IN5020 - Distributed Systems Group Session

Topic 8 – Peer-to-Peer Systems

Praveensankar Manimaran

Praveema@ifi.uio.no

October 13, 2023

Assignment 2 - Presentation

- Choose the timeslot in the following link:
  - https://docs.google.com/document/d/1r4OrgI8LJXYOM7WsghjERB4oapAqHpstKxF1Y9qCEMw/edit
- each group has to choose one timeslot for the presentation. Add the group in the corresponding cell in the row to reserve the timeslot.
- If none of the timeslots work for a group, then let me know through mail.

Question 1 -

What is the key problem in P2P?

Key Problem

- Placement of data objects across many hosts
- Lookup of data objects

Challenges:

- Naming
- Routing table
- Handling node churn

Question 2-

Discuss the functional and non-functional requirements of peer to peer middleware?

Functional and non-functional requirements in p2p middleware

Functional requirements:

- Locate and communicate with nodes
- Add or remove nodes/resources
- Simple API for the resources (irrespective of the type of resource)

Non-Functional requirements:

- Scalability – eg:-handles millions of objects on thousands of nodes
- Load balancing – evenly distributed workload across nodes
- Optimization of local interactions between neighbouring peers
- Handling node churn
- Security, anonymity

Question 3 -

Early file-sharing applications such as Napster were restricted in their scalability by the need to maintain a central index of resources and the hosts that hold them.

What other solutions to the indexing problem can you identify?

Solutions to the indexing problem

1. Distributed Hash Table
    1. Each node maintains a set indexes of other nodes that are responsible for corresponding data objects.
    2. Different algorithm can use different routing mechanisms to achieve efficient lookup
    3. Pastry uses DHT based on GUID and uses the circular routing protocol
    4. Chord uses DHT based on consistent hashing and uses finger table for routing

Question 4 -

When the Squirrel peer-to-peer web caching service was evaluated by simulation, 4.11 hops were required on average to route a request for a cache entry when simulating the Redmond traffic, whereas only 1.8 were required for the Cambridge traffic. Explain this and show that it supports the theoretical performance claimed for Pastry.

Pastry performance

Cambridge data was based on 105 nodes

Redmond data included 36000 nodes.

Pastry is an overlay network and the number of routing hops required in Pastry is **O(log N)** where N is the number of nodes participating in the overlay.

For Cambridge data => $O(\log_{16} 105) = 1.67 \approx 1.8$

For Redmond data =>$O(\log_{16} 36000) = 3.78 \approx 4.11$

Thank You