# 基于微服务系统的数据可视化展示平台软件V1.0

# 说明书

**西安电子科技大学**

## 软件描述

随着信息技术的发展，目前的互联网服务都是使用大规模分布式集群来实现的，其中微服务架构是企业最常用的一种分布式架构。微服务架构围绕业务将服务拆分成一个个模块实现独立部署，以去中心化的方式进行治理，可改善系统的部署和迭代，降低重构成本。但是，微服务架构固有的分布式的复杂性，不同服务间错综复杂的依赖关系，不同服务模块的部署，故障的预警监控等特点，给系统的运维更新带来了极大的困难。

本软件实现面向微服务系统的数据可视化展示，对于后期系统运维，迭代开发，组件和模块管理可提供重要支持。使得不同团队不需要理解其它服务模块上的代码语言和架构配置，就可以及时获取系统运行时的服务数据，以完善系统模块。

## 软件功能

本软件对面向微服务系统的数据可视化展示，后期系统运维，迭代开发，组件和模块管理提供支持，主要包括：

* 日志采集监控功能：通过微服务服务模块实现实时日志监控，历史日志查询。
* App状态监控功能：通过微服务服务模块实现App部署监控，App状态的监控。
* API和第三方请求监控功能：用户通过调用App服务具体方法，实现API实时请求统计，API历史请求统计，第三方实时请求统计，第三方历史请求统计。
* RPC链路跟踪功能：服务产生RPC调用，实现RPC链路跟踪可视化。

## 3.软件的设计与实现

### 3.1 软件结构设计

可视化平台软件整体框架如图1所示。主要通过Zookeeper作为App注册中心，将产生的日志数据上传到Kafka后，再从Kafka转存到ElasticSearch集合做持续化存储，支持用SQL语句对数据进行操作并在Web界面展示处理。RPC调用数据存储到HBase数据库中用于可视化展示。以下是可视化平台设计的整体框架。



图1 基于微服务系统的数据可视化展示平台整体架构图

本系统主要基于SpringBoot框架，借助不同的数据库完成监控数据的存储操作，以Web的形式实现可视化。

### 3.2 软件流程描述

本软件的使用可分为4种组件开发流程来介绍，即日志采集监控，App状态监控，API和第三方请求监控，RPC链路跟踪。

**日志采集监控功能：**首先用户启动微服务系统服务模块，通过Zookeeper注册中心注册App，将基本的应用信息存储到App-info表中，监控平台会根据App-info表显示过滤信息；自定义logback采集器，当App运行时，将日志信息通过Kafka格式化处理后存储到ES集群App-log中；用户在监控平台选择相应的输入后，借助SQL语句获取ES集群中的日志信息实现可视化展示。

**App状态监控功能：**用户启动微服务系统服务模块，通过Zookeeper注册中心注册App，将基本的应用信息存储到App-info表中；当用户查看App运行状态，监控平台获取App-info表中数据实现可视化展示。用户启动或关闭App服务时，会借助Alarm功能模块对上下线应用，用发送邮件的方式向用户报警。

**API和第三方请求监控功能：**用户通过调用App服务具体方法，产生API和第三方调用，将事件日志信息存储到数据库中，并利用Kafka对日志数据格式化处理后存储到ES集群event-log中。

**RPC链路跟踪功能：**服务产生RPC调用，将调用基本信息注册到Zookeeper中心，并存储到数据库中，作为输入调整筛选。根据Dubbox和Google Dapper的原理，在RPC调用过程中采集具体链路数据形成Tracer链路数据存储到HBase数据库中。

