

菲斯科白泽产品

详细设计V1.0



2021-10-13

菲斯科上海（软件）有限公司

上海市延安西路728号华敏翰尊国际4楼J座

V1.0

[1 文档说明 2](#_Toc85409062)

[1.1 文档目的 2](#_Toc85409063)

[1.2 背景 2](#_Toc85409064)

[1.3 读者对象 2](#_Toc85409065)

[1.4 术语和定义 2](#_Toc85409066)

[2 概要设计 0](#_Toc85409067)

[2.1 DMP系统架构 0](#_Toc85409068)

[2.2 DMP微服务架构 0](#_Toc85409069)

[2.3 DMP后台结构 1](#_Toc85409070)

[2.4 DMP系统逻辑架构 2](#_Toc85409071)

[2.5 环境说明 2](#_Toc85409072)

[2.5.1 系统语言 2](#_Toc85409073)

[2.5.2 系统平台 2](#_Toc85409074)

[2.5.3 服务器配置 2](#_Toc85409075)

[3 详细设计 2](#_Toc85409076)

[3.1 数据结构 2](#_Toc85409077)

[3.1.1 数据库设计 2](#_Toc85409078)

[3.1.2 表设计 2](#_Toc85409079)

[3.2 模块设计 2](#_Toc85409080)

[3.2.1 数据接入 3](#_Toc85409081)

[3.2.2 数据建模 3](#_Toc85409082)

[3.2.3 数据可视化 3](#_Toc85409083)

[3.2.4 数据治理 3](#_Toc85409084)

[3.2.5 数据服务 3](#_Toc85409085)

[3.2.6 系统管理 3](#_Toc85409086)

[3.2.7 系统用户 3](#_Toc85409087)

[3.2.8 消息队列 3](#_Toc85409088)

[3.2.9 任务调度 3](#_Toc85409089)

[3.2.10 数据治理 7](#_Toc85409090)

[3.2.11 系统管理 7](#_Toc85409091)

# 文档说明

## 文档目的

本文档编写的目的是说明Fisk白泽产品开发模块的设计，为开发人员编码提供依据。

## 背景

软件系统名称：DMP数据治理平台。

## 读者对象

预期读者包括Fisk产品部开发人员。

## 术语和定义

|  |  |  |
| --- | --- | --- |
| 术语 | 描述 |  |
| Fisk | 菲斯科（上海）软件有限公司 |  |
| DMP | 菲斯科白泽产品 |  |
|  |  |  |

