数据处理平台-技术方案

## 1 概述

随着公司发展，新的产品和项目不断增加，老的项目和产品的功能在不断的迭代，对于错误和异常的种类不断增多，项目管理和异常的排查难度不断增加。数据的类型和格式也在经常的变更，灵活的操作数据转化是目前尚未解决的一个问题。

本技术方案主要涉及对日志（包括业务日志，系统日志）、数据收集并进行分析，对分析的结果进行相应的统计和告警，对收集的数据信息提供查询功能。

## 2 需求分析

日常项目在运行的过成中，对项目异常的响应相对滞后，运维人员无法在第一时间能否感知项目的运行情况。需要出现能够相对正确和及时的告警和预警机制。

在多种数据类型，数据类型变更频发的情况下，经常性修改代码的成本高，切不利于项目本身的维护工作。

### 2.1 需求场景

在日常出现系统异常的时候，排查异常，查看异常日志会前往项目所在服务器，在海量的日志查找异常的信息，还可能会出现找不到异常信息，然后需要复现异常的场景在进行异常的排查。这样传统的方式使排查错误的效率十分低下。如果能过将排查异常的过成从线上服务器转移至平台，通过工具自动进行筛选，能够大大的提高异常排查和处理的效率。

在异常处理的过程中经常会出现系统的异常和告警反复出现，并且会不定期的出现，但是这样的告警服务在业务层面进行根治。当我们对一个产品有了长期的运维经验以后，可以对该产品可能出现的问题已经有了相对的处理方案，在这种情况下，希望是能够将我们所知道的处理过程告知程序，由程序帮助我们进行自动化的处理。

