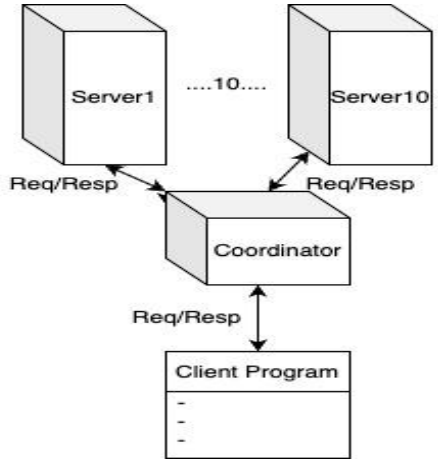
Distributed Log Querier - MP1

Rahul Shivaprasad (rahul11)
Prajwal Kiran Kumar (prajwal5)

Design

Our setup includes ten machines, each of which can be clients/servers. We set up one of the VMs to be a coordinator that handles all requests from the clients and is responsible for abstracting the client from the details of the server IP addresses, and internal functionalities of the grep command. At a high level, the client issues a “distributed grep” command to the coordinator with all the required parameters. The coordinator then performs a grep on all the log files on the servers through individual RPC calls. The servers each send a response to the coordinator, which is consolidated and returned back to the client.



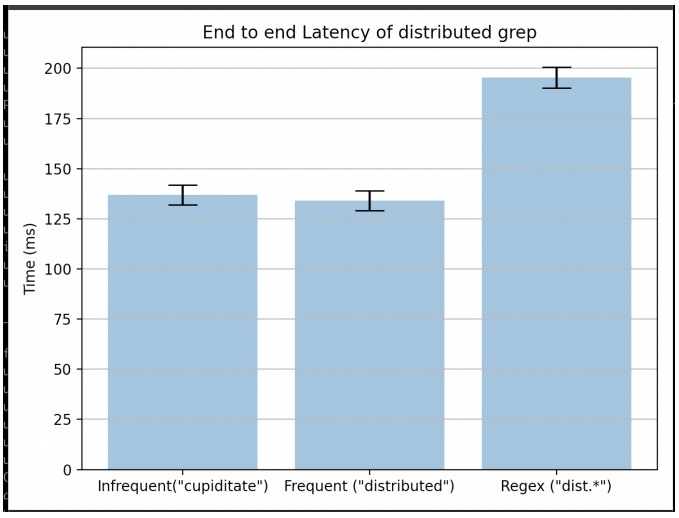
Algorithm

To perform the distributed grep, the client issues a single RPC call to the coordinator with the <pattern> <grep_options> and the <target_files>. The coordinator, which has the information (IP Addresses) of all the members of the distributed cluster, issues concurrent RPC calls to all the servers in its membership list. The coordinator waits on a go channel to receive grep responses from all the servers. Upon receiving all the responses, the coordinator computes the total number of matches of the particular pattern and returns the entire response back to the client. The response has an individual match count of the given patterns along with the total number of matches across all the files in the distributed system.

Test Strategy

The algorithm requires a distributed testing strategy for running integration tests. We have written a test_client that is able to create and delete log files from all the machines of the cluster through RPCs. Once the test_client is run, the original grep function is run, to test for frequent, infrequent and regex patterns. The test_client cleans up by deleting the log files in the cluster after each test is run. The output received by the test script for each test is validated against known values for the pattern.

Results



The results of the experimental test runs indicate that distributed grep heavily depends on the network latency of the distributed cluster for non regex searches. The mean time to grep a frequent pattern is less than that of an infrequent pattern owing to the fact that grep can stop scanning a line once the required search string is found.
The regex mean time is dependent on the complexity of the search pattern. For the pattern “dist.*” vs “dist.*\s”, we see that the latter takes almost 25x the latency.