## 人员安排

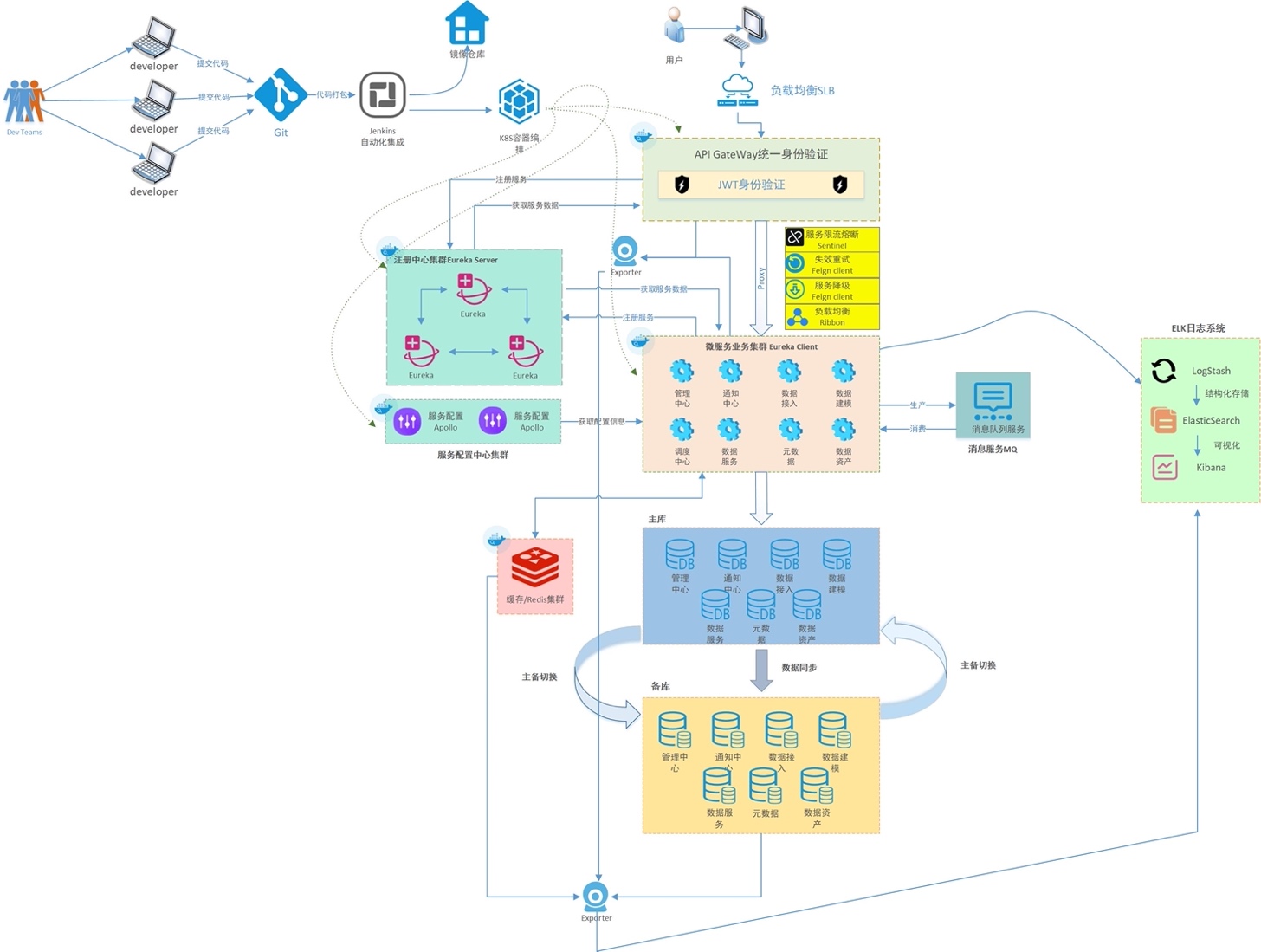
|  |  |  |
| --- | --- | --- |
| **模块** | **前端负责人** | **后端负责人** |
| 数据接入 | 胡卓程 | 赵飞鸿 |
| 数据建模 | 严小韦 | 杨建文 |
| 数据可视化 | 胡丁丁 | 王岩 |
| 数据服务 | 罗丰 | 王岩 |
| 数据工厂 | 罗丰 | 赵飞鸿 |
| 任务调度 | 无 | 崔富凯 |
| 数据治理 | 许俊杰 | 杨建文 |

