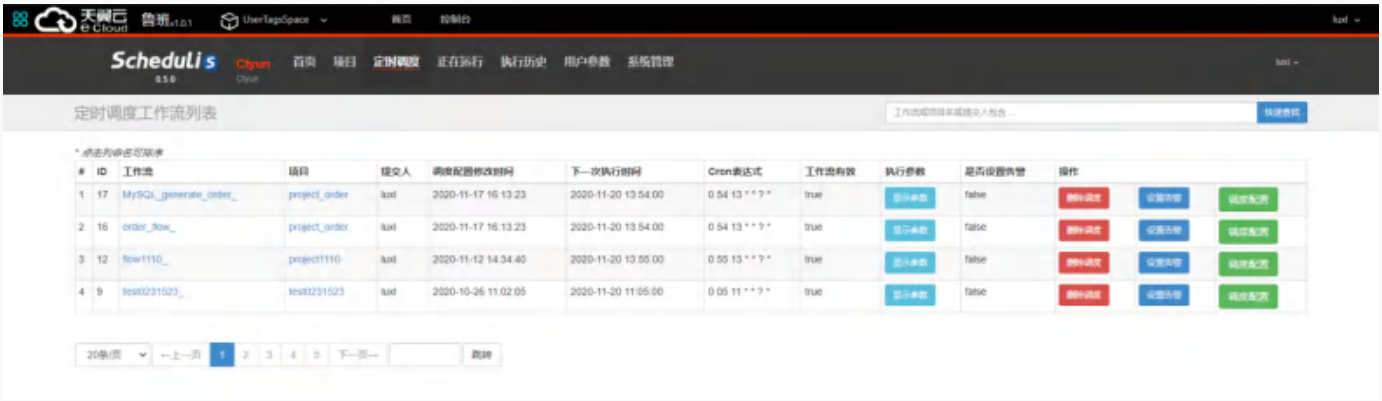
项目由工作流集合组成，对项目的操作其实就是对其底下工作流的操作。

项目页包含快速查找（根据项目名，描述，创建用户包含）、项目名排序、新建项目和项目列表四个部分。项目列表又分为个人项目列表和所有项目列表。



定时调度页

定时调度页包含定时调度工作流列表，列表中展示的信息包含工作流 ID、工作流名称、所属项目、提交人、调度配置修改时间、下一次执行时间、Cron 表达式、工作流是否有效、是否设置告警等信息。提供快速查找（根据工作流名称、所属项目名称或提交人）、按列排序、显示参数、删除调度、设置告警和调度配置等操作。

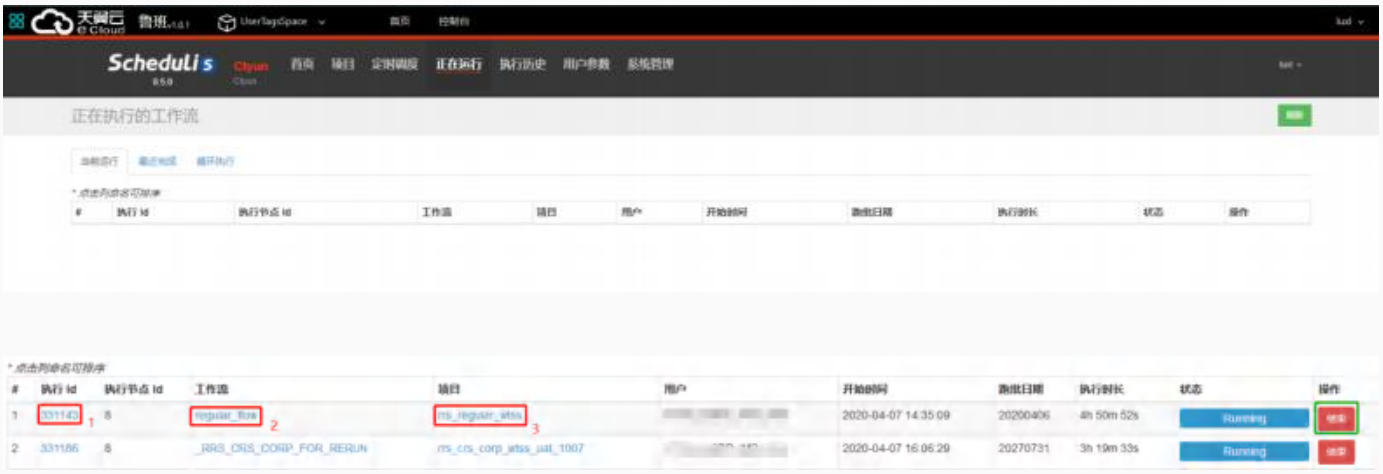


说明：

- 显示参数：以 JSON 字符串方式显示工作流设定的定时调度参数
- 设置告警选项：用来对定时调度工作流的子工作流/任务进行超时告警或者事件告警规则设置。
- 调度配置选项：用来对工作流的定时调度参数重新配置。

正在运行页

正在运行页包含三个子页面，分别是当前运行工作流列表、最近完成工作流列表和循环执行工作流列表，并且对正在执行的工作流状态支持实时刷新功能。



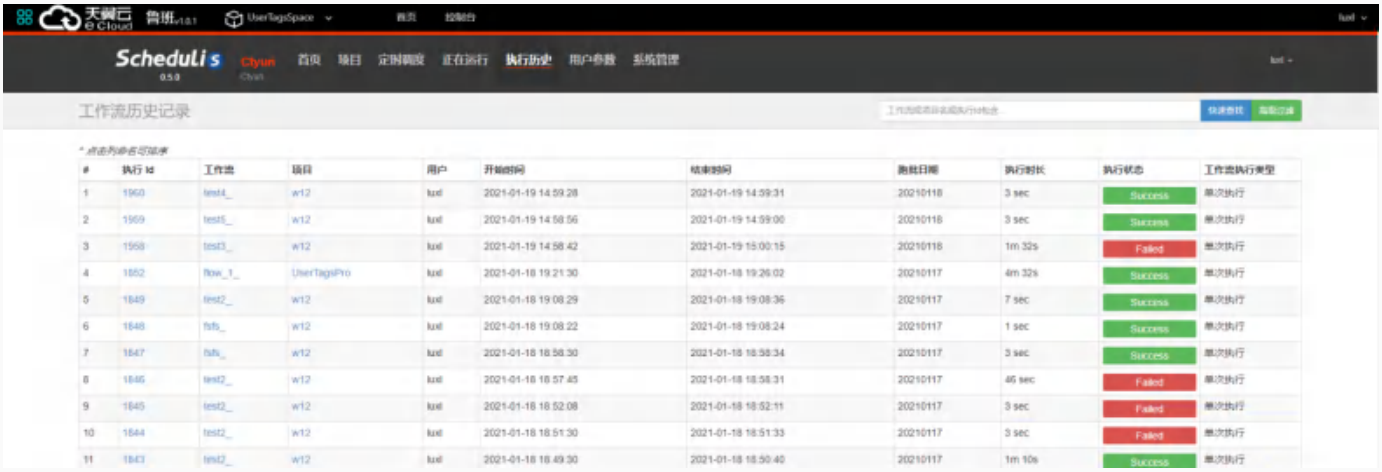
当前运行工作流列表的每一行都列出了当前正在单次运行的工作流运行状态以及相应的基本信息，具体请参考上图。

当前运行工作流列表支持按列排序和手动结束正在运行的工作流。

1. 通过点击执行 Id 可链接到工作流视图，工作流的任务列表，工作流日志以及运行参数。
2. 通过点击工作流名称可链接到工作流视图，工作流执行历史以及执行摘要。
3. 通过点击项目名称可链接到工作流归属的项目视图。

