CS 425 Distributed Systems

MP 3 Report

Anish Shenoy (ashenoy3) & Omkar Lokhande (lokhand2)

For this MP, we had to write a program that implements a Simple Distributed File System (SDFS).

1. Algorithm
   (a) Failure detection and dissemination : We have implemented the gossip protocol to achieve this.
   (b) Leader Election: Most of the file operations were carried out through the leader as will be explained in the sections to follow. So, leader election is critical to our design and the algorithm used was the ring leader election protocol.
   (c) File operations: All the file operations were triggered by a separate program called the client program which could be run on any of the VMs or any other system as long as the leader IP address is known.
      i. For 'put' operation, the client sends this command to the leader, which then hashes the 'sdfsfilename' and finds out which machines (upto 3 machines if they are available) this file should go to (the machine IPs are hashed using the same hashing function). The file is then received by the leader and it then relays the file to appropriate machines.
      ii. The 'delete' command goes through the master which finds out looking at the global file list which machine the file belongs to and sends them messages for deletion.
      iii. The leader upon receiving the 'get' command, goes through the global file list and finds out where the file is, and then sends a message to the client with a list of the machines that have the file. The client then connects to them sequentially until it gets the file that it has requested. If the file does not exist, it throws an error saying file not available.
   (d) Stabilization: Upon re-election, entry and deletion of a member from the membership list of the leader, the leader runs a stabilization protocol to account for any files that may have more than 3 replicas or mainly for the files that do not have even 3 replicas in the system. It re-hashes every file in the global file list and finds out the 3 (or less, if 3 machines are not available) machines it should now belong to and then sends messages to ensure that it goes to those machines (this also ensures load balancing under the simple uniform hashing assumption). The extra (more than 3) replicas are then deleted and the global file list is updated.
   (e) File Lists: Since all the operations go through the leader, it maintains a global file list that maps the sdfsfilenames to the machines that they are present on. In case the leader fails, the global file list is generated at the newly elected leader by way of the 'elected' messages and the stabilization algorithm is called again.

2. Experiments and Results
   (a) Re-replication time and bandwidth:
   (b) Master Failure and Re-election:

      The mean and SD are given by: (6.864, 6.484, 6.004, 5.338, 5.06, 4.746) (0.334, 0.448, 0.537, 0.519, 0.455, 0.372). It can be seen that the mean time for re-election goes down as the number of nodes in the system increases. This can be attributed to the ring election protocol.
   (c) Time for reading and writing files:
   (d) Time for storing Wikipedia corpus with 4 machines:

Table 1: Master Failure and Re-election time

$n = 7$	$n = 6$	$n = 5$	$n = 4$	$n = 3$	$n = 2$
6.91	5.77	6.63	5.22	5.7	4.32
7.31	7.1	5.05	5.4	5.38	4.6
6.31	6.27	5.88	4.42	4.37	5.4
6.74	6.53	6.12	5.91	5.01	4.88
7.05	6.75	6.34	5.74	4.84	4.53