UniCloud 数据管理平台

用户手册

紫光云技术有限公司
www.unicloud.com

资料版本：5W100-20211130
产品版本：UniCloud Data Management Platform (E6101)

前 言

本手册主要介绍了 UniCloud 数据管理平台（UniCloud Data Management Platform）的概述、访问方式、功能介绍、典型案例等内容。

前言部分包含如下内容：

- 读者对象
- 本书约定
- 资料意见反馈

读者对象

本手册主要适用于如下工程师：

- 网络规划人员
- 现场技术支持与维护人员
- 负责网络配置和维护的网络管理员

本书约定

1. 图形界面格式约定

格　式	意　义
< >	带尖括号“< >”表示按钮名，如“单击<确定>按钮”。
[ ]	带方括号“[ ]”表示窗口名、菜单名和数据表，如“弹出[新建用户]窗口”。
/	多级菜单用“/”隔开。如[文件/新建/文件夹]多级菜单表示[文件]菜单下的[新建]子菜单下的[文件夹]菜单项。

2. 各类标志

本书还采用各种醒目标志来表示在操作过程中应该特别注意的地方，这些标志的意义如下：

 警告	该标志后的注释需给予格外关注，不当的操作可能会对人身造成伤害。
 注意	提醒操作中应注意的事项，不当的操作可能会导致数据丢失或者设备损坏。
 提示	为确保设备配置成功或者正常工作而需要特别关注的操作或信息。
 说明	对操作内容的描述进行必要的补充和说明。
 窍门	配置、操作、或使用设备的技巧、小窍门。

3. 端口编号示例约定

本手册中出现的端口编号仅作示例，并不代表设备上实际具有此编号的端口，实际使用中请以设备上存在的端口编号为准。

资料意见反馈

如果您在使用过程中发现产品资料的任何问题，可以通过以下方式反馈：

E-mail：unicloud-ts@unicloud.com

感谢您的反馈，让我们做得更好！

目　录

1 概述 ····· 1-1

1.1 简介 ····· 1-1

1.2 产品架构 ····· 1-1

1.3 术语和定义 ····· 1-3

2 访问数字平台的数据管理平台服务 ····· 2-1

2.1 登录 ····· 2-1

2.2 首页 ····· 2-1

2.3 退出登录 ····· 2-2

3 功能介绍 ····· 3-1

3.1 数据源管理 ····· 3-1

3.2 标准管理 ····· 3-2

3.3 数据开发 ····· 3-4

3.4 数据资产 ····· 3-9

3.5 数据质量 ····· 3-10

3.6 数据脱敏 ····· 3-11

3.7 图引擎 ····· 3-13

3.8 时空引擎 ····· 3-14

3.9 配置管理 ····· 3-14

4 共享自行车案例 ····· 4-1

4.1 案例说明 ····· 4-1

4.2 准备操作 ····· 4-1

4.2.1 创建数据源 ····· 4-1

4.2.2 注册离线表 ····· 4-2

4.3 构建业务流程 ····· 4-3

4.3.1 创建业务流程 ····· 4-3

4.3.2 编辑 SparkSQL 节点 ····· 4-4

4.3.3 提交执行 ····· 4-6

4.3.4 任务监控 ····· 4-6

4.3.5 调度策略 ····· 4-7

4.3.6 调度监控 ····· 4-10

4.3.7 补录数据 ····· 4-13

4.4 结果查看 ····· 4-15

5 疫苗接种监控案例 ····· 5-1
5.1 案例说明 ····· 5-1
5.2 准备操作 ····· 5-1
5.2.1 抽取基础数据 ····· 5-1
5.2.2 创建数据源 ····· 5-2
5.2.3 新建数据表 ····· 5-4
5.3 构建业务流程 ····· 5-11
5.3.1 创建业务流程 ····· 5-11
5.3.2 添加数据清洗作业 ····· 5-12
5.3.3 添加数据计算作业 ····· 5-13
5.3.4 构建完成作业并运行 ····· 5-21
5.4 数据查询 ····· 5-22
5.5 结果数据发布 ····· 5-23
5.6 数据最终呈现 ····· 5-23
6 常见问题解答 ····· 6-1
7 附录 ····· 7-1
7.1 管道字段映射规则 ····· 7-1
7.2 疫苗接种案例业务数据库建表语句示例 ····· 7-7

1 概述

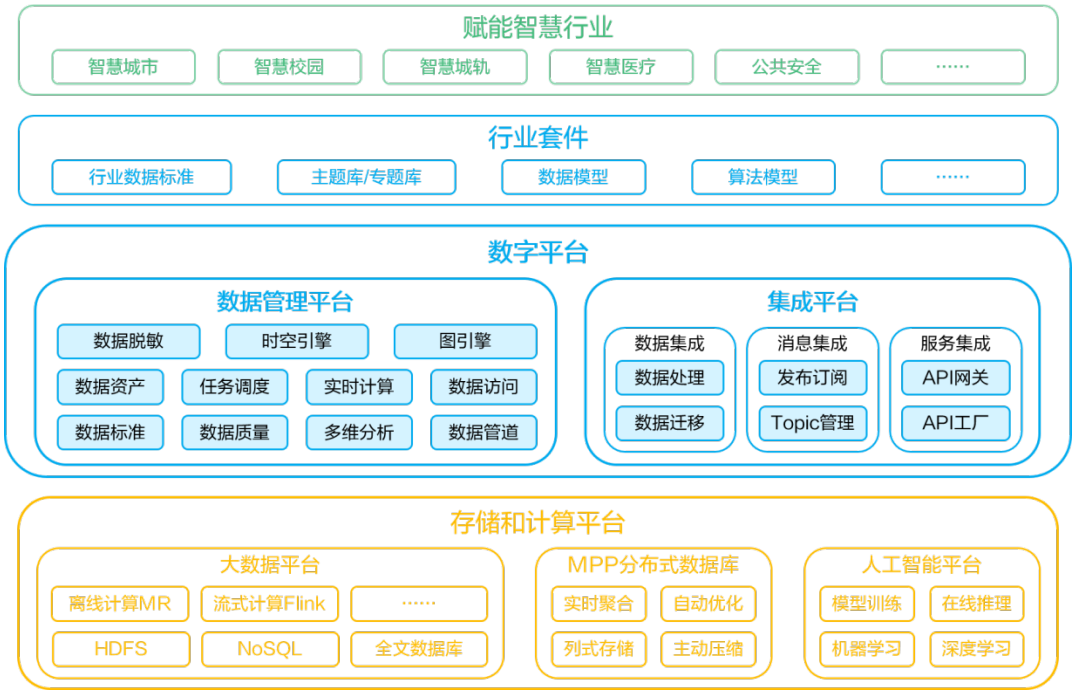
1.1 简介

数据管理平台基于新一代数据开发治理模型，提供了一站式数据开发与治理能力，极大简化治理流程并提高治理效率，为数据管理者提供一站式、自动化的数据治理及数据管控环境，融合了数据访问、数据管道、实时计算、多维分析、业务开发、任务调度、全文检索、时空引擎、图引擎等核心子系统构成。将数据开发的各个环节融合在一套可视化的开发环境中，结合数据标准和数据资产相关能力，实现数据全域开发和治理，可以快速响应业务需求，通过创新带来业务价值。

1.2 产品架构

产品架构如图 1-1 所示：

图1-1 产品架构



数据管理平台产品核心功能模块及其说明如下：

- 数据源管理

  数据是核心资产，在数据管理平台中可以纳管常见的数据源，用户仅需注册进来；可以对于数据源进行增删改查及测试链接等操作；屏蔽多种数据源的客户端差异。

- 标准管理

  将以往文件形式的国家标准和行业标准进行系统化，帮助数据管理者构建自己的数据标准体系。通过定义数据规范，并实现标准的落地，来提升数据的可用性和关联价值。

- 数据开发

  提供数据管道、实时计算、多维分析、业务流程、任务调度、数据查询等数据加工处理端到端的工具集；支持复杂的数据处理模型构建；提供一站式可视化开发与管理界面，支持全托管的作业调度与灵活的调度策略；具有良好的扩展性，支持算子、函数及作业的自定义开发，极大地降低了用户构建数据处理的复杂度，帮助企业专注于数据价值的挖掘和探索。

- 数据资产

  以宏观视角对经过分析和治理的数据进行多种维度的统计，数据支持分层、主题、标签多种维度进行管理，统一管理多种数据源的元数据，拉通数据全生命周期流程形成数据全链路血缘关系，提供数据建模能力。

- 数据质量

  内置多种基础规则模型用于数据质量检测，也支持用户根据业务逻辑定义自己的可复用模型。通过规则模型与数据列进行绑定，建立数据质量指标库，即时或定时监控数据的问题发生率，及时帮助用户发现和分析数据问题。

- 数据脱敏

  数据通常不能直接且全量的暴露给业务使用，往往需要事先对数据中一些隐私、敏感类等信息进行掩盖或加密处理；有效降低或避免数据资产外泄的风险；数据脱敏提供脱敏规则、敏感等级、安全审计等功能，通过对敏感信息识别、数据变形等手段实现对隐私、敏感数据的可靠保护。

- 时空引擎

  时空引擎是一套大规模存取、查询、分析、流动时空数据的工具集合。提供对时空数据的存储、管理和计算调度，能够无缝融合 GIS。

- 图引擎

  图引擎是一个集成图数据库、图计算引擎和图可视化分析的一站式图服务平台。图数据库是一种用图模型来描述知识和建模世界万物之间关联关系的技术方法，旨在从数据中识别、发现和推断事物与事物之间的复杂关系，以及事物关系的可计算模型。

数据管理平台分为三个基础版本和三个特性功能版本，基础版本的差异如表 1-1 所示，特色功能版本如表 1-2 所示。

表1-1 数据管理平台基础版本

功能	标准版	教育增强版	增强版
标准管理	√	√	√
数据资产	√	√	√
数据质量	√	√	√
数据访问	-	√	√
多维分析	-	√	√
实时计算	-	-	√
数据管道	-	-	√

表1-2 数据管理平台特色功能版本

服务组件	特色功能-数据脱敏	特色功能-时空引擎	特色功能-图引擎
数据脱敏	√	-	-
时空引擎	-	√	-
图引擎	-	-	√

数据管理平台不同版本的适用场景如表 1-3 所示。

表1-3 不同版本数据管理平台建议使用场景

版本	建议使用场景
标准版	底层不依赖大数据平台，适合项目预算有限，数据处理复杂度较底且数据量较小的场景
增强版	底层依赖大数据平台，适合有一定数据规模且对数据开发有较高要求的场景
教育增强版	底层依赖大数据平台，主要适用于智慧校园类项目对对数据处理需求的分析场景，比增强版裁减了实时计算及数据管道
特色功能-数据脱敏	必须提前部署增强版或教育增强版。适用于对数据中存在敏感字段，且有查询脱敏需求的场景
特色功能-图引擎	必须提前部署增强版或教育增强版。主要适用于对数据复杂关联关系有处理分析需求的智慧城市类项目
特色功能-时空引擎	必须提前部署增强版。主要适用于对时空数据处理分析需求的智慧城市类项目

1.3 术语和定义

为方便用户理解数据管理平台相关的重要概念，基本术语说明如表 1-4 所示。

表1-4 产品术语

术语	描述
全局索引	全局索引会把HBase二级索引数据放置在HBase中
全文索引	全文索引会把HBase二级索引数据放置在Elasticsearch中
作业	指作业管理模块下的作业。作业是按照系统调度规则生成的，包括可执行的代码程序包
业务流程	将不同作业进行组合生成一个复杂任务执行流程，即可抽象成一个业务流程。一个业务流程包含多个作业，不同作业之间的依赖关系和触发条件都可以在业务流程中配置

术语	描述
流表	流表是流式资源表的简称，是实时计算服务与各种数据源连接的桥梁。通过流表，可以将数据源服务的表注册为实时计算的资源，作为流式计算的输入或输出
实时作业	指实时计算模块下的实时作业。实时作业用于定义一个实时计算的完整流程，包括输入、输出、操作算子及其相关配置，作业开发完成后即可部署运行

2 访问数字平台的数据管理平台服务

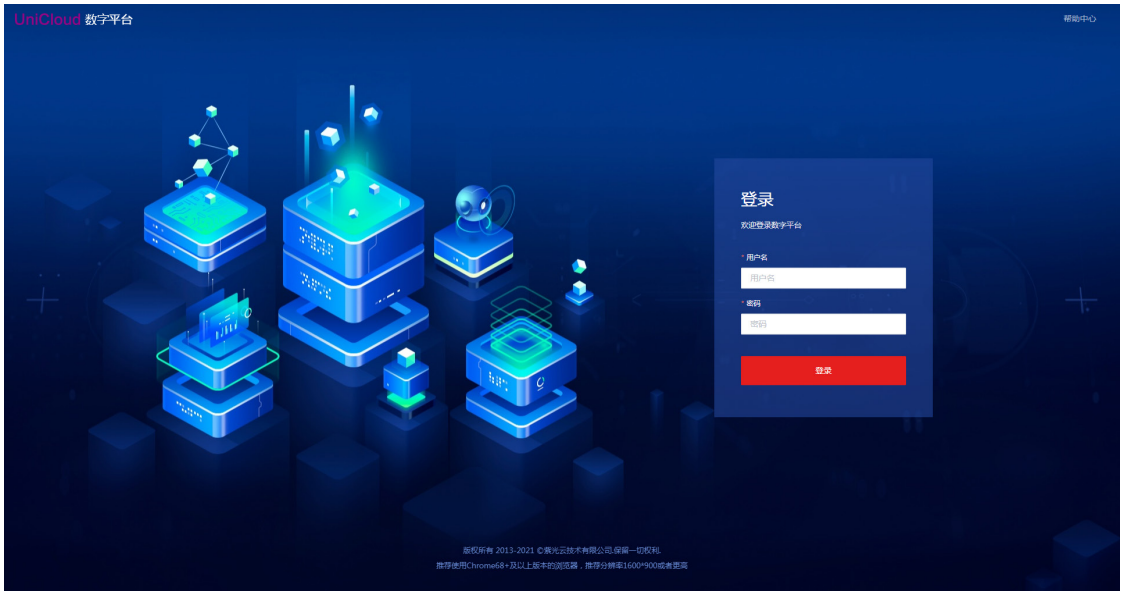
数据管理平台作为数字平台的产品之一，在数字平台中以服务的形式提供数据管理平台的各项功能。

2.1 登录

数字平台的系统服务安装成功以后，数字平台的 URL 地址会在安装脚本成功执行完成后显示，格式为：https://VIP:32015。

在浏览器中输入地址：https://VIP:32015，进入登录页面，如图 2-1 所示。输入正确的用户名和密码，单击<登录>按钮即可登录数字平台。数字平台缺省的超级管理员用户名为 admin，缺省密码为 Passw0rd@_。如果用户名或密码不正确，系统会弹出相应的错误提示。

图2-1 登录页面



登录成功后，跳转至图 2-2 首页。

2.2 首页

首页展示了数字平台的统计信息，首页如图 2-2 所示。

图2-2 首页

2.3 退出登录

登录成功后，在右上角登录的当前用户下拉菜单中选择[退出]菜单项，即可退出系统。

3 功能介绍

说明

- 数字平台包含联机帮助，其中包括对应数据管理平台的联机帮助。在数字平台中数据管理平台的各功能页面中，单击左上角的帮助按钮，弹出帮助窗口，窗口中提供了各功能的详细配置操作说明及注意事项等，可以帮助用户更好地使用数据管理平台。
- 本章仅对数据管理平台各功能进行概括说明，以便用户快速了解数据管理平台各服务提供的主要功能，关于各功能的详细说明请参见数据管理平台的联机帮助。

3.1 数据源管理

数据源管理是对当前用户所在权限下的数据源记录进行管理，该模块管理的数据是数据开发中建表选源的来源。包含以下几个基本功能：查看数据源列表、查看数据源详情、新增数据源、编辑数据源、复制数据源、删除数据源。

表3-1 数据源管理功能特性

特性	描述
查看数据源列表	• 数据源列表中显示当前用户已在系统中增加的所有数据源，展示信息包括数据源名称、数据源类型、描述信息、创建用户、创建时间，对每条记录可进行查看、编辑、删除操作 • 用户可通过上方的搜索条件展开不同维度的搜索，其中 IP 地址代表该源创建时所属的机器 IP
查看数据源详情	在查看数据源列表中单击<查看>按钮，可跳转至数据源详情页面，查看该条记录的详细信息。包含数据源的名称、类型、描述等基本信息及属性列表信息
新建数据源	• 查看数据源列表中单击<新建>按钮，可跳转至新建数据源页面。数据管理平台支持的数据源类型有：Elasticsearch、Greenplum、HBase、HDFS、Hive、Kafka、MySQL、Oracle、PostgreSQL、Redis、STDB、Vertica、达梦 • 新增数据源时，部分数据源需要填写目标 IP 地址、端口号、用户名及密码等信息，页面提供了“测试连接”的功能，点击“测试连接”按钮来测试该条记录是否可以通过连接 • 在新建 Greenplum、Hive、MySQL、Oracle、PostgreSQL、Vertica 这几种类型的数据源时，如勾选了属性“是否采集元数据”，那么在数据资产采集任务模块会创建该源的采集任务

特性	描述
编辑数据源	在查看数据源列表页单击<编辑>按钮，跳转至编辑数据源页面，用户可以修改除“数据源类型”外的任何数据 为了保护用户信息的安全性，对于敏感信息会做脱敏处理，如用户密码只能通过重新输入的方式来覆盖已有的密码。所有信息输入完毕后，便可以进行测试连接并完成提交入库
复制数据源	在已经创建的数据源后面单击<复制>，输入新的数据源名称，就可以复制出一个相同数据源
删除数据源	对于系统中无意义的数据源记录，在数据源列表页中可自行删除。为了防止用户意外删除，除了单击<删除>按钮提示删除确认提示外，还会检查该数据源是否正在被其他模块使用，若存在使用的情况则无法删除，否则将删除该条记录的所有信息

3.2 标准管理

标准管理提供了对国家标准、部级标准、行业标准、以及本地标准进行管理和维护的功能。基于标准，可以解决数据格式不统一、数据内容不规范的问题，帮助用户对现有数据进行梳理，对新数据进行规范约束。从系统层面建立标准体系，用流程化的方式构建新的标准。缺失标准管理这一环节，容易遇到因格式不一致，值域范围不统一导致的数据之间无法有效关联的问题，增加数据价值挖掘的难度，降低数据的可用性。因此，在全局范围内建立标准体系，可以有效保障新数据的质量和使用价值。

表3-2 标准管理功能特性

特性	描述
概述	概述页面包括功能介绍、功能模块、产品术语等区域，各区域的说明如下： • 功能介绍：介绍了数据标准管理的功能及各子功能模块的作用 • 功能模块：通过卡片形式展示了数据元管理、数据项管理和数据集管理的概念，并提供了相应功能模块的入口。单击区域中各模块卡片中的<立即进入>按钮，即可跳转至对应的功能模块页面 • 产品术语：通过卡片形式介绍了数据元相关、数据项相关、数据集相关的产品术语。点击区域中各卡片，会弹出相关的数据说明窗口。单击窗口底部的<确定>按钮，即可返回概览页面