## 3 总体设计

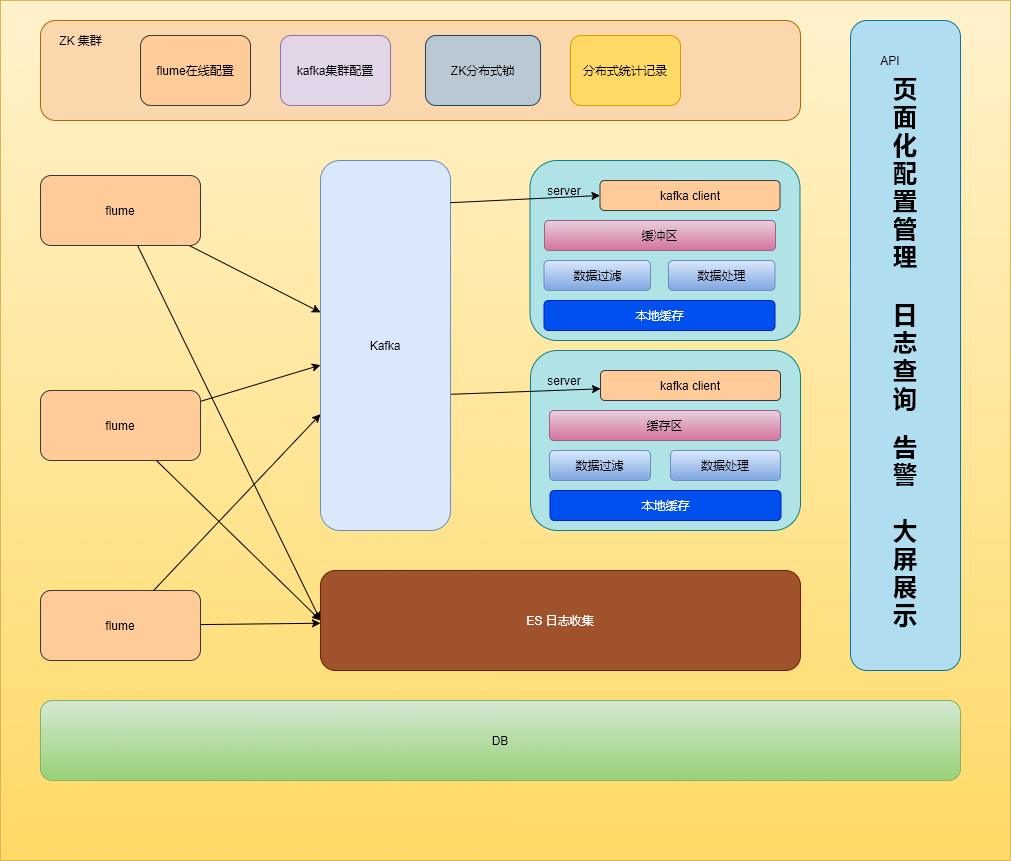
### 3.1 设计思路

本技术方案在设计之时尽可能的增强他的灵活性和可扩展性。

主要涉及的技术栈有：Apache-Flume、Apache-Kafka、Apache-Zookeeper、ElasticSearch、Mysql

系统架构图如下：

**架构图**

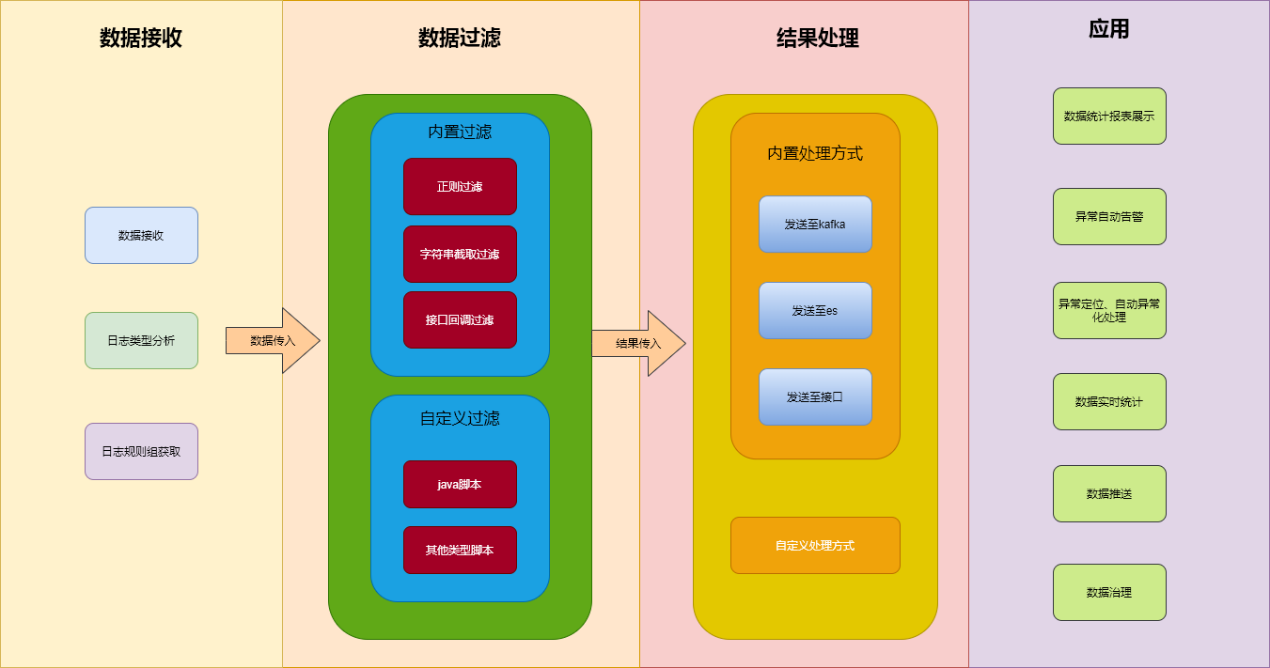


（图3.1）

各组件模块功能说明如下：

1. Flume：对项目日志，系统日志进行采集，然后上传至Kafka和ElasticSearch
2. Kafka：用于数据的传输，中间件存储
3. ElasticSearch：用于后期日志信息的查询
4. Server：对收集的日志进行分析，将分析的结果进行告警规则和自动化处理规则进行比对，将比对结果存储于数据库
5. API：为告警的规则和Flume配置的规则进行页面化的配置。提供日志查询的功能。定时监控flume运行情况。提供大屏对一些基本的信息进行展示。告警组的设置和告警信息的发送
6. Zookeeper：将Flume和Kafka的配置托管至Zookeeper中，在集群的情况下由Zookeeper来实现分布式锁
7. DB：存储相关的配置和告警的信息

