相关架构，描述一下对比实验的相关架构。一页。

本文基于微服务思想构建读写分离的以太坊后台服务系统，底层基于Docker容器构建多Geth客户端的以太坊服务平台，并基于状态空间反馈设计合理的负载均衡策略将任务合理分发到对应客户端。为了验证本系统在性能上的进步，本文对照单体应用架构的以太坊服务系统做对照实验。

单体应用在具有不同的逻辑模块化架构，但应用程序被作为一个单体进行打包和部署，系统使用Java实现，基于SpringBoot框架构建并编译成WAR包直接运行于应用服务器Tomcat中。底层客户端的调用同样使用单客户端模型，因此无需负载均衡策略。



如图所示为对照实验的单体应用架构，该应用内部同样区分为读和写逻辑模块，但是在实际的应用架构中是作为一个整体部署的。底层的Geth客户端的服务也是单个的Geth客户端对整个应用提供服务。