DISTRIBUTED MONOTONICITY

Kevin Cheek
Alex Maskovyak
Joseph Pecoraro

AGENDA

- Research Paper Analysis (all three)
  - Pastry
  - PAST
  - Erasure / RAID
- Progress
- Demonstration

OVERVIEW

- We plan on building a distributed backup system.
- The system will offer a RAID-like data redundancy on a distributed system of computers.
- We will make use of an Open Source Peer-to-Peer System Framework to keep track of file divisions, reliable redundancy across the “cloud,” and efficient routing, and network maintenance chores.

PAPER #1 - PASTRY

- A. Rowstron and P. Druschel, "*Pastry: Scalable, decentralized object location and routing for large-scale peer-to-peer systems*". IFIP/ACM International Conference on Distributed Systems Platforms (Middleware), Heidelberg, Germany, pages 329-350, November, 2001.
- http://research.microsoft.com/en-us/um/people/antr/PAST/pastry.pdf
- Understand how to leverage a Distributed Hash Table in a large Peer-to-Peer application.

ABSTRACT

- "All nodes have identical capabilities and responsibilities and all communication is symmetric."
- Scalable routing scheme that are fault resilient and take advantage of locality properties.
- General Substrate that is easy to build upon.

ROUTING TABLE

- Routes: $\lceil log_{2^b} N \rceil$
- Table Parameters
  - (b) is solely configurable
  - (L) leaf, (M) neighbor sets
- Always route to the node ID that is closer to the node ID, by using node ID prefixes.

