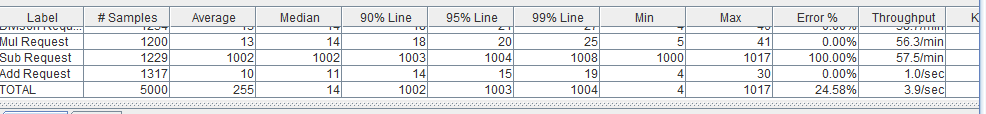
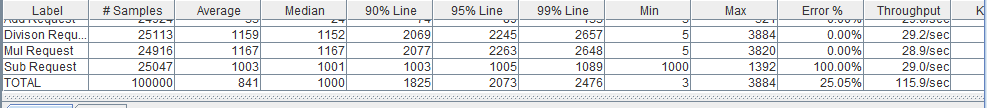


For 1000 random calls



For 5000 random calls



For 100 users 1000 random calls (100000 calls)

1. 1000 calculator calls: Invoking 1000 calculator calls from single user.
2. 5000 calculator calls: Invoking 5000 calculator calls from single user.
3. 10000 calculator calls: Invoking 1000 calculator calls from 100 users.

With number of requests from the same user increases doesn’t affect the average response time, but when the number of concurrent users increased, there is significant delay in the average response time, Since with concurrent users all the concurrent requests at same time must be queued and processed , we can find latency in the response.

**Performance in rest service for calculator.**

1. 1000 calculator calls: **1**: Invoking 1000 calculator calls from single user.
2. 5000 calculator calls: 1: Invoking 5000 calculator calls from single user.
3. 10000 calculator calls: **505:** Invoking 10000 calculator calls from 100 users.

During a request to server from same user, the average time of response doesn’t vary with the number of requests, but as number of requests from different user’s increases (like 100 users) then the performance of the node gradually decreases.

Thus Restful web services perform same way as in JAX-WS with less number of users but with increase in the number of users, Restful web services scale well when compared to JAX-WS service.

Rest has much better performance because it runs on top of HTTP which doesn’t have any overhead like in JAX-WS service. Soap needs handlers and parsers in the exchange.

Data is encoded in rest service which takes less bites of memory than the XML data. Data exchanged in the REST service is JSON which is very lean data without any xml overheads.