## Jmeter压测SpringBoot SpringBootWebFlux应用

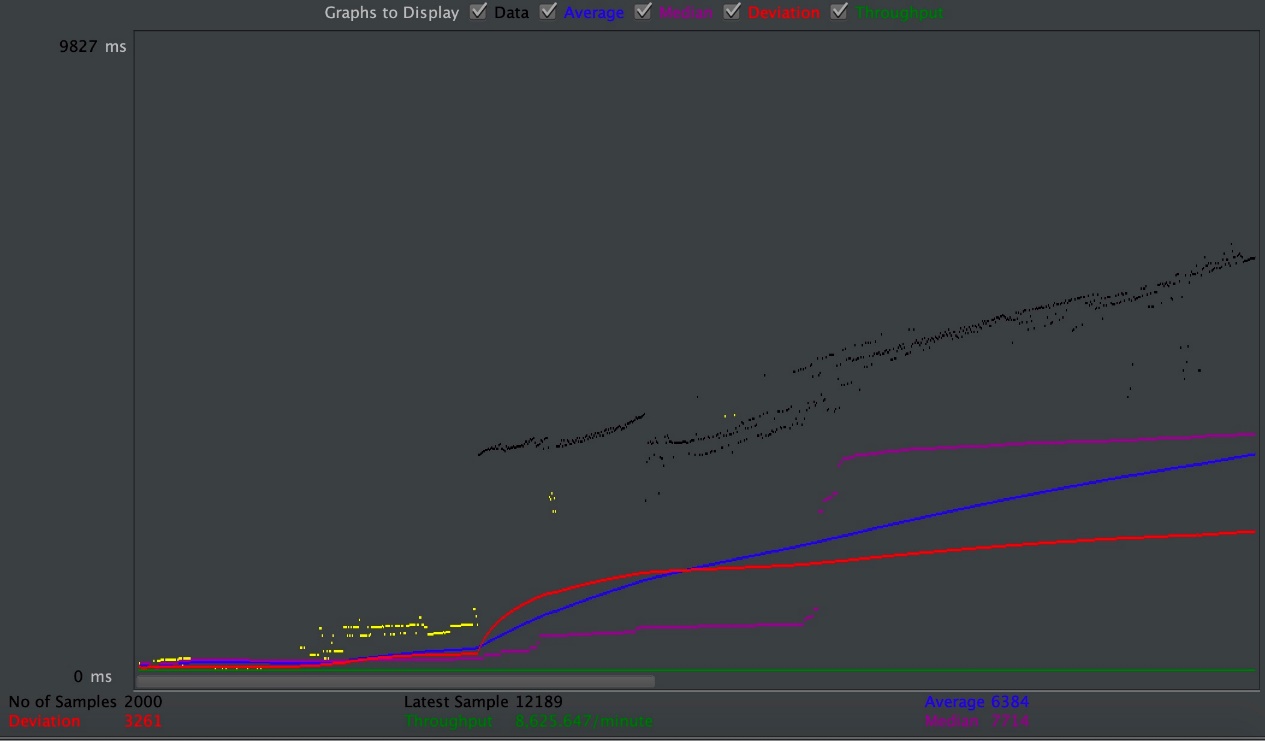
评价指标：

1、两者在达到临界error率时的线程数大小

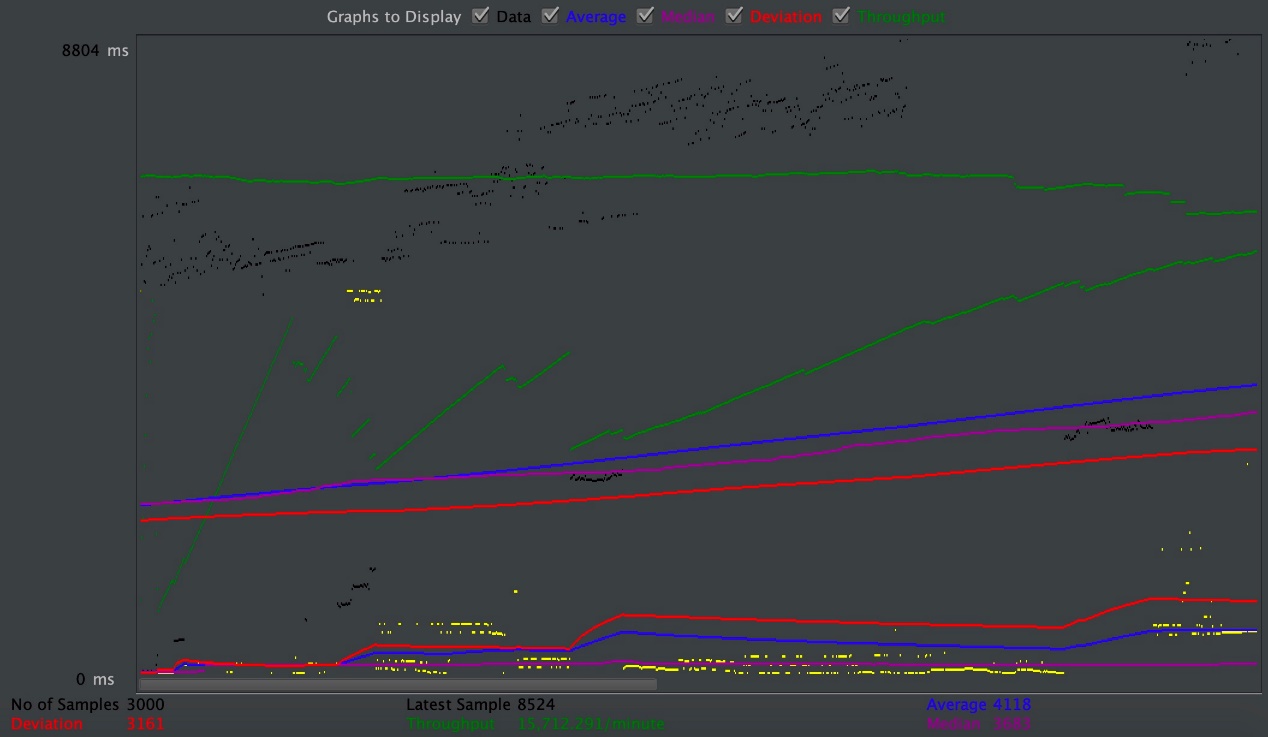
2、图表中的吞吐量比较（绿线）、平均值比较（蓝线）、中间值比较（紫色值）、偏离比较（红色）

3、聚合报告中的90%Line和KB/sec

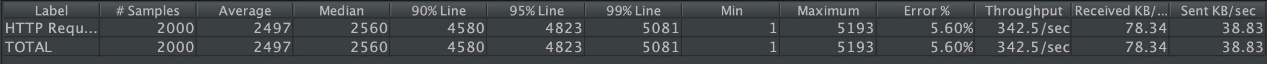
经过压测对比发现，在Dragonwell JDK上运行时（本地机器环境），SpringWebFlux应用在1S内2000线程数的情形下，error率为14.9%，平均响应时间3364，吞吐量8625.9 / min