**Server数据流转图**

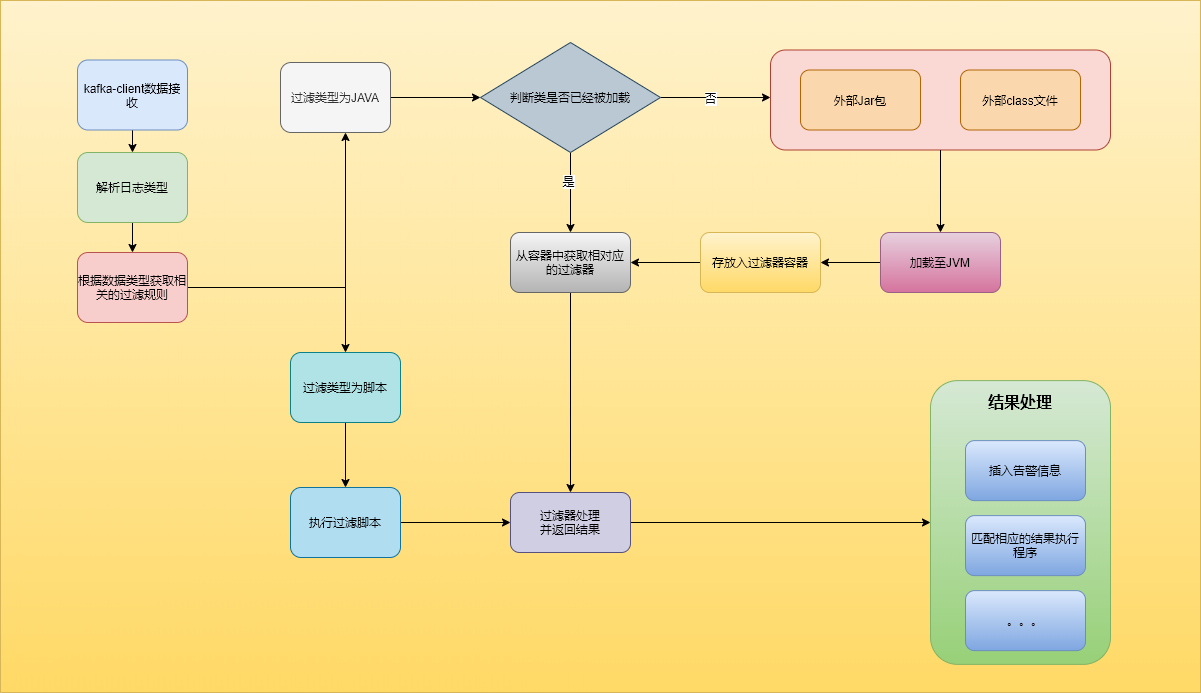


（图3.2）

1. 数据接收
   1. 数据通过kafka-client获取数据。获取数据时通过不同的数据类型将数据存放至不同的线程池中处理，确保不同的数据类型之间处理时不相互影响。
   2. 根据不同类型的数据类型，匹配出不同的过滤规则，并将数据流转至过滤区
2. 数据过滤
   1. 内置过滤
      1. 正则过滤：通过配置好的正则表达式，在数据中提取符合规则的字符串
      2. 字符串截取过滤：通过配置好的截取长度，开始，结束。来截取符合的字符串
      3. 接口回调过滤：通过调用接口的方式完成逻辑过滤的过成
   2. 自定义过滤
      1. 支持继承java的过滤类完成过滤条件
      2. 支持其他脚本完成过滤逻辑
3. 结果处理
   1. 将过滤好的结果流转至结果处理区，完成对结果的处理
   2. 内置结果处理
      1. 发送至kafka
      2. 发送至ES
      3. 接口回调：支持将过滤信息通过接口方式发送其他地方
   3. 自定义结果处理
      1. 通过继承方式实现结果处理