执行历史页

执行历史页中列出了所有工作流的执行历史记录。支持对列表进行按列排序、快速查找（根据执行 Id ，工作流名称或者项目名称）、通过输入更详细的信息进行高级过滤等操作。



工作流执行历史记录列表每一行都列出了工作流的运行状态，执行类型以及相应的基本信息。

用户参数页

用户参数页包含用户参数列表，支持新增用户参数、修改已有用户参数操作。

此功能模块实现了一个用户级别的静态全局变量，避免了用户每次修改参数时都得手动更改项目文件脚本，再次上传项目文件和发版。此变量只有变量拥有者可以编辑，变量拥有者可以分享变量给其他用户使用(只读)。

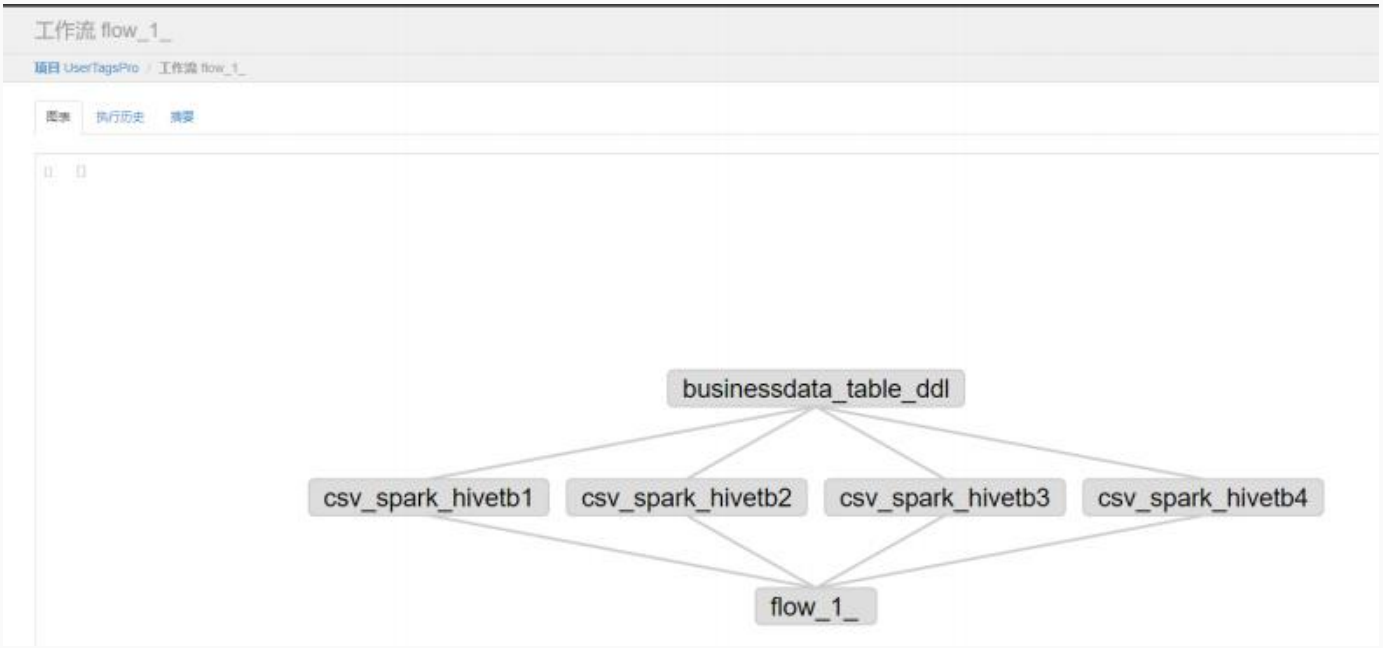
系统管理页

系统管理页需要管理员权限才可以使用。系统管理页包含用户管理、部门管理、资源分组管理这三个子页面，提供快速查找（根据用户名包含，部门名包含）、新增用户、修改已有用户、新增部门、修改已有部门、新增资源分组、修改已有资源分组等功能。

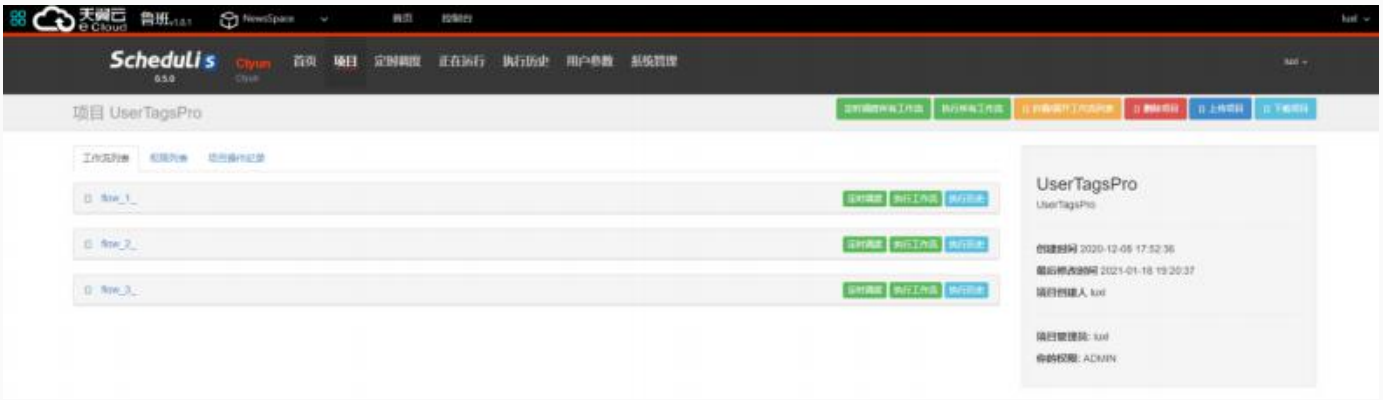
项目视图

工作流发布后,可进入项目,找到对应工程进入详情页面，分为工作流列表、权限列表、项目操作记录三个子页。

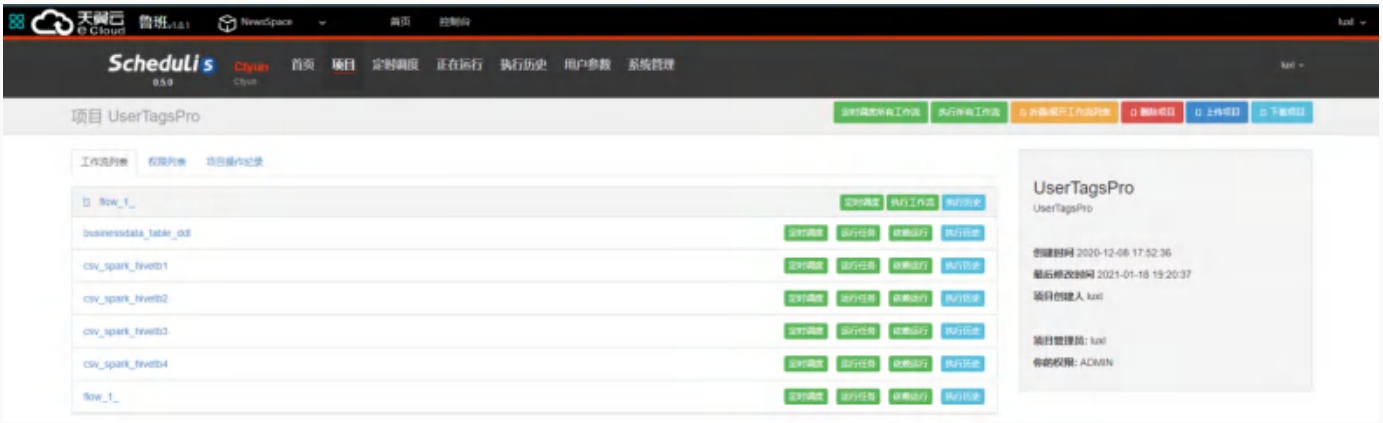
一个工作流是子工作流和任务的集合。这些任务和子工作流可以相互关联组成 DAG (Directed Acyclic Graph)，其中 DAG 末端节点的名称即为该工作流的名称。



