CS4262 - Distributed Systems

Distributed Content Searching System

Final Report

150041X : H.B.G.C. Ariyarathna

150097X : H.M.S.U. Dayaratahna

150106D: K.T.S. De Silva

150213D : H.B.W.S. Heenatigala

Table of Content

1. Project Introduction

1.1. Topology

Unstructured overlay topology has been used to implement file-sharing among distributed nodes as it provides highly available connections among peers. It also performs well in large scale and highly dynamic environments compared to structured overlay topologies. In this project, we have used 10 nodes for implementation purposes. The following diagram shows the peer to peer network. (Note that only 5 peers have been drawn)

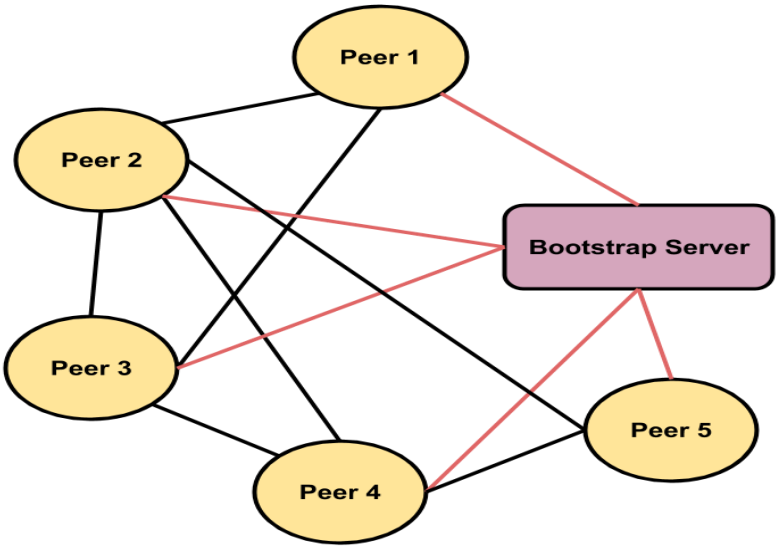

Fig. Peer to Peer Network Structure

1.2. Communication Among Nodes

All the nodes must connect to the bootstrap server first in order to join the network. Each node register at the bootstrap by giving its IP address, port number, and user name. The 1st node receives an acknowledgment only from the bootstrap server and the 2nd node receives an acknowledgment and the details of the first node. 3rd node receives details of the first two nodes. 4th node receives details of the first three registered nodes, and so on. From the 4th node onwards, only to 2 randomly selected nodes from the list are joined.

After forming the initial overlay, nodes begin to communicate with each other to search for shared content. The provided queries for the files will be flooded through the network to search files.

1.3. Performance Parameters & Measurements

- **Query Hop Count** - Number of nodes engaged in the search to get the results. This measure is important as a performance parameter as, when the number of involved hops is high, the significant load was there to communicate and node's resource utilization.

- **The latency of search** - This is the requirement of the time of a query to respond to the query results. For that, it gets the difference between the delivery time of the result and the trigger time of the query using the requested node's local timestamp.

- **Successful search rate** - This can be calculated by getting the number of successful queries out of the issued total number of queries. Furthermore, test queries can be used for the calculations.

- **Query time network utilization** - The network utilization at the time of query execution

2. Configuration steps

- A network with 10 nodes and the Bootstrap server was configured where the 10 separate instances used for the distributed content searching program.
- In the given Queries.txt file, 50 queries were given and 3 randomly picked nodes issued those queries at the same time.
- The statistics were gathered based on different performance metrics by the log file analysis.
- The algorithm of random walk search was used and set the maximum hop count to 5 hops

3. Performance Analysis

The system performance was measured using a successful and unsuccessful search count.

- Successful - searched files found within defined max hop count
- Unsuccessful - searched files not found due to,
    - Results for the query exists, but the defined max hop count exceeded
    - No results exist for the given query.

	Node1	Node2	Node3
Successful	45	46	43
Unsuccessful	5	4	7

Statistics for both unsuccessful and successful searches

	All queries		Successful Searches	
	Hop Count	Latency (ms)	Hop count	Latency (ms)
Total	155	245	120	198
Min	0	3	0	3
Avg	3.8	5.25	2.2	4.21
Max	5	7.24	6	8.11
Std. deviation	0.88	1.22	2.12	1.86

	UDP Messages per Node		Node degree
	Sent	Received	
Min	7895	7889	3
Avg	15987.21	15979.23	6.14
Max	21459	21467	8
Std. deviation	5147.3	5149.5	1.12

4. Discussion

Discuss how your solution will behave if the number of queries (Q) is much larger than the number of nodes (N) (Q >> N) and vice versa (N >> Q). Comment on how to improve the query resolution while reducing messages, hops, and latency.

When a query is submitted for searching the content, a random walk is performed. The overlay network is created by having a higher number of messages as heartbeat messages, however, these messages do not exist in the network for a long time avoiding congestion. It is designed in such a way that a search query will terminate after 15 hops.

Let's consider the scenario where Q >> N. Under the current setup, it would require a large number of queries to make a significant impact. After reaching the hops limit, a search query is terminated. Therefore the messages do not remain forever in the network. Since the random walk algorithm only picks one random node at a time, we can expect the system to be robust even for a large number of queries compared to other implementations. When N >> Q, When the number of nodes is much larger than the number of queries to be issued, there will be a decrease in the number of successful searches. It is because the maximum number of hops will be exceeded quickly when the number of nodes in the system is high. Currently, in the random walk algorithm, we initialized only one path for search. Because, when the number of nodes is high and the search queries are low, we can easily initialize more than one distinct random walk path for one search query. Then the search coverage and the success rate of searching will be increased. When the number of nodes in the system increases, it will be needed to increase the maximum number of hops a query would run to keep the number of successful searches intact. Additionally, it may be required to increase the limit on the number of entries in routing tables to avoid failures due to partitioning in the distributed system.

1. Register the nodes at the bootstrap server by providing the node's IP address, port number, and user name.

2. Send acknowledgments to each node as provided in the design.
3. Implementation of JOIN messages and query searching
4. Implementation of node un-registration before attempting to register again

5. Individual Contribution

Name	Index Number	Contribution
H.B.G.C. Ariyarathna	150041X	❑ Designing the peer to peer overlay network structure ❑ Node Registration ❑ Content Search by setting hops ❑ Worked on performance analysis ❑ Final Report Preparation
H.M.S.U. Dayaratahna	150097X	❑ Designing the peer to peer overlay network structure ❑ Node Registration ❑ Content download with validation ❑ Worked on performance analysis ❑ Final Report Preparation
K. T. S. De Silva	150106D	❑ Designing the peer to peer overlay network structure ❑ Node Registration ❑ Content download with validation ❑ Worked on performance analysis ❑ Final Report Preparation
H.B.W.S.	150213D	❑ Designing the peer to peer overlay network

Heenatigala		structure ❏ Node Registration ❏ Content Search by setting hops ❏ Worked on performance analysis ❏ Final Report Preparation