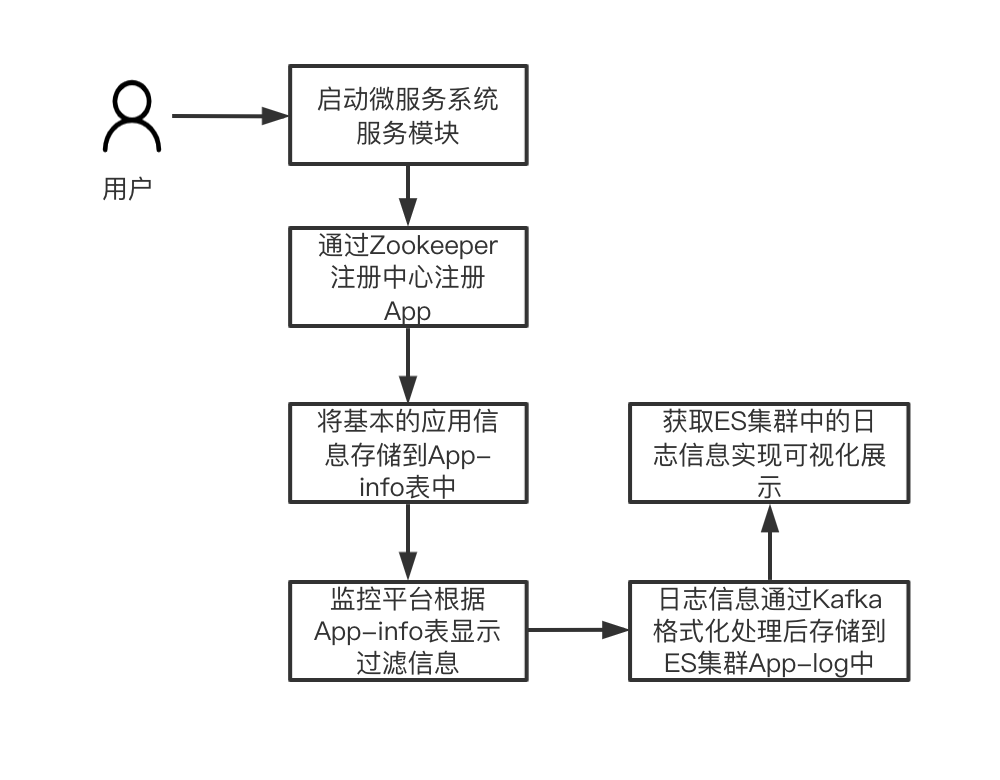


图2 封日志采集监控单元流程图

日志采集监控单元的流程如图2所示。用户在软件中选择“日志采集监控”向导，然后按照向导的一步步提示和要求就可以实现日志采集监控。

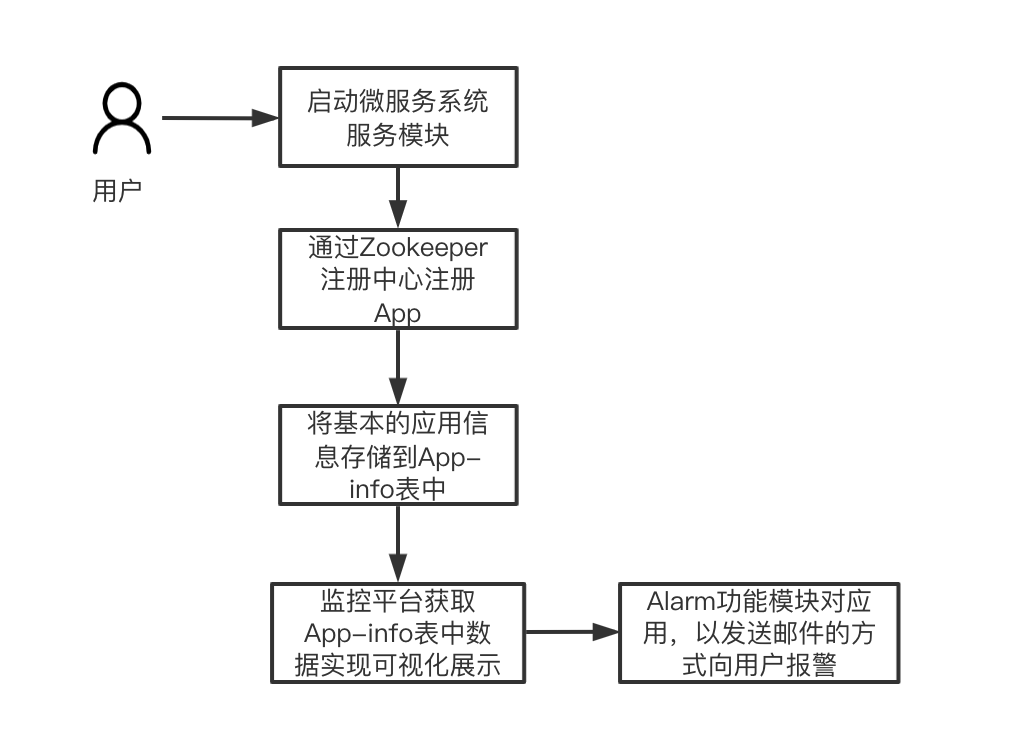


图3 App状态监控功能流程图

App状态监控功能流程如图3所示。用户在软件选择App状态监控，然后按照向导的一步步提示和要求就可以实现App状态监控。

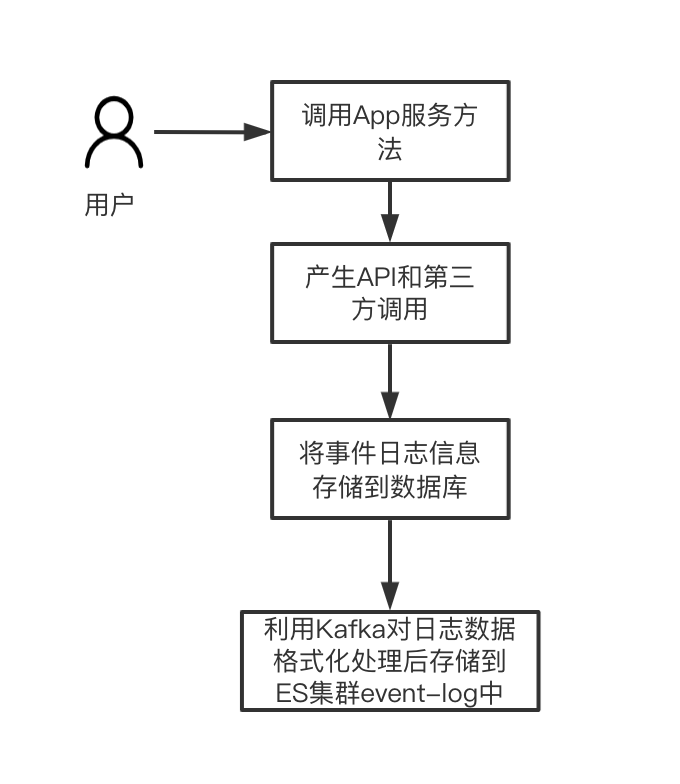


