# VMAX大数据ADMA功能说明

## 任务管理

大数据管理平台提供任务管理机制，所有任务统一建模、由平台统一调度、分配资源、已简化开发流程、提高开发效率、系统能够高效运转。

### 1.1 任务分类

1、定时任务：任意指定任务运行时间，已满足不同用户的定制化需求；

2、数据驱动任务：依据数据实际情况驱动具体任务执行，避免数据到达不及时运行或

数据未到空跑或未到全运行，产生垃圾数据；

3、指定日期运行：指定某一天的某些任务重新运行，适用范围如下：

1. 新任务或版本升级后需要把旧数据利用起来，都可以使用这种方式

2）数据异常引起的没跑或跑过的（失败和成功）任务

### 1.2 任务运行环境分类

基于通用的大数据组件，如Spark、gbase、Hbase等。

### 1.3 任务时效性

系统支持多种任务粒度，分类如下：

1) 支持离线任务——小时、天、周、月业务的数据清洗，实时性要求低 ；

2) 支持准实时任务—— 5分钟任务 ；

3) 支持实时任务——时效性要求高，如流处理任务。

### 1.4 VIP任务

支持已优先级方式保证重点任务优先运行。

### 1.5 资源分配

系统把集群资源进行统一分配管理，已保证资源利用率达到最优。

1、静态资源分配

每个任务可根据实际需要指定所需资源。

2、动态资源优化

由于各任务指定资源存在不合理性，为此系统根据任务历史运行资源耗费情况，寻

找任务资源敏感度因素，择优选取不同策略动态调整资源，使得系统内资源利用率

达到最优。

### 1.6 稳定性-系统级监控

系统需要有完善的自检自愈机制，已保证系统可不间断运行，减少人为干预。系统监控的主要功能包括：

1）进程级看门狗

2）进程内的线程状态监控，心跳检测；

1. 重要事件（任务下发、调度执行等）的自检；
2. 异常集群节点的检测
3. 检测结果的持久化

自愈方式：会根据异常严重程度，采取不同自愈措施如告警、重启等及时恢复系统。

### 1.7 热补丁

由于算法错误导致的异常，需要及时修正算法，但是又不能影响系统，则通过热补丁方式在不重启系统前期下及时加载变更后的算法。

### 1.8 任务可视化监控

系统内的任务按照分组进行可视化监控，包括如下4部分，见图1：

1、监控对象选择