工作流列表



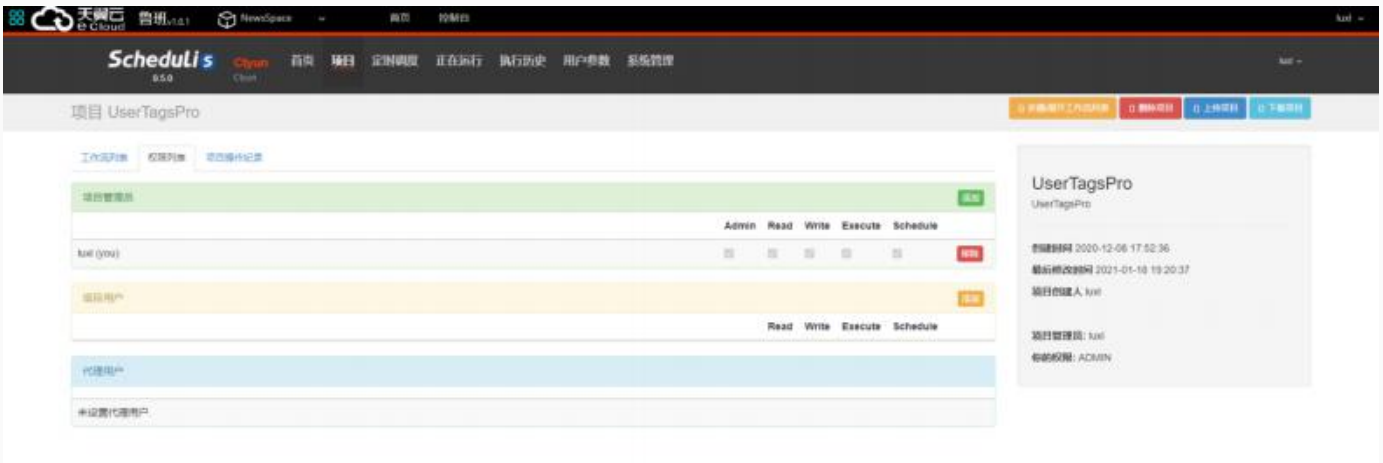
可展开工作流下的所有任务列表



对于每个工作流，可以对之执行定时调度、立即执行和查看执行历史操作。对于工作流下面的任务列表，可以对每一个任务执行定时调度、立即执行、依赖运行和查看执行历史操作。

权限列表

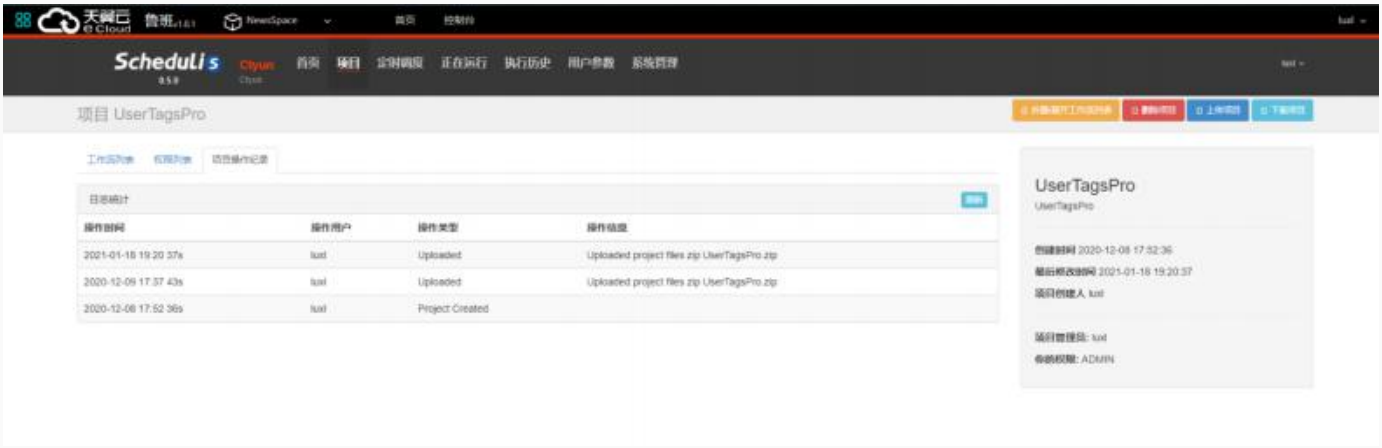
对于项目，Schedulis 包含3个角色和5种权限。3个角色分别为：项目管理员、项目用户和代理用户。5个权限分别为：管理权限、读权限、写权限、执行权限和调度权限。



- 如果是项目管理员，则拥有操作该项目的所有权限(管理，读，写，执行，调度)。
- 项目管理员可以为项目添加项目用户，并为之分配权限，同时可以修改已有项目用户的权限，删除已有项目用户。
- 项目管理员还可以添加新的项目管理员和移除已有的项目管理员。