图4 API和第三方请求监控流程图

API和第三方请求监控功能流程如图4所示。用户在软件选择API和第三方请求监控，然后按照向导的一步步提示和要求就可以实现API和第三方请求监控功能。

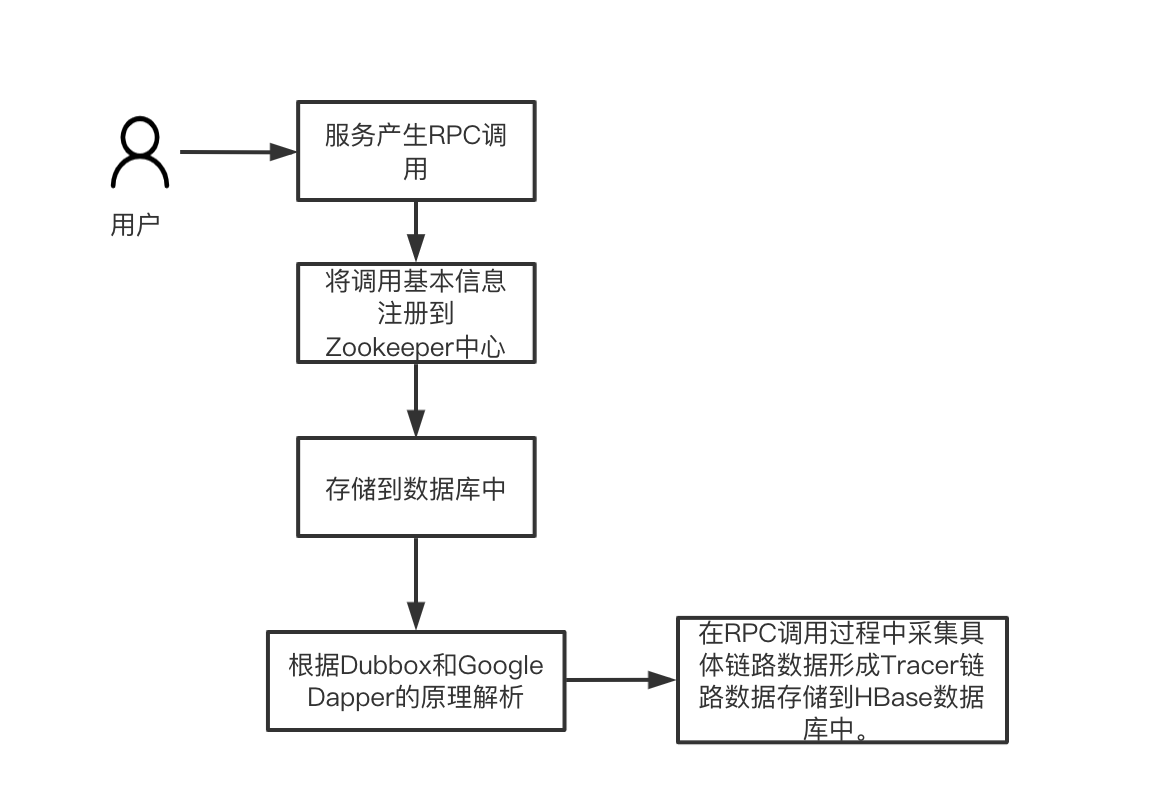


图5 RPC链路跟踪功能流程图

RPC链路跟踪功能流程如图5所示。用户在软件选择RPC链路跟踪，然后按照向导的一步步提示和要求就可以实现RPC链路跟踪功能。

### 3.3 典型应用

本软件为用户提供了基于微服务系统的数据可视化展示平台软件，可以广泛的应用于基于微服务的软件系统的开发中。

现以边电子商城为例，介绍本软件的应用场景。

1. 用户发生购物事件之后可以实时用户操作日志监控，用户操作历史日志查询。
2. 可以实现电子商城部署监控，商城状态的监控。用户实现购物之后，可以用户通过调用App服务具体方法，统计API实时请求，API历史请求，第三方实时请求，第三方历史请求的次数。
3. 用户实现购物之后，可以用户通过调用电子商城服务具体方法，实现API实时请求统计，API历史请求统计。
4. 对服务产生RPC调用，实现RPC链路跟踪可视化。

