Distributed “Peer-to-Peer” Systems

Jasleen Kaur

November 9, 2009

Object Discovery Protocols

- How to design a distributed system that helps members share objects (by exchanging copies of objects between peers)?
  - Objects could be files, music, video, ...
  - e.g., distributed file systems, named, naming service, etc.
  - Object could be available at one, many, or none of the peers

- How to quickly find (and retrieve) the object?
  - Data is important (location isn't)
    - Query refers to data (and not location)
    - Data placement unrelated to overlay topology
Gnutella – Search Flooding

Gnutella – Rules for Flooding

- Hop-count in messages limits horizon for forwarding
  - May cause search to fail even if target exists
- Incoming Ping or Query messages are forwarded on all connections except the one that sent it
- Incoming Ping or Query messages that are duplicates should be discarded
- Response messages should be forwarded only on the connection sending the original Ping or Query
- Response messages should be discarded if the node has not received the matching Ping or Query

Alternatives to flooding?
Can Gnutella Be Made Scalable?
Random Walk vs Flooding

Issues:
- Termination
  (TTL vs explicit)
- State maintained
- Number of walkers

Reduces number of messages by two orders of magnitude!

Can Gnutella Be Made Scalable?
Degree of Replication

- Replication should be proportional to square-root of query rate
- Distributed implementation:
  - # of replicas created should be proportional to length of search
Can Gnutella Be Made Scalable?

Topology Adaptation

- High degree nodes observe high query loads
- Random walks that seek out high-degree nodes limit its superstar scaling in large systems

- Gnutella overlay organization:
  - High capability nodes have high out-degree (many neighbor links)
  - Low capability nodes are “swan” high capability nodes
  - High capability nodes are likely to answer query

KaZaA (Fasttrack)