特性	描述
数据集管理	数据集是数据项的集合，用于将数据项按照业务特性进行分类管理： • 数据集管理与我的数据集左侧均提供目录的新增、编辑、删除、搜索功能 • 数据集管理列表上方提供搜索、批量删除、批量移动、批量导出、全部导出、全部删除、查看导出记录功能 • 数据集管理列表内提供数据集的查看详情、移动、删除、共享功能 • 我的数据集列表上方提供新增、导入、模板下载、搜索、批量删除、批量审批和批量移动、批量录入，全部删除功能 • 我的数据集列表内提供数据集的查看详情、编辑、删除、提交审批、直接录入、移动功能 • 我的数据集可以通过引用数据项管理中审批通过的数据项、引用已审批通过的数据集中的数据项两种方式来添加数据项 • 新增或者编辑数据集时，可以通过删除、批量删除两种方式删除数据集中的数据项
数据项管理	数据项是构建数据集的最小单元，数据项可根据数据元进行创建，数据项管理页面展示初始化数据和审批通过的数据，我的数据项页面展示新创建的数据项信息： • 数据项管理与我的数据项左侧均提供目录的新增、编辑、删除、搜索功能 • 数据项管理列表上方提供搜索、批量删除、批量移动、批量导出、全部导出、全部删除、查看导出记录功能 • 数据项管理列表内提供数据项的查看详情、移动、删除、共享功能 • 我的数据项列表上方提供新增、导入、模板下载、搜索、批量删除、批量审批和批量移动、批量录入、全部删除功能 • 我的数据项列表内提供数据项的查看详情、编辑、删除、提交审批、直接录入、移动功能
数据元管理	数据元管理分为部标和本地标，部标数据是依据国家规定初始化的数据，不可编辑删除，本地标的数据是用户在我的数据元页面自己添加的数据，审批通过后会在数据元管理页面中展示，在我的数据元页签中通过审批的数据不可进行编辑/删除操作： • 数据元管理与我的数据元左侧均提供目录的新增、编辑、删除、搜索功能 • 数据元管理提供搜索、批量删除、批量移动、批量导出、全部导出、全部删除、导出记录查看功能 • 数据元管理提供数据元的查看详情、移动、删除、共享功能 • 我的数据元提供新增、导入、模板下载、搜索、批量删除、批量审批和批量移动、批量录入、全部删除功能 • 我的数据元提供数据元的查看详情、编辑、删除、提交审批、直接录入、移动功能 • 我的数据元对系统管理员提供字典管理功能，允许系统管理员新增、编辑、查看、搜索字典

特性	描述
值域管理	值域管理是对标准取值范围的管理，数据元、数据项可以引用值域来描述自身的取值范围： • 值域管理提供目录的新增、编辑、删除、搜索功能 • 值域列表提供新增、导入、模板下载、搜索、批量删除、批量导出、全部导出、全部删除、查看导出记录功能 • 值域列表内提供查看详情、编辑、删除、共享功能

3.3 数据开发

数据开发涵盖数据访问、数据管道，多维分析、实时计算、业务流程、数据建模、以及全文检索等能力集，打造出全域数据开发和全链路的数据监控，让用户轻松能够看到整个开发链条上每个节点的开发状态和统计监控，最终通过数据资产、数据质量、数据脱敏等，方便用户掌控数据资产，为企业政府的发展决策提供依据。

数据开发模块主要包括数据模型、表管理、作业开发、调度中心、调度运维、数据查询、全文检索等。数据开发支持创建实时数据同步作业、将关系库日志数据采集入库等数据接入能力；此外，还支持用户对函数开发扩展与注册使用，以及自行编写 Flink、Spark 和脚本作业，系统可对作业进行全托管地调度和监控。

表3-3 数据开发功能特性

特性	描述
概览	概览通过插图的形式简单展示了数据处理流程图和数据生命周期图，让用户大致了解数据开发
数据模型	数据模型支持可视化创建逻辑表模型，逻辑表可以在多种数据源上一键创建物理表。数据模型支持新建、映射管理、导入、导出、全部导出、导出记录查询等功能 • 单击<新建>按钮，新建数据模型 • 单击<编辑>按钮，编辑数据模型 • 单击<删除>按钮，删除数据模型 • 单击<复制>按钮，配置新数据模型的名称，复制数据模型

特性	描述
表管理	表管理提供了可视化的建表能力，支持按本组织表、已申请表、已发布表、已上架表和已订阅表对表资源进行展示，表结构支持根据标准管理中定义好的行业数据模板进行选择，统一元数据定义，便于数据质量分析和治理 表管理提供了Kafka、HBase、MySQL、STDB、PostgreSQL、Elasticsearch、达梦、Vertica、Greenplum、Hive、Oracle数据源的建表功能，其中，Hive、Greenplum、Vertica、MySQL、PostgreSQL、Oracle、达梦类型的表，支持对字段名、类型、描述的增删改，其他数据源的表只支持追加字段的功能 表管理提供了如下功能： • 根据业务需求对表创建不同的主题和分层，方便管理 • 建表过程全程可视化操作，无需在线编写建表 SQL • 建表关联国家、行业、企业标准，自顶向下，自成一体 • 多种维度对表进行管理和分类，支持搜索 • 提供表的查看、编辑、删除、清空、共享操作 • 表的发布申请功能可以实现表中数据的跨组织共享 • 导出功能可以批量导出平台中的表，然后导入其他数据管理平台 • 支持通过上传 SQL 文件进行批量建表 • 支持详情、列属性、预览、血缘、索引等信息查看 • 数据上架功能支持将表上架到资产市场

特性	描述

作业开发	• 作业管理支持创建实时、数据同步两种类型作业：
	  ○ 实时作业：支持创建 FLINK_GRAPH、FLINK_JAR 和 FLINK_SQL 三种作业类型，支持对作业进行编辑、删除、查询、共享等功能
	  ○ 数据同步作业：支持创建同步任务，将 Kafka 中的数据实时同步到不同的数据源中，支持的目标数据源包括 ES、MySQL、PostgreSQL、Greenplum、达梦、Vertica、STDB、HBase 以及文件中。支持作业的删除、查询、共享等功能
	• 函数管理支持对 Spark 和 Flink 引擎的用户函数进行管理以及 Switch 函数的列表展示：
	  ○ 提供 Jar 包管理，包括对 Jar 包上传、Jar 包查看、Jar 包删除、Jar 包更新
	  ○ 函数管理对函数进行创建、函数编辑、函数查看、函数删除、函数导入导出
	  ○ Switch 函数在数据同步作业中使用
	• 任务管理是对离线任务模板的管理，支持内置作业管理和自定义作业管理，任务模板在[调度中心/业务流程]的离线分析组件中使用时，通过调整参数作为具体的作业运行：
	  ○ 任务管理支持的离线任务类型有 5 种，分别是：SparkJar 任务、Java 任务、MR（MapReduce）任务、Shell 和 PySpark 任务
	  ○ 所有任务的操作类型均包括新建、禁用、删除，不同的任务类型所需要的参数及任务配置项有所区别
	  ○ 内置作业包括 HBase 表数据统计、HBase 表数据批量加载、时空表数据批量导入导出

调度中心	调度中心的核心是业务流程数据管理，一个业务流程数据对象可以统筹操作多个类型的作业，如同步作业、实时作业等，多个不同类型作业各自完成不同的目标，但最终是为了实现一个实际业务目的 • 新建业务流程 • 业务流程画布中支持数据集成、离线分析、实时计算、控制节点 • 业务流程画布中，除数据同步与 StreamingJob 外的组件节点支持一键成组，组内节点串行执行，支持手动从失败节点开始执行 • 业务流程支持简单调度和高级调度，高级调度使用 Cron 表达式控件 • 业务流程支持批量删除、查看批量操作记录功能 • 业务流程支持根据业务流程名称和更新时间、创建时间等列进行排序 • 单个节点支持数据补跑，对 SQL 中的时间变量界面中可以传递具体的值 • 支持 MR、SparkSQL、SparkJar、Sqoop、pySpark、RDSSQL 作业指定 Yarn 的执行队列 • 支持 Java、Shell 作业输出执行结果；Java、Shell、SparkJar、PySpark 作业接收上游作业节点的结果作为参数参与作业执行

特性	描述
SQL调试	SQL调试可用于调试对SPARKSQL或HIVE数据源执行的SQL语句： • 支持查看表字段信息 • 支持对 SQL 进行执行、SQL 上传、选中执行、格式化、语法校验等便捷操作 • SQL 编辑器支持多个 SQL 串行执行 • 支持执行记录、动态日志、执行结果、表信息、字段信息的查看
调度运维	调度运维页面展示了业务流程的状态监控信息，并通过列表的形式展示了所有业务流程： • 调度运维概览查看业务作业的运行状态、业务流程的总量和新增数量 • 调度运维支持业务流程的批量提交和停止 • 调度运维可以概览业务流程的健康状况和异常节点信息 • 业务流程支持分组和标签管理 • 调度运维中支持对启动后的业务流程进行监控和调度信息查看，其中监控页面可以对节点的运行情况、日志、数据进行查看和预览；调度页面支持查看任务调度执行记录，包含已执行、正在执行和计划执行的记录 • 支持查看批量操作记录日志
数据查询	数据查询提供了简单易用的数据查询工具，简单容易上手，不需要掌握大数据组件的原始用法同时具备多种便捷操作，例如历史SQL查看，多窗口、查询结果导出等： • SQL 编辑器支持对 HBase、MySQL、Oracle、PostgreSQL、Elasticsearch、Greenplum、达梦、Kafka、Vertica、hive 类型表进行查询 • 支持对数据进行 insert、update 等更新操作 • 支持对 SQL 语句进行格式化、保存 SQL、查看历史 SQL 列表等便捷功能 • 查询结果以结构化表格展示，支持翻页和查询结果导出 • 左侧导航树以数据源列表展示，选中可查看已有表中数据量、字段等详情 • SQL 编辑器支持多窗口，方便同时执行多种 SQL • SQL 编辑器中执行 SQL 语句添加 limit 限制，查询结果最多返回 10000 条

特性	描述

全文检索	• 数据搜索：
	  ○ 主要用来对[表管理]中创建的 Elasticsearch 类型表中的数据进行检索
	  ○ 对 Elasticsearch 集群中所有 Elasticsearch 表的结构化和非结构化数据的整体全局搜索，以及对某些指定主题下的所有 Elasticsearch 表的全局搜索。指定数据源下的一个或多个表进行表的全局搜索。如果搜索到的数据本身含有附件类型的数据结构，还提供文件的下载功能
	• 数据上传：
	  ○ 支持对[表管理]中创建的 Elasticsearch 类型表插入数据
	  ○ 支持对 int、keyword、text、double、date、long、boolean 类型数据的上传
	  ○ 支持对 attachment 类型字段的文件、图片、视频的上传
	• 数据导入：用于将 MySQL、PostgreSQL、Greenplum、达梦和 Vertica 类型数据源中某张表的数据导入到 Elasticsearch 类型数据源中的目标 ES 表，并且支持持续监控来源表中的数据新增情况，同步将新增数据更新到对应的目标 ES 表中。这部分的数据导入需要在数据源中定义 ES 类型的索引表（不包含附件类型的表）；另外，对于外部表需要提供时间类型和主键字段的支持，事件类型用来判断数据的增加情况，主键主要用来进行数据的去重
	• 集群监控：集群监控以图表形式展示 Elasticsearch 集群总览信息、集群节点信息、Elasticsearch 索引信息、全文检索服务实例信息和服务接口调用数据
	• 自定义词库
	  ○ 自定义词库用于全文检索自定义分词的管理，在此可以添加行业或业务等需要的特殊词汇，优化索引和搜索效果
	  ○ 用户可以通过文件方式去批量导入，也可选择单个词汇进行单条上传
	• 快照管理：快照管理用来备份某一时刻 HBase 及 Elasticsearch 的数据状态，以便异常情况下恢复 HBase 及 Elasticsearch 的数据，防止数据丢失
	• 时间序列：时间序列功能可以根据用户的时间相关数据，渲染出某些指标在某段时间的变化情况，从而便于用户根据现有趋势做出决策

数据管道	• 监控概览：对用户指定的 Kafka 数据源进行监控，监控的内容包括历史趋势、总数据量以及数据量信息，其中数据量信息中含有处理数据量、管道类型分布、生产和消费速率
	• 数据采集
	  ○ 文件上传系统：支持通过文件上传、Flume 采集、NIFI 采集三种方式写入
	  ○ 数据接入接口：支持通过 Java SDK 方式进行数据采集

文件管理	提供与数据管理平台关联的大数据集群中HDFS组件的操作。其通过列表的方式展示了HDFS文件系统中的目录和文件，并提供了常用的管理操作

3.4 数据资产

数据资产以宏观维度展示了客户数据资产全貌，实现多种维度的资产图表统计展示，并提供数据血缘和智能搜索，以盘清数据资产、理清数据链路、管妥数据分层和搞懂数据价值为使命。

表3-4 数据资产功能特性

特性	描述
总览	统计MySQL、Oracle、PostgreSQL、Greenplum、HBase、Elasticsearch、Hive2(Embedded Http)、达梦和Vertica5这9种数据源的数据，以图表的形式展示多维度的数据统计情况，包括：数据源总数、表总数、数据字段总数、数据总存储大小、数据总条数、同前一天的比较。按照主题或分层统计数据表总数分布、数据字段总数分布、数据条数总数分布、数据存储量总数分布、数据量趋势(按主题或分层)。对于每一种数据源类型，统计上述指标项，表来源于两部分，采集的表和数据开发中创建的表。将不同数据源下的表挂载到自定义的主题以及分层下，可以从主题和分层的维度按上述指标项对挂载表的数据进行统计以及展示
元数据采集	支持对MySQL、PostgreSQL、Oracle、Greenplum、Hive、Vertica六种关系型数据库中的表信息、字段信息的采集，不采集视图、函数、存储过程 • 采集任务：支持周期性调度，任务运行后支持查看采集任务的日志信息。目录树中将展示任务个数的统计数量 • 任务监控：展示了采集任务的运行记录和日志，并可以对采集任务进行重跑操作
血缘管理	数据开发中涉及到数据处理、流转的服务包括数据采集、实时计算、多维分析、数据管道等，在数据处理过程中记录数据的关系，从中解析出完整的数据血缘关系 在界面支持按照数据源、表以及作业类型、作业名这两种维度进行搜索，结果以可视化的方式在界面展示表和字段的血缘关系。对于自动解析作业中的数据血缘关系失败的情况，可以通过手动添加的方式生成作业中的血缘关系
表类别管理	表类别管理提供对表的快速分类功能。使用者可以根据已经注册的数据源，对该数据源下的表进行批量分类，可以按照自定义的主题以及分层批量添加以及删除 在表类别管理页面可以根据数据源、是否已经关联等条件对表信息进行过滤，方便查看某个主题或分层下已经挂载的表，或者某个数据源下挂载在指定主题或分层下的表。挂载到不同主题和分层下的表将在总览中统计并展示。目录树中对挂载表的个数进行统计并展示

特性	描述
配置管理	配置管理提供了对主题、分层和标签的管理能力，表管理中建表时按照具体业务指定主题和分层，便于管理： • 标签管理：标签作为一种带有业务属性的标记，可以很好地对系统内的资源进行检索和分类；可以给资源打上多种业务属性信息。标签管理支持标签的新建、更新、删除、导入、导出等操作 • 主题管理：支持 3 级，借助主题可以清晰地对资源进行归类及查找，主题管理提供主题的新建、更新、删除等操作 • 分层管理：旨在将表进行不同层次的划分，可以按照数据仓库的定义将数据分层，提供对分层的新建、更新、删除等操作 • 统计管理：展示了数据源的统计配置项，支持按不同数据源类型配置数据的统计情况和统计周期

3.5 数据质量

质量管理模块通过为数据字段绑定规则模型，来定义字段指标。多个指标可以挂载到某一个质量监控任务中，形成一套质量监控方案，根据质量监控任务的执行结果形成质量报告，从字段的维度统计各指标的准确率。

表3-5 数据质量功能特性

特性	描述
规则模型	内置的校验规则，在配置指标时会用到这些内置规则，目前包含的内置规则有以下： • 空值校验 • 值域校验 • 格式校验 • 长度校验 • 唯一约束校验 • SQL 条件校验 此外，还支持自定义创建规则模型，导入、导出、查看导出记录、全部导出、模板下载功能
指标管理	指标通过对数据字段绑定规则模型，来定义字段指标，指标管理提供指标的新增、全部删除、批量删除、导入、导出、导出记录查看、全部导出、模板下载等功能 同时提供新增指标保存为草稿、更新草稿、删除草稿、草稿保存为指标功能

特性	描述
质量监控	多个指标可以挂载到某一个质量监控任务中，形成一套质量监控方案，质量监控提供对监控任务的创建任务、编辑、删除、共享、搜索、详情查看、执行结果查看、导入、导出、导出记录查看、全部导出、模板下载功能 手动调度提供立即执行、停止任务功能，自动调度提供启动任务、结束调度任务功能
质量报告	一个质量监控任务的执行结果可形成一个质量报告，从表、字段的维度出发，分为数据表质量报告与指标趋势报告，以图表的形式展示各指标执行情况（展示检核数据量、错误数据量、错误率），支持导出数据表质量报告
质量评估	质量评估用于对指定的数据源进行扫描检测，分为评估配置和评估报告： • 评估配置：用于管理数据源的评估配置任务：用户通过创建不同的评估配置，评估配置即可以任务的形式根据指定的调度方式对数据源进行质量评估，并将结果呈现在评估报告中 • 评估报告：提供了数据源的质量评估报告：报告中提供了数据源的质量评级、接入率、表活跃度等信息，并提供了表详情列表

3.6 数据脱敏

数据脱敏是指对数据中的敏感信息通过规则进行识别和脱敏处理，使敏感数据得到有效保护。其通过对敏感信息自动发现、分级分类、数据变形、安全审计等功能实现对敏感隐私数据的可靠保护。数据脱敏包含数据发现、数据访问、风险审计、扫描状态、配置管理、系统配置等模块，支持数据源包括 MySQL、Oracle、PostgreSQL、达梦、Vertica、HBase、Hive、Elasticsearch、Greenplum。

