Distributed Hash Table (DHT) User Manual

Contents

1. System Introduction

Distributed Hash Table ((hereinafter referred to as “DHT”) System simulates the behavior of Hash Table Data Structure across network by taking advantages of the Distributed Systems. This concept will help to achieve scalability if resources are increased based on the demand. DHT concept can be used as File Indexing Servers, Domain Name Servers, etc. It utilizes the advantages of decentralized distribution system by providing Dictionary like interface. Nodes of this Dictionary like interface can be spread across the network. It provides:

- A distributed decentralized database where users can store the data using Key – Value pair concepts
- DHT Client can perform Put, Get and Delete operations on DHT database
- Log File at Client and Server ends will record the operations done on each of the nodes

Some of the features available in the current System are,

- **Easy to use console based interface**

  A console based interface allows user to select through various operations (Put, Get, and Delete). Depending upon the operation selected by user, System prompts for the required inputs (like Key Value, Key Path, etc.).

- **Highly parameterized system**

  The “.properties” file allows administrator or user to define the various options (like DHT Server IP Address, Port, No. of Servers in DHT Environment, No. of Data Points for Testing purpose, etc.) for keeping the System scalable and parameterized.

- **TCP Support**

  System uses TCP / IP Protocol to ensure that the data is downloaded with consistency. The reasons for choosing TCP / IP over UDP are listed below.

  - TCP is connection oriented – once a connection is established, data can be sent bidirectional.
  - It is also widely used by other protocols like HTTP, HTTPS, FTP, SMTP, and Telnet which makes this protocol widely popular for Net Traffic.
  - TCP is more reliable since it manages message acknowledgment and retransmissions in case of lost parts. Thus there is absolutely no missing data. UDP does not ensure that communication has reached receiver since concepts of acknowledgment, time out and retransmission are not present.

- TCP transmissions are sent in a sequence and they are received in the same sequence. In the event of data segments arriving in wrong order, TCP reorders and delivers application. In the case of UDP, sent message sequence may not be maintained when it reaches receiving application. There is absolutely no way of predicting the order in which message will be received.
- It has its own Error checking mechanism ensuring the data order and consistency.

2. Prerequisites and Constraints

In order for the effective and proper functioning of System following details needs to be ensured.

- Since the System uses connection oriented TCP / IP Protocol, network connectivity must be ensured among DHT Servers, and DHT Clients as and when required.
- Please ensure that ".properties" file is available in the working folder of each module to run the System smoothly.

3. DHT Server Module

DHT Client communicates with DHT Server in order to put, get, and delete various files. This module maintains the database of all the Key – Value pairs put by various DHT Client instances.

A. Inputs / Constraints / Prerequisites

i. Network connectivity between various DHT Clients and DHT Server will have to be ensured during all operational time.

B. Process

a. Put

i. This operation accepts Key and Value data from Client and stores it into DHT Server database.
ii. If the DHT Server database already has entry of the referred Key, then the new entry will be created with the latest details provided by client machine.
iii. If the details are not existing already then the data will be inserted into DHT Server database bucket.
iv. Final status of the operation is passed to Client.

b. Get

i. This operation accepts Key from Client and searches the Key into DHT Server database.
ii. If the DHT Server database already has occurrence(s) of the referred Key then the Value will be sent back to Client.
iii. If the search returns no result, then similar message is sent back to Client.

c. Delete

i. This operation accepts Key from Client, searches the file name into Indexing Server database.
ii. If the DHT Server database already has occurrence(s) of the referred Key, then the referred entry or occurrence is deleted or removed from DHT Server database
iii. Final status of the operation is passed to Client.

C. Output

i. Depending upon the type of operation, DHT Server database is either updated (in case of Put or Delete operation) or available existing data (in case of Get operation) is sent back to Client.

4. DHT Client Module

DHT Client communicates with DHT Server in order to put, get, and delete Key – Value pair data.

A. Inputs / Constraints / Prerequisites

i. Network connectivity between various DHT Clients, and DHT Servers will have to be ensured during all operational time.

B. Process

a. Put

i. This operation sends either user accepted or randomly generated Key – Value pair data DHT Server for put operation.
ii. Final status of the operation is will be displayed on DHT Client console.

b. Get

i. This operation sends either user accepted or randomly generated Key data DHT Server for get operation.
ii. If the DHT Server database already has occurrence(s) of the referred Key, then the Value data will be sent back to Client.
iii. If the search returns no result, then similar message is sent back to Client.

c. Delete

i. This operation sends either user accepted or randomly generated Key data DHT Server for delete operation.
ii. Final status of the operation is passed to Peer Client.

C. Output

i. The final status message of each operation requested by user will be displayed on console.