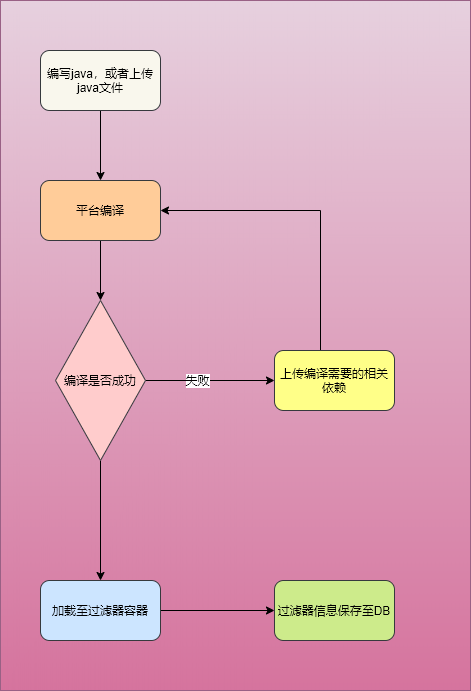
### 3.2 可行性分析

**数据分析流程图**



（图3.3）

**Java脚本加载过成**

（图3.4）

日志分析细节实现如下：

为了增加系统对日志分析的灵活性和准确性，系统将会在开发的过程中预留分析的接口，外部程序可以通过继承接口的方式对不同的项目日志进行个性化的分析。

为了提高程序分析的效率，需要在对日志在采集器进行标记，指定被标记的日志需要经过部分判断条件而不是所有的。同一条日志信息采用多线程的方式异步处理不同的判断的添加，不同的日志信息将会同时进行分析来提升分析的速度。

为了减轻对使用者的使用难度，增加对各种不同的脚本来支持日志的分析过程，前提是需要脚本返回相应的可以进行比对的结果值。

系统可靠性：

1. 系统日志的采集和传输依赖Flume和Kafka来保证数据不丢失
2. Server通过记录Kafka的offset来保证日志分析不丢失，不重复
3. Server 集群方式来提高系统的容错能力

系统分析正确性：

1. 系统日志分析主要由使用自定义脚本，使用本系统的前提必须是在使用了解被监听的程序的日志的格式

系统告警方式多样：

1. 告警方式可以支持邮件、钉钉、微信

系统查询：

1. 依靠ElasticSearch来完成日志查询。保证数据检索的速度和正确率

### 开发工作预览

|  |  |  |
| --- | --- | --- |
| 序号 | 所属模块 | 工作内容 |
| 1 | flume | flume 快速部署脚本 |
| 2 | flume 支持多种日志推送方式 |
| 3 | server | 数据接收后的缓冲区开发，考虑是否需要数据丢弃机制 |
| 4 | 对数据库中规则配置的读取 |
| 5 | 增加本地缓存减少对数据库的访问 |
| 6 | 心跳注册至ZK便于监控 |
| 7 | 内置过滤器开发 |
| 8 | 内置处理器开发 |
| 9 | api | 页面配置flume信息推送至ZK，包括flume配置回显 |
| 10 | 过滤规则配置 |
| 11 | 告警规则配置 |
| 12 | 告警组信息配置 |
| 13 | 项目信息和数据信息和flume的绑定配置 |
| 14 | ES日志查询开发 |
| 15 | 监听flume心跳，监听server心跳 |

## 4 后期功能升级和扩展

1. 更改数据接入源。可以不仅仅只监听日志文件，可以是其他数据，只要需要传入kafka就可以将数据进行提取和分析
2. 增强日志分析能力。可以接入第三工具或者其他分析方式来增强数据分析的能力，如：
   1. 可以通过提供外部接口的方式将过滤器类加载是JVM
   2. 可以通过外部接口来实现过滤器筛选
   3. 在线编写java类实现过滤器功能
3. 增加告警方式。比如调用接口进行短信发送
4. 增加日志存储和查询的方式。目前可能支持ES，后期可以增加第三方工具比如SLS
5. 增加数据过滤结果处理功能。可以增加其他的结果处理操作。如：
   1. 结果比对后执行脚本
   2. 结果比对后进行数据格式修改
6. 性能优化
   1. 过滤与处理分离
      1. 数据传输问题
   2. 增加缓冲区
      1. 缓存区存储方式

6.3 本地缓存

# 1、数据字典

## 1.1、t\_ds\_alert

* 告警表

| **字段** | **类型** | **允许空** | **默认** | **注释** |
| --- | --- | --- | --- | --- |
| id | int(11) | NO | NULL | key |
| title | varchar(64) | YES | NULL | title |
| show\_type | tinyint(4) | YES | NULL | send email type,0:TABLE,1:TEXT |
| content | text | YES | NULL | Message content (can be email, can be SMS. Mail is stored in JSON map, and SMS is string) |
| alert\_type | tinyint(4) | YES | NULL | 0:email,1:sms |
| alert\_status | tinyint(4) | YES | 0 | 0:wait running,1:success,2:failed |
| log | text | YES | NULL | log |
| alertgroup\_id | int(11) | YES | NULL | alert group id |
| receivers | text | YES | NULL | receivers |
| receivers\_cc | text | YES | NULL | cc |
| create\_time | datetime | YES | NULL | create time |
| update\_time | datetime | YES | NULL | update time |