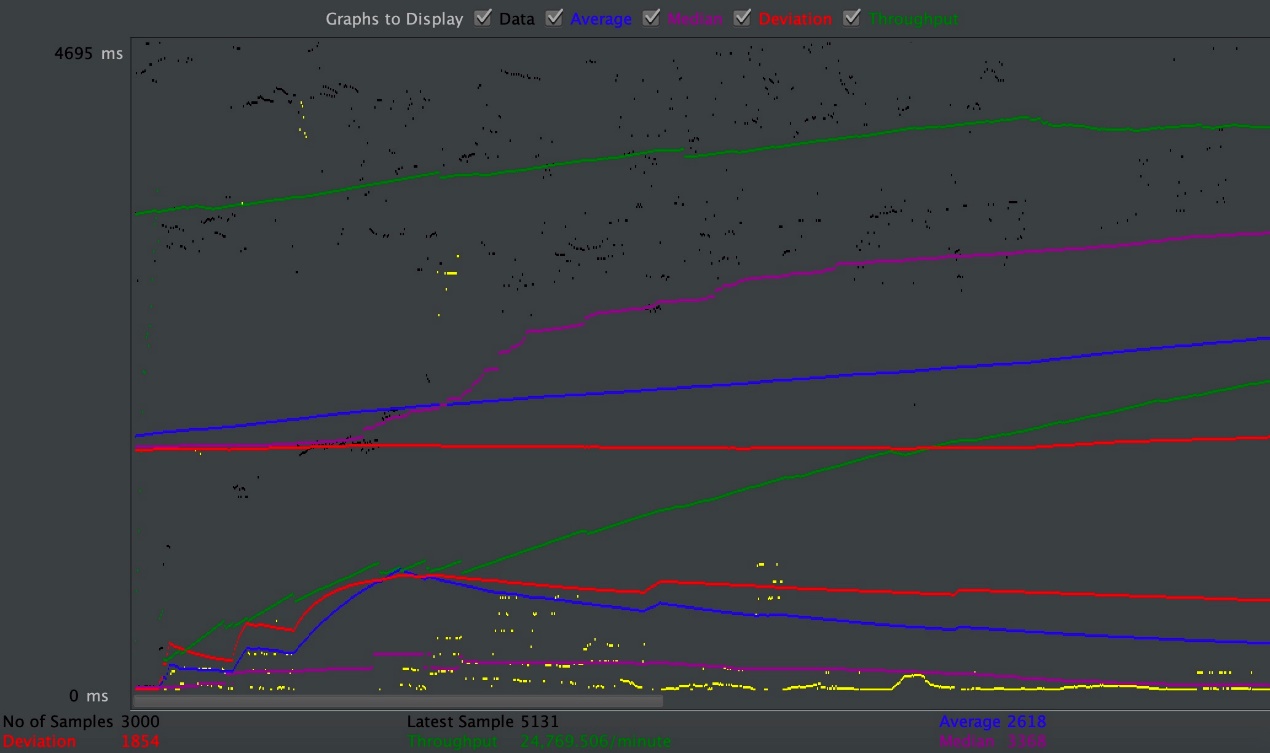
在3000线程数情形下，error率为33.4%，平均响应时间3683，吞吐量15712.9



在Oracle JDK下执行同样的应用时，2000线程数情形下，error率为5.6%，平均响应时间为2497，吞吐量为20547.9/min



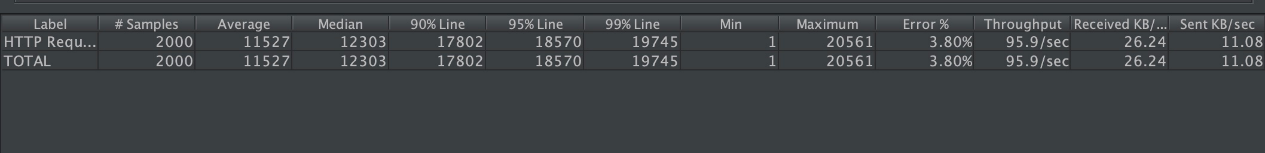
3000线程数情形下，error率为33.4%，平均响应时间2518，吞吐量24769.5 /min