2、系统完成情况概览

3、具体功能列表

4、系统资源信息



图1、元数据功能监控界面

1、监控对象选择，见图2：

元数据功能监控，支持地区和时间选择，地区以接入省份为主，时间支持最近3天。



图2、地区时间选择

2、系统完成情况概览

1）提供系统完成情况概览，见图3；

2）支持自定义功能的监控，见图3；

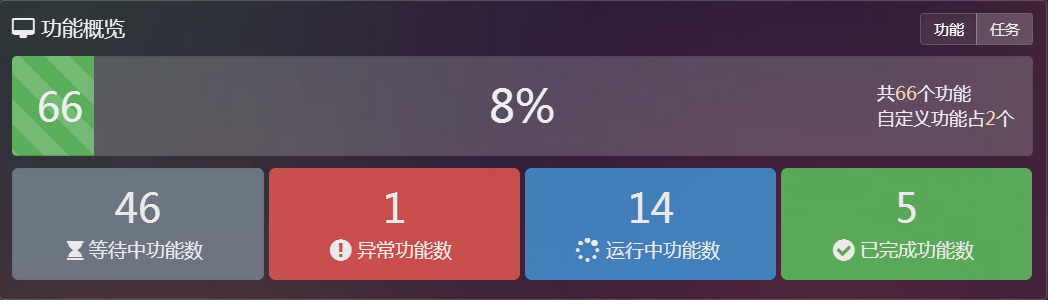


图3、系统完成情况概览

3）支持按状态联动具体的功能，见图4~8。



图4、按自定义功能联动



图5、按已完成功能联动



图6、按进行中功能联动



图7、按异常功能联动



图8、按等待功能联动

3、具体功能列表

1）支持搜索、排序

2）支持查看具体功能的完成情况，见图9



图9、查看具体功能的完成情况

4、系统资源信息

1）支持查看系统的资源使用趋势

2）支持查看当前的可用资源信息

3）支持查看历史的资源使用信息

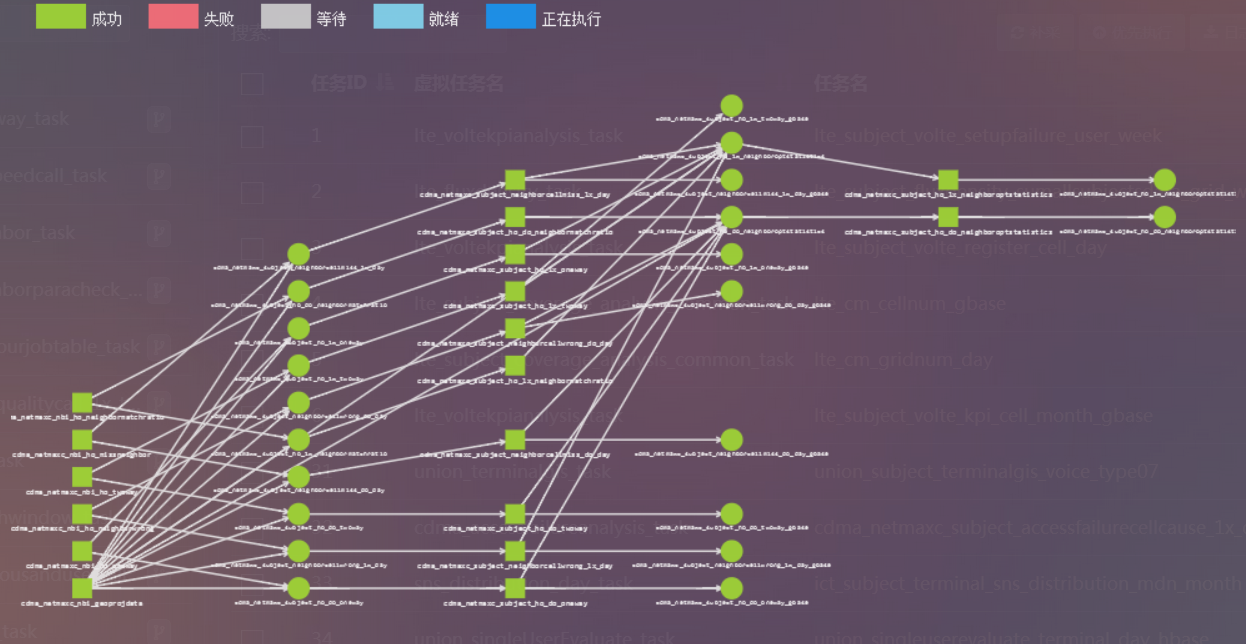
## 2、数据管理

支持数据统一建模，定义数据属性形成数据元数据。

### 2.1血缘分析

基于表和任务之间的关系建立的血缘图，可以通过血缘分析了解系统内任务依赖关系、数据依赖关系、任务数据间依赖关系，对于系统有全面了解。

**血缘图示例：**



**血缘分析应用：**

1) 可视化监控拓扑图

2) 补采功能

3) 优先级同步反转

### 2.2数据质量检查

原始数据进入系统后，会经过一系列数据处理流程，为了保证整个数据流中的数据质量，支持数据质量检查。

**检查粒度**：包括周期巡检、实时检查。

**检查过程**：中间结果检查、最终结果检查。

**检查方式**：对检查策略进行建模，用户可随意配置检查规则，实现不同策略的检查。

**检查结果**：质量检查不合格的数据，会终止后续数据流的处理，以便及时发现数据质量问题，采取补救措施。

### 2.3数据异常恢复

系统中，常存在由于如数据异常、或算法错误等原因引起的一些数据处理结果异常，导致系统中存在无效数据，为此需要有异常恢复机制，保证系统中数据能及时纠错，主要通过如下三种方式实现，包括补采、指定日期运行。

