Machine Programming 3 – Simple Distributed File System

ysaboo2 - Yash Saboo and pihess - Hess Parker I

Design (algorithm used and replication strategy)

Max # of failures at a time: 3 (Given)
Master Selection: Bully Algorithm, elects minimum available IP available (only 1 Master present)
Failure Detection and Group Membership: MP2
Replication Strategy: Passive; If failure occurs, Master will take care of re-replication of the files.
Replication Factor (N): 4 (means there are 4 copies of a file in the system)
Write Quorum(W) + Read Quorum(R) > N: W+R>4 --> W+R=5 --> W=4 (>N/2), R=1; Note, we don't explicitly use quorums, since we choose our replicas intelligently with respect to membership list which lists alive and reliable nodes.
File Sharding: Not Used
Modules/Arch.: Client-Server, where client interacts with the user and contacts server for file transfers and system topology queries over TCP connection (because reliability is important in file transfers).
Operations: What happens when Client receives the following requests: First Step - Client sends query JSON packet to Master with that operation tagged. Then:

Put: client receives the server IPs where the file needs to be replicated. Client transfers the files to those replica IPs. These server IPs were intelligently selected by Master by checking every system's file load. Also, write-write conflict is taken care by storing the history of file edit operations with timestamps.
Get: client receives the IP, which has agreed to send the file. Client then connects to that IP and file transfer takes place.
Delete: Master updates the 'replica tracking structure' and thus notifies the nodes.
ls: client receives the replica IPs which has that file, which is then displayed.

Note:- The nodes IPs with it's alive status and the files which it stores are stored together in a data structure called 'replica tracking structure' to help facilitate above operations. This structure is available to all the nodes.

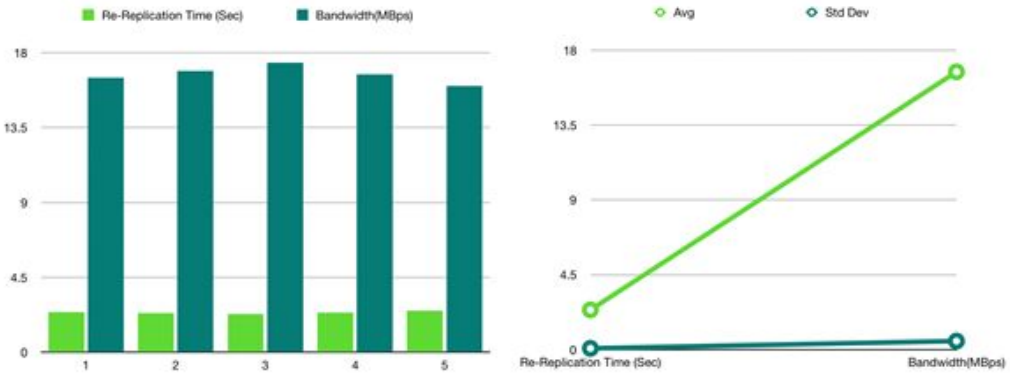
Usefulness of MP1

MP1 was highly beneficial since it provides a better mechanism to mine logs and debug the code.

Measurements
Time in seconds

(i) Re-replication time and Bandwidth upon a failure:

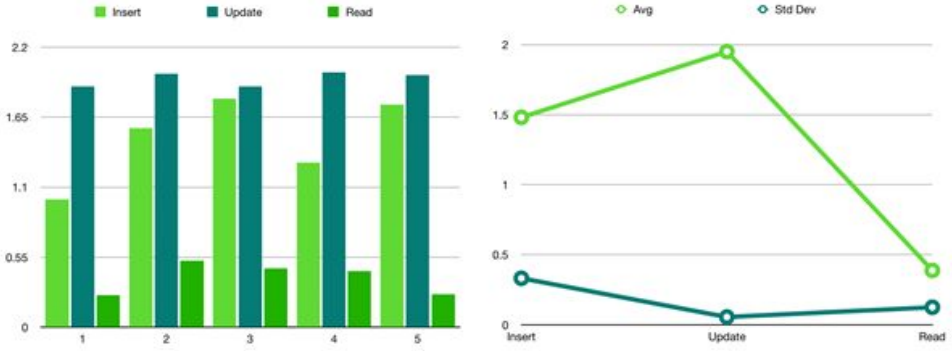
	1	2	3	4	5	Avg	Std Dev
Re-Replication Time (Sec)	2.41	2.35	2.29	2.4	2.5	2.39	0.077782
Bandwidth(MBps)	16.5	16.9	17.4	16.7	16	16.7	0.514782



It takes 2.39 seconds to re-replicate 40MB file with a bandwidth of 16.7MBps for 3 failures.