表3-6　数据脱敏功能特性

特性	描述
首页	首页展示了数据脱敏系统状态的总览信息，包含近一周新增敏感字段总数、近一周捕获的风险总数、敏感数据访问量趋势、风险操作量趋势、用户授权的等级分布和数据源授权占比，旨在让用户掌握系统敏感数据的分布及访问状况，调整措施，提高数据保护能力。该模块提供展示和切换组织的功能
数据发现	数据发现基于不同维度展示识别到的敏感字段的统计信息，包含识别字段数、识别表总数、敏感信息在各分级分布、数据源分布以及明细统计。该模块提供了展示、搜索和切换组织的功能
数据访问	数据访问记录并展示对敏感信息访问的统计和详细信息，包含访问量趋势图、访问量、访问人数和访问详细记录的展示。该模块提供了展示、搜索、和切换组织的功能
风险审计	系统支持自定义风险审计规则，对潜在危险操作进行识别并记录。风险审计模块展示识别的风险操作统计信息和详情展示，支持用户对结果进行审计操作，确认或排除风险项。通过此项，管理员能够发现数据风险项，进而采取针对性操作提高系统的数据安全。该模块提供了展示、搜索、审计、删除和切换组织的功能

特性	描述
扫描状态	扫描状态展示了敏感数据识别日志和调度任务的日志。敏感数据识别日志以表粒度展示扫描的表信息、扫描状态、结果信息和扫描时间等内容；针对扫描失败的表，可以按结果信息进行处理并重试，该模块提供了展示、搜索和切换组织的功能。调度日志展示调度任务的启动时间、与单次调度扫描的表数量，该模块提供了展示和切换组织的功能
配置管理	配置管理提供了创建并维护各类规则的能力，包含分级信息管理、数据识别规则、数据脱敏规则、风险识别规则、敏感信息维护，同时具备授权功能。通过这些配置，决定了数据脱敏的最终结果 • 分级信息管理：用来定义和管理数据的敏感级别，实现根据数据的敏感程度对敏感数据进行分级分类，确定所属的保护级别，方便进行权限控制，并实现根据分级进行资产打标，安全管控等。该模块提供新建、导入、导出、展示、搜索、编辑、管理分级规则的能力 • 数据识别：是基于数据识别规则，针对系统中存在的表进行敏感字段的自动发现，数据识别规则针对不同敏感数据进行识别，内置常用敏感字段识别模板，并支持用户根据其行业特点自定义规则，根据配置的识别规则对存储的数据进行扫描、分类、分级。该模块提供新建、展示、搜索、编辑、删除、启用或者禁用规则的功能 • 数据脱敏：是基于脱敏算法对数据进行变形、掩盖、打乱、加密等，实现对敏感信息的脱敏。数据脱敏组件通过数据脱敏规则定义数据的脱敏方式，将数据脱敏规则和数据识别规则进行关联，是进行数据脱敏的基础，数据脱敏规则用来配置对应的数据识别规则对系统识别到敏感数据如何进行脱敏。该模块提供新建、展示、搜索、编辑、删除、启用或者禁用规则的功能，能够实时预览脱敏效果 • 风险识别规则：对用户的行为进行分析，将触发到规则的事件进行记录，推送给安全管理员进行风险审计，支持针对数据源、表、分级、敏感数据类型、操作数量、操作时间等场景进行布控。该模块提供新建、展示、搜索、编辑、添加配置、删除、启用或者禁用规则的功能 • 敏感信息维护：用于展示系统识别到的敏感字段信息，包括敏感字段所属的表以及数据源信息，敏感字段对应的识别规则等。支持对敏感信息的搜索、编辑与删除，方便对识别结果进行手动修正 • 授权配置：支持对数据源授权和用户授权两种。数据源授权支持对数据源进行扫描权限、脱敏权限、审计权限的配置，细分对数据源的操作权限，支持新建、展示、删除数据源授权。用户授权目的是给用户指定访问等级，以及授权角色，授权用户仅能访问到小于等于自身等级的未脱敏信息，支持开发、审计、普通三种角色的选择，支持展示、新建、编辑、删除用户授权
系统配置	系统配置提供了调整敏感数据扫描过程中关键参数和配置定时任务的能力。通过调整扫描参数，可以在扫描效率与准确率间取得平衡 配置定时调度任务，可以在指定时间启动或者周期性启动扫描任务，对系统配置的数据源进行增量扫描。该模块支持展示、编辑的功能

3.7 图引擎

图引擎是一个集图数据库、图计算、图可视化为一体的一站式图服务平台。针对高度互联数据的存储和查询场景进行设计，提供一种更好的组织、管理和理解海量信息的能力。适用于数据之间存在复杂或深度关联关系的场景，利用高度连接的数据中复杂、动态的关系来产生洞察力和竞争优势。

表3-7 图引擎功能特性

特性	描述
图谱	图库中所有图集中管理，提供图的创建、修改、删除以及配置管理。支持同组织间共享/取消共享图
图概览	展示图的业务流程以及图的用户信息，并提供了各种操作的入口： 提供了清空、备份/还原、提交统计、定时统计等配置功能
图模型	图模型页面中展示了图中所有标签模型，包括模型管理、模型管理（表格视图）和索引管理： • 模型管理，包括可视化建模、模型管理（增删改查），从数据中提取实体、关系、属性要素，进行模型构建，满足业务场景建模需求 • 模型管理（表格视图），列表方式展示属性、顶点标签和边标签，并提供了更新等管理功能 • 索引管理，支持通过索引加速点、边的查询效率，提供索引状态监控以及索引的分布式重建、删除功能 此外，模型管理和索引管理均提供模型批量加载与导出Schema文件功能
图入库	图入库功能包括图模型展示以及数据入库管理。针对不同场景对数据接入的需求，提供单机、分布式、实时三种数据导入工具 可支撑亿级数据的高效入库，支持增量/全量两种入库模式。可通过任务中心实时监控入库任务状态，支持入库过程中点数据校验、错误数据记录功能
图检索	图检索用于图数据的查询，包含点查询、边查询、路径查询、扩展查询以及Gremlin查询五种查询方式 针对点、边查询，封装了基于属性条件过滤的查询功能。查询结果可视化展示，提供树形、圆形、力导向图三种布局方式，支持查询结果的过滤及导出。Gremlin查询支持输入Gremlin语句检索图数据，提供历史查询语句保存及查看功能
图算法	图算法功能用于关系数据的推理运算，挖掘隐藏关系 内置丰富的图算法库，包含PageRank、PersonalPageRank、连通体、增强连通体、三角计数、单源L-hop、多源L-Hop、标签传播、最短路径、单源最短距离、最短距离，基于Spark GraphX提供分布式的图数据挖掘，满足各种场景的算法分析需求。提供可视化的算法分析界面，通过界面选择相应算法、数据进行分析，提供算法分析任务的可视化监控，并对分析结果可视化展示

图任务	图任务提供图的任务监控功能，可监控的任务包括数据统计、备份/还原、入库、算法分析，可以实时查看任务状态，任务失败可以检查异常信息。针对入库任务，提供任务取消及任务详情监控功能，实时查看入库数据量

3.8 时空引擎

时空引擎以大数据分布式技术为基础，提供稳定、弹性和高效的时空存储和计算服务，通过对时空数据进行高效索引和管理，实现海量数据快速查询和分析。适用于对包含空间位置信息的数据进行查询和复杂分析的场景。

表3-8 时空引擎功能特性

特性	描述
时空概览	展示时空引擎整体运行情况，包含了时空数据源状态预警，时空数据量分布情况，热点时空数据表访问量，重点概览数据展示。以上功能均按照时空数据源进行分类，可按照时空数据源切换数据
数据目录	根据时空数据表结构，展示时空数据源和时空catalog的树结构，根据时空数据源和catalog展示了时空数据表的基本信息，可对时空表进行查看、离线表注册、扩展表、清空表数据等操作 时空表详情展示时空表的基本表信息和所有字段信息，可对表进行数据导入操作，查看时空表在物理集群上的状态信息
时空查询	对时空表进行数据查询，查询方式支持查询转换、实时状态、聚合统计、数据导出、轨迹查询： • 查询转换对数据使用 CQL 查询语法进行数据过滤查询，支持对查询结果函数转换，针对时空表的 HBase 数据源进行查询 • 实时状态查询是对时空对象的最新位置信息进行查询，针对时空表 Redis 数据源进行查询 • 聚合统计查询是根据 CQL 过滤条件对数据进行多种聚合分析 • 数据导出功能是使用分布式导出方式根据 CQL 过滤条件对大数据量进行导出 • 轨迹查询是根据 CQL 条件过滤的位置信息重组成轨迹信息的查询方式
运行维护	对时空引擎进行总体配置，设置时空数据统计周期，配置时空引擎对接的地图引擎信息，以及对时空概览重点数据的配置

3.9 配置管理

配置管理用于管理数据管理平台的参数管理、字典管理、告警管理和行业套件。

表3-9 配置管理功能特性

特性	描述
参数管理	支持对数据管理平台服务运行参数的动态管理，内置的参数包括业务流程的重试参数、SQL调试执行记录的保存周期、离线分析是否加载时空函数、以及操作日志保留时长，并可以添加自定义的属性。仅系统管理员可见
字典管理	字典管理目前主要是对数据元中组织机构、对象类词、表示词、计量单位数据的管理： • 字典管理对所有用户提供查看功能 • 字典管理只对系统管理员提供编辑、删除功能，对组织机构、对象类词、表示词、计量单位根目录不提供编辑、删除功能
告警管理	以表格的形式展示数据管理平台中各功能的告警情况。包括当前告警信息查询、历史告警信息查询、当前告警信息确认、告警级别查询和编辑、告警通知管理。其中，系统会根据告警通知管理中所设置的邮件地址，将对应租户的告警信息周期发送至已配置的邮件地址
行业套件	行业套件主要用于对智慧园区、智慧教育等场景的DO数据模型在数据管理平台中进行安装初始化工作，支持安装、版本升级和卸载操作

4 共享自行车案例

数据管理平台是通过数据技术，对海量数据进行采集、清洗、加工，成为标准化、统一化的数据存储，形成大数据资产，进而为客户提供高效的、创新的服务。其中又以数据开发模块最为核心，将实际业务需求抽象为一个个实体，提供基于任务类型的代码组织方式。

本章旨在以业务流程中的“SparkSQL”节点为核心创建离线任务，展示开发及运维等过程的完整使用步骤。

4.1 案例说明

自行车共享系统是一种租赁自行车的方法，注册会员、租车、还车都将通过城市中的站点网络自动完成。通过该系统，人们可以根据需要从一个地方租赁一辆自行车然后骑到自己的目的地归还。

但在共享单车的运维过程中，不时出现一些共享单车长时间未有运行轨迹上报或无法准确监控到位置的情况。这就导致，在 APP 中可以看到很多可用单车，但走到对应位置时，才发现该位置没有单车或无法使用。这些单车中，有些单车不知道被藏到了哪里；有些车或许是在高楼的后面，因 GPS 的误差而找不到；有些车被放到了小区内，一墙之隔使骑车人无法取到单车，甚至使单车直接从 GPS 定位中消失等等。

为了获得这些单车的数据，确认这些车是否已经变成了“僵尸车”，本案例将通过业务流程中的 SparkSQL 节点将单车基本数据信息与实时更新的数据信息进行关联分析，筛出“僵尸车”。

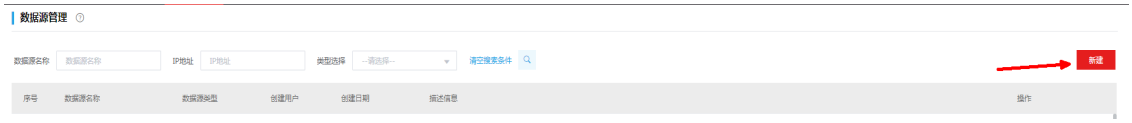
4.2 准备操作

4.2.1 创建数据源

在通过业务流程对需要分析的数据源进行操作之前，需要将待分析的数据源注册到数据管理平台中。当前数据管理平台支持丰富的数据源类型，用户可选择合适的数据源并进行配置。本说明文档将以 Hive 存储类型的数据源为例说明数据源创建的过程。

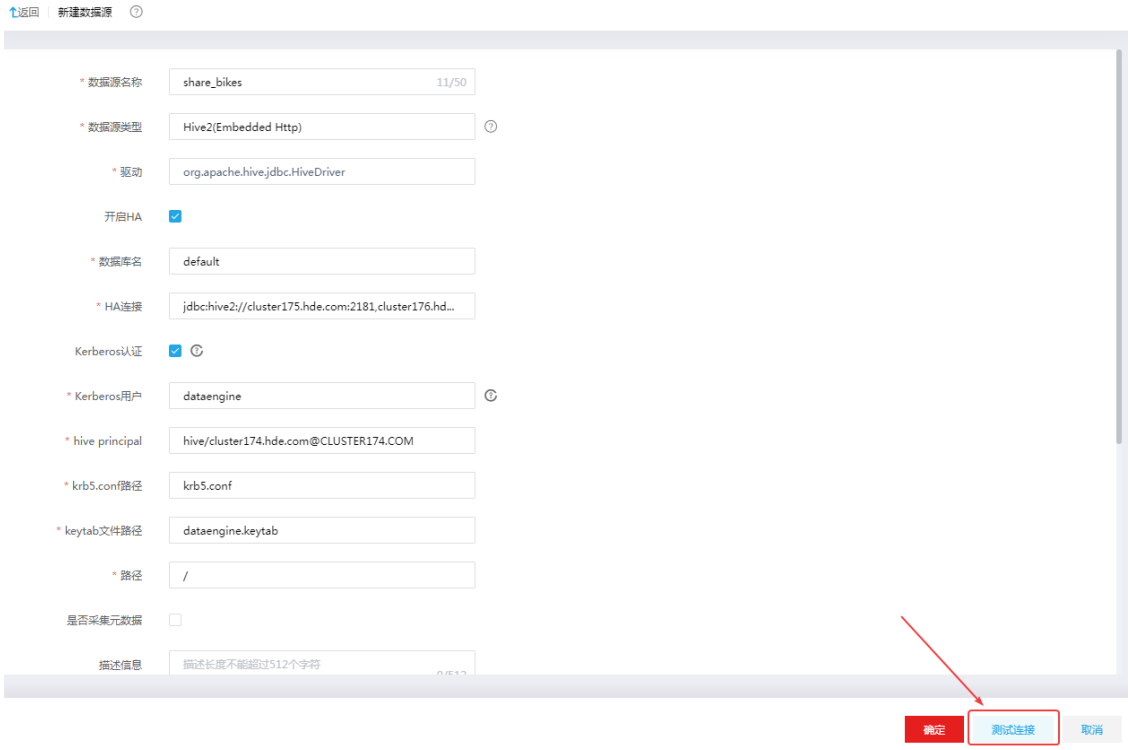
(1) 在[数据源管理]模块中，单击右上角<创建>按钮，进行数据源的创建操作，如图 4-1 所示。

图4-1 数据源配置页面



(2) 选择 Hive 数据源，并配置参数，图 4-2 所示为创建 Hive 数据所需填写的信息模板。

图4-2 新增 Hive 数据源页面



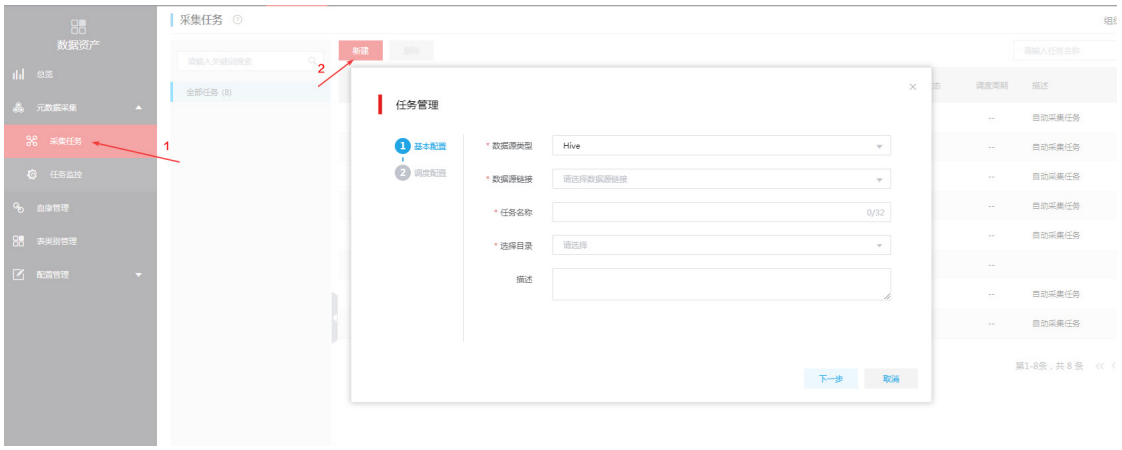
(3) 填写完毕注册数据源所需要的信息之后，可以单击<测试连接>按钮，测试数据源连通性。

(4) 提示“连接测试成功”信息，单击<确定>按钮，执行注册数据源。之后即可在数据源列表中看到注册成功的数据源概要信息。

4.2.2 注册离线表

(1) 成功创建数据源之后，进入[数据资产]，按照图 4-3 中所示的 1/2 步骤，进入元数据采集的任务创建页面，并创建元数据采集任务。

图4-3 元数据采集配置

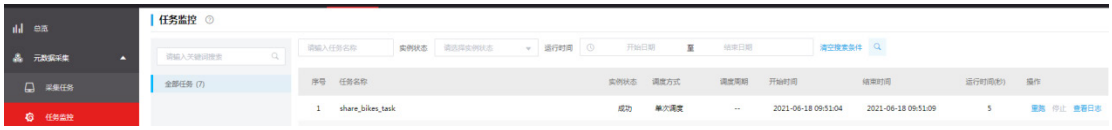


(2) 创建完成后，在列表中单击对应操作列中的<运行>按钮，启动采集任务，如图 4-4 所示，将数据源 share_bikes 中的所有 Hive 表采集到数据管理平台中。

图4-4 启动元数据采集

(3) 采集任务启动之后，可在左侧导航树中选择[元数据采集/任务监控]菜单项，进入采集任务监控列表查看采集任务执行进度，如图 4-5 所示。

图4-5 元数据任务监控



采集任务也可以设置调度周期为定期或周期性的对某个数据源执行采集操作，更新已采集的数据表信息。Hive 数据源的表在创建采集任务时，已配置了“注册离线表”操作，即已被注册为离线表，其他类型数据源中的表（MySQL/Oracle/PostgreSQL 等）需要执行“注册离线表”操作后才可在业务流程的 SparkSQL 节点中使用。