权限列表的中的代理用户非常重要，在执行任务的时候需要用到代理用户，即后台实际执行任务的用户 。（注意：代理用户配置之后需要重新登录之后才能生效）

项目操作记录

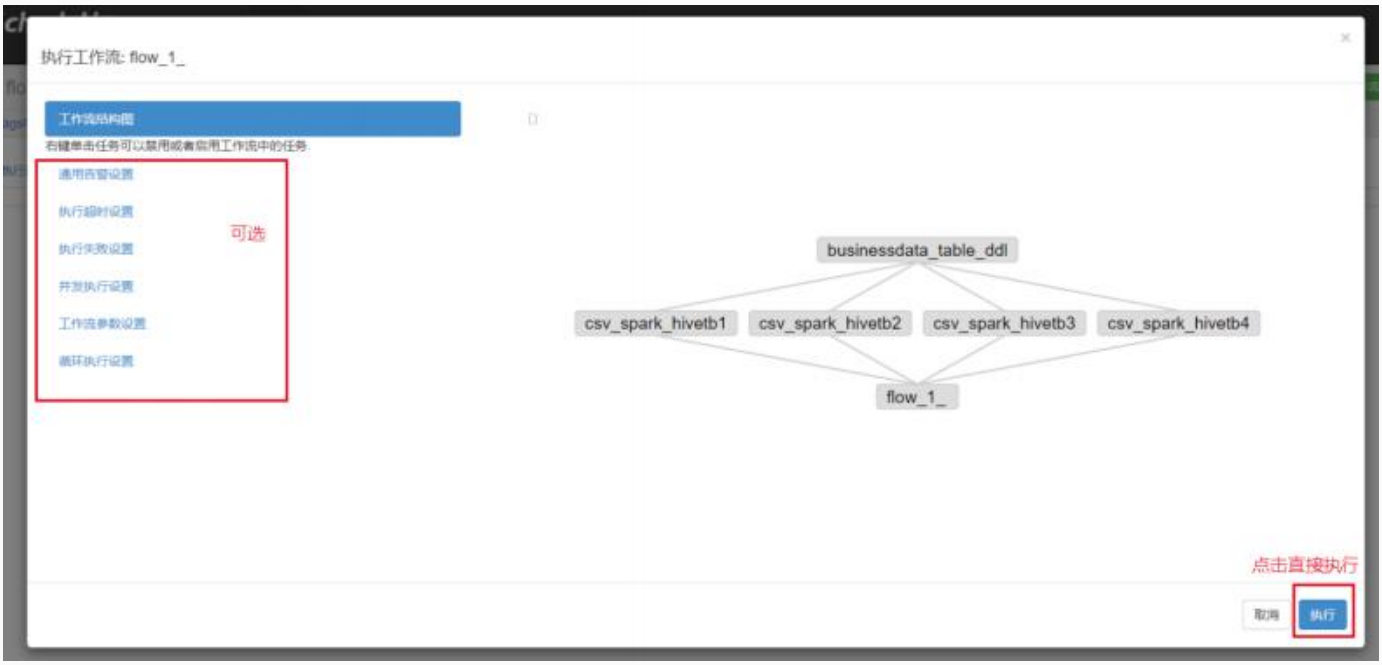


该页面列出了该项目的所有操作记录，并支持实时刷新项目操作。

执行工作流

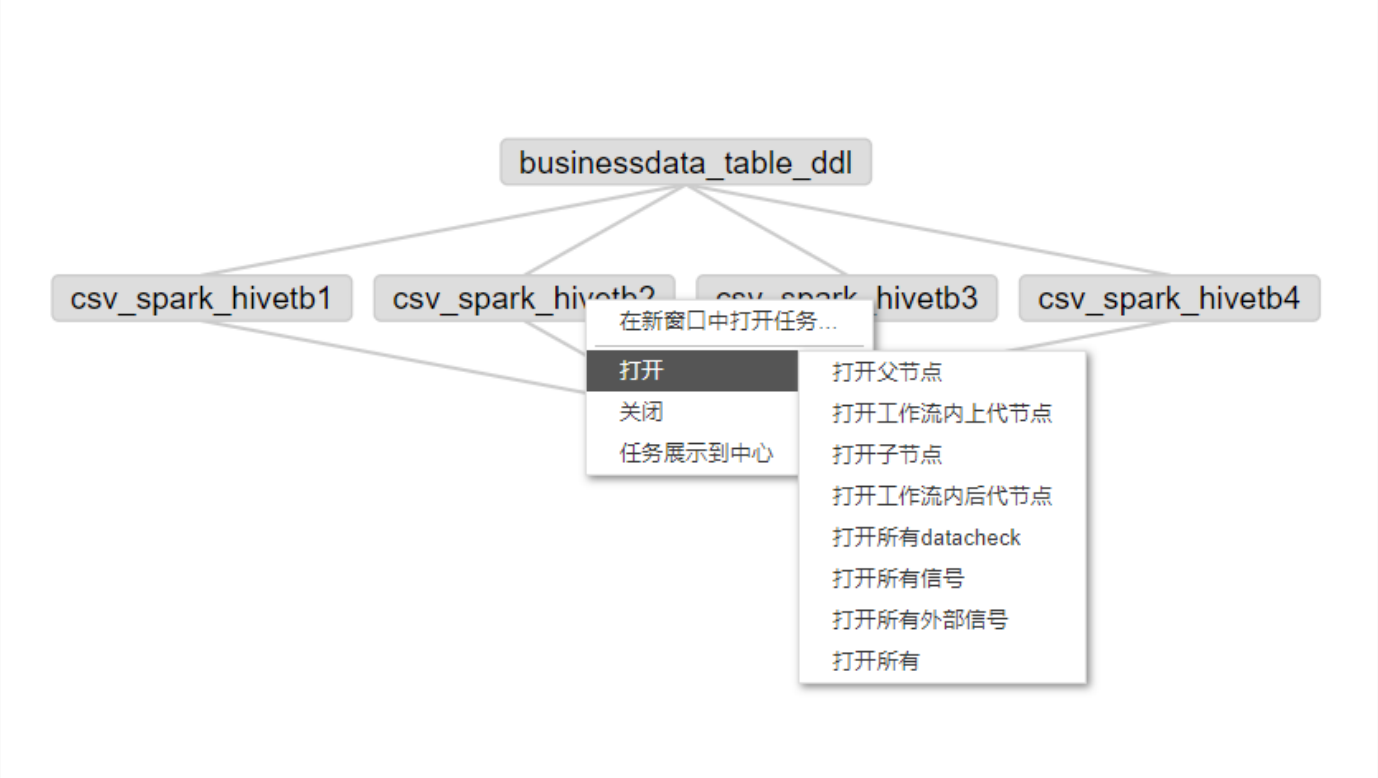
在执行工作流的视窗中可以查看工作流结构图，对工作流进行通用告警设置、执行超时设置、执行失败设置、并发执行设置、工作流参数设置和循环执行等操作，然后点击执行按钮，触发工作流立即执行；也可选择使用系统默认参数，直接点击执行按钮，使工作流一键执行。

