依照实验设计，本文通过模拟请求测试并发性，并同时检测各个容器的内存和CPU的占用情况。如图5. 为并发测试结果。

图5.10并发测试结果

从上图中可以看出，在200并发量的时候，单客户端的完成速度高于多客户端，这是因为在多客户端之间的切换需要消耗时间。因此在低并发情况下，单客户端架构性能优于多客户端架构。在并发量到600及往上的时候，多客户端架构的优越性逐渐体现出来，多客户端的任务完成时间小于单客户端的任务完成时间，而且随着客户端的数量的增加，任务完成的时间变得更短。在并发量达到1200的时候，单客户端因为阻塞问题导致无法正常完成功能。但是多客户端架构依然在没有影响的情况下完成任务。

本文在对并发性能测试的同时，为了测试在资源消耗上的优势，对容器的CPU占用率和内存占用率也进行了检测。本文涉及的容器较多，本测试仅选取其中最有代表性的容器进行说明。对于单体应用架构，本文考虑检测两个容器，首先是以太坊Geth客户端运行的容器，其次是以太坊服务系统的服务端所运行的容器。对于本文设计的架构，考虑在3个Geth客户端提供服务的情况下，其中一个Geth客户端运行的容器，一个读业务组件运行的容器和一个写业务组件运行的容器。如图5.11是CPU占用情况，5.12为内存占用情况。