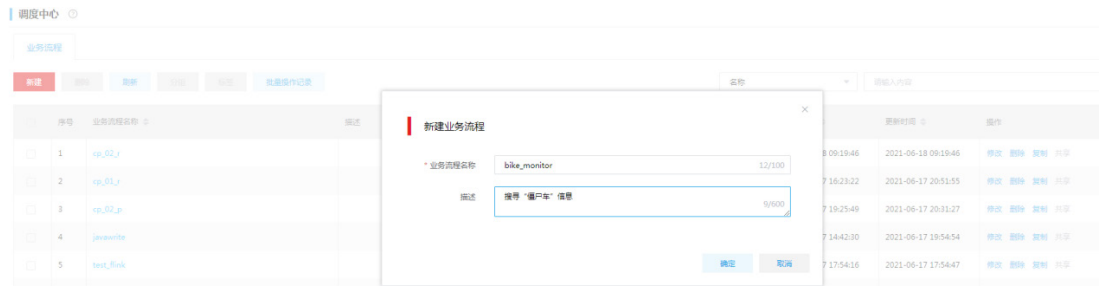
4.3 构建业务流程

业务流程是按照业务的种类将相关的不同类型的节点任务组织在一起所构成的有向无环图。本章中以 SparkSQL 节点为例介绍离线分析任务在业务流程中的使用步骤。

4.3.1 创建业务流程

(1) 进入[调度中心]模块，单击<新建>按钮，弹出新建业务流程窗口，如图 4-6 所示。

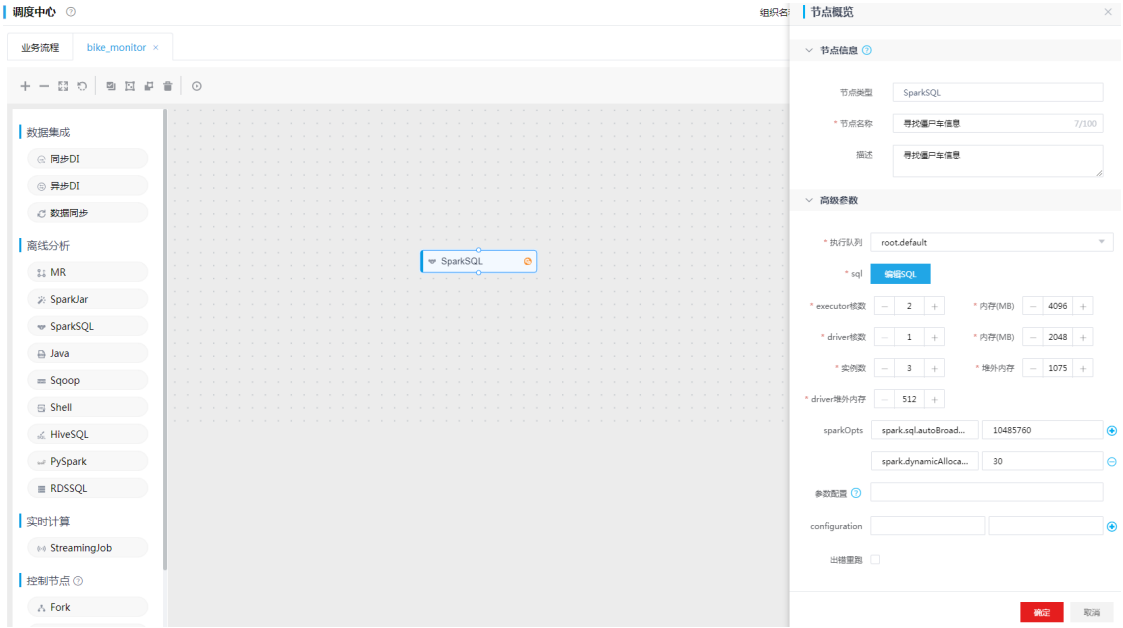
图4-6 新建业务流程



(2) 填写业务流程及名称，单击<确定>按钮，新建业务流程。

(3) 进入业务流程画布编辑页面，拖拽一个 SparkSQL 节点到画布中，双击弹出右侧边栏，如下图 4-7 所示。

图4-7 查看 SparkSQL 节点信息



4.3.2 编辑 SparkSQL 节点

(1) 编辑 SparkSQL 节点信息，根据提示填写必要的节点名称等信息。

(2) 单击<编辑 SQL>按钮，弹出 SQL 智能编辑器窗口，如图 4-8 所示。

图4-8 SQL 智能编辑器开发 SQL

(3) 在此窗口中编写符合 SparkSQL/HiveSQL 语法规范的 SQL 语句，所使用的分析表即为通过元数据采集操作注册进来的 Hive 数据表。

本例所述的识别“僵尸车”信息的分析 SQL 语句，其执行的操作是对单车基本数据信息表 share_bikes.bikes_record 和实时更新的数据信息表 share_bikes.bikes_history 做 left join 操作，获取近 30 天内未有实时数据的车辆即推测为“僵尸车”。

一般而言，对于需要通过 SparkSQL 节点任务执行的离线分析操作，建议将分析结果保存至数据表中。将分析结果保存至结果表可以便于后续查看分析结果或作为其他操作的数据源使用。本例中，在保存执行识别僵尸车的分析 SQL 语句之前，对其添加结果表 share_bikes.zombie_bke_info。

将 SQL 语句改写为“create... select...”方式，如图 4-9 所示。

图4-9 通过 SQL 创建结果表

(4) 编辑完成 SQL 语句后，可以通过单击<语法校验>按钮，校验 SQL 语法是否正确规范。在不追加结果表保存的情况下，也可以单击<执行>按钮或<选中执行>按钮，直观地查看结果。

(5) SQL 智能编辑器窗口的各参数填写完毕后，单击<确定>按钮，退出编辑 SQL 窗口，返回业务流程画布编辑页面。

(6) 在页面右侧边栏中配置其他几项可选填的参数，具体说明请参阅表 4-1。

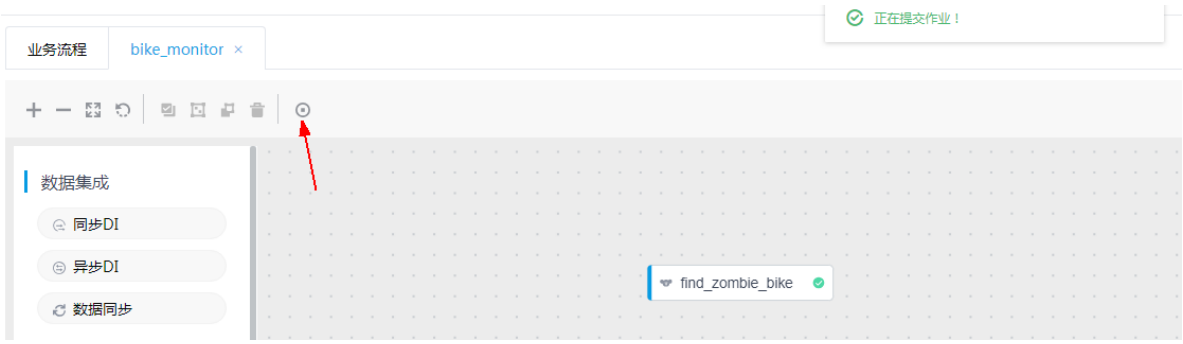
表4-1 SparkSQL 节点选填信息说明汇总

参数名	参数说明	示例值
配置参数	配置执行节点的基本配置参数	• executor 核数：2 • 内存(MB)：4096 • driver 核数：1 • 内存(MB)：2048 • 实例数：3 • 堆外内存：1075 • driver 堆外内存：512
sparkOpts	通过业务流程的方式提交任务给Spark集群时的资源参数	• spark.sql.autoBroadcastJoinThreshold=10485760 • spark.dynamicAllocation.maxExecutors=30
参数配置	如在SparkSQL中需要写入动态参数，如EL表达式等，该参数配置用于传递实际的参数名和参数值	如果SparkSQL代码为“select * from default.table where start_date='$biztime' and end_date='$cyctime'”，其参数配置值为“$biztime= 2021-06-10 $cyctime=2021-06-11”
configuration	用于传递给SparkJob任务的配置属性，一般为hadoop的配置项	key: mapred.compress.map.output value: true

4.3.3 提交执行

保存 SparkSQL 节点之后，单击业务流程画布左上方的 图标，启动业务流程，如图 4-10 所示。

图4-10 启动业务流程



4.3.4 任务监控

启动业务流程之后，可通过画布右上角<进入监控页面>按钮，或通过在左侧导航树中选择[数据开发/调度运维]菜单项，在调度运维页面。上述名为 bike_monitor 的业务流程提交之后在调度监控列表显示如图 4-11 所示。

图4-11 业务流程任务列表

单击<监控>按钮，进入该业务流程的监控画布页面，提交的 SparkSQL 节点上显示监控属性的实时变化如图 4-12 示。在画布中右键单击该节点，弹出菜单中会显示该节点相关的“运行详情”、“查看日志”等监控属性，可查看该节点的具体监控信息，如图 4-13 所示。

图4-12 业务流程执行状态信息展示

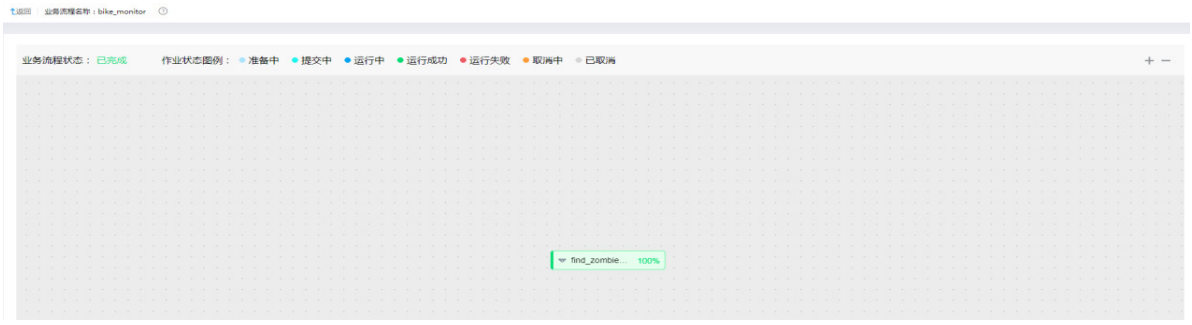


图4-13 业务流程节点执行信息查询

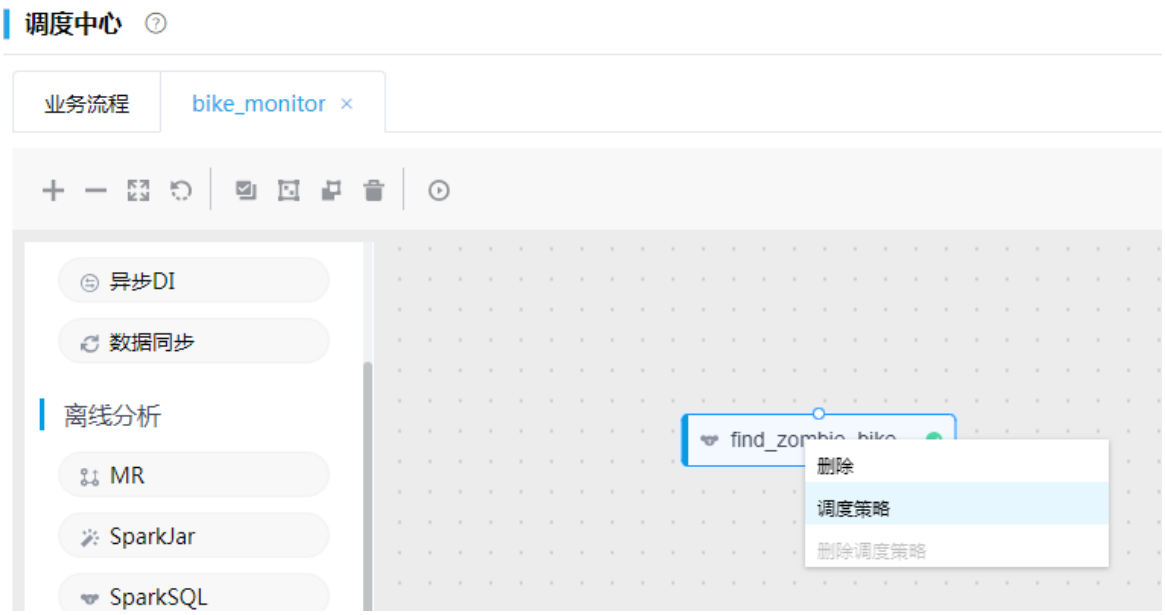
4.3.5 调度策略

1. 配置单个节点的调度策略

对单个节点而言，可以在业务流程画布编辑页面中，右键单击具体节点，配置调度策略。以图 4-14 中的 find_zombie_bike 节点为例：

右键单击该节点，在弹出菜单中选择“调度策略”菜单项，即可为该节点设置调度策略。配置完毕后，单击左上角的 图标，启动业务流程，在该过程中调度策略将被触发执行。

图4-14 业务流程节点调度策略配置



2. 配置多个节点的调度策略

当需要对多个节点配置调度策略顺序并周期执行时，可以通过业务流程的成组功能将多个节点设置成组，之后批量为其添加调度策略。

在介绍成组操作前，对照图 4-15 的标记序号对业务流程编辑页面的左上角各个按钮的说明如表 4-2 所示。

图4-15 业务流程操作按钮

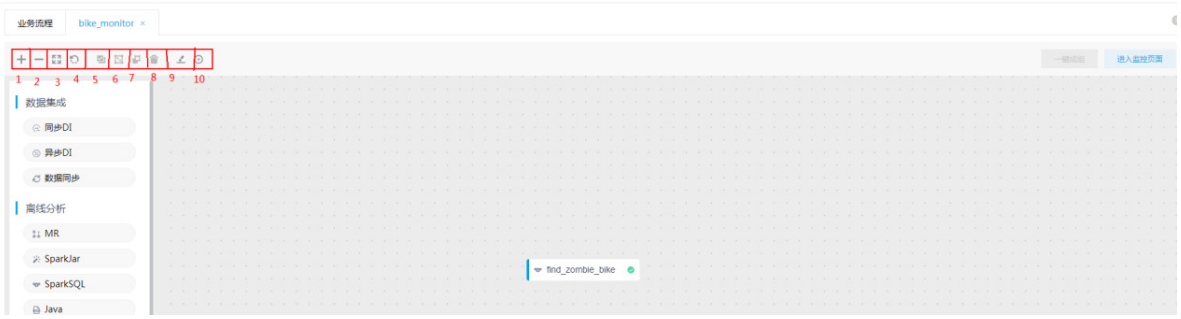


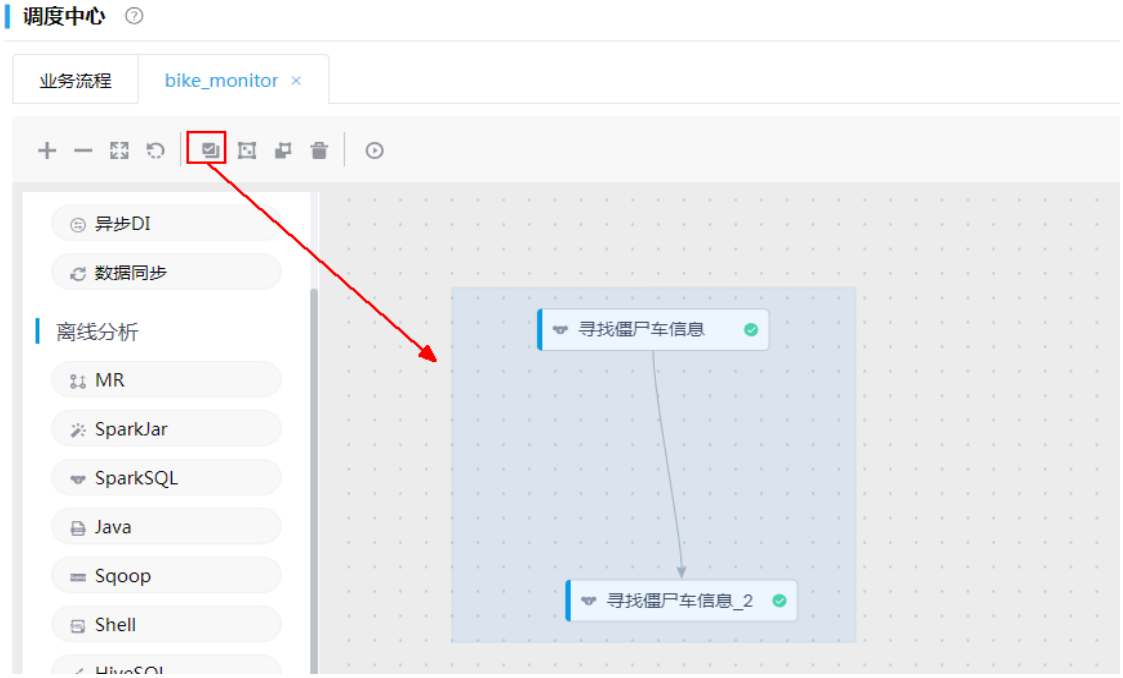
表4-2 业务流程按钮功能说明

符号	名称	说明
	放大	放大画布页面。
	缩小	缩小画布页面。
	画布自适应	画布页面显示采用自适应的方式。
	刷新	提交任务执行之后，可点击该按钮刷新任务状态。
	多选	在业务流程处于可编辑状态时，成组前需要先使用该按钮将多个节点一起选中。
	成组	在业务流程处于可编辑状态时，在选中多个节点之后，点击该按钮将多个节点成一个组，对应一个工作流执行单元，在内部无控制节点的情况下，多个连接接节点将会顺序执行任务。
	解组	当业务流程处于可编辑状态时，选中某个已成组的单元点击该按钮进行解组。
	删除	删除业务流程中的某个节点。
	编辑	对处于执行结束状态的业务流程才有该按钮，点击该按钮，业务流程状态恢复会READY状态。
	启动	启动业务流程。
	停止	停止业务流程。

为多个节点配置调度策略的步骤如下：

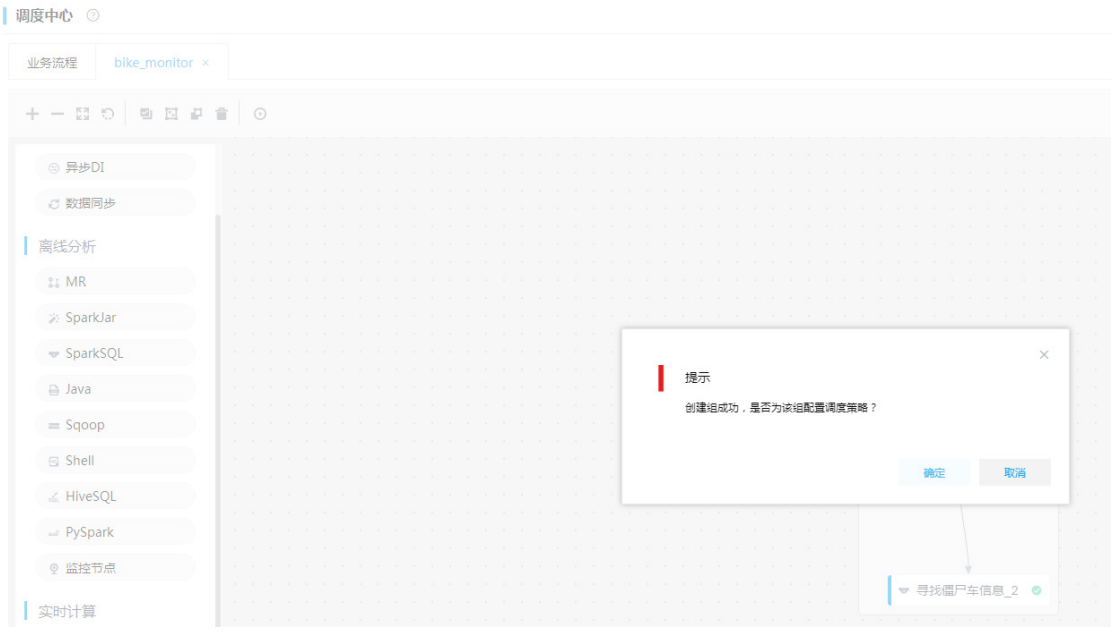
(1) 在业务流程 bike_monitor 处于可编辑状态时，对 2 个 SparkSQL 节点执行“多选”操作，如图 4-16 所示。