工作流结构图



用户可以选择手动对工作流进行设置，或者使用系统默认设置。

对于工作流结构图中的每个节点，用户可以通过右键点击选定的节点，手动地对其进行一系列操作：



通用告警设置

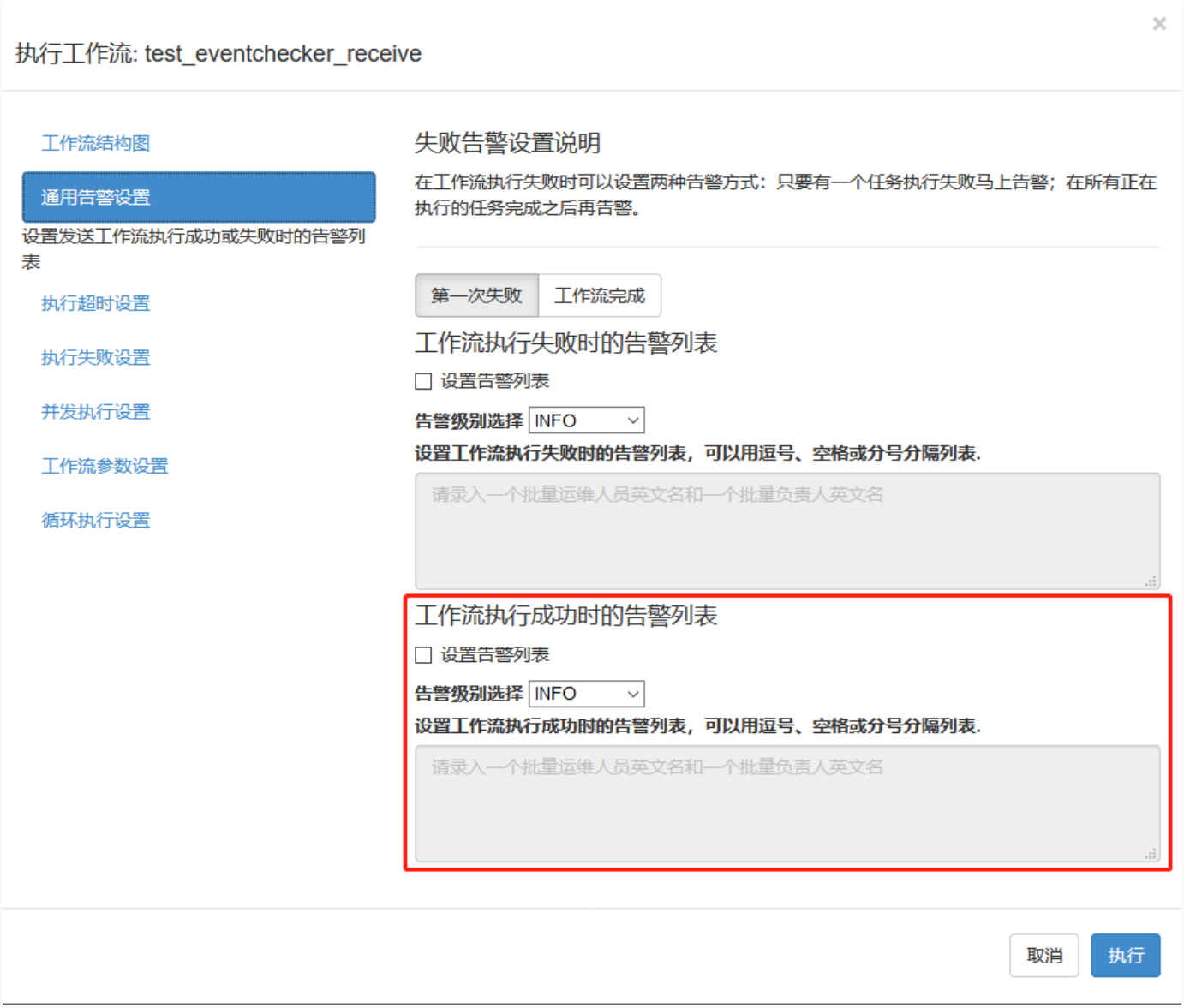
通用告警设置可以分为两种：工作流执行失败告警设置和工作流执行成功告警设置。

工作流执行失败告警

工作流执行失败告警可以设置的告警策略，目前有两种：一种是在工作流中某一个任务失败的时候立即触发告警；另一种是在所有正在执行的工作流完成之后再触发告警。可以自由选择告警策略，设置告警级别，填写告警通知的用户列表。系统会根据告警的级别以邮件、通讯消息、电话等多种方式通知用户。

工作流执行成功告警

工作流执行成功时可以自由选择是否告警，设置告警级别，填写告警通知的用户列表。具体告警级别相对应的告警方式，请参考附录 7.2 告警级别与对应通知方式参照表


执行工作流: test_eventchecker_receive
工作流结构图
通用告警设置
设置发送工作流执行成功或失败时的告警列表
执行超时设置
执行失败设置
并发执行设置
工作流参数设置
循环执行设置
失败告警设置说明
在工作流执行失败时可以设置两种告警方式：只要有一个任务执行失败马上告警；在所有正在执行的任务完成之后再告警。
第一次失败
工作流完成
工作流执行失败时的告警列表
设置告警列表
告警级别选择 INFO
设置工作流执行失败时的告警列表，可以用逗号、空格或分号分隔列表.
请录入一个批量运维人员英文名和一个批量负责人英文名
工作流执行成功时的告警列表
设置告警列表
告警级别选择 INFO
设置工作流执行成功时的告警列表，可以用逗号、空格或分号分隔列表.
请录入一个批量运维人员英文名和一个批量负责人英文名
取消
执行

执行超时设置

单次执行工作流支持设置超时告警，用户可以自由选择当工作流没有在指定的执行时间内完成，是否选择触发超时告警。 告警规则分为：SUCCESS 和 FINISH，即工作流是否在指定时间内成功或者结束。用户指定超时时间，告警级别，根据需求选择发送超时告警邮件（填上告警邮件地址）或者终止工作流。 超时时间的格式是 HH:MM，单位是小时和分钟，例如18分钟任务还未完成触发超时告警，时间格式为00:18。

执行失败设置

可以在该页面设置一些工作流执行失败时的策略：完成当前正在运行的任务、结束正在执行的任务和完成所有可以执行的任务。 另外，用户还可以对一些指定的任务（可以是全部任务列表）或者子工作流，进行失败重跑设置和失败跳过设置。 当系统已经完成了用户设置的失败重跑以及失败跳过操作后，工作流仍然处于执行失败状态的，系统就会根据用户选择的三个失败策略中的一个，进行相应的后续操作。

并发执行配置

可以在该页面配置策略来处理重新提交的同名工作流，在上一次提交的工作流未运行完成时，对于新一次提交的同名工作流，并发执行策略包含跳过执行、同时运行和设置管道，管道还可以分为 Level 1 和 Level 2 等级。

除了按照系统默认的工作流并发度来运行工作流之外，用户还可以按照自己的设计来控制工作流中的 Job 并发数目。可以在工作流参数设置里面，添加参数 flow.num.job.threads 。例如设置10个并发作业度：

工作流参数配置

在该页面可以设置工作流的临时全局参数，该参数会覆盖项目包中 properties 参数配置文件中的 KEY 值相同的参数。

参数配置优先级：工作流动态全局变量 > 系统临时全局参数（本次需求） > properties 参数配置文件 > 用户级别静态全局变量 > 系统默认值

循环执行设置

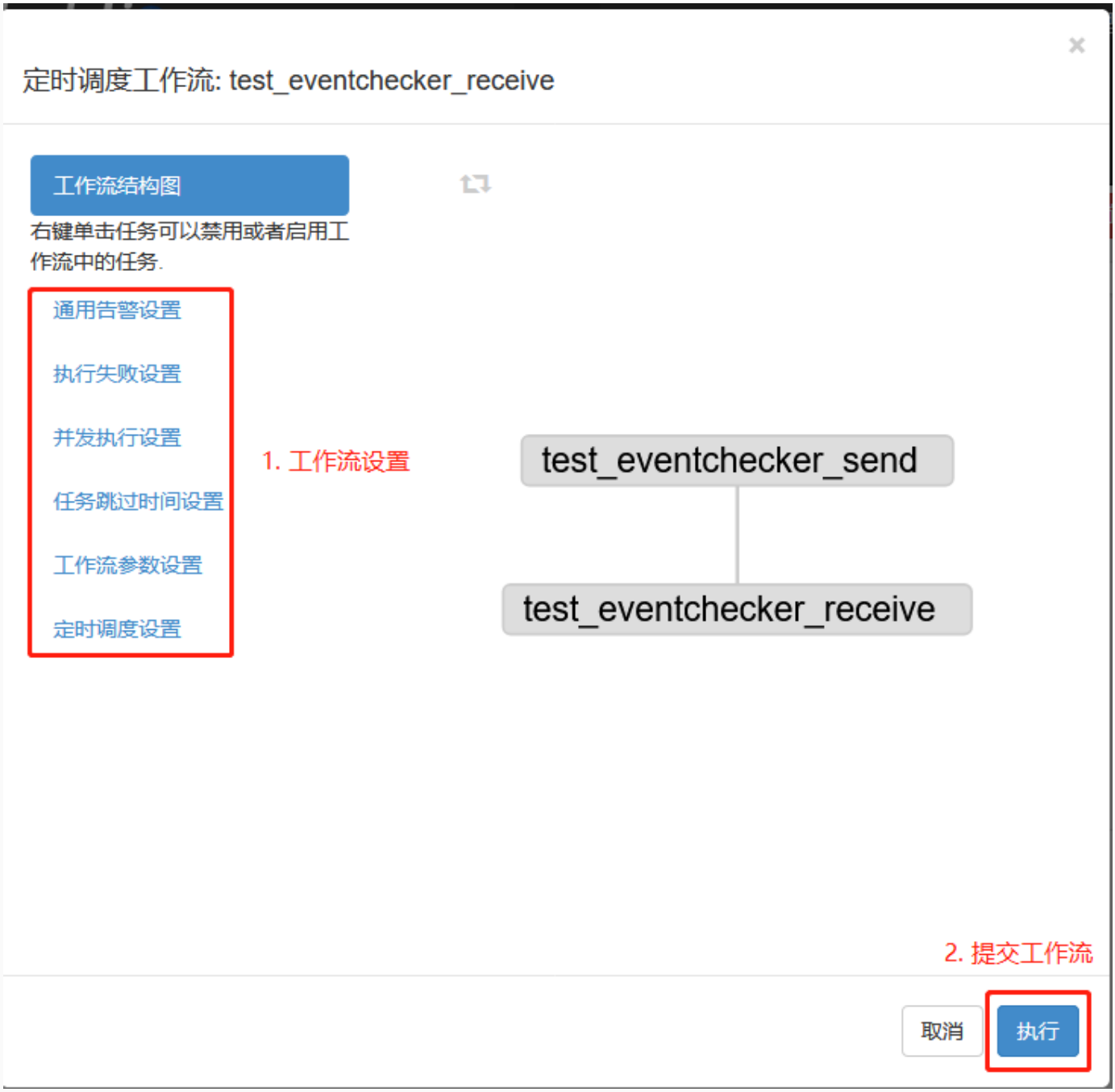
可以在一次工作流结束后，立即开始下一次的执行，并且需要带上上一次执行的相关参数。

