MATHEMATICAL MODEL

**A] Mapping Diagram:**

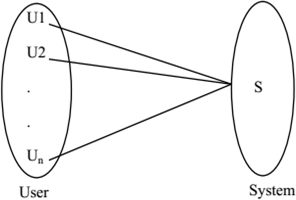


Figure:2 Mapping Diagram

Where,

U1, U2…. Un =Users. S = System

**B] Set Theory Input:** File or Image.

**Output:** Whenever user wants to upload the file on system, then check or test the duplication.

# Process:

Step 1: Open account.

Step 2: Upload file on storage.

Step 3: System checks for the duplicate file available on storage system. Step 4: If found then remove the duplication and maintains index co.

Step 5: On non-duplicate data, check for deltas.

Step 6: Store unique and deltas on system in encrypted form. Mathematical model contains five tuples –

S = {s, e, X, Y, ϕ}

Where, the following conditions are satisfied-

{s} = Start of the program

1. Log in with webpage.

To access the facilities of system such as store on system, user has to log into system.

1. Upload text Files on system.

Upload files on system in text format.

{X} = Input of the program.

Input should be any text file.

{Y} = Output of the program.

{e} = End of the program.

ϕ = Success and failure conditions

File will be first fragmented then it is encoded and the fragments are allocated.

{X, Y € U}

Let U be the Set of System.

{U} = {Client, F, S, T, M, D, R, DC}

Where,

Client, F, S, T, M, D, R, DC are the elements of the set.

{Client} = Data Owner, User.

{F} = Fragmentation

{T} = Generates fingerprints for file and blocks.

{D} = Check for duplicate file or block.

{R} = Detects similarity by using existing information of a deduplication system.

{DC} = Delta compression module takes each of the blocks detected previously, and reads its base-chunk, and then delta encodes their differences.

# Chunking:

Before storing the files on system, Files are broken down into chunks such as, F = {FC1, FC2….Fcn}

# Deduplication Checking:

H (New chunk) = h H (Old n chunks)

If H (New chunk) == H (Old n chunks [])

Chunk is duplicate and do not store it, instead provide link.

Else Chunk is not duplicate, and then stores it.

# Success Condition

File splitting and storing it on multiple nodes. User gets result very fast according to their needs.

# Failure Condition

* + Hardware failure.
  + Software failure.
  + Maintaining indexing leads to more time consumption to get the proper file stored on system.

# Space Complexity

More the storage of data more is the space complexity.

# Time Complexity

Time complexity of system depends on following factors: time taken to upload file, time taken during file level and block level deduplication, delta calculating, storing deltas and non-duplicate data in encrypted format on different nodes.

OBJECTIVE FUNCTION

1. Upload File

* This function uploads the file on the system.

1. Download File

* This function downloads the file from the system.

1. Delete File

* This function deletes the existing file on the system.

1. Delete User

* This function deletes the existing user account on the system.

1. updateFile

* This function updates the existing file as per the user requirement on the system.