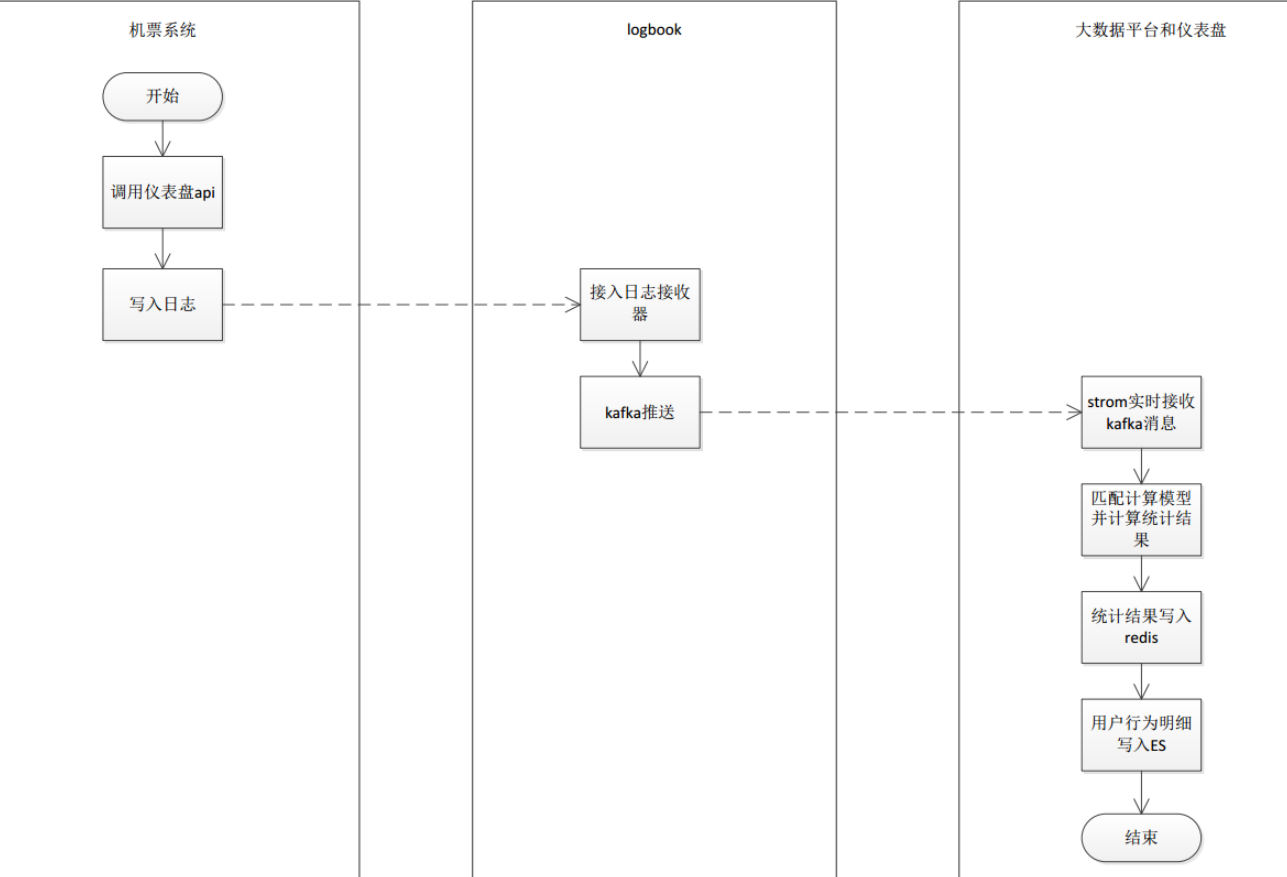
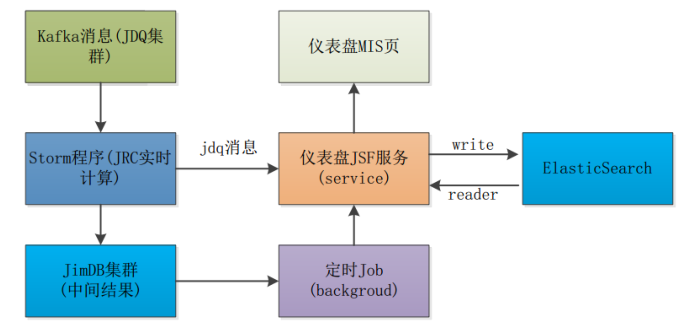
架构





