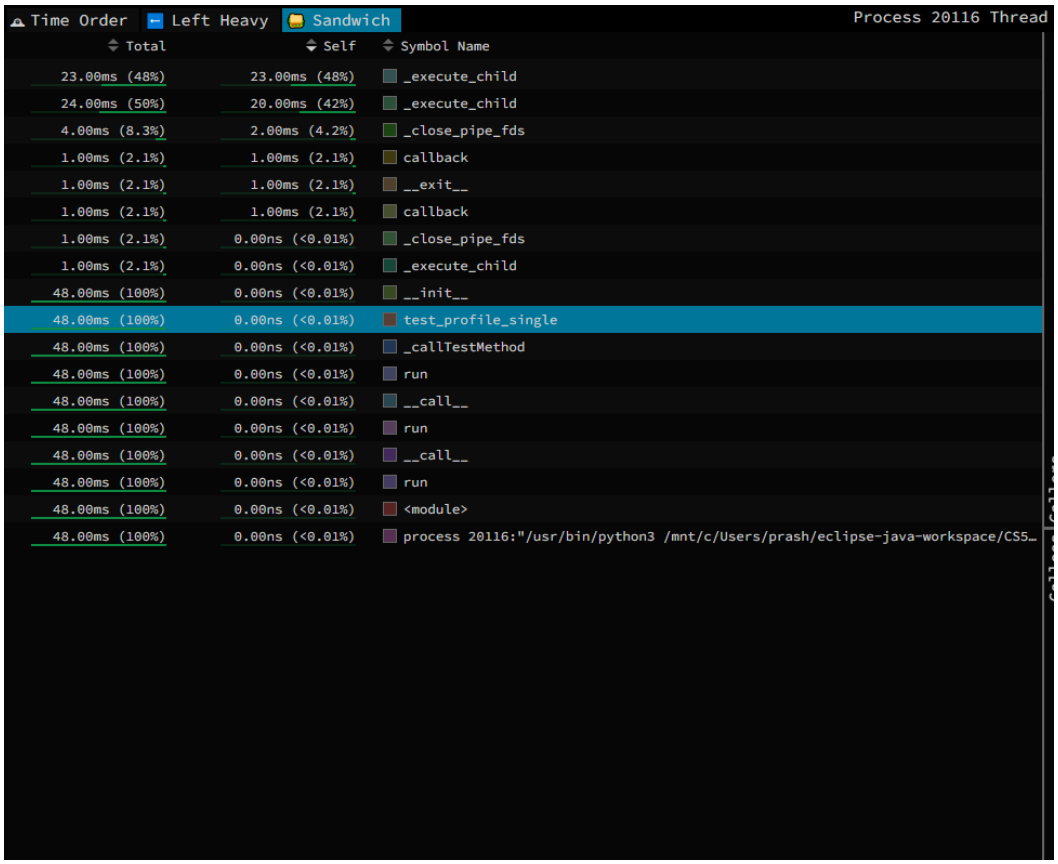
Single Client

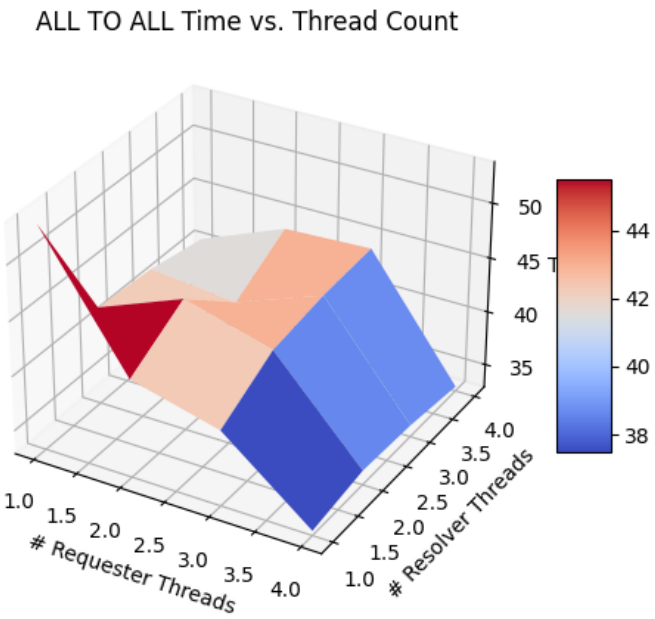
Here is the flamegraph of executing as a single client. The test_process_single takes **40ms** to complete under no other load with a single QUERY_MESSAGE request in the linear topology.



Multiple Clients

All - to - All

The RMI Server is already multithreaded but here we see that by adding more clients to the system the time taken per request reduces below 38. It may owing to the fact that requests can be served with fewer TTLs in the all-to-all topology.



Linear

The following is the plot of several clients being used in the server with Linear topology. The number of clients were added from 1 up to 5 and plotted in this 3d diagram. As we can see the round trip time for the QUERY_MESSAGE increases by adding more processes. The linear approach may incur linear time delay in finding the SuperPeer that has the designated key.

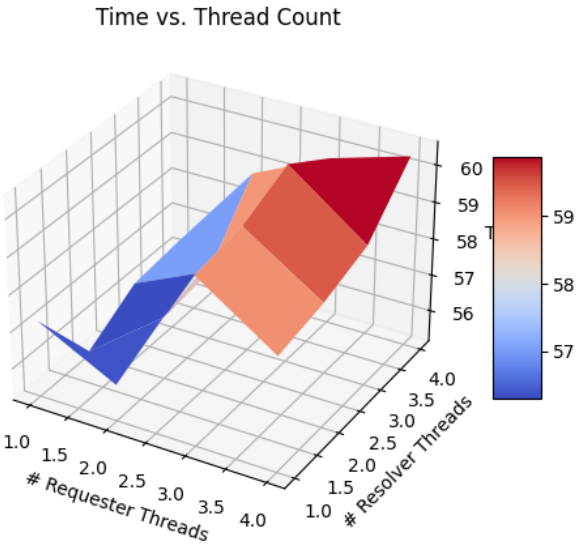


Fig. Linear time taken Plot.

Real-World

In the real world , all -to- all topology takes constant amount of time to find the SuperPeer that has the key associated. In just within 2-3 TTLs decrements it is possible to find the SuperPeer that can serve this client's request. Whereas, the linear topology makes O(n) requests searching along the chain in order to find the SuperPeer. This adds a lot of latency due to message delays and the reason why it is not used in the real world is because if even one SuperPeer is unreachable there will be a network partition and there is no builtin redundancy for this scenario.