该功能主要适用于这样的场景：业务团队在开发工作流时有需要在一次执行结束之后立即进行下一次任务，而每一次任务的执行时长是不确定的，没办法设置固定时间拉起下一次任务。
用户可以对循环执行错误时选择失败策略：终止循环执行流程或者继续执行循环执行流程。
此外，用户可以在页面上手动终止循环执行的工作流。当循环执行中断时，可以选择告警级别以及告警人列表。

定时调度工作流

在定时调度工作流的视窗中可以查看工作流结构图、对工作流进行通用告警设置、执行失败设置、并发执行设置、任务跳过时间设置、工作流参数设置和定时调度等操作，点击执行调度按钮，触发调度任务的提交；也可选择使用系统默认参数，直接点击执行按钮，使工作流一键调度。

工作流结构图



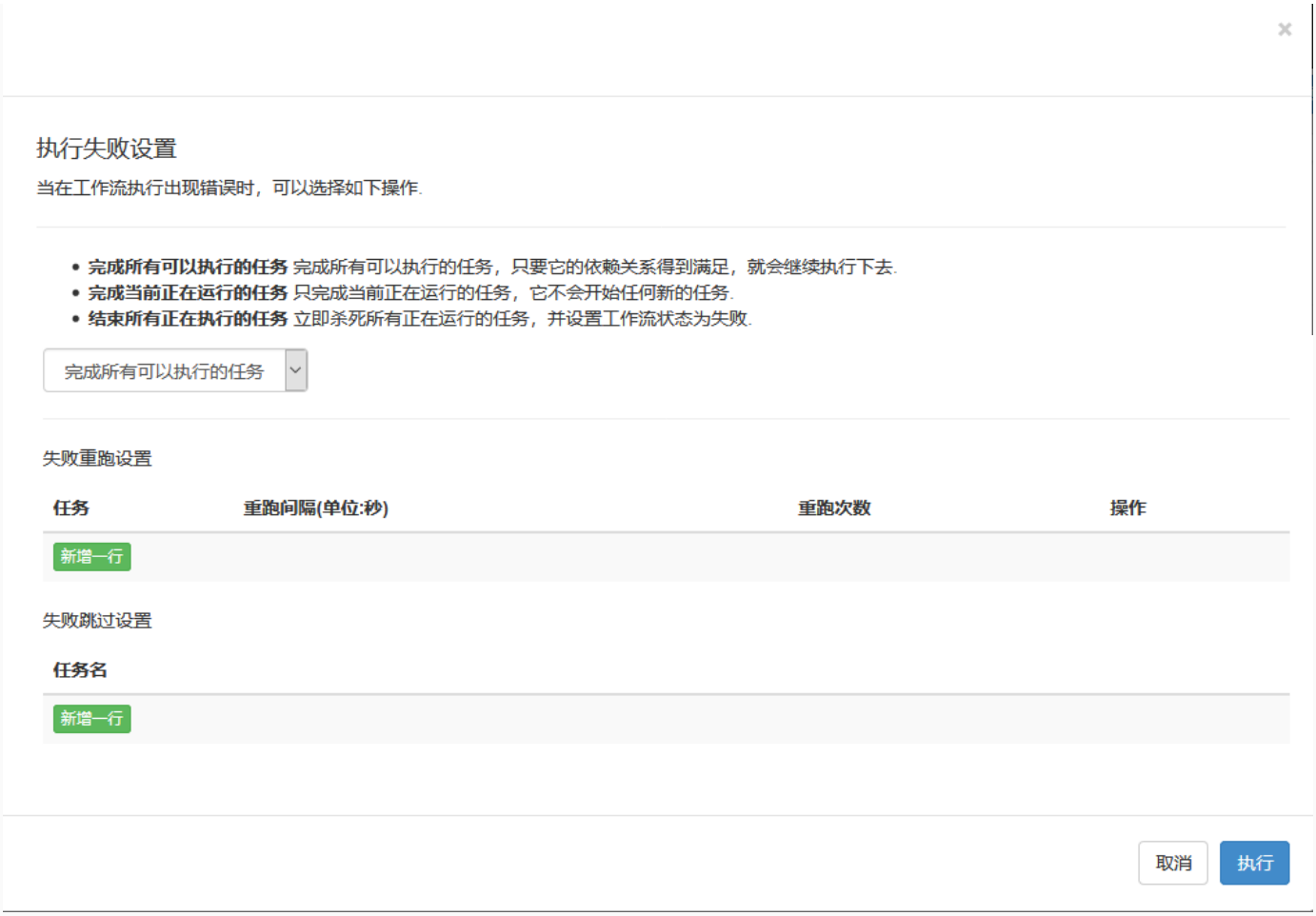
通用告警设置

通用告警设置可以分为两种：工作流执行失败告警设置和工作流执行成功告警设置。

工作流执行失败告警

工作流执行成功告警

执行失败设置



并发执行设置

任务跳过时间设置

在定时调度的工作流中，为一些 Job 提供 Cron 表达式，如果定时调度工作流的提交时间和指定 Job 的 Cron 时间一致（只精确到年月日），该 Job 就会被跳过执行。

- 适用场景：

适用于某些场景下，用户需要定时调度作业流中的某些 Job 在指定日期不执行。

- 使用说明：

选择要跳过执行的 Job，输入正确的 Cron 表达式（只精确到年月日），当 Cron 表达式输入正确后，页面会展示 TOP 10 时间，否则表示 Cron 表达式输入有误。

工作流参数设置

以上内容详情参见上一节；

定时调度设置

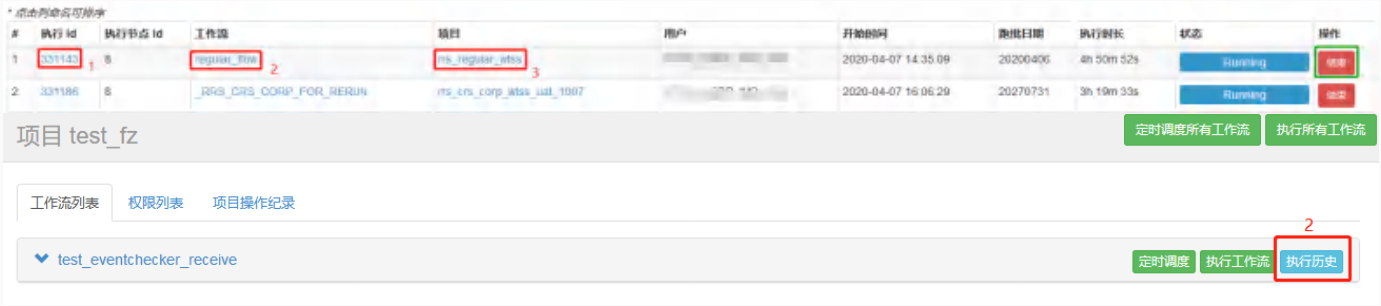
定时调度以 Asia/Shanghai 为参考时区，以 Quartz 的 Cron 表达式来设置定时任务的调度时间。当 Cron 表达式输入正确后，页面会展示 TOP 10 时间，否则表示 Cron 表达式输入有误。