# 概要设计

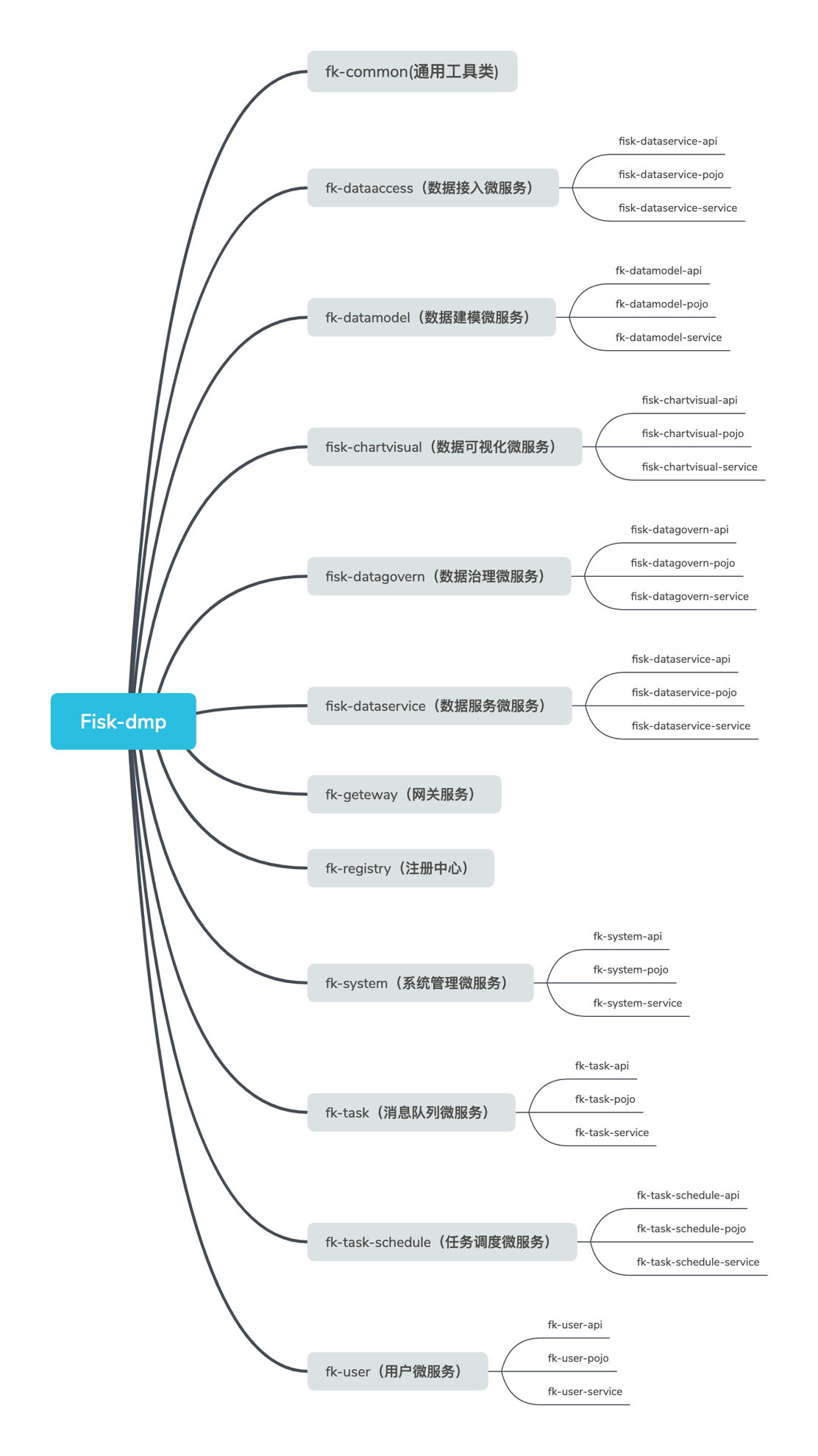
## DMP系统架构

DMP考虑到后期对系统的升级和维护，功能的水平扩展以及开发技术的更新等因素，故后台使用JAVA微服务根据系统业务按照功能拆分为更加细粒度的服务，所拆分的每一个服务都是一个独立的应用，这些应用对外提供公共的API，可以独立承担对外服务的职责。

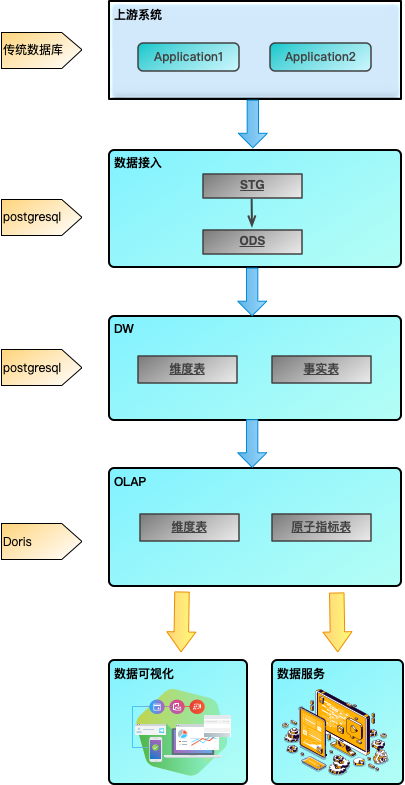
## DMP微服务架构



## DMP后台结构



## DMP系统逻辑架构



## 环境说明

### 系统语言

### 系统平台

### 服务器配置

# 详细设计

## 数据结构

### 数据库设计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **数据库** | **数据库类型** | **所属微服务** | **描述** | **备注** |
| dmp\_chartvisualt\_db | mysql |  | 数据可视化数据库 |  |
| dmp\_datagovern\_db | mysql |  | 数据治理数据库 |  |
| dmp\_datainput\_db | mysql |  | 数据接入数据库 |  |
| dmp\_datamodel\_db | mysql |  | 数据建模数据库 |  |
| dmp\_dataservice\_db | mysql |  | 数据服务数据库 |  |
| dmp\_system\_db | mysql |  | 系统管理数据库 |  |
| dmp\_task\_db | mysql |  | 消息队列数据库 |  |
| dmp\_ods | postgresql |  | 数据同步的ods表 |  |
| dmp\_dw | postgresql |  | 数仓 |  |
| dmp\_dw | doris |  | 分析模型 | 分析模型 |