图4-16 节点多选



(2) 对选中后的多个节点执行“成组”操作，弹出是否需要添加调度策略的弹框，如图 4-17 所示。

图4-17 节点成组



(3) 单击<确定>按钮，进入调度策略配置页面，如图 4-18 所示。

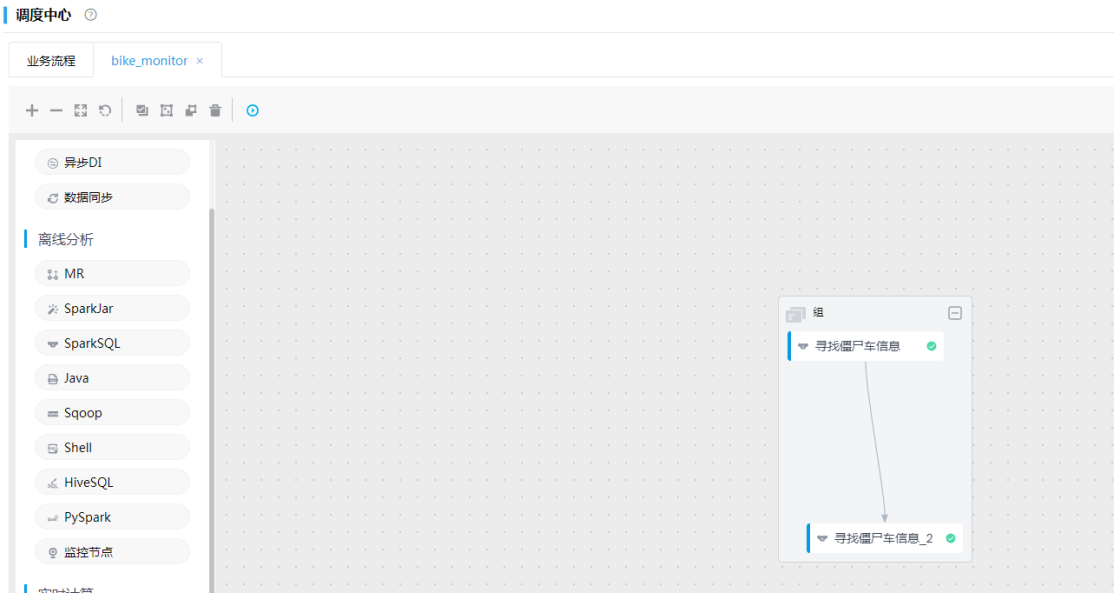
图4-18 节点配置调度信息

(4) 根据提示信息填写调度策略参数。

(5) 调度策略配置完成后，单击<确定>按钮，返回业务流程画布。

(6) 单击画布左上角的图标，启动已配置调度策略的业务流程，如图 4-19 所示。

图4-19 启动业务流程



4.3.6 调度监控

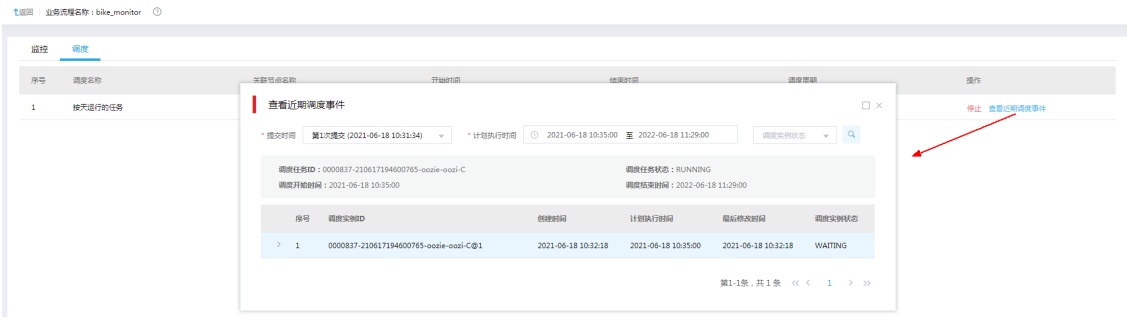
(1) 单击右上角<进入监控页面>按钮，跳转到调度运维界面。

(2) 在页面列表中，单击业务流程操作列中的<调度>按钮，进入调度监控页面，如图 4-20 所示。

图4-20 业务流程调度展示

(3) 在业务流程调度页面中，单击操作列中的<查看近期调度事件>按钮，弹出近期的调度列表，如图 4-21 所示。

图4-21 调度事件查询

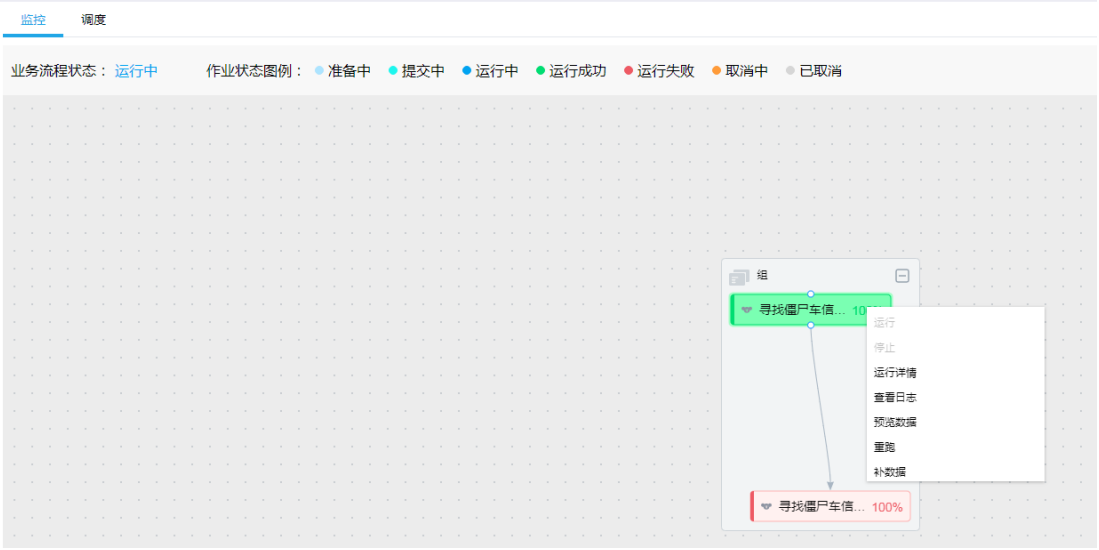


(4) 再次回到监控画布页面，可以看到成组的节点已处于提交中，当时间达到调度设定时间，任务会自动触发执行，如图 4-22 所示。

图4-22 任务流程执行中

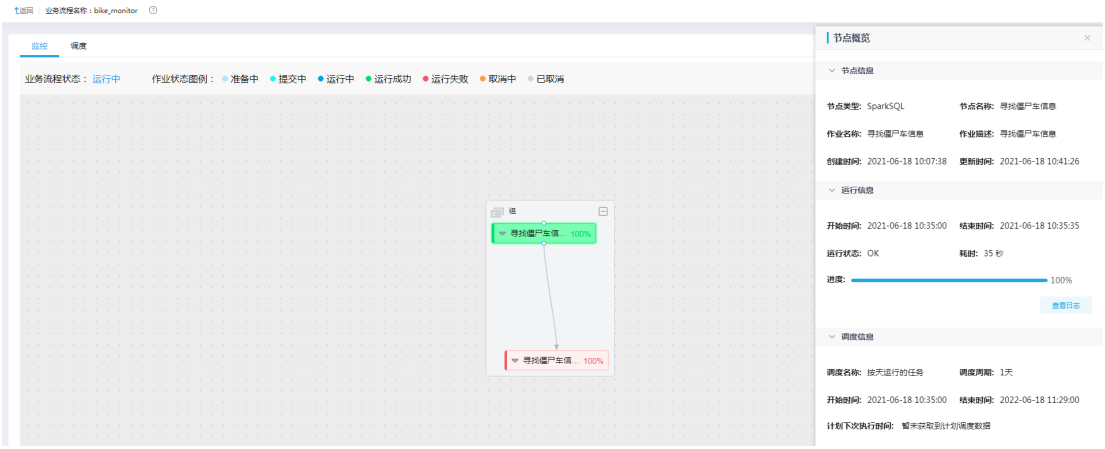
图 4-23 展示了一次调度执行完毕之后节点的状态以及业务流程的状态，业务流程的状态在整个调度周期中都是“运行中”,但其中的各个节点状态会随着周期性的调度状态不断发生变化。图 4-23 中的 2 个节点状态一个运行成功，一个运行失败。

图4-23　业务流程调度过程中状态



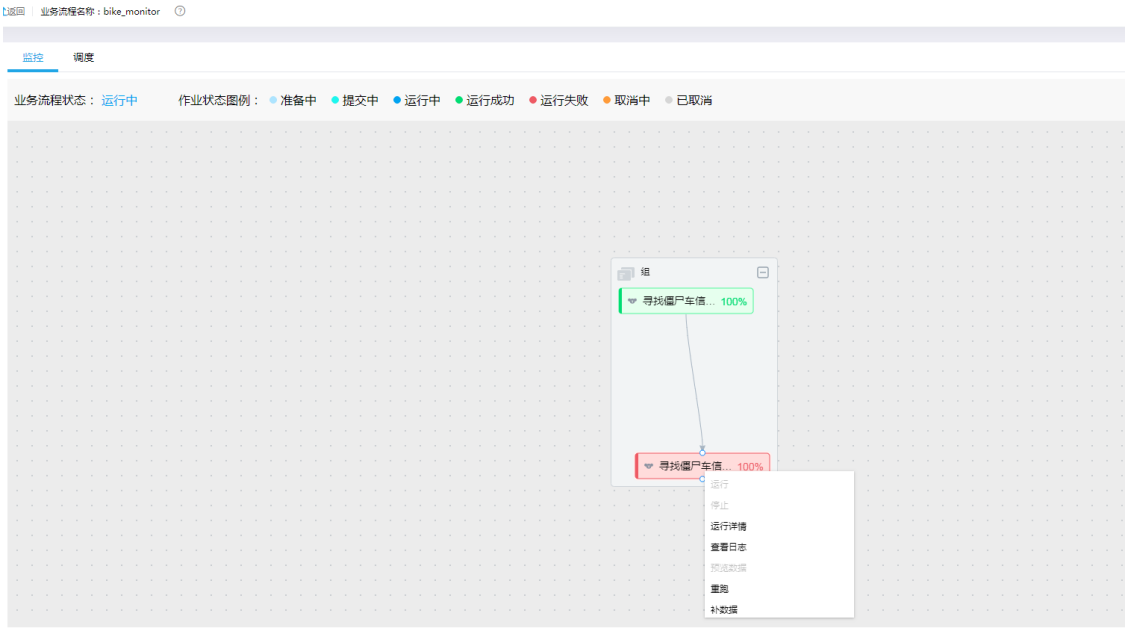
右键单击运行成功的节点，可以查看运行详情和日志，并可以执行预览数据、重跑和补数据操作。选择“运行详情”菜单项，出现图 4-24 所示的右侧弹框。

图4-24 节点调度信息



右键单击运行失败的节点，可以查看运行详情和日志，并可以执行重跑和补数据操作，如图 4-25 所示。

图4-25 调度失败节点信息查询



4.3.7 补录数据

对于执行完毕（成功或失败）的节点均可执行“补数据”操作。以对执行失败的节点“寻找僵尸车信息_2”执行补数据操作为例，步骤如下：

(1) 右键单击该节点，在弹出菜单中选择“补数据”菜单项，弹出补数据窗口，如图 4-26 所示。

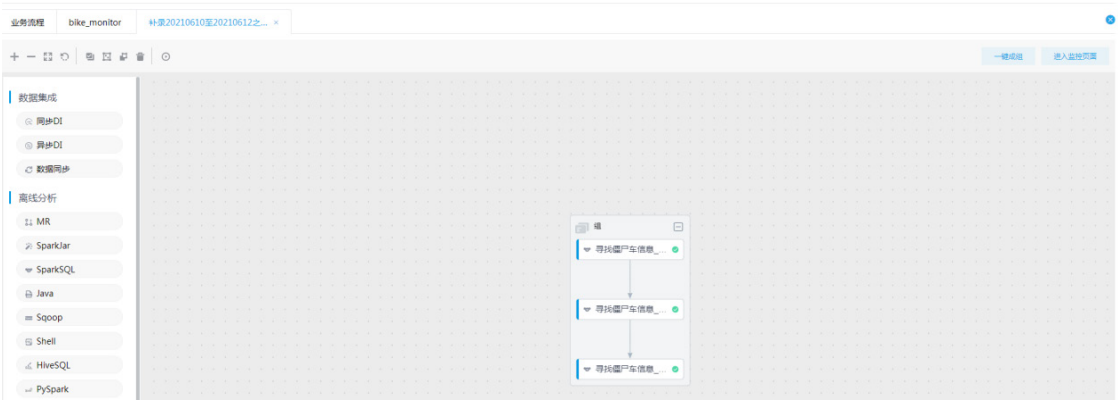
(2) 根据提示配置补数据所需的信息。

(3) 单击<确定>按钮，补数据操作完成。

图4-26 节点补数据

补数据操作会生成新的业务流程，如图 4-27 所示，列表中显示了本次补录数据操作之后自动生成的业务流程，其根据时间范围按照天为单位生成数个成组的 SparkSQL 节点。

图4-27 补数据自动生成业务流程

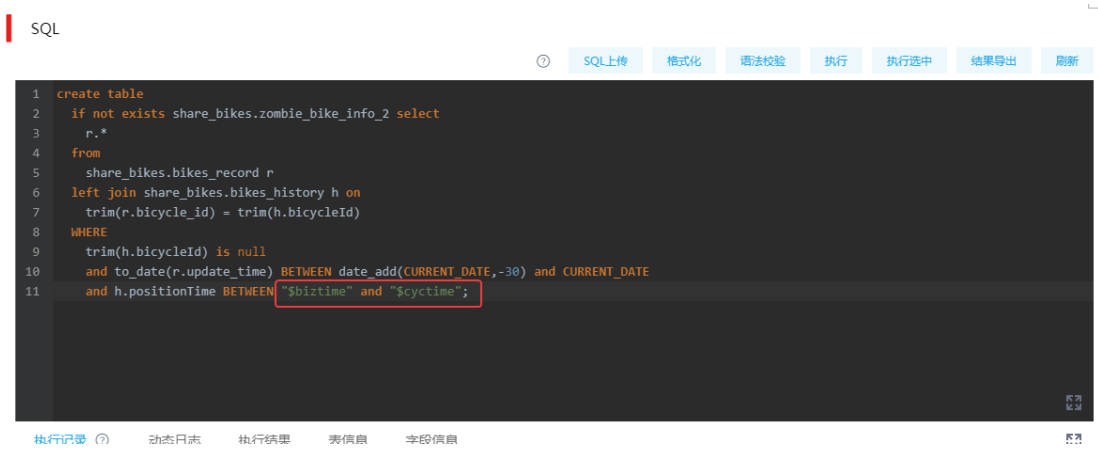


(4) 双击第一个节点，弹出侧边栏，如图 4-28 所示，其中“参数配置”项中的值，表名为补录日期为“2021-06-10”的数据。

图4-28 补数据节点信息展示

需要对应修改 SparkSQL 的 SQL 语句，单击<编辑 SQL>按钮，弹出 SQL 编辑窗口，如图 4-29 所示，即应用 EL 表达式作为指代条件。

图4-29 添加 EL 表达式 SQL



依次修改完这 3 个节点的 SQL 语句并保存后，提交运行。运行过程中，SQL 语句中的$biztime 和$cyctime 将会被动态替换为“参数配置”中的值。以图 4-29 中 SQL 语句为例，其对应实际运行的 SQL 语句如下所示：

```
create table
  if not exists share_bikes.zombie_bike_info_2 select
        r.*
  from
        share_bikes.bikes_record r
  left join share_bikes.bikes_history h on
        trim(r.bicycle_id) = trim(h.bicycleId)
  WHERE
        trim(h.bicycleId) is null
        and to_date(r.update_time) BETWEEN date_add(CURRENT_DATE,-30) and CURRENT_DATE
        and h.positionTime BETWEEN "2021-06-10" and "2021-06-11";
```

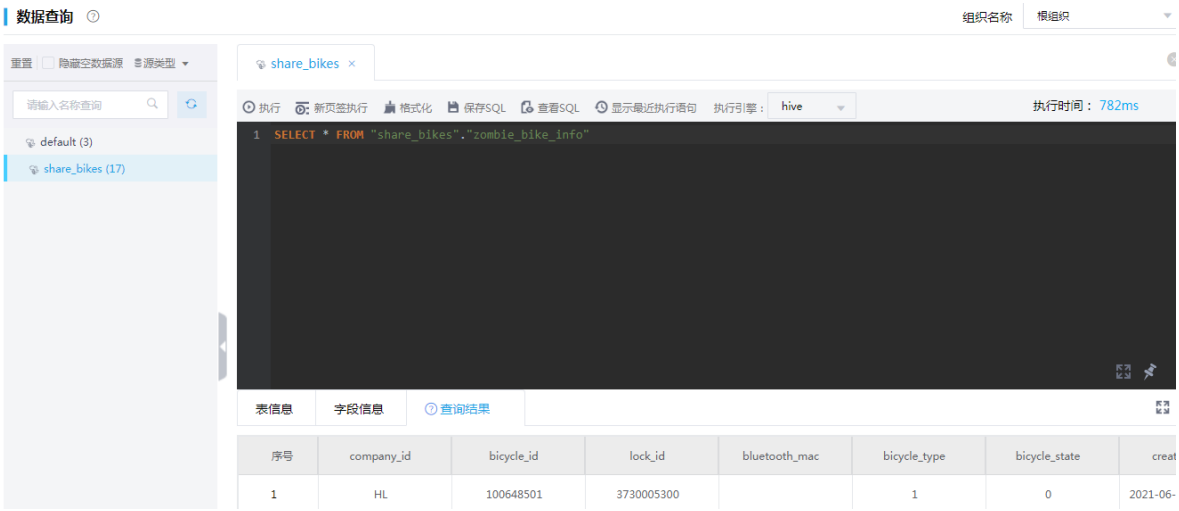
4.4 结果查看

对于未设置“结果导出”操作但带有结果表的 SparkSQL 节点（即 SQL 语句以“create...”或“insert...”等开头）执行成功之后，可以通过[数据开发/数据查询]直接查询，如图 4-30 所示。