Cron 表达式的使用说明可以参考如下链接：

http://www.quartz-scheduler.org/documentation/quartz-2.3.0/tutorials/tutorial-lesson-06.html

任务详情查看

在导航栏的定时调度功能模块中可以查询所有的定时调度工作流； 在导航栏的正在运行功能模块中可以查询：正在运行的工作流、最近完成的单次运行的工作流； 在导航栏的执行历史功能模块中可以查询所有的工作流执行状态。

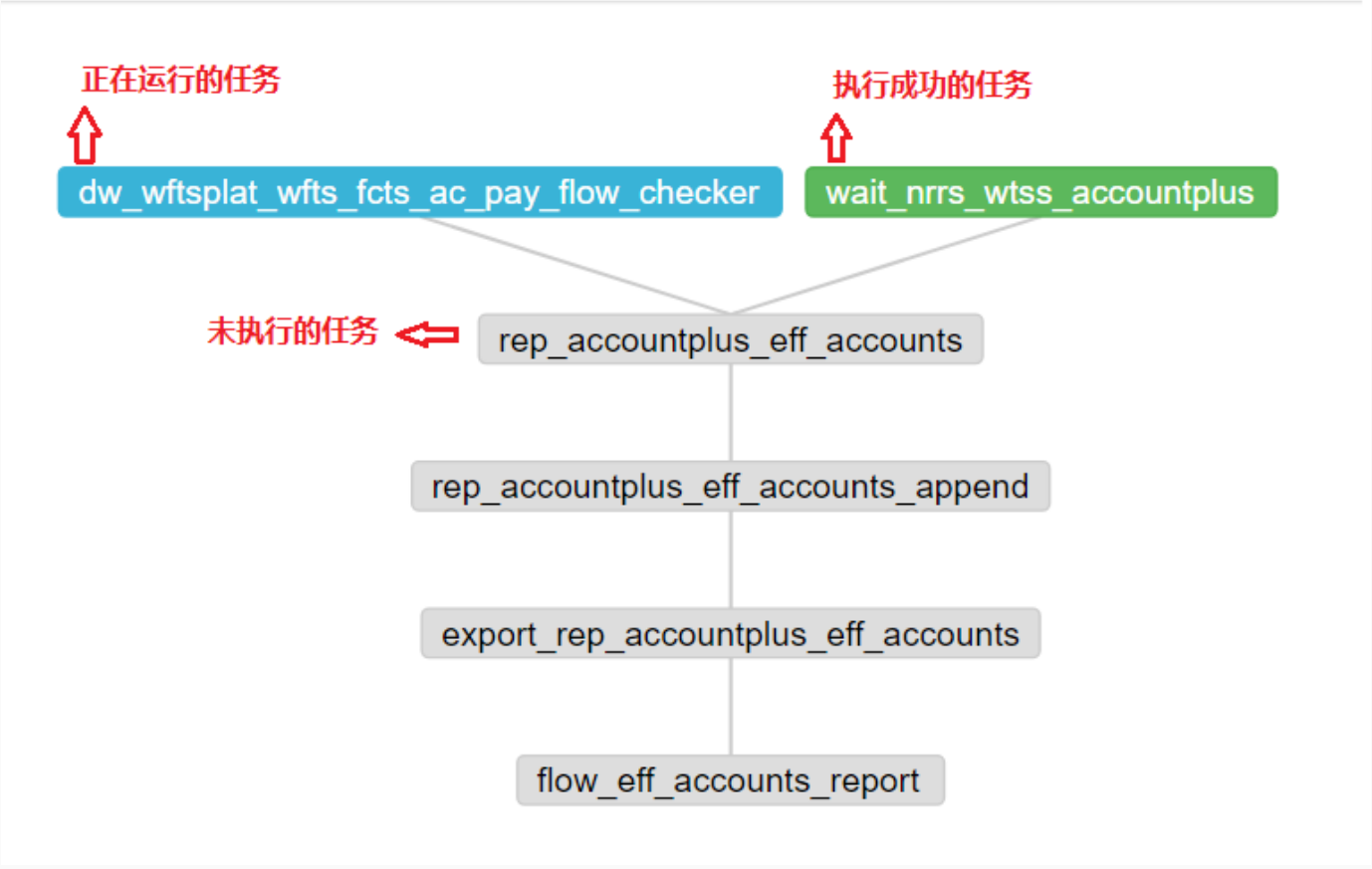


在这些页面中，用户都可以通过一下操作查看相应的信息：

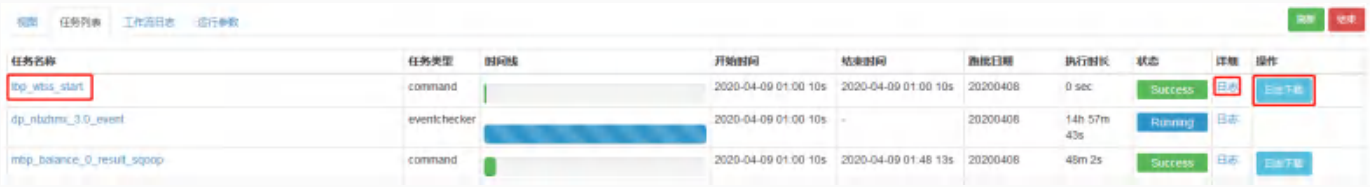
1. 通过点击执行 Id 可链接到工作流视图，工作流的任务列表，工作流日志以及运行参数。
2. 通过点击工作流名称或者项目视图中的执行历史选项可链接到工作流视图，工作流执行历史以及执行摘要。
3. 通过点击项目名称可链接到工作流归属的项目视图。

工作流视图

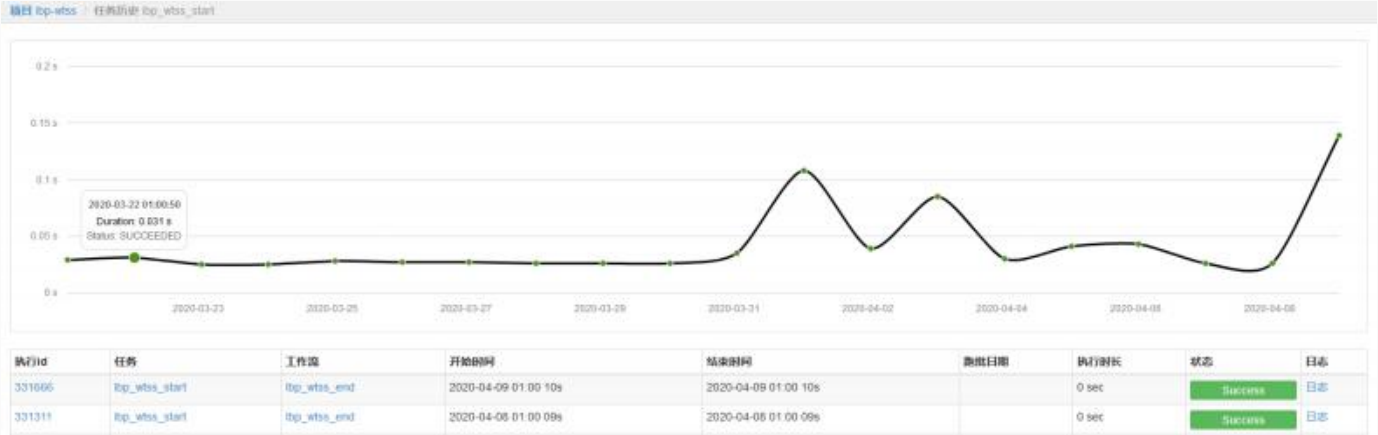
工作流视图可以用来表现整体工作流的执行情况，灰色的表示未执行的任务、绿色的表示执行成功的任务、蓝色表示正在执行的任务、红色表示执行失败的任务。