# Storm消费jdq

消费的时候有限流，通过redis的incr限流1000条

打日志：达到阀值1000,丢弃

Storm消费jdq消息时候，异步多线程将数据写入redis缓存（1.查询次数，2查询明细使用list构建的队列，1000条）

默认没有写入es（若要写入，是通过jsf接口调用）

消息体：

{

"time": 1557466849000,

"msg": "{\"dimension\":null,\"key\":\"userQueryLine\",\"value\":\"d804e0a9dec846d2b9c685ccad0527371557466845571,279774460\_m,mobile,XIY,WUX,2019-05-13,1,1557466847610,,,\"}",

"host": "ae9f1bb8-ba5e-4b63-8f5f-2cc76db92f62",

"ip": "11.20.16.44",

"app-name": "air.flight.server",

"file": "/export/Logs/flight.skyeye/user\_behavior\_dash\_point.log",

"microsecond": 1557466849160322,

"group": "马驹桥"

}

# 定时任务

执行任务的机器是固定的，在properties中配置

## 1.1处理用户查询次数

定时将redis中的数据写入es，并删除redis中数据CalculateOfflineDataJob

执行时间：每天00:30执行

功能：

将用户查询次数缓存的数据从redis取出（hGetAll，key是当前日期的前一天，若今天是2019-5-9，那么key后缀是2019-5-8），组装后写入es（出发城市三字码、到达城市三字码、查询次数、查询日期2019-05-08、是否有结果，出发城市中文名、到达城市中文名），然后删除redis

## 1.2处理用户查询明细数据

执行时间：每30s执行一次

功能：

限流key减去1

While队列中有数据，异步多线程将中间队列的数据rPop取出并移除，写入es中

# 借助Redis

## 2.1查询次数缓存

结构：hash

Key：USER\_FLIGHT\_QUERY\_LINE\_查询日期(yyyy-MM-dd) 日期如：2019-5-8

Field：出发三字码:到达三字码:是否有结果0或1

value: 查询次数

有效时间：1天

数据示例：

Key：USER\_FLIGHT\_QUERY\_LINE\_QUERY\_NUM\_2019-5-9（只能查到当天的）

{"SYX:TSN:1":"13","XNN:YUS:1":"4","INC:SHA:1":"2","SYX:HET:1":"4","KWE:WUH:1":"8","CAN:NNG:1":"7"}

## 查询数据统计缓冲队列

结构：list

Key：USER\_FLIGHT\_QUERY\_LINE\_QUERY\_QUEUE

Value：用户查询明细对象的json

有效时间：7天

使用lTrim，只保留了1000个数据，队列中

数据示例

Value：["{\"arriCode\":\"DLC\",\"depDate\":\"2019-05-09\",\"deptCode\":\"SZX\",\"displayVenderIds\":\"\",\"doingVenderIds\":\"\",\"doneVenderIds\":\"\",\"extendTxt\":\"\",\"isHaveResult\":1,\"queryTime\":1557388989756,\"sourceType\":\"mobile\",\"userPin\":\"554242-98761043\",\"uuid\":\"377b4aa5b23041afb9d135d15c635bde1557388988408\"}"]

# 3.3限流

Key：USER\_FLIGHT\_QUERY\_LINE\_QUERY\_LIMITER

# 页面搜索

在controller中提供接口，.vue中使用.do访问

## 4.1对用户搜索航线时产生的数据，按航线和天统计查询(物理分页)

/flight/skyeye/userQueryLine/statsListPhysical.do

# 技术点

申请日志接收器：<https://cf.jd.com/pages/viewpage.action?pageId=74205018>

本项目中没有直接消费JDQ消息，而是通过storm消费，storm消费消息后将数据存入redis中。有worker会将数据写入es：每天凌晨将用户查询航线数据写入es并删除redis，每隔30s将用户明细数据从redis缓冲队列中拉取写入es中。