(ii) Times to insert, read, and update, file of size 25 MB, 500 MB (6 total data points), under no failure:
25MB:

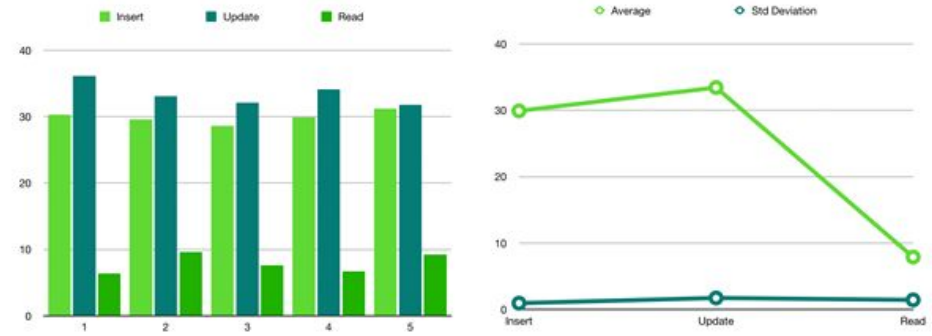
	1	2	3	4	5	Avg	Std Dev
Insert	1.004	1.564	1.796	1.293	1.75	1.4814	0.33217
Update	1.891	1.992	1.891	2.001	1.981	1.9512	0.055409
Read	0.251	0.524	0.464	0.442	0.26	0.3882	0.124841



500MB:

500MB

	1	2	3	4	5	Average	Std Deviation
Insert	30.3	29.6	28.6	29.9	31.2	29.92	0.952365
Update	36.1	33.1	32.1	34.1	31.8	33.44	1.740115
Read	6.4	9.6	7.6	6.7	9.2	7.9	1.445683

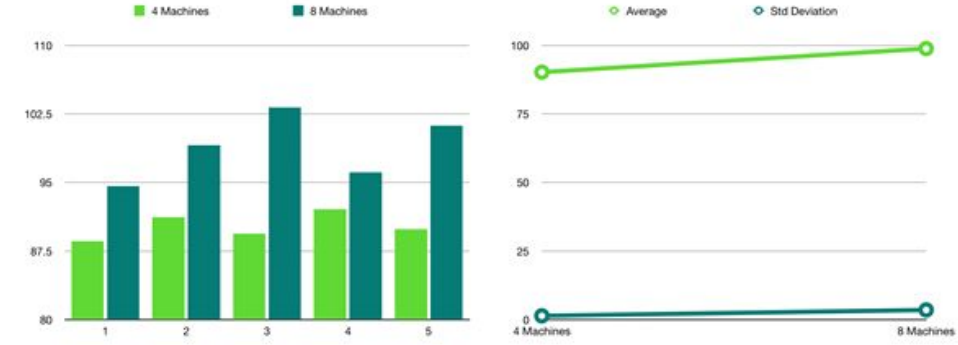


Insert and Update takes almost the same time because insert and update is quite identical in its operation. In update, master will have to search for already assigned replica IPs while in insert, it has to assign new ones. Extracting the assigned nodes from our replica tracking structure, along with replacing the existing file takes a little more time than just creating new file, which is the reason for update being a little higher than insert. Read takes less time since it is supposed to just read from just one IP, while insert and update takes place for 4 IPs. Naturally, as the size increases, the time taken to perform insert, update and read will increase.

(iii) Time to store the entire English Wikipedia corpus into SDFS with 4 machines and 8 machines

English Dictionary

Question 3	1	2	3	4	5	Average	Std Deviation
4 Machines	88.6	91.2	89.4	92.1	89.9	90.24	1.404635
8 Machines	94.6	99.1	103.2	96.1	101.2	98.84	3.540198



The time taken for 8 machines is a little higher than 4 machines because it has to transmit the replica tracking structure to more machines (replica tracking structure is present on all machines). The standard deviation is larger for 8 machines because more amount of network is used.