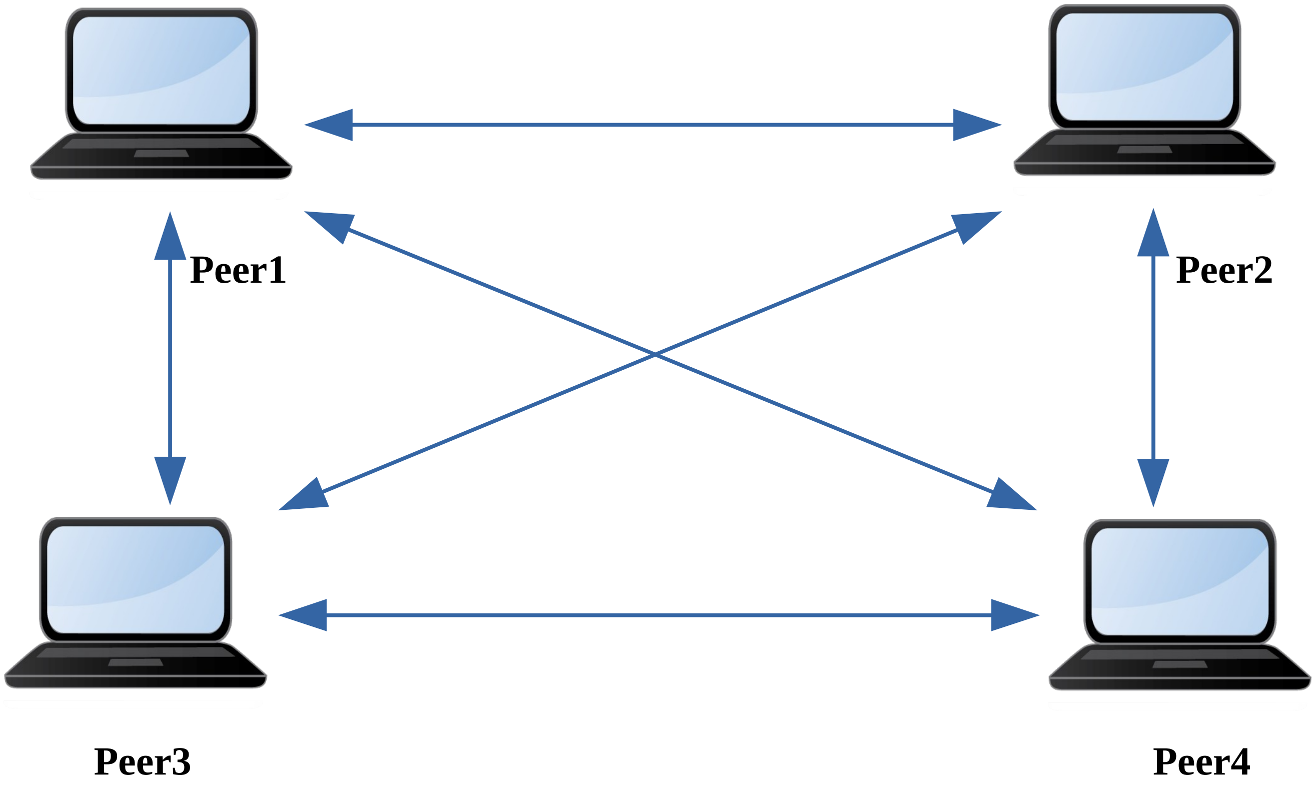
Aditya Kale – A20355807

Design Document :



Decentralized Peer to Peer File Sharing System.

Its a Distributed Peer to Peer file sharing system where all the all the Clients (Peer) are connected to all other clients (Peer). Also Each Peer can communicate with another Peer while putting Key-Value pair (ex. File name and Machine address) in distributed hash table Or getting values from distributed hash table.

Classes in Project:

Client.java

1.Client.java

It's a main class which runs the Client and also start as server to wait for other client requests to store values in hash table. It also start as another file transfer server which sends requested files to another peers.

It calls methods of **ClientImpl.java** . Which does four tasks.

ClientImpl.java

2.ClientImpl.java

It has four methods.

i. Register Files : It register all files into a distributed hash table. It takes a file name applies hash function and store it on particular server.
ii.Search File: Accept file name from client, applies hash function to get server number and retrieves value from that server to get machine address or location of that file.
iii.Get all file names: Displays all file names which are registered and available for download.
iv.Download a File: It Accept file name from user and process it to return an integer which is the server number on which that file is present. It will create a connection with that client and downloads file in Downloaded folder.