由于本例所使用的案例的结果表存于数据源 share_bikes 中，所以可以再执行一次元数据采集任务，将该表采集到元数据信息中，从而可以在[数据查询]中引用到该表名，再对其执行如下命令同样可以查询结果。

```
select * from "share_bikes"."zombie_bike_info"
```

图4-30　任务结果查询


数据查询
组织名称
根组织
重置
隐藏空数据源
源类型
请输入名称查询
default (3)
share_bikes (17)
share_bikes
执行
新页签执行
格式化
保存SQL
查看SQL
显示最近执行语句
执行引擎：
hive
执行时间： 782ms
1 SELECT * FROM "share_bikes"."zombie_bike_info"
表信息
字段信息
查询结果
序号
company_id
bicycle_id
lock_id
bluetooth_mac
bicycle_type
bicycle_state
1
HL
100648501
3730005300
1
0

5 疫苗接种监控案例

5.1 案例说明

在疫情防控中，接种疫苗作为重要的一环，是控制病毒传染扩散的重要手段。随着疫苗的推广，接种人员越来越多，为准确迅速地掌握疫苗接种情况，需要对登记的人员信息、人员类别信息、辖区信息、人员接种信息等进行汇总计算，并将结果提供给大屏展示。

为提供大屏展示所需的数据，需在数据库建立业务数据表（创建业务数据表的参考 SQL 语句请参见 7.2 疫苗接种案例业务数据库建表语句示例），记录原始数据，并对这些数据进行处理和计算，得出如下 4 个指标：

- 行业接种统计数据
- 社区街道接种统计数据
- 各年龄段接种统计数据
- 一针接种至今各个时间段接种人数统计

主要步骤及说明如下：

(1) 本例中的数据来源于业务库中的原始数据，而为了保证这些原始数据信息不受影响，需要通过 DI 将业务库中的原始数据（存量和增量）抽取至数字平台的 ODS（Operation Data Store）层中，作为基础数据。

(2) 将 ODS 层中保存的基础数据，在数据管理平台中注册为数据源，以便后续步骤中使用。

(3) 在数据管理平台中，创建对应基础数据的表，表的结构需要与基础数据存储表的结构一致，使系统能够正确读取识别基础数据。此外，还需创建存放清洗后数据的表和统计结果的表，以便在后续步骤中使用。

(4) 在数据管理平台中，对基础数据进行清洗处理，并将清洗后的数据存放至专用的数据表中。

(5) 在数据管理平台中，对清洗后的数据进行计算等处理，得出统计结果数据，并存放至预先准备好的表中。

(6) 对统计结果数据进行查询验证，无误后即可通过集成平台的服务集成进行发布和授权。第三方应用可以调用数据结果用于大屏展示等。

5.2 准备操作

5.2.1 抽取基础数据

基础数据的抽取需要通过集成平台的数据集成服务完成，涉及在集成平台-数据集成中执行，完成对基础数据的全量抽取及增量抽取。简要说明如下：

- 注册 MySQL 数据源（用户业务库，存储基础数据），注册 HDFS 数据源（承载从用户业务库中抽取的基础数据）。
- 创建 ETL 任务将用户业务库（MySQL）中的数据抽取至 HDFS 中的 Hive 数据文件中（ODS 层数据）。
- 创建 DI 作业运行 ETL 任务。

详细操作配置请参见《UniCloud 集成平台 用户手册 E6101》中的疫苗接种监控案例。

5.2.2 创建数据源

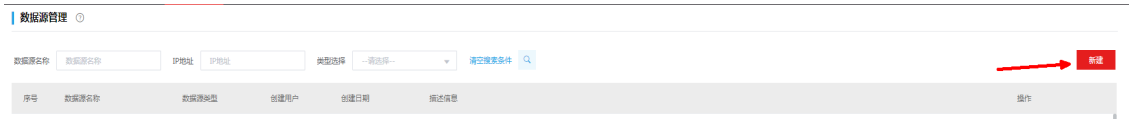
在数据管理中需创建 Hive 数据源和 Greenplum 数据源，其中：

- Hive 数据源：将前序步骤中，集成平台的 HDFS 数据源中承载数据的 Hive 数据文件，作为 ODS 层数据源。
- Greenplum 数据源：用于存放进行计算处理后的结果数据，作为 DWS 层数据源。

1. 创建 Hive 数据源

(1) 在[数据管理平台/数据源管理]模块中，单击右上角<创建>按钮，进行数据源的创建操作，如图 5-1 所示。

图5-1 数据源配置页面

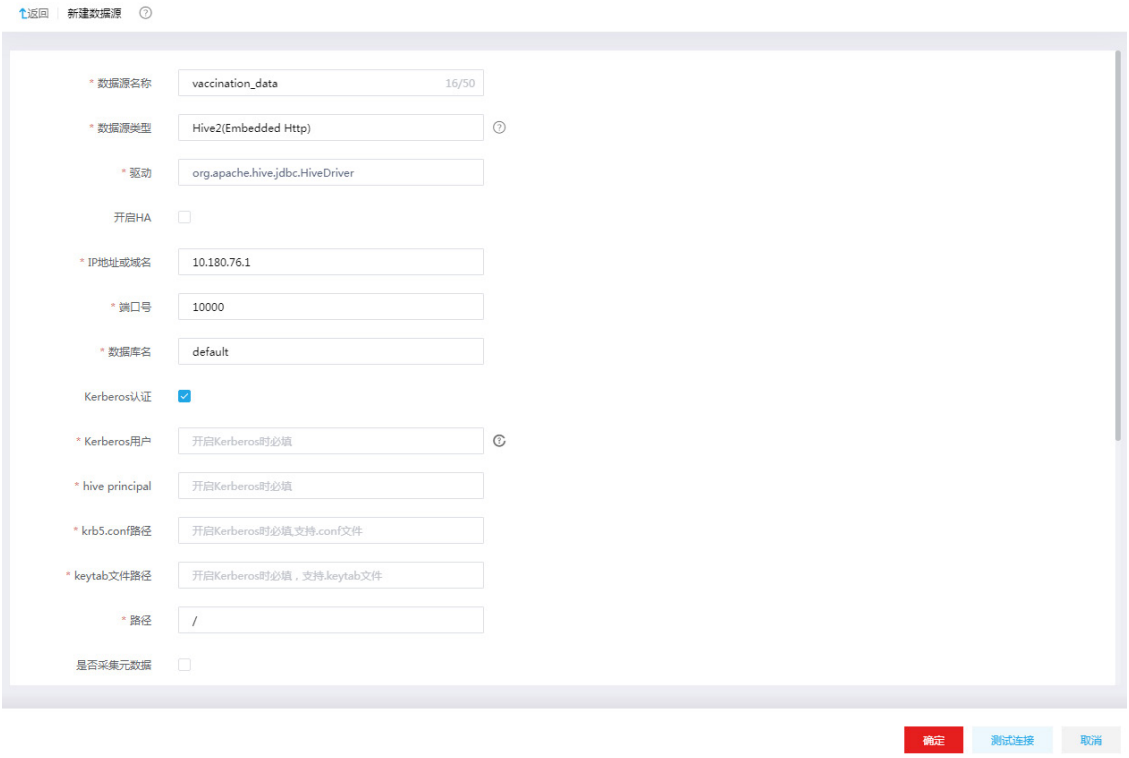


(2) 选择 Hive 数据源，并配置参数，如图 5-2 所示。其中：

- Kerberos 用户等信息可以在数据管理平台所使用的大数据平台管理页面中查看。
- hive principal 参数格式为：hive/*IP 地址对应节点的主机名@集群名称大写*.COM。
- krb5.conf 和 Keytab 文件为 Kerberos 认证文件，需要从大数据平台中的集群管理页面下载。

(3) 在大数据平台获得 Kerberos 的相关信息和认证文件后，返回数据管理平台的新建数据源页面，如图 5-2 所示，填写各个 Kerberos 参数并上传所需的认证文件即可。

图5-2 新增 Hive 数据源



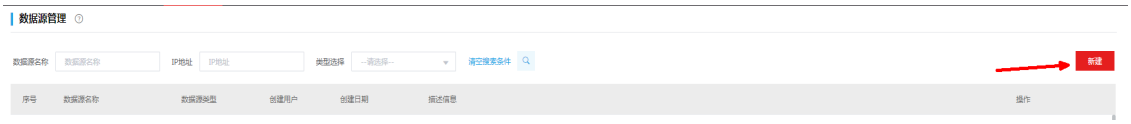
(4) 填写完毕注册数据源所需要的信息之后，可以单击<测试连接>按钮，测试数据源连通性。

(5) 提示“连接测试成功”信息，单击<确定>按钮，执行注册数据源。之后即可在数据源列表中看到注册成功的数据源概要信息。

2. 创建 Greenplum 数据源

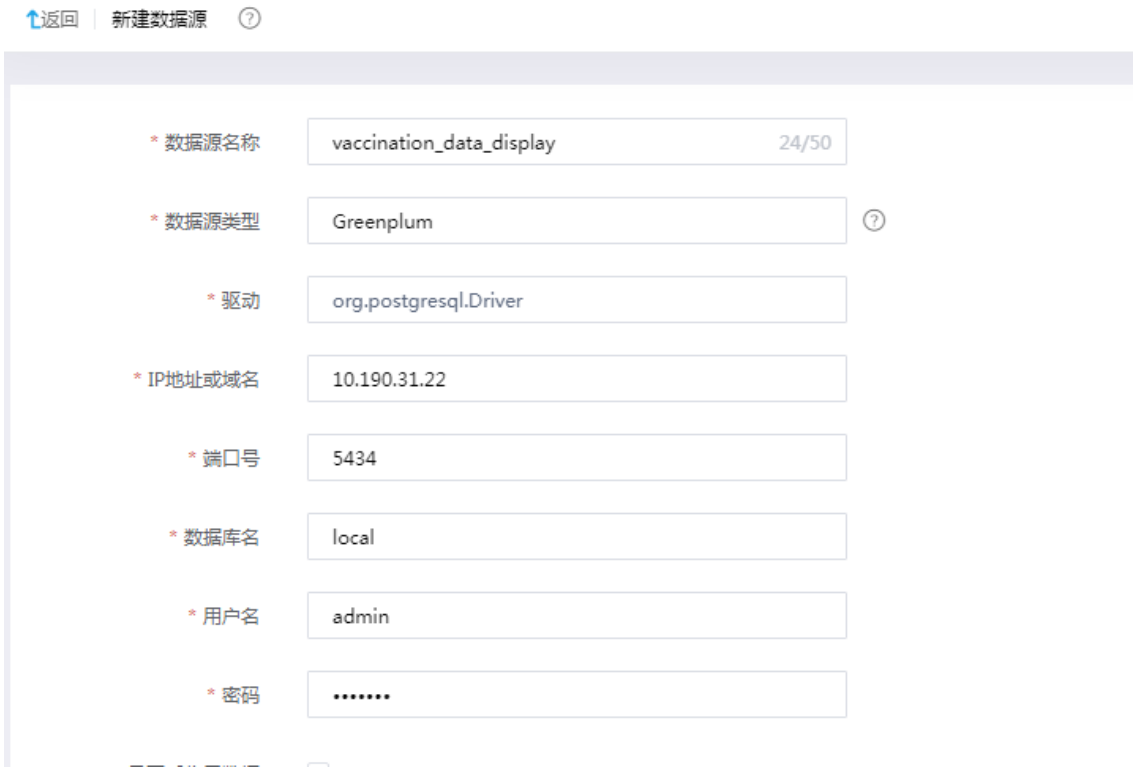
(1) 在[数据管理平台/数据源管理]模块中，单击右上角<创建>按钮，进行数据源的创建操作，如图 5-3 所示。

图5-3 数据源配置页面



(2) 选择 Greenplum 数据源，并配置参数，如图 5-4 所示。

图5-4 新增 Greenplum 数据源



(3) 填写完毕注册数据源所需要的信息之后，可以单击<测试连接>按钮，测试数据源连通性。

(4) 提示“连接测试成功”信息，单击<确定>按钮，执行注册数据源。之后即可在数据源列表中看到注册成功的数据源概要信息。

5.2.3 新建数据表

本例中，需要根据 Hive 数据源中的基础数据新建对应的数据表，并提前创建针对人员接种信息数据的清洗表以及后续存储数据处理结果的各结果表。

1. 新建基础信息表

为使系统能够正确识别 Hive 数据源中的基础数据，并检测到数据源的表结构，方便后续业务流程中作业的 SQL 处理，需要对应 Hive 数据源中的基础信息数据表在[数据管理平台/数据开发]的表管理中新建数据表，这些表为 ODS 层的表。

(1) 在[数据管理平台/数据开发]模块中，选择左侧导航树中的[表管理]菜单项，进入表管理页面。

(2) 在页面右上角选择组织，本例中选择“根组织”。

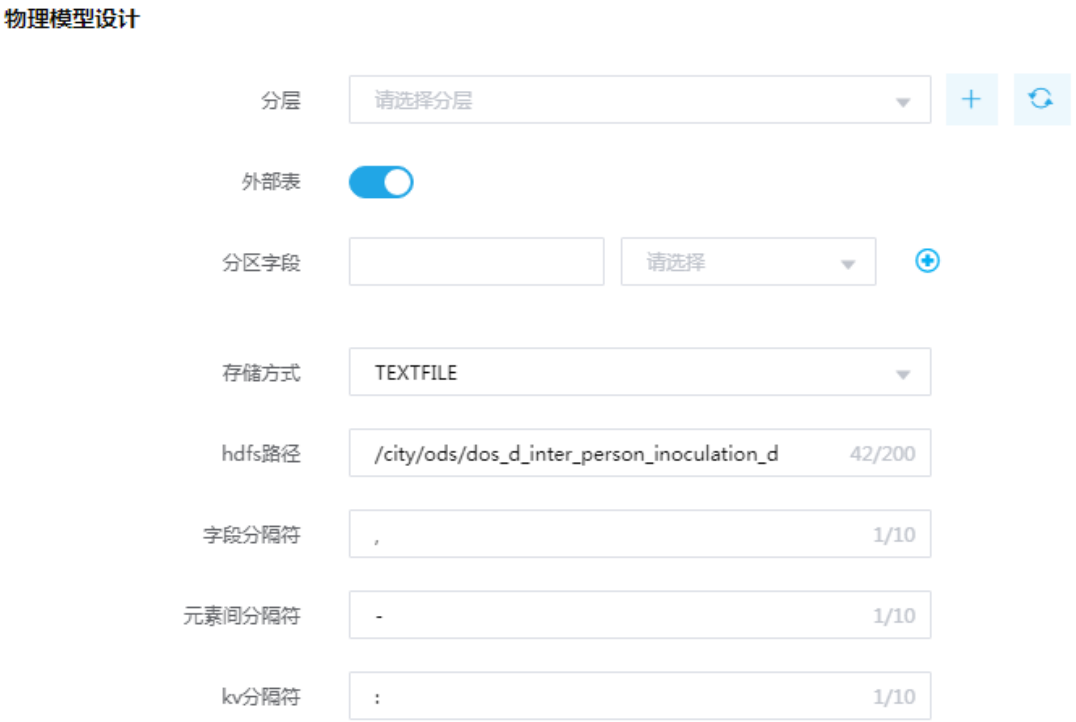
(3) 单击左上角的<新建>按钮，进入新建表页面。

(4) 选择 Hive 数据源类型，并选择 5.2.2 创建数据源中创建的数据源。

(5) 配置表名等基本属性参数和物理模型设计参数。其中，表名根据实际情况配置，本例中为“ods_d_inter_person_inoculation_d”（人员接种信息）；物理模型设计的“外部表”参数需设为  状态，并指定 Hive 数据源中数据文件存放的 HDFS 路径。

图5-5 基本属性配置

图5-6 物理模型设计配置



(6) 单击<下一步>按钮，进入数据结构配置页面。

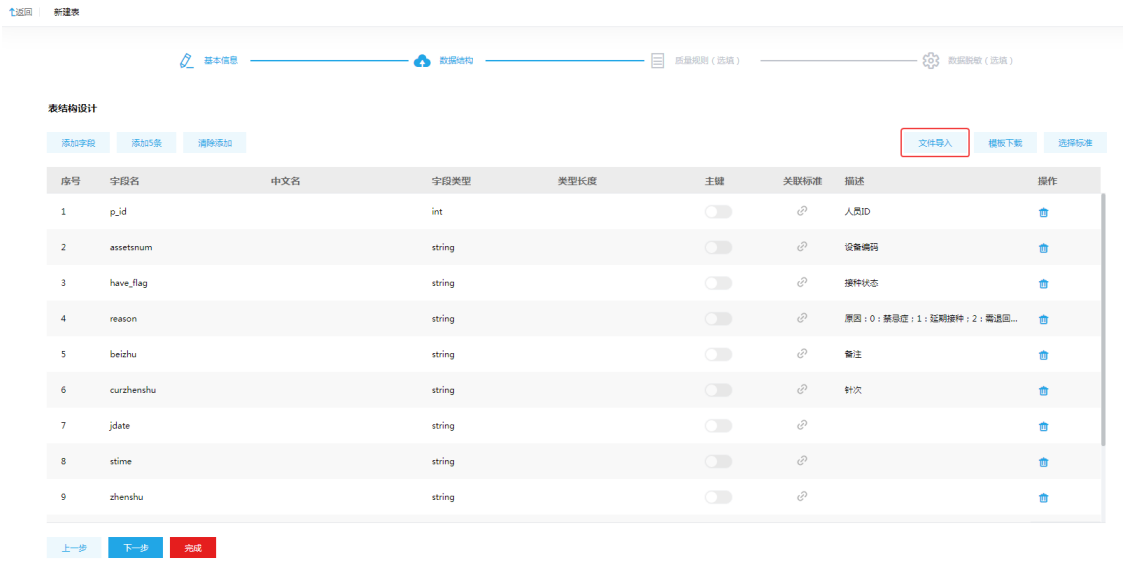
(7) 在数据结构配置页面中，可通过<模板下载>按钮下载模板，然后填入字段等信息，再通过<文件导入>按钮进行导入。人员接种信息表示例字段如表 5-1 所示，通过文件导入后如图 5-7 所示。

表5-1 人员接种信息表示例字段信息

字段名称	字段类型	描述
p_id	int	人员ID

assetsnum	string	设备编码
have_flag	string	接种状态
reason	string	原因：0：禁忌症；1：延期接种；2：需退回上级重新分配；3：不符合接种人群范围
beizhu	string	备注
curzhenshu	string	针次
jdate	string	-
stime	string	-
zhenshu	string	-
yimiao	string	疫苗种类
jinjizheng	string	禁忌症
created	string	接种时间