## 5.1 Kafka

## 5.2 storm

<http://bdp.jd.com/jrdw/jrctask/storm/overview.html>

可以查询运行日志

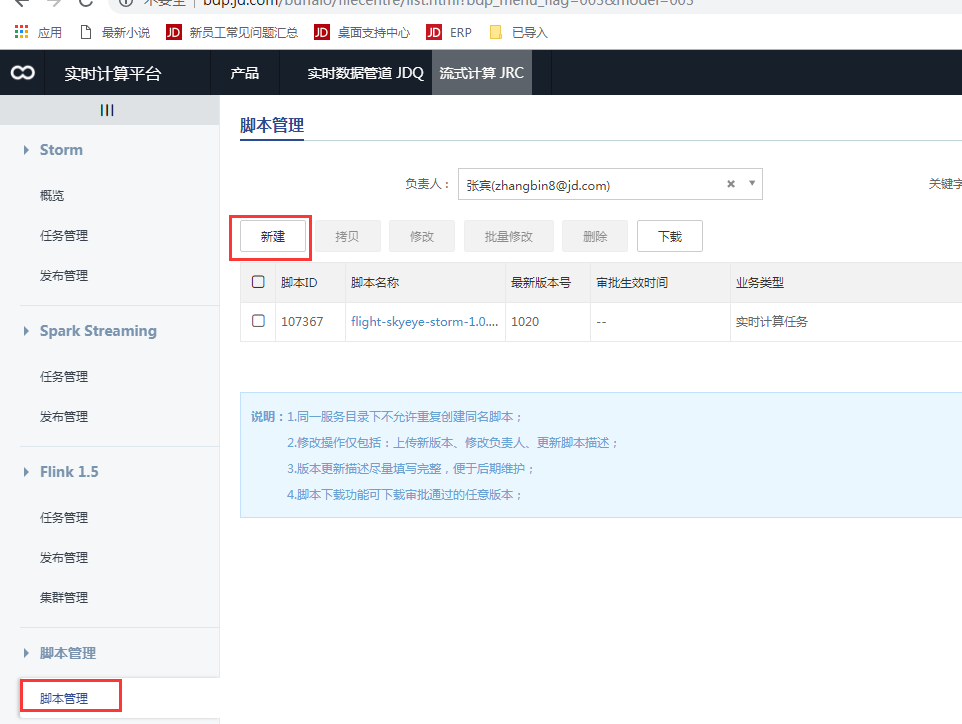
<http://bdp.jd.com/jrdw/jrctask/storm/listStorm.html?status=1>