## 4.运行安装说明

由于本软件是以IDEA插件形式开发的，因此本软件的安装要依赖于IDEA软件的安装。

用户安装完成IDEA之后，才能安装本软件。IDEA插件安装的方式有许多中，本说明书就其中一种进行说明：

1. 准备好需要安装的本软件的插件文件。
2. 找到IDEA软件的安装目录，插件文件复制到自己的文件夹下。
3. 重新启动软件。

## 5.使用说明

### 5.1数据可视化展示平台软件主界面

图6 数据可视化展示平台软件主页面

数据可视化展示平台软件主界面，其界面如图6所示。

### 5.2 平台功能使用

5.2.1日志查询

首日志查询的实现界面主要分为实时日志监控，历史日志查询和运用SQL语句进行日志的过滤检索三个界面。

其中实时日志监控界面如图7所示，输入机器名，应用名和相关关键字对日志信息进行筛选；

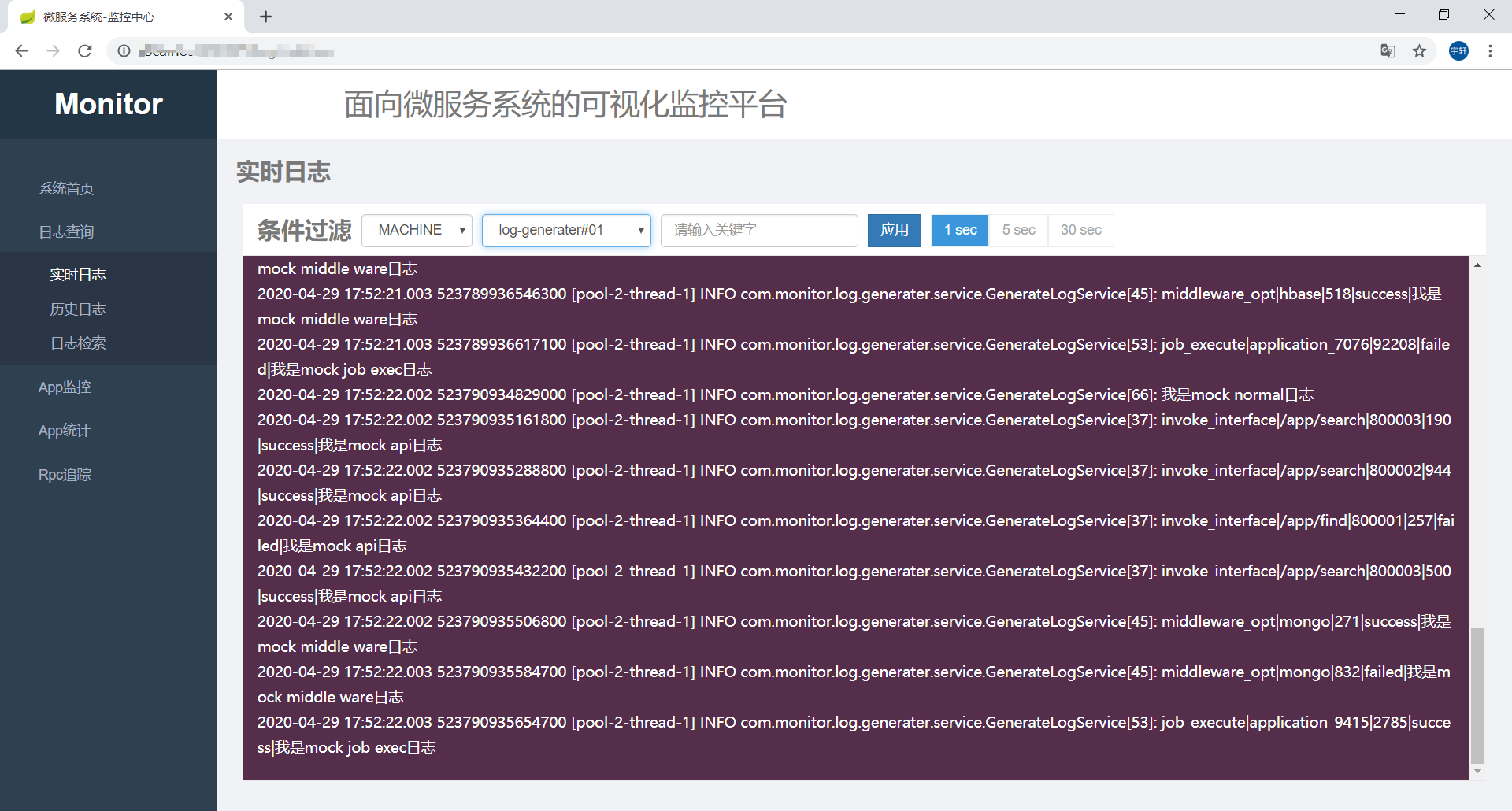


图7 实时日志查找

历史日志监控界面如图8所示，输入相关机器名，应用名，日期范围以及日志类型对日志进行查询；

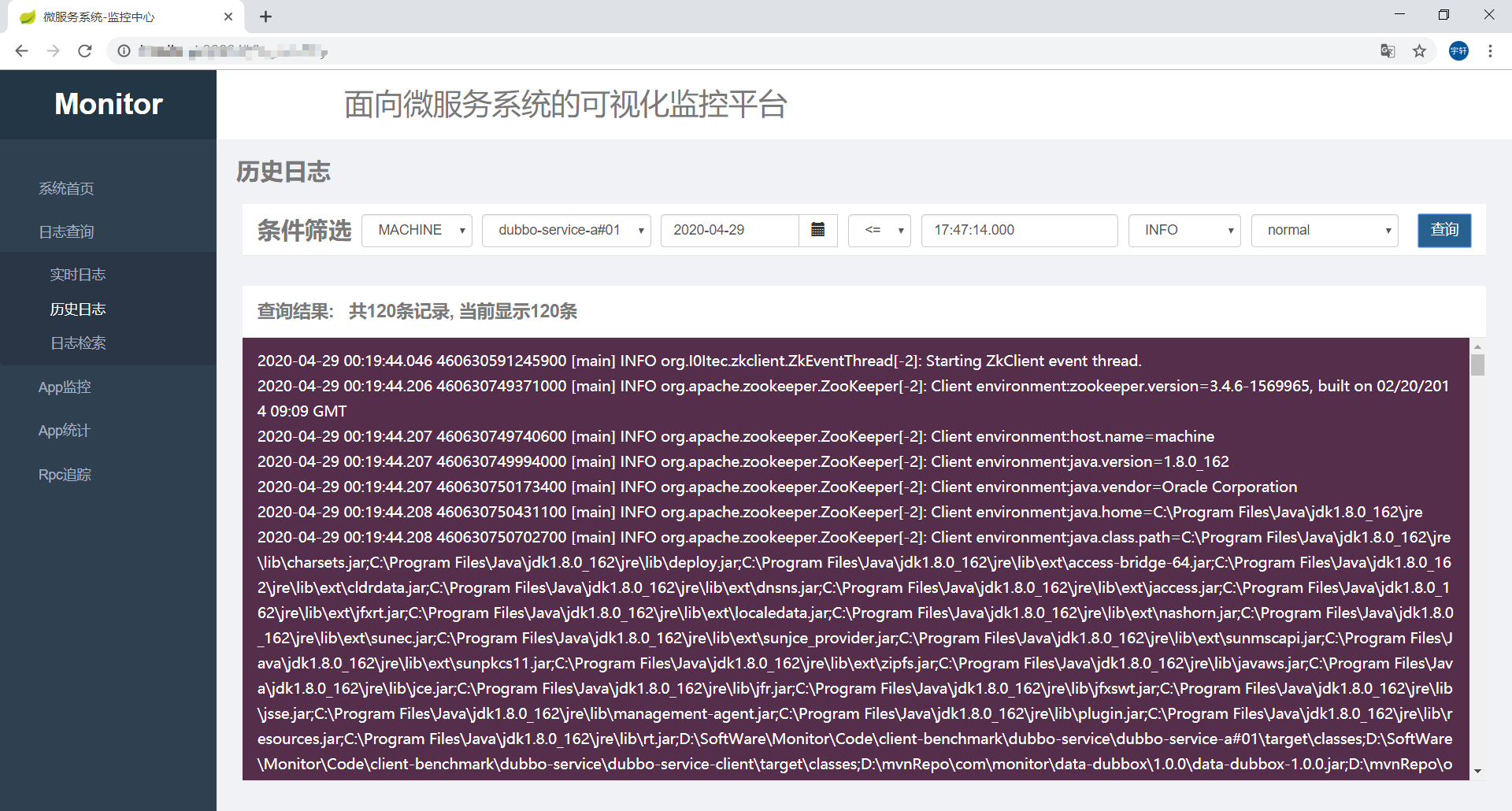


图8 历史日志查找

5.2.2 App监控

App监控功能主要包括业务模块运行状态和部署位置的监控两个页面。其中App状态监控界面如图9所示可输入机器名和应用名对App状态进行查询。App部署监控界面如图10所示也可输入机器名和应用名进行搜索，通过查询反转改变输入检索顺序。