## 1.2、t\_ds\_alertgroup

* 告警组

| **字段** | **类型** | **允许空** | **默认** | **注释** |
| --- | --- | --- | --- | --- |
| id | int(11) | NO | NULL | key |
| group\_name | varchar(255) | YES | NULL | group name |
| group\_type | tinyint(4) | YES | NULL | Group type (message 0, SMS 1...) |
| description | varchar(255) | YES | NULL | 无 |
| create\_time | datetime | YES | NULL | create time |
| update\_time | datetime | YES | NULL | update time |

## 1.3、t\_ds\_relation\_user\_alertgroup

* 用户告警组关系绑定

| **字段** | **类型** | **允许空** | **默认** | **注释** |
| --- | --- | --- | --- | --- |
| id | int(11) | NO | NULL | key |
| alertgroup\_id | int(11) | YES | NULL | alert group id |
| user\_id | int(11) | YES | NULL | user id |
| create\_time | datetime | YES | NULL | create time |
| update\_time | datetime | YES | NULL | update time |

## 1.4、t\_ds\_user

* 用户表

| **字段** | **类型** | **允许空** | **默认** | **注释** |
| --- | --- | --- | --- | --- |
| id | int(11) | NO | NULL | user id |
| user\_name | varchar(64) | YES | NULL | user name |
| user\_password | varchar(64) | YES | NULL | user password |
| user\_type | tinyint(4) | YES | NULL | user type, 0:administrator，1:ordinary user |
| email | varchar(64) | YES | NULL | email |
| phone | varchar(11) | YES | NULL | phone |
| tenant\_id | int(11) | YES | NULL | tenant id |
| create\_time | datetime | YES | NULL | create time |
| update\_time | datetime | YES | NULL | update time |
| queue | varchar(64) | YES | NULL | queue |
| state | int(1) | YES | 1 | state 0:disable 1:enable |

## 1.5、t\_hs\_deal\_rules\_group

* 处理规则组

| **字段** | **类型** | **允许空** | **默认** | **注释** |
| --- | --- | --- | --- | --- |
| id | int(11) | NO | NULL | 主键 |
| uuid | varchar(16) | YES | NULL | 规则组ID |
| group\_name | varchar(255) | YES | NULL | 规则组名称 |
| note | varchar(255) | YES | NULL | 规则组说明 |
| data\_uuid | varchar(16) | YES | NULL | 数据类型ID |
| is\_effective | tinyint(2) | YES | 0 | 是否有效：0有效，1无效 |
| create\_time | datetime | YES | NULL | 创建时间 |

## 1.6、t\_hs\_deal\_rules\_info

* 处理规则具体信息

| **字段** | **类型** | **允许空** | **默认** | **注释** |
| --- | --- | --- | --- | --- |
| id | int(11) | NO | NULL | 主键 |
| deal\_group\_uuid | varchar(16) | YES | NULL | 处理规则组uuid |
| uuid | varchar(16) | YES | NULL | 规则uuid |
| rules\_name | varchar(255) | YES | NULL | 规则名称 |
| note | varchar(255) | YES | NULL | 备注 |
| type | varchar(255) | YES | NULL | 规则类型（java，shell） |
| config | varchar(255) | YES | NULL | 过滤规则，json格式 |
| context | text | YES | NULL | 过滤详细内容 |
| create\_time | datetime | YES | NULL | 创建时间 |
| is\_effective | tinyint(2) | YES | 0 | 是否有效：0有效，1无效 |

## 1.7、t\_hs\_ex\_flume\_source\_info

* flume资源信息记录

| **字段** | **类型** | **允许空** | **默认** | **注释** |
| --- | --- | --- | --- | --- |
| id | int(11) | NO | NULL | 自增主键 |
| project\_uuid | varchar(16) | YES | NULL | 一个项目和一个flume相互绑定 |
| log\_uuid | varchar(16) | YES | NULL | 日志的类型，与flume下的一个source相互绑定 |
| note | varchar(255) | YES | NULL | 日志类型说明 |
| create\_time | datetime | YES | NULL | 创建时间 |