A **List of alll sockets and streams are generated and stored in an array. This operation is performed only once**. Then afterwards same streams are used for communication.

3. ClientServer.java

Its a simple class which starts client as a server. It starts listening at given port.

4. ClientServerImpl.java

It accepts the key-value pair from clients and perform Register and Search operations. This class also downloads and store a file in its registered folder and the time of Replication.

5. ClientToSendFiles.java

Its a simple class which starts client as a server for sending registered files to other servers when requested.

6. SendFiles.java

This class simply make a connection to a client which has requested a file and send file contents to that client.

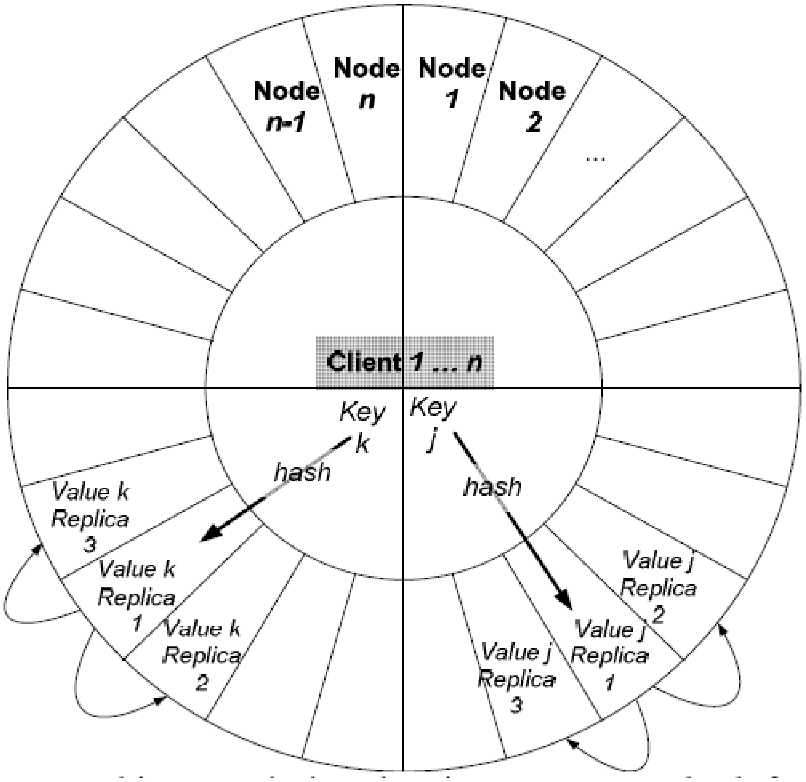
Overview Of a System:

- When any Client gets started, it **creates** a **ServerSocket connection**. It is **continuously running** and ready to accepts continuous requests from different Client. It also act as a server for sending the registered files to different clients.

- When any Client connects to a server for performing operation on distributed hash table, **first time it accepts the connection and create a new Thread to process a request**. Further all communication is done through same socket on that same thread.

- Here **every client has his own hashtable** where values will be stored. And **all clients has a same HashFunction** which is used to find the server number where that key-value pair (FileName, MachineAddress) will be stored. Similarly the same hash function will be used to retrieve records from distributed hashtable.

