Overview

- To discuss two papers on consistent routing in structured peer-to-peer systems
  1. Performance and dependability of structured peer-to-peer overlays (Pastry)
  2. Robust and efficient data management for a distributed hash table (Chord)
- General Discussion

Definition for Consistent Routing

- Routing is consistent if no overlay node ever delivers a lookup message when it is not the current node for the message's destination key
  + storage consistency when updating files
  + correct routing between nodes
- easy satisfaction is to have no movement nor duplication of data for root nodes

Pastry Recap

- NodeID's are random 128-bit unsigned integers

MSPastry

- Based on Pastry
- Addresses the issue of high churn rates
  - nodes join and leave the overlay continuously and do not stay for long
- Provides dependable routing

MSPastry Mechanisms

- Periodic routing table maintenance
  - active liveness probing (linear)
  - per hop-ack
  - lazy node repair
- Low overhead proximity neighbor selection
- distance probing
MSPastry Evaluation Metrics

- Relative delay penalty (RDP) – average ratio between the delay achieved by MSPastry when routing two nodes and the network delay between the same nodes
- Control traffic (c) – average number of control messages sent per second per node

DHash using IDA erasure code

- Hashes SHA-1 hash
- Storage by blocks (812b each)
- Each block is fragmented into 14 pieces using Rabin's Information Dispersal Algorithm (IDA) erasure code
- Fragments are sufficient to reconstruct the block

DHash Ideal State

1. Multiplicity – 14 or 1 fragments
2. Distinctness – All fragments are distinct with high probability
3. Location – Each of the 14 nodes succeeding the block's key store a fragment

Fragment Maintenance

- Global DHash Maintenance
  - Push misplaced fragments to the correct location
  - Delete extra copies of fragments
- Local DHash Maintenance
  - Recreate missing fragment
  - Synchronize database with each of its 13 successors

Database Synchronization

- Fragments indexed by the fragment's key (SHA-1) in form of Merkle Tree

Consistent DHT structures
**DHash Valuation 1**

**DHash Valuation 2**

**DHash Tradeoffs**
- Fault tolerance – Measure vs. Duplicates
- Network traffic latency – Max transfer rate vs. slowest sender
- Memory usage – Redundancy vs. fault tolerance
- Load balance – network load vs. space utilization

**MSPastry DHash Discussion**