## 模块设计

根据DMP后台微服务的拆分，主要分为以下模块：数据接入，数据建模，数据可视化，数据治理，数据服务，系统管理，系统用户，消息队列，任务调度。

### 数据接入

### 数据建模

### 数据可视化

#### 表设计

#### 数据可视化结构（dmp\_chartvisual\_db）

* 可视化图表（tb\_char）

|  |  |  |
| --- | --- | --- |
| **字段** | **类型** | **备注** |
| id | int | 主键 |
| fid | int | 文件夹id |
| name | varchar(50) | 报表名称 |
| content | varchar(10000) | 图表格式（json字符串） |
| detail | varchar(200) | 报表描述 |
| image | longblob | 报表缩略图 |
| create\_user | varchar(20) | 创建人 |
| create\_time | datetime | 创建时间 |
| update\_user | varchar(20) | 更新人 |
| update\_time | datetime | 更新时间 |
| del\_flag | tinyint | 逻辑删除标识（1未删除，0删除） |

* 数据连接表（tb\_datasource\_con）

|  |  |  |
| --- | --- | --- |
| **字段** | **类型** | **备注** |
| id | int | 主键 |
| name | varchar(50) | 连接名称 |
| con\_str | varchar(500) | 连接字符串 |
| con\_ip | varchar(50) | ip |
| con\_port | int | 端口 |
| con\_cube | varchar(50) | 模型 |
| con\_dbname | varchar(50) | 数据库名称 |
| con\_type | tinyint | 连接类型 |
| con\_account | varchar(50) | 连接账号 |
| con\_password | varchar(50) | 连接密码 |
| create\_time | datetime | 创建时间 |
| update\_user | varchar(20) | 更新人 |
| update\_time | datetime | 更新时间 |
| del\_flag | tinyint | 逻辑删除标识（1未删除，0删除） |

### 数据治理

### 数据服务

### 系统管理

### 系统用户

### 消息队列

### 数据工厂

#### 表设计

#### 数据工厂结构（dmp\_datafactory\_db）

* NIFI组件类型表（tb\_nifi\_components）

|  |  |  |
| --- | --- | --- |
| **字段** | **类型** | **备注** |
| id | int | 主键 |
| name | varchar(50) | 组件名称 |

* 自定义NIFI数据管道表（tb\_nifi\_custom\_workflow）

|  |  |  |
| --- | --- | --- |
| **字段** | **类型** | **备注** |
| id | int | 主键 |
| workid | varchar(50) | GUID |
| workname | varchar(200) | 管道名称 |
| status | tinyint | 状态：已发布（1）、未发布（0） |
| create\_user | varchar(100) | 创建人 |
| create\_time | datetime | 创建时间 |
| update\_user | varchar(100) | 更新人 |
| update\_time | datetime | 更新时间 |
| del\_flag | tinyint | 逻辑删除（1未删除，0删除） |

* 自定义NIFI数据管道详细表（tb\_nifi\_custom\_workflow\_detail）

|  |  |  |
| --- | --- | --- |
| **字段** | **类型** | **备注** |
| id | int | 主键 |
| workflowid | varchar(50) | tb\_nifi\_custom\_workflow > id |
| componentsid | int | tb\_nifi\_components > id |
| tableid | varchar(50) | 关联的表id |
| schedule | tinyint | 调度类型，1 timer driven、2 cron表达式 |
| script | varchar(50) | 调度的脚本，如果是timer driven 则值应是秒数，如果是cron driven则值应是cron表达式 |
| inport | varchar(50) | 串行入口 |
| outport | varchar(50) | 串行出口 |
| create\_user | varchar(100) | 创建人 |
| create\_time | datetime | 创建时间 |
| update\_user | varchar(100) | 更新人 |
| update\_time | datetime | 更新时间 |
| del\_flag | tinyint | 逻辑删除（1未删除，0删除） |