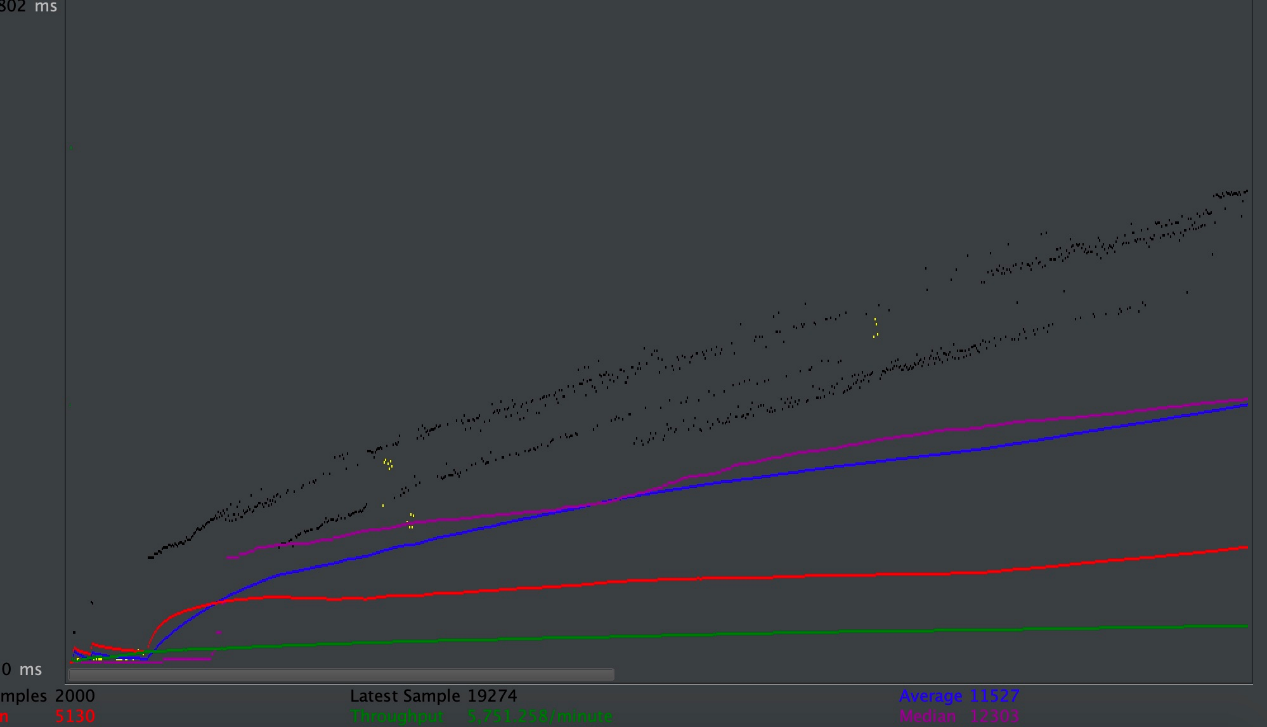


对SpringBoot应用类似情形的压测，数据如下：

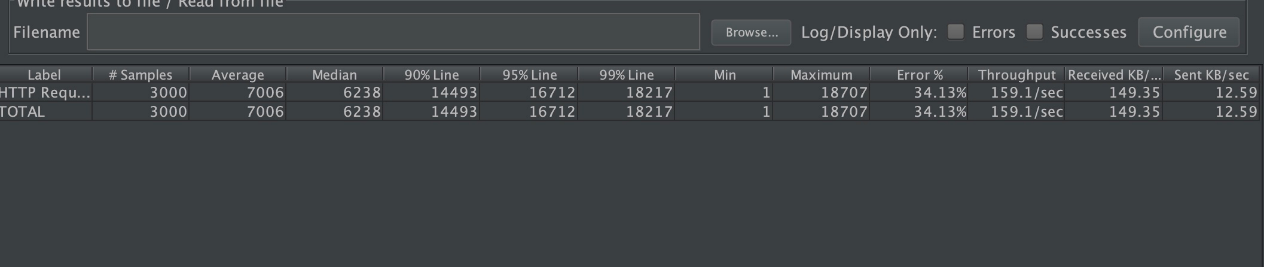
Dragonwell JDK的表现为：

2000线程数下，error率为 3.8%，平均响应时间11357，吞吐量5751.2 /min

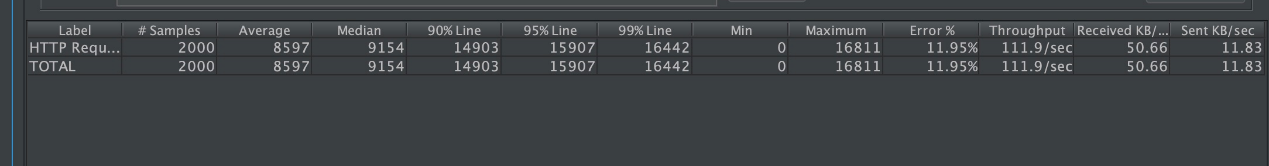


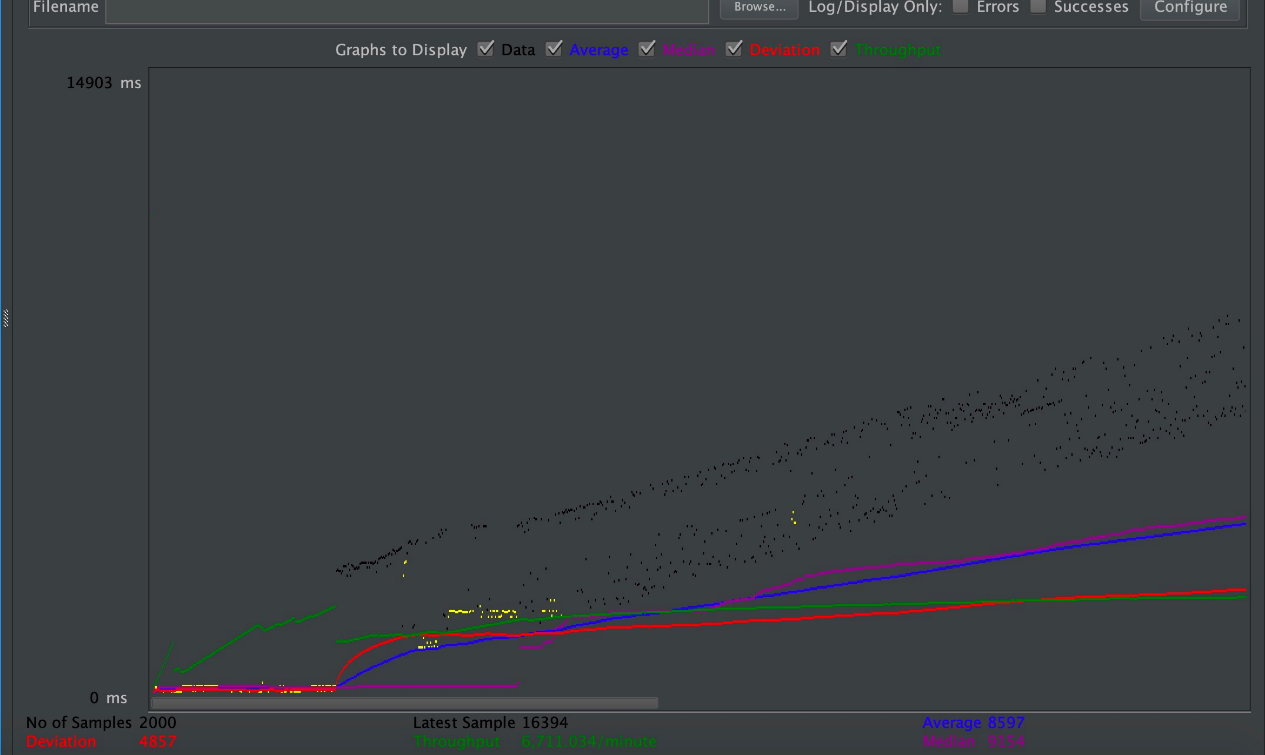


3000线程数下，error率为34.13%

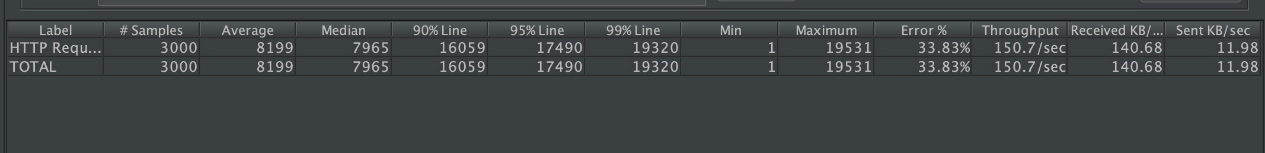


Oracle JDK下，2000线程数时，error率为11.95,平均响应时间8597，吞吐量6711 /min





3000线程数时，error率为33.83%



测试结论：

SpringBootWebflux应用 oracle jdk性能优于dragonwell jdk，体现在高并发情形下error率更低，吞吐量更大，平均响应时间更短。但是极限性能没有太大差别。

SpringBoot crud应用 dragonwell jdk性能优于oracle jdk，同样体现在error率，吞吐量和平均响应时间三点，极限性能差异不大。