## 1.8、t\_hs\_ex\_project\_info

* 项目信息

| **字段** | **类型** | **允许空** | **默认** | **注释** |
| --- | --- | --- | --- | --- |
| id | int(11) | NO | NULL | 主键 |
| uuid | varchar(0) | YES | NULL | 项目主键 |
| project\_name | varchar(255) | YES | NULL | 项目名称 |
| note | varchar(255) | YES | NULL | 备注 |
| zk\_node | varchar(255) | YES | NULL | 给项目分配的zk节点 |
| create\_time | datetime | YES | NULL | 创建时间 |
| update\_time | datetime | YES | NULL | 心跳时间 |
| context | text | YES | NULL | 心跳内容 |
| is\_survive | tinyint(2) | YES | NULL | 是否存活：0正常。1离线 |

## 1.9、t\_hs\_ex\_rel\_user\_project

* 用户项目关系绑定

| **字段** | **类型** | **允许空** | **默认** | **注释** |
| --- | --- | --- | --- | --- |
| id | int(11) | NO | NULL | 自增主键 |
| user\_id | int(11) | YES | NULL | 用户ID |
| project\_uuid | varchar(16) | YES | NULL | 项目主键 |
| permis | enum('admin','user') | YES | admin | 权限 |

## 1.10、t\_hs\_filter\_rules\_group

* 过滤组

| **字段** | **类型** | **允许空** | **默认** | **注释** |
| --- | --- | --- | --- | --- |
| id | int(11) | NO | NULL | 主键 |
| uuid | varchar(16) | YES | NULL | 规则组ID |
| group\_name | varchar(255) | YES | NULL | 规则组名称 |
| note | varchar(255) | YES | NULL | 规则组说明 |
| log\_uuid | varchar(16) | YES | NULL | 日志ID |
| is\_effective | tinyint(2) | YES | 0 | 是否有效：0有效，1无效 |
| create\_time | datetime | YES | NULL | 创建时间 |

## 1.11、t\_hs\_filter\_rules\_info

* 过滤规则配置

| **字段** | **类型** | **允许空** | **默认** | **注释** |
| --- | --- | --- | --- | --- |
| id | int(11) | NO | NULL | 主键 |
| filter\_group\_uuid | varchar(16) | YES | NULL | 过滤规则组uuid |
| deal\_group\_uuid | varchar(16) | YES | NULL | 处理规则组uuid |
| uuid | varchar(16) | YES | NULL | 规则uuid |
| rules\_name | varchar(255) | YES | NULL | 规则名称 |
| note | varchar(255) | YES | NULL | 备注 |
| type | varchar(255) | YES | NULL | 规则类型（java，shell） |
| config | varchar(255) | YES | NULL | 过滤规则，json格式 |
| context | text | YES | NULL | 过滤详细内容 |
| create\_time | datetime | YES | NULL | 创建时间 |
| is\_effective | tinyint(2) | YES | 0 | 是否有效：0有效，1无效 |

## 1.12、t\_hs\_kafka\_log\_info

* kafka数据接收记录

| **字段** | **类型** | **允许空** | **默认** | **注释** |
| --- | --- | --- | --- | --- |
| id | int(11) | NO | NULL | 主键 |
| kafka\_url | varchar(255) | YES | NULL | kafka连接地址 |
| topic | varchar(255) | YES | NULL | 主题 |
| partition | int(255) | YES | NULL | 分区号 |
| offset | int(255) | YES | NULL | 偏移位 |
| createtime | datetime | YES | NULL | 创建时间 |
| is\_effective | tinyint(2) | YES | 0 | 是否有效：0有效，1无效 |

## 1.13、t\_hs\_tool\_statisti

* 统计记录

| **字段** | **类型** | **允许空** | **默认** | **注释** |
| --- | --- | --- | --- | --- |
| id | int(11) | NO | NULL | 自增主键 |
| deal\_uuid | varchar(0) | YES | NULL | 处理器主键 |
| field | varchar(255) | YES | NULL | 统计字段 |
| count | int(11) | YES | NULL | 次数 |
| period | int(11) | YES | NULL | 时间段，单位为秒 |
| create\_time | datetime | YES | NULL | 创建时间 |