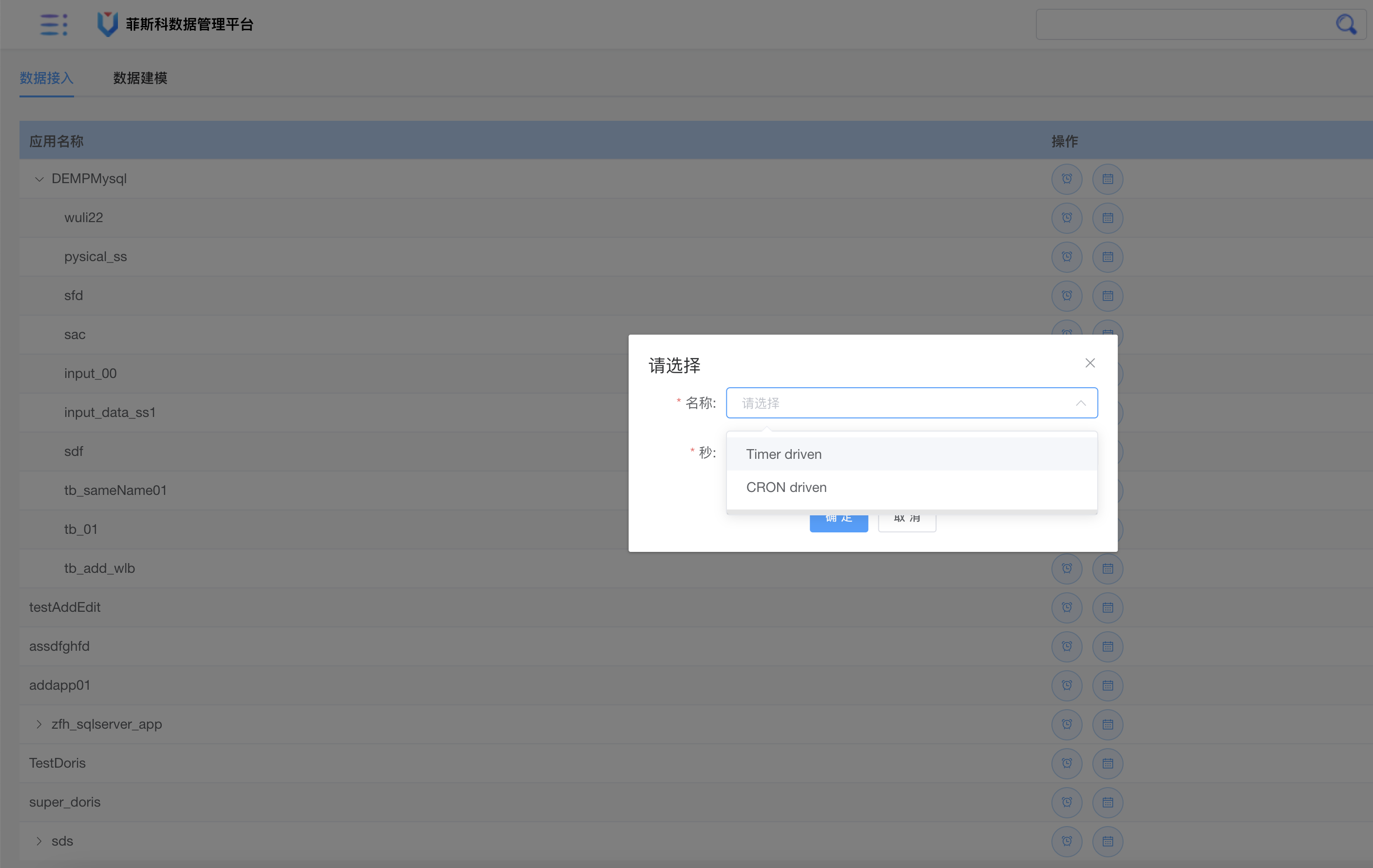
#### 单表调度

在DMP中完成数据接入、数据建模等一系列操作之后即可到数据调度页面（图1、图2）对之前接入的数据表设置定时同步频率。

图1



图2



对于数据表的调度频率有两种方式：

1. 时间驱动（Timer Driven）
2. Cron 表达式（Cron Driven）

时间驱动可以设置每隔多少秒执行一次调度。

Cron表达式可以设置定时调度（图3）

图3



#### 数据管道

在DMP管道设置中可以针对多表串行数据同步进行配置（图4、5），能够按照设定的流程从上游系统直接流入到最终的分析建模中，以便报表访问人员能够在报表中直观的看到数据的变化，确保了数据流的连续性。

图4

图形用户界面, 应用程序

描述已自动生成

图5

图形用户界面, 表格

描述已自动生成

### 数据治理

### 系统管理