任务列表



任务列表页面中列出了已经执行结束的任务（包括成功或者失败）和正在执行的任务列表。可以查看单个任务的执行详细信息，譬如执行时长，也可以查看或下载单个任务的执行日志。 也可以通过点击任务名称，查看以图表和表格的形式显示的单个任务运行状况。



工作流日志

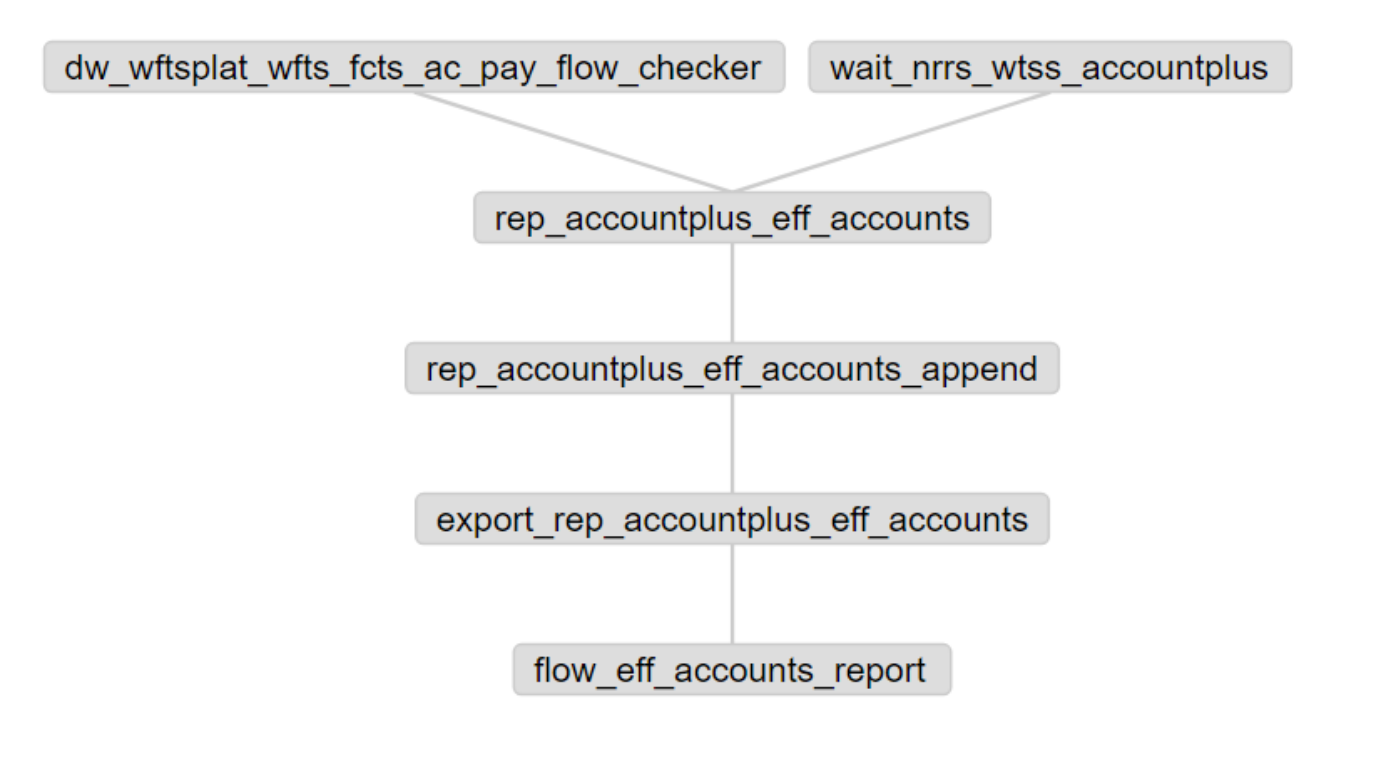
在工作流日志页面可以查工作流的执行日志，对于正在运行的任务也可以实时刷新该工作流的执行日志或者手动结束该工作流。

工作流运行参数

用户可以通过这个页面，查看工作流的临时系统参数以及工作流运行时输出的动态全局变量。

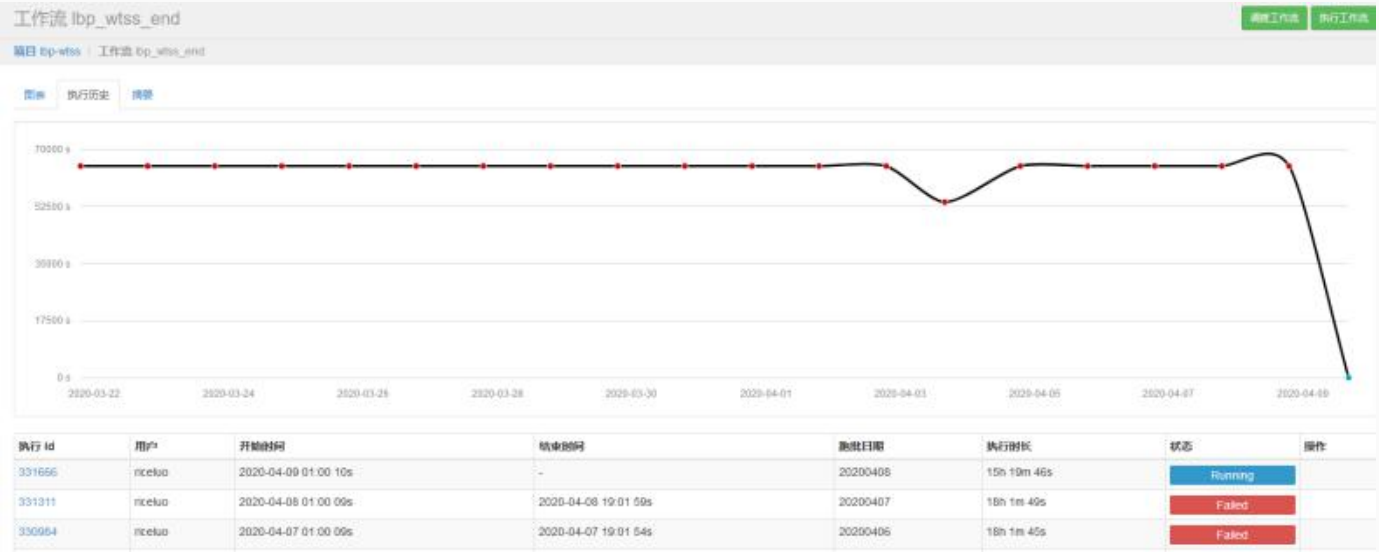
工作流图表

工作流图表和工作流视图类似，都是工作流的 DAG 图，和工作流视图不同的是，工作流图表只描述了任务之间的依赖关系，没有任务的执行状态。



执行历史

执行历史页面，以图表和表格的形式，显示出了工作流的运行状况。



任务摘要

工作流摘要页面列出了工作流的摘要信息。

项目 account_plus / 工作流 flow_eff_accounts_report

图表 执行历史 摘要

Project name	account_plus
Job Types Used	eventchecker datachecker command

Scheduling

Schedule ID	75	**Submitted By**	zuhengshi
First Scheduled to Run	2019-06-24 15:06:48	**Cron Expression**	0 * * ? * *
Next Execution Time	2019-07-18 20:35:00	**SLA**	false

Last Run Stats