#### 2.3.1 补采

**补采**：就是根据任务血缘关系，分析出受其影响的任务，备份选中任务以及被影响的任务，生成对应新任务链的过程。

**补采适用范围：**数据异常引起的运行过的任务，包括成功和失败

**对系统的影响**：补采会屏蔽旧任务，加入新任务并进行调整，并把补采信息同步到数据库

**补采使用场景**

场景1：对失败任务进行补采

场景2：认为不合格的任务

**补采操作界面**：见图10



图10、补采操作确认

#### 2.3.2 指定日期运行

与补采区别：不会通过血缘寻找被指定补采任务相关的上下游任务链，指定那个任务运行，就单纯运行该任务。

详见1.1节。

### 2.4 数据可视化监控

系统内所有原始表数据都可以在界面分组可视化监控，见图10~12。

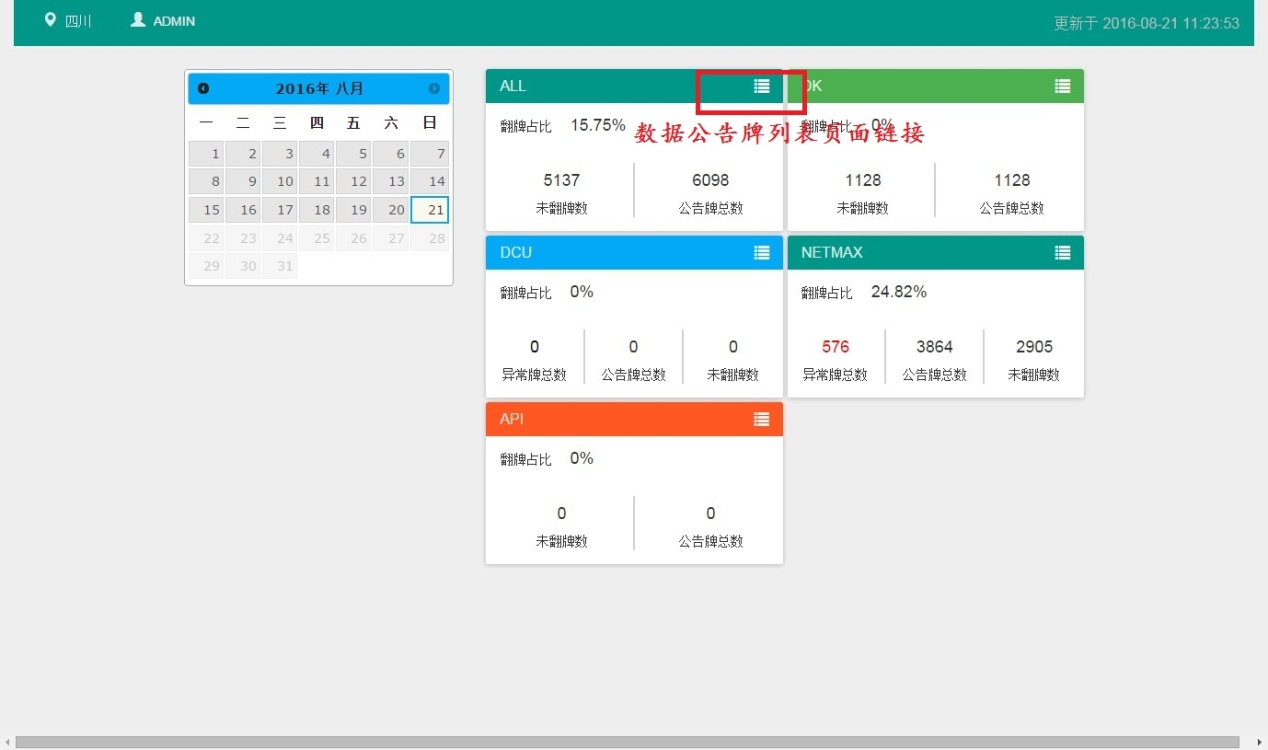


图10、数据分组显示首页

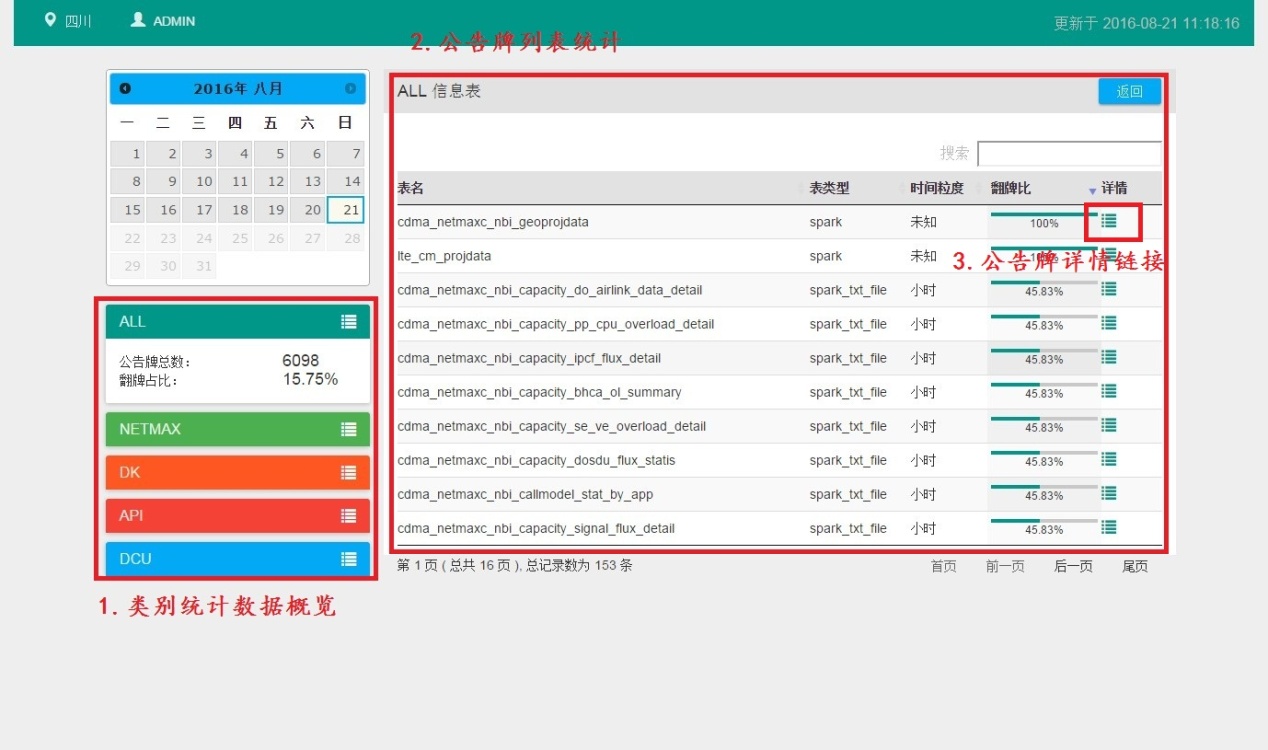


图11、某一分组下数据列

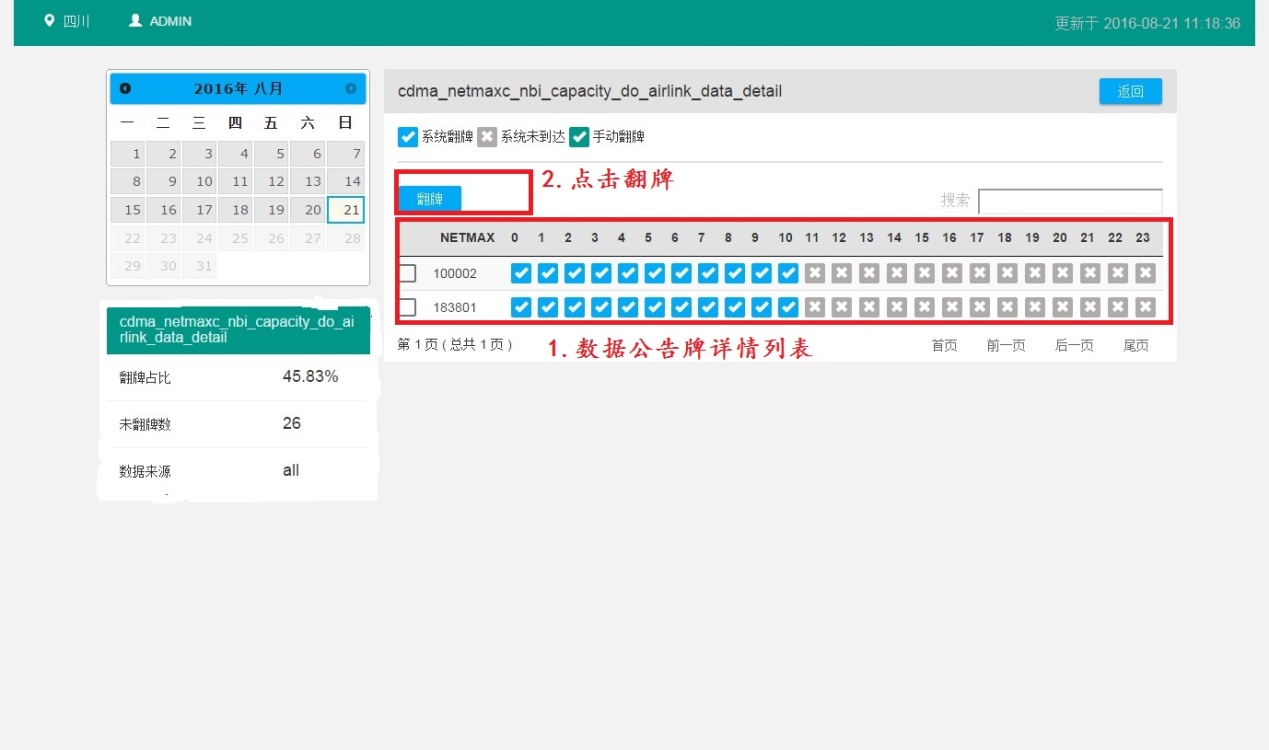


图12、某一张表数据到达详情

### 2.5 数据生命周期管理

海量数据下，系统运行一定的时间后，集群中存储的数据会成几何级数的增长，系统的资源又是有限的，会严重影响系统的运行时间周期，因此针对一些历史无效数据，要进行定时的删除，减轻系统的负载，提高系统的运行周期。

## 3、算法探索

### 3.1 任务通用算法模型

为了减轻基于平台的应用开发工作量，使得算法人员更关注算法实现，而不聚焦于平台任务调度，平台提供通用算法模型包括以下几种：

1. SparkSQL类型算法模型：用户只需编写SparkSQL的配置文件就可以完成业务清洗平台实现SQL的应用的封装
2. GbaseSQL算法模型

### 3.2 动态加载

算法专家可基于系统之上做算法探索，为了满足探索需求，实现动态加载功能，可以在不重启系统前提下，随时提交新的算法。