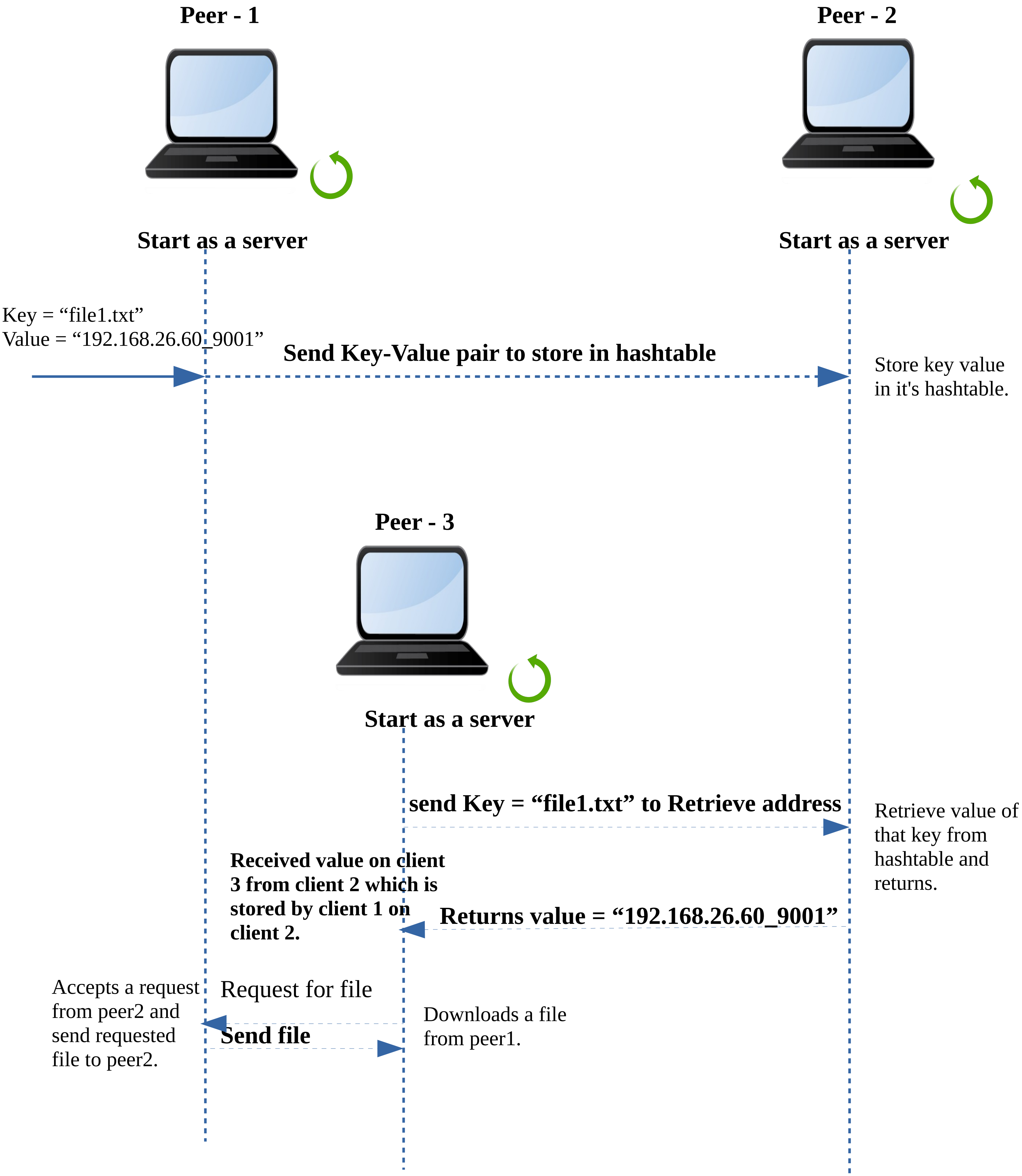
- **System is Fault tolerant**. Both metadata and file replication is handled by system. If any node fails still we can retrive all its hash table entries from its previous or next server. Also files which are registered by that client are also replicated on other servers.
- System also supports Binary files of large size upto 4GB.
- **Replication factor:** System supports maximum of two level replication. This can be achieved by just **changing** the **parameter values** of '**replica1' and 'replica2'** to '**true'** or '**false'** in **config.properties**.
  - If replica1 = true, metadata and files are replicated on the next server.
  - If replica2 = true, metadata and files are replicated on the previous server.
  - User can change replication factor by changing these parameter values to false to reduce replication factor.
  - If its a last server then replica will be stored on its previous server and next i.e first server of a system. And vice versa.



Task : This system register files on sever, replicates its metadata and files on another server and returns machine address of particular file when requested. And allows user to download files which are registered on this distributed file sharing systemm.

Aditya Kale – A20355807

<u>Sequence Diagram :</u>



Peer - 1

Start as a server

Peer - 2

Start as a server

Key = “file1.txt”
Value = “192.168.26.60_9001”

Send Key-Value pair to store in hashtable

Store key value
in it's hashtable.

Peer - 3

Start as a server

send Key = “file1.txt” to Retrieve address

Retrieve value of
that key from
hashtable and
returns.

Received value on client 3 from client 2 which is stored by client 1 on client 2.

Returns value = “192.168.26.60_9001”

Accepts a request
from peer2 and
send requested
file to peer2.

Request for file

Send file

Downloads a file
from peer1.