

从所有教程的词条中查询...

首页 > 慕课教程 > Go工程师体系课全新版 > 3. jaeger安装和架构

- 全部开发者教程
1. go最常用的设计模式 - 函数选项
2. 单例模式和懒加载
3. 测试金字塔
- 第23周 protoc插件开发、cobra命令行
1. protoc调试源码
2. protoc自定义gin插件
- 第24周 log日志包设计
- 日志源码
- 第25周 ast代码生成工具开发
- 错误码
- 第26周 三层代码结构
- 通用app项目启动



bobby · 更新于 2022-11-16

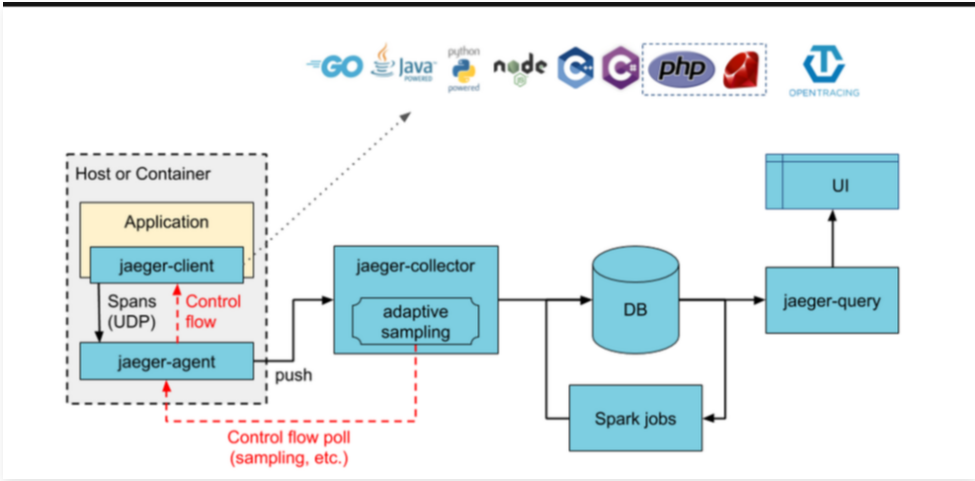
上一节 2. 链路追踪技术... 4. opentracing... 下一节

## 1. 安装

<> 代码块

```
1 docker run \  
2   --rm \  
3   --name jaeger \  
4   -p6831:6831/udp \  
5   -p16686:16686 \  
6   jaegertracing/all-in-one:latest
```

## 2. 架构



## Jaeger组成

**Jaeger Client** - 为不同语言实现了符合 OpenTracing 标准的 SDK。应用程序通过 API 写入数据，client library 把 trace 信息按照应用程序指定的采样策略传递给 jaeger-agent。

**Agent** - 它是一个监听在 UDP 端口上接收 span 数据的网络守护进程，它会将数据批量发送给 collector。它被设计成一个基础组件，部署到所有的宿主机上。Agent 将 client library 和 collector 解耦，为 client library 屏蔽了路由和发现 collector 的细节。

**Collector** - 接收 jaeger-agent 发送来的数据，然后将数据写入后端存储。Collector 被设计成无状态的组件，因此您可以同时运行任意数量的 jaeger-collector。

**Data Store** - 后端存储被设计成一个可插拔的组件，支持将数据写入 cassandra、elastic search。

**Query** - 接收查询请求，然后从后端存储系统中检索 trace 并通过 UI 进行展示。Query 是无状态的，您可以启动多个实例，把它们部署在 nginx 这样的负载均衡器后面。

分布式追踪系统发展很快，种类繁多，但核心步骤一般有三个：代码埋点，数据存储、查询展示

 我要提出意见反馈