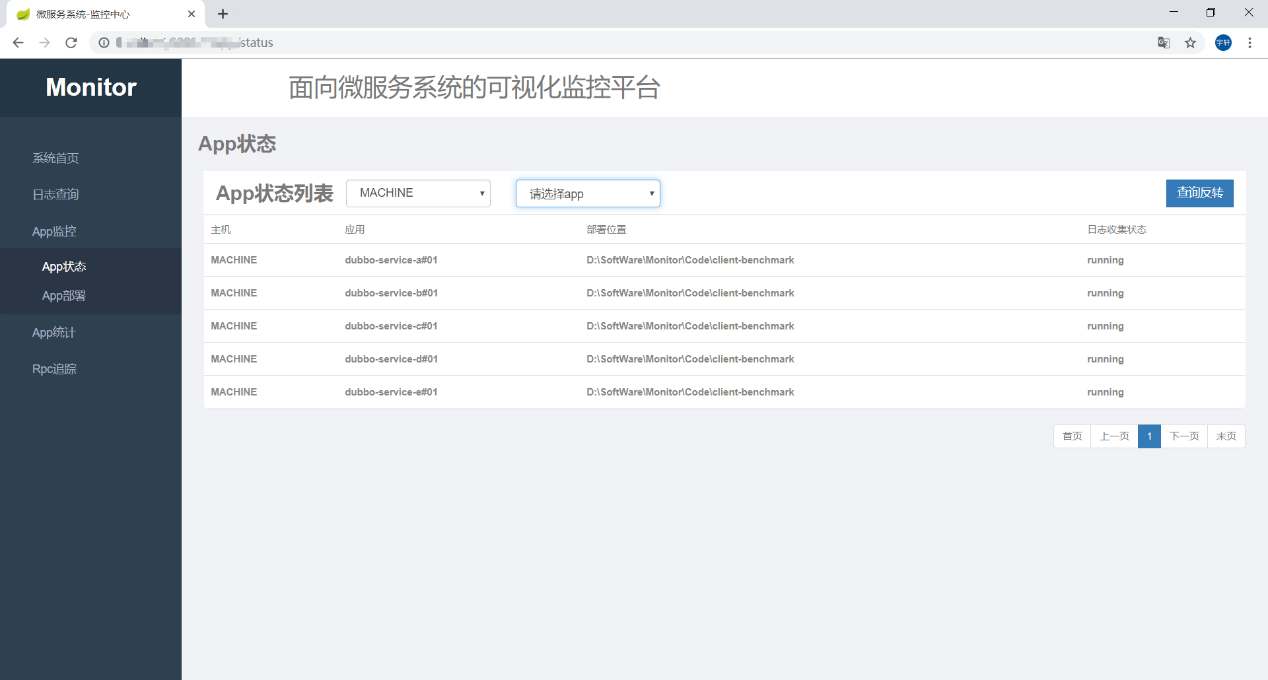


图9 App状态监控

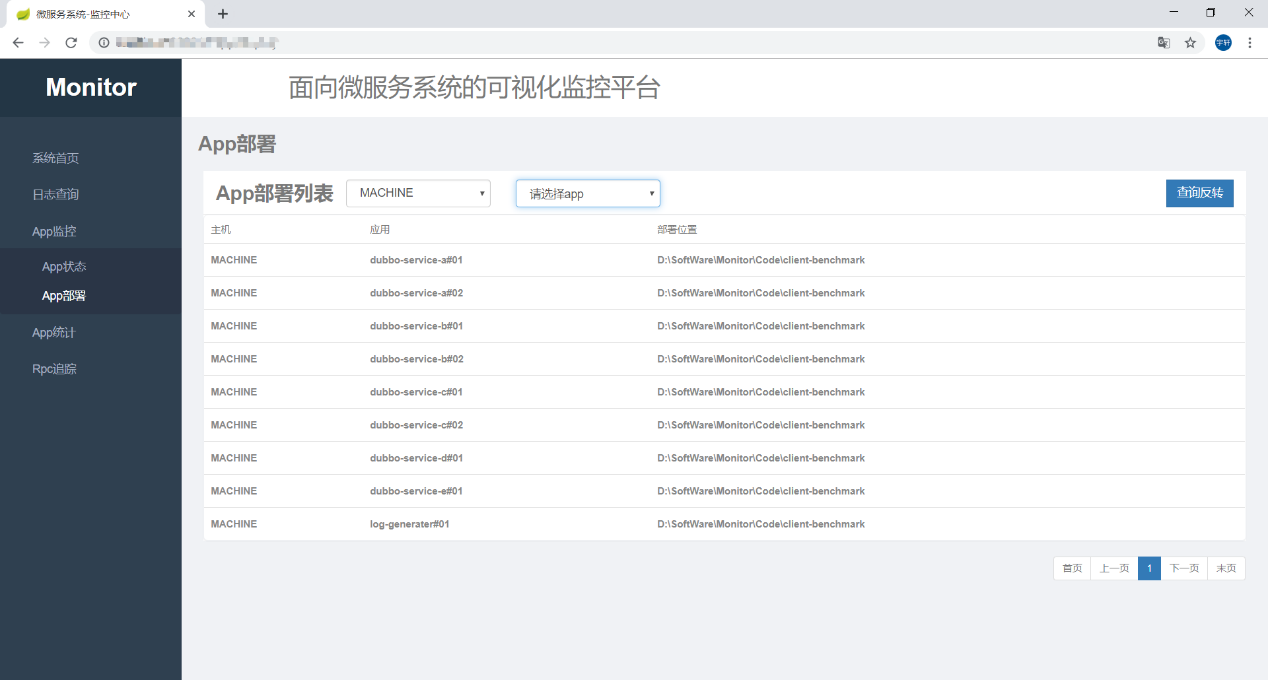


图10 App部署监控

App调用统计主要是对业务模块API调用和第三方请求监控统计的可视化展示，通过折线图，柱状图等形式直观的展示出来。分为API调用统计和第三方请求统计两个页面，API请求统计界面可分为API实时统计和API历史统计两个功能如图11和图12所示，通过选择请求类型和展示方法实现可视化统计；第三方请求统计分为第三方实时统计和第三方历史统计两个功能如图13和图14所示，可输入相关请求类型和展示图表形式实现可视化统计。