Storm的jar包上传

<http://bdp.jd.com/jrdw/jrctask/storm/stormPublishList.html>

先上传脚本





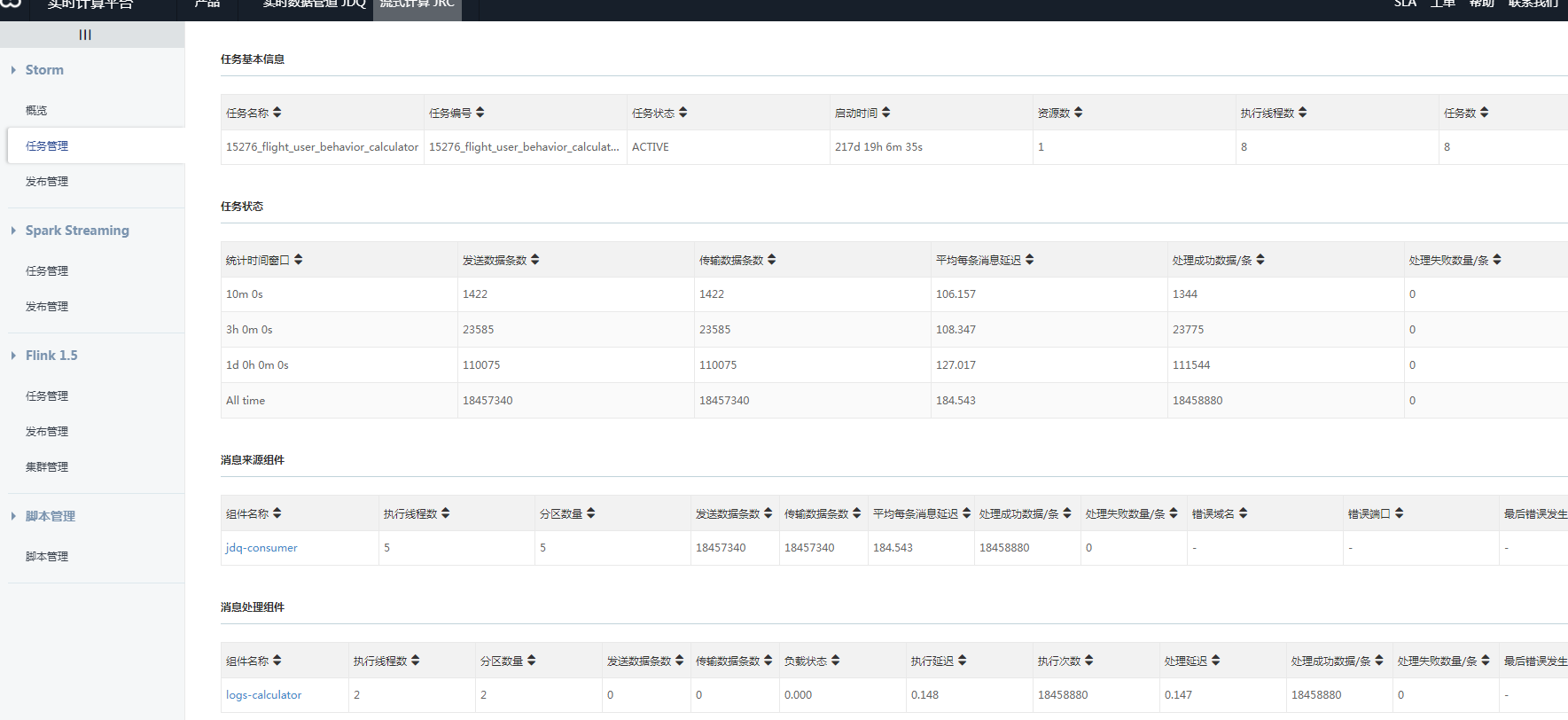
然后新建发布信息：



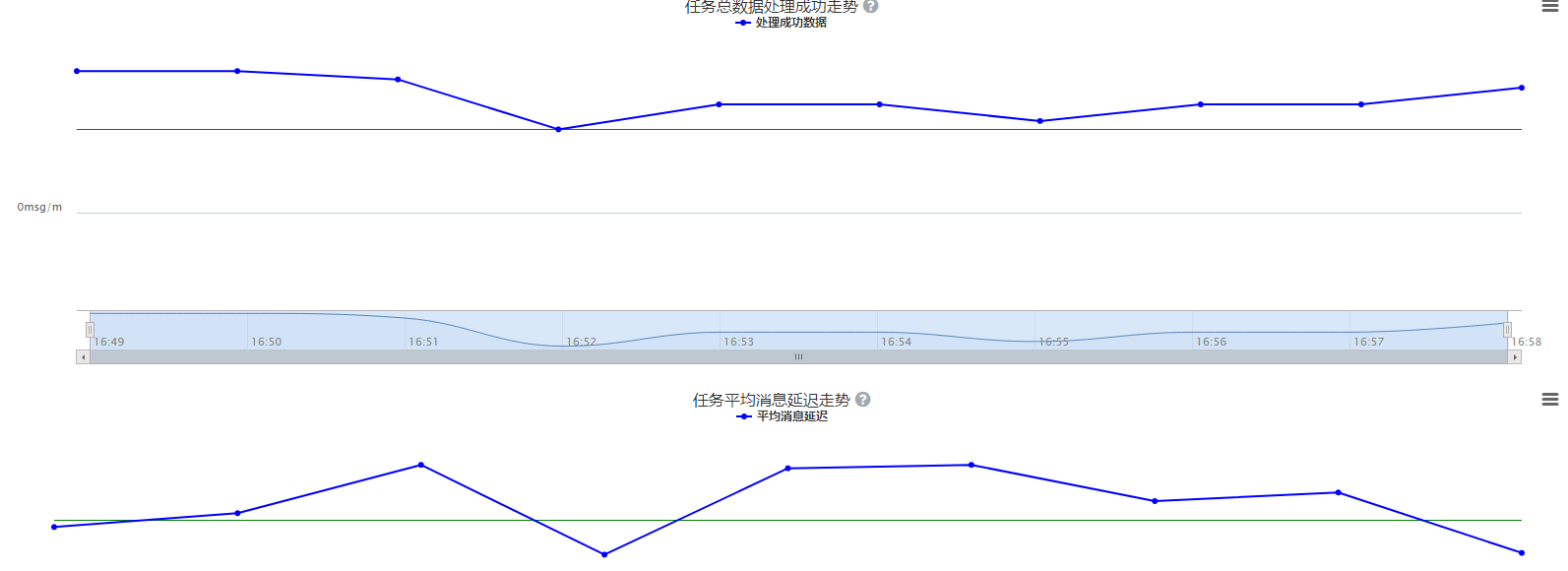
发布按钮发布任务



查看任务的基本信息



任务监控



## 5.3 es

## 5.4 redis