图5-7 表结构设计



(8) 单击<完成>按钮，表新建完成。

(9) 重复步骤(2)-步骤(8)，依次新建人员信息表（字段如表 5-2 所示）、辖区（街道）字典表（字段如表 5-3 所示）、人员分类字典表（字段如表 5-4 所示）。

表5-2 人员信息表示例字段信息

字段名称	字段类型	描述
id	int	序号
name	string	姓名
sex	string	性别
age	int	年龄
mobile	string	手机号
cardno	string	身份证号
classfication_id	int	人员分类一级
content	string	备注
company_id	int	单位ID
region_id	int	辖区ID对应属地的摸底工作部门
declare_department_id	int	申报部门ID
created_time	string	创建时间
modified_time	string	修改时间
uuid	string	UUID
addr	string	现住址
streetId	string	街道
provinceid	string	省ID
cityid	string	市ID
disctrictid	string	区域ID
flag	string	是否本市住户
area	string	小区名字
subclass_id	int	人群分类二级

表5-3 辖区（街道）字典表示例字段信息

字段名称	字段类型	描述
id	int	序号

region	string	辖区
streetId	string	街道
userid	string	User_id
category	string	有没有二级分类（1表示有）
sort	int	顺序编号

表5-4 人员分类字典表示例字段信息

字段名称	字段类型	描述
id	int	序号
classsfication	string	人员分类
created_time	string	创建时间
modified_time	string	修改时间
category	string	级别

2. 新建人员接种信息数据清洗表

在基础的人员接种信息表中，可能存在错误或不完整的数据，为保证后续的数据处理可以正常进行，需要对基础信息表中的人员接种信息表进行清洗处理。人员接种信息数据清洗表即用于存放清洗后的人员接种信息数据，需要在数据清洗操作执行前，先新建该表。该表结构与基础信息表中的人员接种信息表相同。该表为 DWB 层的表。

(1) 在[数据管理平台/数据开发]模块中，选择左侧导航树中的[表管理]菜单项，进入表管理页面。

(2) 在页面右上角选择组织，本例中选择“根组织”。

(3) 单击左上角的<新建>按钮，进入新建表页面。

(4) 选择 Hive 数据源类型，并选择 5.2.2　创建数据源中创建的的数据源。

(5) 配置表名等基本属性参数和物理模型设计参数。其中，表名根据实际情况配置，本例中为“dwb_filtered_person_inoculation_d”（人员接种信息数据清洗表）；物理模型设计的“外部表”参数需设为  状态，将该表设置为内部表，以便于管理和使用。

(6) 单击<下一步>按钮，进入数据结构配置页面。

(7) 在数据结构配置页面中，表的字段信息与人员接种信息表示例字段一致，如表 5-1 所示，可通过文件导入，导入后如图 5-7 所示。

(8) 单击<完成>按钮，表新建完成。

3. 新建结果表

为便于存储数据处理后的结果数据，需要先新建各结果数据表，这些表为 DWS 层的表。

(1) 在[数据管理平台/数据开发]模块中，选择左侧导航树中的[表管理]菜单项，进入表管理页面。

(2) 在页面右上角选择组织，本例中选择“根组织”。

(3) 单击左上角的<新建>按钮，进入新建表页面。

(4) 选择 Greenplum 数据源类型，并选择 5.2.2 创建数据源中创建的数据源。

(5) 配置表名等基本属性参数和物理模型设计参数。其中，表名根据实际情况配置，本例中为“dws_region_inoculation_day_statistics”（社区街道疫苗接种按天统计结果表）；物理模型设计部分，存储模式使用默认值“row”，模式选择 public。

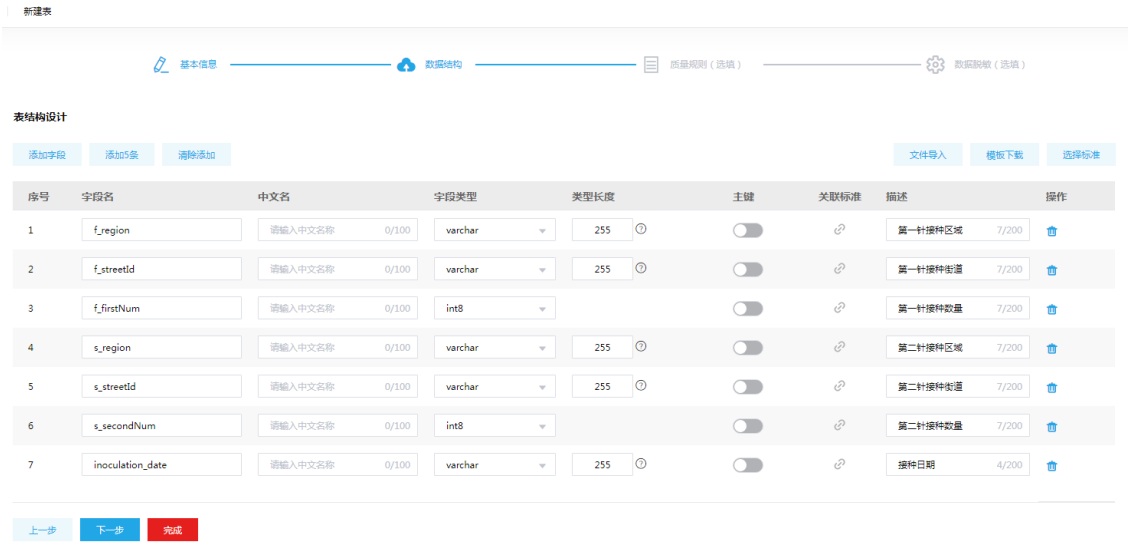
(6) 单击<下一步>按钮，进入数据结构配置页面。

(7) 在数据结构配置页面中，表的字段信息与人员接种信息表示例字段一致，如所示，可通过文件导入后如所示。

表5-5 社区街道疫苗接种按天统计结果表字段信息

字段名称	字段类型	描述
f_region	varchar(255)	第一针接种区域
f_streetId	varchar(255)	第一针接种街道
f_firstNum	int8	第一针接种数量
s_region	varchar(255)	第二针接种区域
s_streetId	varchar(255)	第二针接种街道
s_secondNum	int8	第二针接种数量
inoculation_date	varchar(255)	接种日期

图5-8 表结构设计



(8) 单击<完成>按钮，表新建完成。

(9) 重复步骤(2)-步骤(8)，依次新建社区街道疫苗接种全量统计结果表（字段信息如表 5-6 所示）、行业疫苗接种按天统计结果表（字段信息如表 5-7 所示）、行业疫苗接种全量统计结果表（字段信息如表 5-8 所示）、各年龄段疫苗接种按天统计结果表（字段信息如表 5-9 所示）、各年龄段疫苗接种全量统计结果表（字段信息如表 5-10 所示）、第一针接种至今各个时间间隔人数统计结果表（字段信息如表 5-11 所示）。

表5-6 社区街道疫苗接种全量统计结果表字段信息

字段名称	字段类型	描述
f_region	varchar(255)	第一针接种区域
f_streetId	varchar(255)	第一针接种街道
first_total_num	int8	第一针接种数量
s_region	varchar(255)	第二针接种区域
s_streetId	varchar(255)	第二针接种街道
second_total_num	int8	第二针接种数量

表5-7 行业疫苗接种按天统计结果表字段信息

字段名称	字段类型	描述
f_classification	varchar(255)	第一针行业分类
firstNum	int8	第一针接种数量
s_classification	varchar(255)	第二针行业分类
secondNum	int8	第二针接种数量
inoculation_date	varchar(255)	接种日期

表5-8 行业疫苗接种全量统计结果表字段信息

字段名称	字段类型	描述
f_classification	varchar(255)	第一针行业分类
firstNum	int8	第一针接种数量
s_classification	varchar(255)	第二针行业分类
secondNum	int8	第二针接种数量

表5-9 各年龄段疫苗接种按天统计结果表字段信息

字段名称	字段类型	描述
f_ageRange	varchar(255)	第一针年龄段
firstNum	int8	第一针接种数量
s_ageRange	varchar(255)	第二针年龄段
secondNum	int8	第二针接种数量
inoculation_date	varchar(255)	接种日期

表5-10 各年龄段疫苗接种全量统计结果表字段信息

字段名称	字段类型	描述
f_ageRange	varchar(255)	第一针年龄段
first_total_num	int8	第一针接种数量
s_ageRange	varchar(255)	第二针年龄段
second_total_num	int8	第二针接种数量

表5-11 第一针接种至今各个时间间隔人数统计结果表字段信息

字段名称	字段类型	描述
interval_period	varchar(255)	时间间隔
person_num	int8	人数

5.3 构建业务流程

准备工作完成后，即可开始构建业务流程，包括创建业务流程，并在业务流程画布中增加数据清洗作业和各类数据计算作业。

5.3.1 创建业务流程

(1) 在[数据管理平台/数据开发]模块中，选择左侧导航树中的[调度中心]菜单项，进入调度中心页面。

(2) 在页面右上角选择组织，本例中选择“根组织”。

(3) 单击左上角的<新建>按钮，进入新建业务流程。

(4) 输入业务流程名称和描述信息，本例中名称为“疫苗接种数据统计”。

(5) 单击<确定>按钮，业务流程创建成功，页面进入该业务流程的画布编辑页签。

(6) 将左侧的作业组件拖入画布中，生成业务流程中的作业节点。双击该作业节点，可在弹窗中配置节点参数。

本例中需要增加人员接种数据清洗作业和各数据计算作业。

5.3.2 添加数据清洗作业

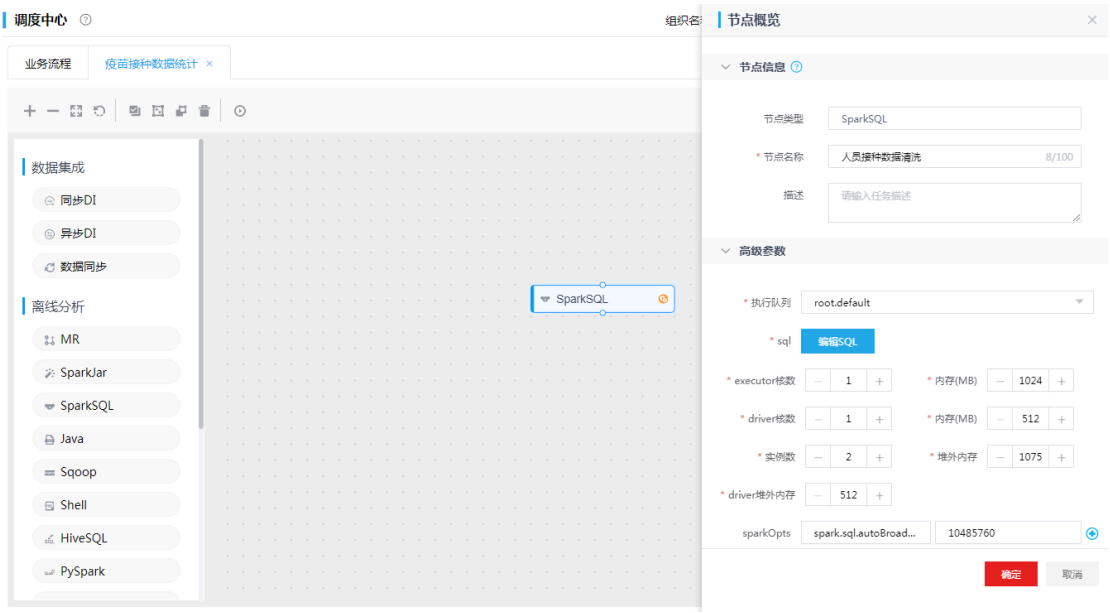
业务流程创建后，需要在业务流程的画布中增加人员接种数据清洗作业。

(1) 在业务流程的画布编辑页签中，选择左侧离线分析下的 SparkSQL 组件，并拖入画布中。

(2) 双击画布中的 SparkSQL 作业节点，弹出作业节点参数编辑窗口。

(3) 本例中，配置节点名称为“人员接种数据清洗”，选择执行队列为缺省队列。

图5-9 配置节点参数



(4) 单击<编辑 SQL>按钮，在弹出框中编写 SQL 语句，示例如下：

```
insert into
   default.dwb_filtered_person_inoculation_d
select
   *
from
   default.ods_d_inter_person_inoculation_d
where
   p_id is not null
   and haveflag is not null
   and curzhenshu is not null
   and created is not null;
```

图5-10 配置 SQL 语句

(5) 编写完成，并通过语法校验后，单击<确定>按钮，保存 SQL 语句。

(6) 单击<确定>按钮，数据清洗作业节点配置完成。

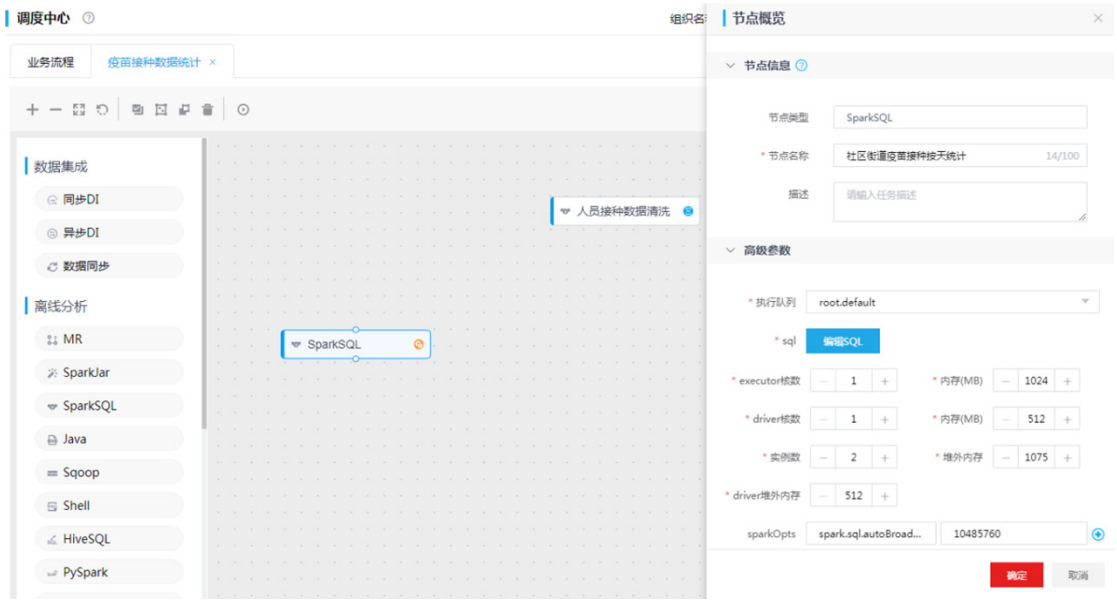
5.3.3 添加数据计算作业

(1) 在业务流程的画布编辑页签中，选择左侧离线分析下的 SparkSQL 组件，并拖入画布中。

(2) 双击画布中的 SparkSQL 作业节点，弹出作业节点参数编辑窗口。

(3) 本例中，配置节点名称为“社区街道疫苗接种按天统计”，选择执行队列为缺省队列。

图5-11 配置节点参数



(4) 单击<编辑 SQL>按钮，在弹出框中编写 SQL 语句，示例如下：

```
select
    f.region as f_region,
```

```
    f.streetId as f_streetId,
    f.firstNum as f_firstNum,
    s.region as s_region,
    s.streetId as s_streetId,
    s.secondNum as s_secondNum,
    f.f_inoculation_date as inoculation_date
from
    (
        select
            case
                when rd.region is not null then rd.region
                else '无区域'
            end as region,
            case
                when p.streetId is not null then p.streetId
                else '无街道'
            end as streetId,
            pi.f_inoculation_date,
            count(pi.p_id) as firstNum
        from
            (select *, substring_index(created, ' ', 1) as f_inoculation_date from
default.dwb_filtered_person_inoculation_d) pi
            left join ods_d_inter_person_d p on pi.p_id = p.id
            left join ods_d_inter_region_dict_d rd on rd.id = p.region_id
        where
            pi.haveflag = 'true'
            and curzhenshu = '0'
        group by
            rd.region,
            p.streetId,
            pi.f_inoculation_date
    ) f full
    join (
        select
            case
                when rd.region is not null then rd.region
                else '无区域'
            end as region,
```

```
        case
          when p.streetId is not null then p.streetId
          else '无街道'
        end as streetId,
        pi.s_inoculation_date,
        count(pi.p_id) as secondNum
      from
        (select *, substring_index(created, ' ', 1) as s_inoculation_date from
default.dwb_filtered_person_inoculation_d) pi
        left join ods_d_inter_person_d p on pi.p_id = p.id
        left join ods_d_inter_region_dict_d rd on rd.id = p.region_id
      where
        pi.haveflag = 'true'
        and curzhenshu = '1'
      group by
        rd.region,
        p.streetId,
        pi.s_inoculation_date
    ) s on f.region = s.region
    and f.streetId = s.streetId and f.f_inoculation_date = s.s_inoculation_date;
```

(5) 编写完成，并通过语法校验后，单击<确定>按钮，保存 SQL 语句。

(6) 单击<确定>按钮，社区街道疫苗接种按天统计结果节点配置完成。

(7) 依次增加其他数据计算作业，各作业使用的 SQL 语句如所示。

表5-12 各数据计算作业使用的 SQL 语句

作业	SQL 语句
社区街道疫苗接种全量统计	select f.f_region as f_region, f.f_streetId as f_streetId, f.first_toal_num as first_total_num, s.s_region as s_region, s.s_streetId as s_streetId, s.second_toal_num as second_total_num from (select f_region, f_streetId, sum(f_firstNum) as first_toal_num from default.dws_region_inoculation_day_statistics group by f_region, f_streetId) as f full join (select s_region, s_streetid, sum(s_secondNum) as second_toal_num from default.dws_region_inoculation_day_statistics group by s_region, s_streetid) as

作业	SQL 语句
	s on f.f_region = s.s_region and f.f_streetId = s.s_streetId ;
行业疫苗接种按天统计	select f.classification as f_classification, f.firstNum as firstNum, s.classification as s_classification, s.secondNum as secondNum, f.f_inoculation_date as inoculation_date from(select case when pc.classification is not null then pc.classification else '未分类人员' end as classification, pi.f_inoculation_date as f_inoculation_date, count(pi.p_id) as firstNum from (select *, substring_index(created, ' ', 1) as f_inoculation_date from default.dwb_filtered_person_inoculation_d) pi left join ods_d_inter_person_d p on pi.p_id = p.id left join ods_d_inter_person_classification_d pc on pc.id = p.classification_id where pi.haveflag = 'true' and pi.curzhenshu = '0' group by pc.classification, pi.f_inoculation_date) f full join (select case when pc.classification is not null then pc.classification else '未分类人员' end as classification, pi.s_inoculation_date as s_inoculation_date,