图11 API实时请求统计



图12 API历史请求统计



图13 第三方请求实时统计



图14 第三方请求历史统计

5.2.3RPC追踪

RPC追踪主要是参考Zipkin的表现形式，链路跟踪界面如图15所示，输入RPC调用的模块和方法名称，发生的日期范围，追踪RPC调用具体的实现过程，包括该方法调用的下级方法，调用耗时等，实现过程的可视化。

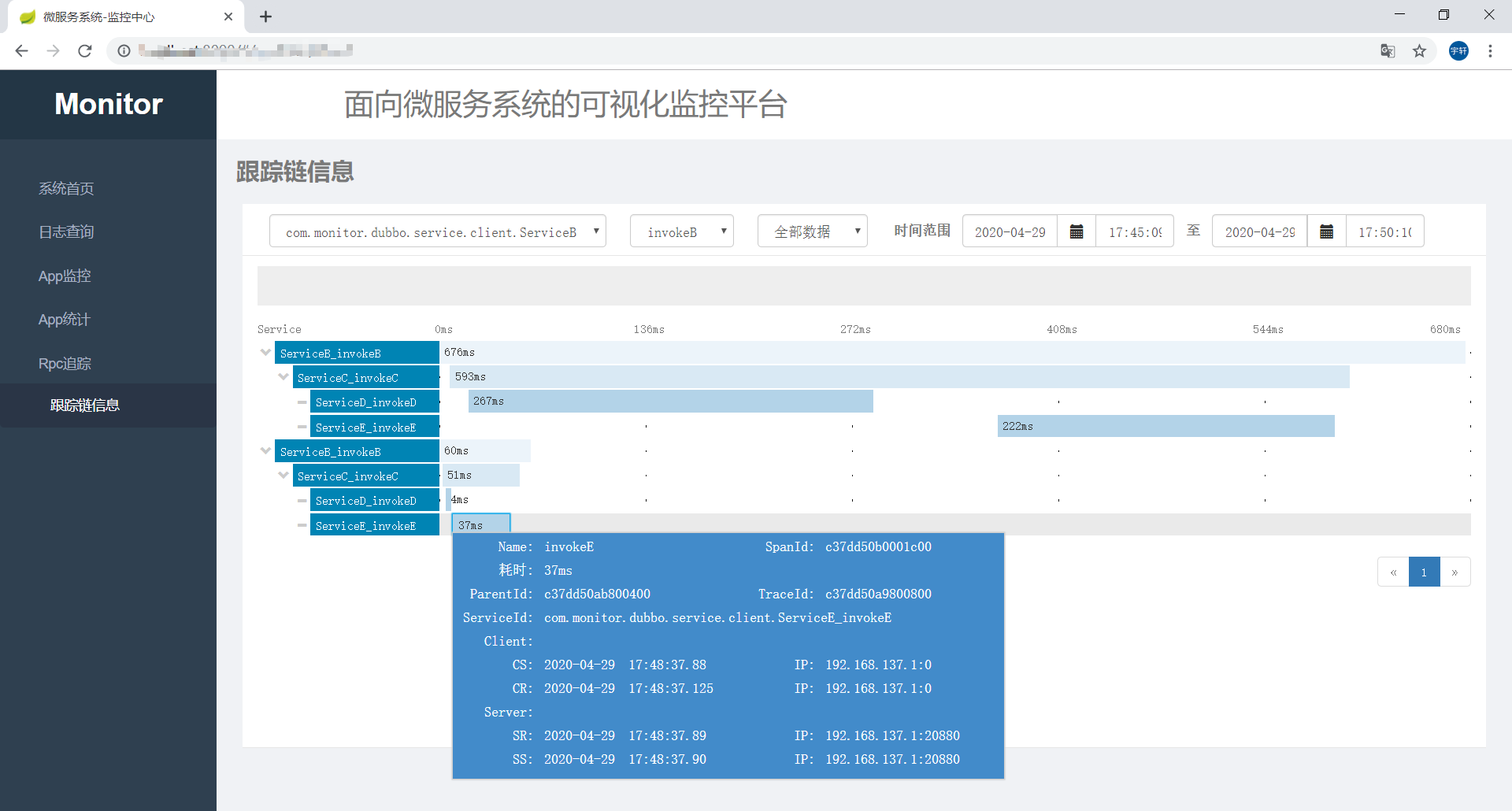


图15 RPC链路跟踪