Joseph Sawaya 1004760537

Johnathon Martin 1004350230

Lunjun Zhang 1003937412

Milestone 2: Design Document

Storage Server

- ECS now controls the storage server
    - Control variables are implemented such as "stopped" and "Writelock" inorder to give ECS more control of each server.
- Servers hold metadata in a navigable map.
    - This allows for the implementation of a circular consistent hashing using built in functions such as higher(key) to find the first key within the map greater than the inputted key.
- Movedata
    - Movedata is implemented by moving all data that is no longer within the servers range. This means that no range is passed into the function or server. This was implemented for simplicity.
- Storage
    - The server's storage is located in a json file with the name according to the servers name. This allows for no write conflicts when using multiple servers on the same machine.
    - Ex server1 -> server1.json

ECS

- Controls the storage servers by setting and getting data
- Znodes are ephemeral, therefore the ECS can detect when servers go down
- The flow for creating a new node is:
    - Remote ssh to the new node
    - Remote process connects to zookeeper and the ECS is aware of new node
    - ECS passes metadata to the new node and node acknowledges the data was received.
    - ECS tells successor to move any data that it should move and passes metadata to successor at the same time
    - Successor signals being done receiving data, predecessor officially removes the data from its server and the ECS sends updated metadata to everyone.

KVStore

- Forwards each client request to the storage server responsible for the associated key
- Maintains the metadata about the storage service by processing error messages from the server
- Method of reconnecting to a new server:
    - Parse the returned message about metadata, and update the maintained map for metadata
    - Disconnect from the current server

    - Find the next responsible server by getting the higher position from the consistent hashing ring, and try to connect.
- Method of finding the next responsible server:
    - Hash the key using MD5, get the higher entry of this hashed key on the ring. If the higher entry does not exist, get the first entry.
    - This ECS node becomes the new responsible server, and we update the port and address of this KVStore instance.
- For put and get operations:
    - After receiving a message, we check whether the status type of the message corresponds to NOT_RESPONSIBLE.
    - If the server is indeed not being responsible, we initiate the operation of reconnecting to a new server as described above.

Performance Report

Each of the following tests were measuring the time it takes to complete 5000 requests.

Server	Client	Throughput
1	1	8.5436 req/s
1	10	5.1026 req/s
1	20	25866 req/s

- Note that there is a point of diminishing returns when it comes to multiple clients being run on one computer as the computer has a limited number of threads.
- During testing we encountered a bug with mass requests on multiple servers and thus we have no data to report on multiple servers for this section.
- The performance eval code can be found in our files.

Appendix A: Additional Test Cases

Test functions	**What it does**	**Test Report**
test_file_creation_by_server_name	Tests the servers ability to generate a json storage file based on its name	Passing
test_text_messege	Tests the creation of text messages and the setting of relevant flags.	Passing
test_file_creation_with_invalid_input	Tests whether the server can deal with server names with invalid characters such as '.'	Passing
test_server_clear	Tests that the server clear function clear the current contents of the server without removing the file	Passing
Test_server_clear_file_doesn't_exist	Tests the servers ability to cope with a file being deleted	Passing
test_store_teardown_connection	Tests KVStore's ability to correctly disconnect from a server	Passing
test_client_disconnect	Test the client's ability to get running and then correctly disconnect from the store.	Passing
test_stored_key	Test whether KVStore is able to successful hash and get the stored key	Passing