作业	SQL 语句
	count(pi.p_id) as secondNum from (select *, substring_index(created, ' ', 1) as s_inoculation_date from default.dwb_filtered_person_inoculation_d) pi left join ods_d_inter_person_d p on pi.p_id = p.id left join ods_d_inter_person_classification_d pc on pc.id = p.classification_id where pi.haveflag = 'true' and pi.curzhenshu = '1' group by pc.classification, pi.s_inoculation_date) s on f.classification = s.classification and f.f_inoculation_date = s.s_inoculation_date;
行业疫苗接种全量统计	select f.f_classification as f_classification, f.first_toal_num as first_toal_num, s.s_classification as s_classification, s.second_toal_num as second_toal_num from (select f_classification, sum(firstNum) as first_toal_num from default.dws_classfication_inoculation_day_statistics group by f_classification) as f full join (select s_classification,

作业	SQL 语句
	sum(secondNum) as second_toal_num from default.dws_classfication_inoculation_day_statistics group by s_classification) as s on f.f_classification = s.s_classification;
各年龄段疫苗接种按天统计	select f.age_range as f_ageRange, f.firstNum as firstNum, s.age_range as s_ageRange, s.secondNum as secondNum, f.f_inoculation_date as inoculation_date from (select p.age_range as age_range, pi.f_inoculation_date as f_inoculation_date, count(pi.p_id) as firstNum from (select *, substring_index(created, ' ', 1) as f_inoculation_date from default.dwb_filtered_person_inoculation_d) pi left join (select case when age <= 18 then '18岁以下' when age <= 59 and age > 18 then '18岁至59岁'

作业	SQL 语句
	when age > 59 then '大于59岁' else '未找到年龄信息' end as age_range, id from ods_d_inter_person_d) p on pi.p_id = p.id where pi.haveflag = 'true' and pi.curzhenshu = '0' group by p.age_range, pi.f_inoculation_date) f full join (select p.age_range as age_range, pi.s_inoculation_date as s_inoculation_date, count(pi.p_id) as secondNum from (select *, substring_index(created, ' ', 1) as s_inoculation_date from default.dwb_filtered_person_inoculation_d) pi left join (select case when age <= 18 then '18岁以下' when age <= 59

作业	SQL 语句
	and age > 18 then '18岁至59岁' when age > 59 then '大于59岁' else '未找到年龄信息' end as age_range, id from ods_d_inter_person_d) p on pi.p_id = p.id where pi.haveflag = 'true' and pi.curzhenshu = '1' group by p.age_range, pi.s_inoculation_date) s on s.age_range = f.age_range and f.f_inoculation_date = s.s_inoculation_date;
各年龄段疫苗接种全量统计	select f.f_ageRange as f_ageRange, f.first_total_num as first_total_num, s.s_ageRange as s_ageRange, s.second_total_num as second_total_num from (select f_ageRange, sum(firstNum) as first_total_num from default.dws_age_range_inoculation_day_statistics group by f_ageRange) as f full join (select s_ageRange,

作业	SQL 语句
	sum(secondNum) as second_total_num from default.dws_age_range_inoculation_day_statistics group by s_ageRange) as s on f.f_ageRange = s.s_ageRange;
第一针接种至今各个时间间隔人数统计	select case when datediff(date_format(CURRENT_DATE, 'yyyy-MM-dd'), replace(substring_index(created, ' ', 1), '/', '-')) <= 21 then '三周以内' when datediff(date_format(CURRENT_DATE, 'yyyy-MM-dd'), replace(substring_index(created, ' ', 1), '/', '-')) > 21 and datediff(date_format(CURRENT_DATE, 'yyyy-MM-dd'), replace(substring_index(created, ' ', 1), '/', '-')) <= 56 then '三周至八周' else '超过八周' end as interval_period, count(p_id) as person_num from default.dwb_filtered_person_inoculation_d where curzhenshu = 0 and haveflag = 'true' group by interval_period;

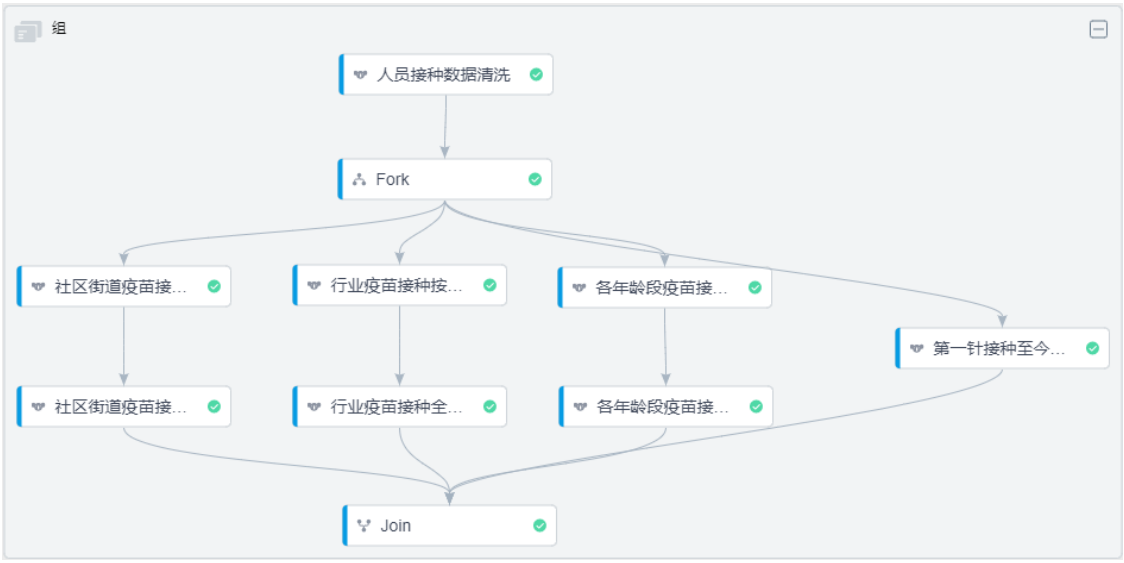
5.3.4 构建完成作业并运行

各作业创建完成后，需要通过控制节点下的 Fork 组件和 Join 组件进行连接，构建完整的业务流程。

(1) 在业务流程的画布编辑页签中，选择左侧控制节点下的 Fork 组件和 Join 组件，并拖入画布中。

(2) 依次连接各作业，连接结果如图 5-12 所示。

图5-12 关联作业



(3) 连接完成后，即可单击画布上方的 按钮，运行该业务流程。针对业务流程中的各作业，可配置调度策略，使作业定期自动运行。

5.4 数据查询

当业务流程运行完成后，统计结果数据会存入预先创建的统计结果表中。

数据管理平台中提供了数据查询功能，可查询统计结果数据。

(1) 在[数据管理平台/数据开发]模块中，选择左侧导航树中的[数据查询]菜单项，进入数据查询页面。

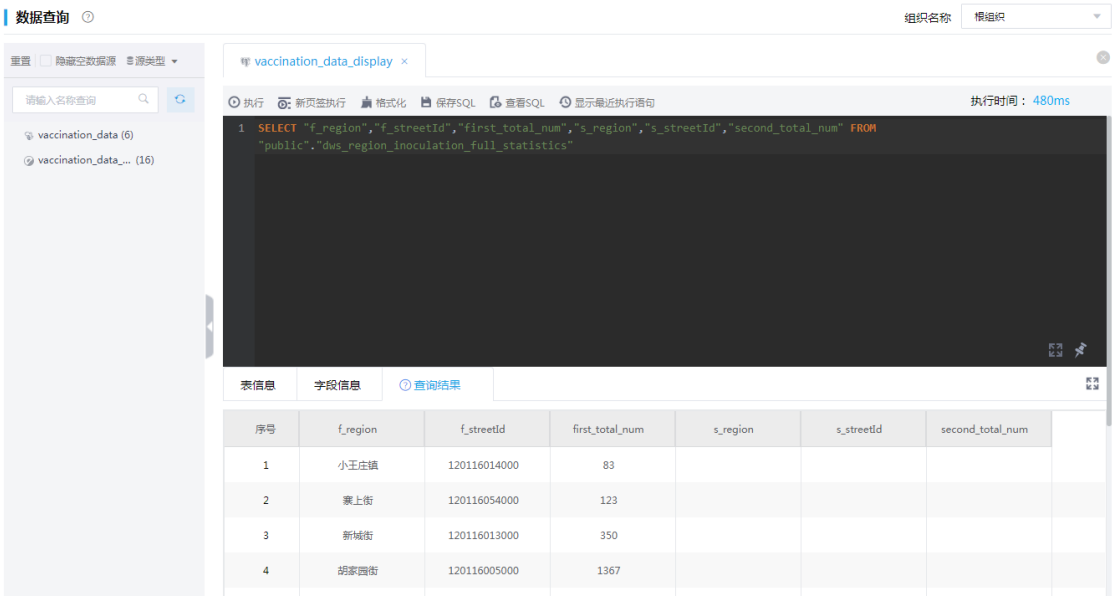
(2) 在左侧目录中选择数据源，右侧出现该数据源的数据查询页签。

(3) 在输入框中输入 SQL 查询语句，查看各统计结果表中的数据。示例语句如下：

```
SELECT "f_region","f_streetId","first_total_num","s_region","s_streetId","second_total_num"
FROM "public"."dws_region_inoculation_full_statistics"
```

(4) 单击<执行>按钮，执行该查询语句，下方的查询结果页签中，会展示该统计结果表中的数据。

图5-13 查询结果



5.5 结果数据发布

通过数据计算得出的统计数据存入了统计结果表中，数字平台支持以表为单位，在集成平台的服务集成功能中，将 Greenplum 数据源中的统计结果表发布，并授权给特定的工作空间，以便于第三方应用通过 URL 获取数据。

详细操作配置请参见《UniCloud 集成平台 用户手册 E6101》中的疫苗接种监控案例。

5.6 数据最终呈现

该案例中的统计数据发布后，支持通过第三方调用展示，如图 5-14 和图 5-15 所示。

图5-14 疫苗接种情况展示（一）

图5-15 疫苗接种情况展示（二）

6 常见问题解答

1. 通过 DI 抽取到 HBase 表的数据支持在数据查询中访问吗

(1) 由于 DI 是一套独立完整的数据抽取工具，不和具体的业务强相关，HBase 作为 NoSQL 数据库，在表结构设计、查询等方面比较灵活，因此需要根据具体的业务进行设计。

(2) 数据查询组件针对 HBase 数据的访问有一套自己的业务逻辑，因此不支持数据查询对 DI 抽取到 HBase 表数据的访问。

(3) 如果存在此需求，可以先将数据通过 DI 抽取到管道，然后通过管道将数据写入到 HBase 中，管道针对 HBase 的表结构设计，索引、以及入数据的过程是和数据查询的业务逻辑保持一致的。

2. 将数据通过 DI 抽取到管道后，新建数据同步任务运行报错以及数据采样存在数据缺失

若存在需求通过 DI 将数据抽取至管道中，针对 CSV 及 JSON 类型管道，请确保抽取过程中，数据格式满足目标管道预设数据结构。

7 附录

7.1 管道字段映射规则

表7-1 字段映射类型

数据源类型	字段类型	Kafka 字段类型
HBase	string	string
	array	
	double	double
	integer	integer
	long	long
Hive	bigint	bigint
	boolean	boolean
	decimal	double
	double	
	float	float
	int	integer
	smallint	
	tinyint	
	array<string>	string
	binary	
	char	
	date	
	map<string,string>	
	string	
	varchar	
	timestamp	timestamp
MySQL	smallint	integer

数据源类型	字段类型	Kafka 字段类型
	mediumint	
	int	
	tinyint	
	bigint	bigint
	decimal	double
	double	
	float	float
	char	string
	varchar	
	varbinary	
	tinyblob	
	tinytext	
	blob	
	text	
	mediumblob	
	mediumtext	
	longblob	
	longtext	
	date	
	time	
	datetime	
	binary	
	timestamp	timestamp
	bit	boolean
Elasticsearch	keyword	string
	text	
	ip	

数据源类型	字段类型	Kafka 字段类型
	geo_point	
	array_text	
	array_long	
	array_keyword	
	array_integer	
	array_double	
	array_date	
	attachment	
	integer	integer
	double	double
	date	timestamp
	boolean	boolean
	long	long
PostgreSQL	int8	long
	timestamp	timestamp
	bit	boolean
	bool	
	int2	integer
	int4	
	float4	float
	float8	double
	numeric	
	bpchar	string
	text	
	bytea	
	date	

数据源类型	字段类型	Kafka 字段类型
	interval	
	varchar	
	time	
Greenplum	int8	long
	bit	boolean
	bool	
	int2	integer
	int4	
	timestamp	timestamp
	float4	float
	float8	double
	numeric	
	varchar	string
	text	
	bpchar	
	bytea	
	date	
	interval	
	time	
STDB	boolean	boolean
	double	double
	float	float
	integer	integer
	long	long
	timestamp	timestamp
	bytes	string
	string	

数据源类型	字段类型	Kafka 字段类型
	uuid	
	date	
	point	
	linestring	
	polygon	
	multipoint	
	multilinestring	
	multipolygon	
	geometrycollection	
	geometry	
	list[a]	
	map[a,b]	
Vertica	boolean	boolean
	numeric	double
	float	float
	integer	integer
	timestamp	timestamp
	binary	string
	char	
	geography	
	date	
	geometry	
	long varbinary	
	long varchar	
	uuid	
	time	
	varchar	

数据源类型	字段类型	Kafka 字段类型
	varbinary	
达梦	char	string
	character	
	varchar	
	varchar2	
	binary	
	varbinary	
	bfile	
	blob	
	byte	
	clob	
	date	
	image	
	time	
	text	
	integer	int
	int	
	tinyint	
	smallint	
	bit	boolean
	datetime	double
	decimal	
	double	
	double precision	
	number	
	numeric	
	timestamp	timestamp

数据源类型	字段类型	Kafka 字段类型
	bigint	bigint
	float	float

7.2 疫苗接种案例业务数据库建表语句示例

在业务数据库中，可通过 SQL 语句创建记录原始数据的表，本节提供了参考示例。

1. 创建人员信息表的 SQL 语句示例

```
CREATE TABLE 'person' (
  'id' int(11) NOT NULL AUTO_INCREMENT COMMENT '序号',
  'name' varchar(255) DEFAULT NULL COMMENT '姓名',
  'sex' varchar(255) DEFAULT NULL COMMENT '性别',
  'age' int(11) DEFAULT NULL COMMENT '年龄',
  'mobile' varchar(255) DEFAULT NULL COMMENT '手机号',
  'cardno' varchar(20) DEFAULT NULL COMMENT '身份证号',
  'classification_id' int(11) DEFAULT NULL COMMENT '人员分类(一级)',
  'content' varchar(255) DEFAULT NULL COMMENT '备注',
  'company_id' int(11) DEFAULT NULL COMMENT '单位名称',
  'region_id' int(11) DEFAULT NULL COMMENT '辖区 id，对应属地的摸底工作部门',
  'declare_department_id' int(11) DEFAULT NULL COMMENT '申报部门 id',
  'created_time' datetime DEFAULT NULL COMMENT '创建时间',
  'modified_time' varchar(255) DEFAULT NULL COMMENT '修改时间',
  'uuid' varchar(50) DEFAULT NULL COMMENT 'UUID',
  'addr' varchar(500) DEFAULT NULL COMMENT '现住址',
  'streetId' varchar(20) DEFAULT NULL COMMENT '街道',
  'provinceId' varchar(20) DEFAULT NULL COMMENT '省 ID',
  'cityId' varchar(20) DEFAULT NULL COMMENT '市 ID',
  'districtId' varchar(20) DEFAULT NULL COMMENT '区域 ID',
  'flag' varchar(10) DEFAULT NULL COMMENT '是否本市住户',
  'area' varchar(255) DEFAULT NULL COMMENT '小区名字',
  'subclass_id' int(11) DEFAULT NULL COMMENT '人群分类（二级)',
  PRIMARY KEY ('id') USING BTREE,
  UNIQUE KEY 'class_index' ('id','classification_id') USING BTREE,
  KEY 'compant_index' ('company_id') USING BTREE,
  KEY 'region_id_index' ('region_id') USING BTREE
```

) ENGINE=InnoDB AUTO_INCREMENT=1649514 DEFAULT CHARSET=utf8 ROW_FORMAT=DYNAMIC COMMENT='人员信息'

2. 创建人员分类字典表的 SQL 语句示例

```
CREATE TABLE 'person_classification' (
  'id' int(11) NOT NULL COMMENT 'id',
  'classification' text COMMENT '人员分类',
  'created_time' datetime DEFAULT NULL COMMENT '创建时间',
  'modified_time' datetime DEFAULT NULL COMMENT '修改时间',
  'category' varchar(255) DEFAULT NULL COMMENT '级别',
  PRIMARY KEY ('id') USING BTREE
) ENGINE=InnoDB DEFAULT CHARSET=utf8 ROW_FORMAT=DYNAMIC COMMENT='人员分类字典表'
```

3. 创建辖区字典表的 SQL 语句示例

```
CREATE TABLE 'region_dict' (
  'id' int(11) NOT NULL AUTO_INCREMENT COMMENT '序号',
  'region' varchar(255) DEFAULT NULL COMMENT '辖区',
  'userid' varchar(50) DEFAULT NULL COMMENT 'userid',
  'category' varchar(255) DEFAULT NULL COMMENT '有没有二级分类(1 是有)',
  'sort' int(11) DEFAULT NULL COMMENT '顺序编号',
  PRIMARY KEY ('id') USING BTREE,
  UNIQUE KEY 'region' ('region') USING HASH COMMENT 'regoion 索引'
) ENGINE=InnoDB AUTO_INCREMENT=27 DEFAULT CHARSET=utf8 ROW_FORMAT=DYNAMIC COMMENT='辖区字典表'
```

4. 创建人员接种信息表的 SQL 语句示例

```
CREATE TABLE 'person_inoculation' (
  'p_id' int(11) NOT NULL COMMENT 'ID',
  'assetsNum' varchar(100) DEFAULT NULL COMMENT '设备编码',
  'haveflag' varchar(10) DEFAULT NULL COMMENT '接种状态  true/false',
  'reason' varchar(800) DEFAULT NULL COMMENT '状态：0:禁忌症,1:延期接种,2:需退回上级重新分配,3:不符合接种人群范围',
  'beizhu' text COMMENT '备注',
  'curzhenshu' varchar(20) DEFAULT NULL COMMENT '针次',
  'jdate' varchar(20) DEFAULT NULL,
  'stime' varchar(20) DEFAULT NULL,
  'zhenshu' varchar(20) DEFAULT NULL,
```

```
  'yimiao' varchar(10) DEFAULT NULL COMMENT '疫苗种类  0:北京生物,1:北京科兴,2:武汉生物,3:康希诺',
  'jinjizheng' varchar(255) DEFAULT NULL COMMENT '禁忌症',
  'created' datetime DEFAULT NULL COMMENT '接种时间',
  KEY 'class_index' ('p_id') USING BTREE
) ENGINE=InnoDB DEFAULT CHARSET=utf8 ROW_FORMAT=DYNAMIC
```