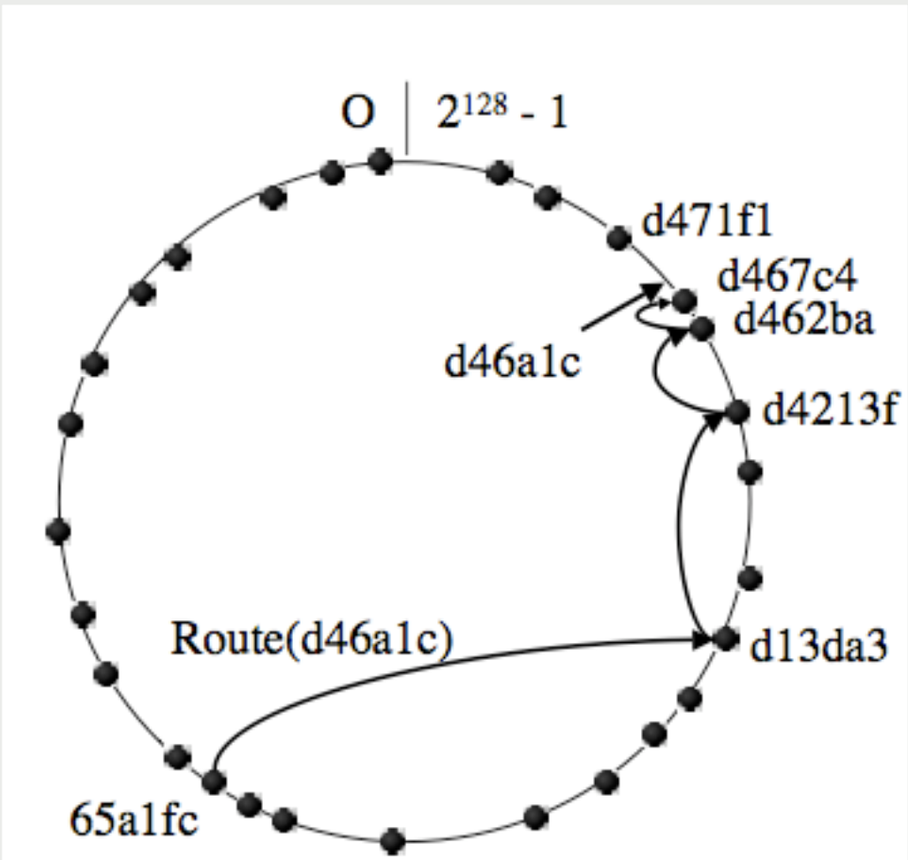
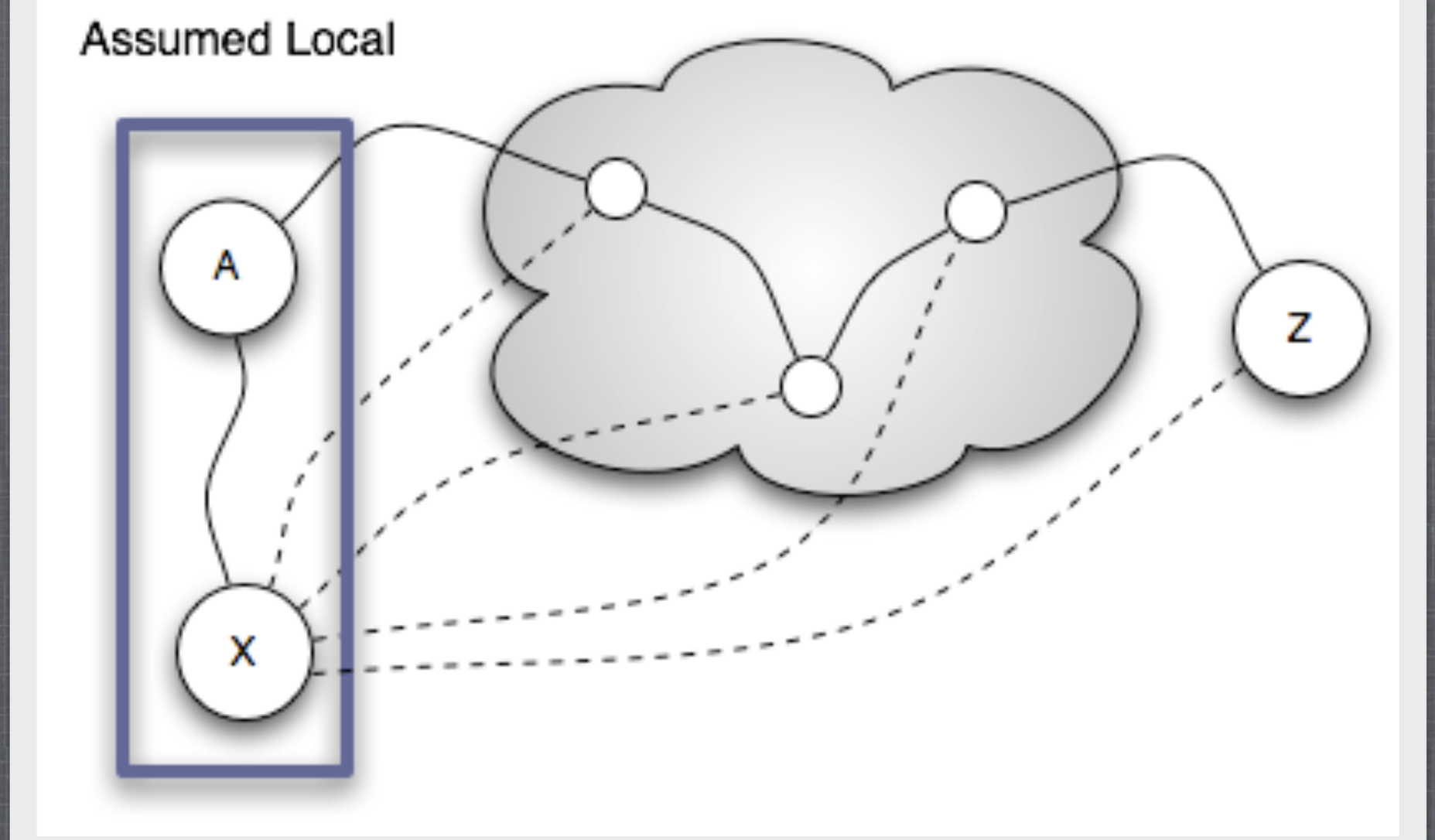


Figure 2: Routing a message from node $65a1fc$ with key $d46a1c$. The dots depict live nodes in Pastry's circular namespace.

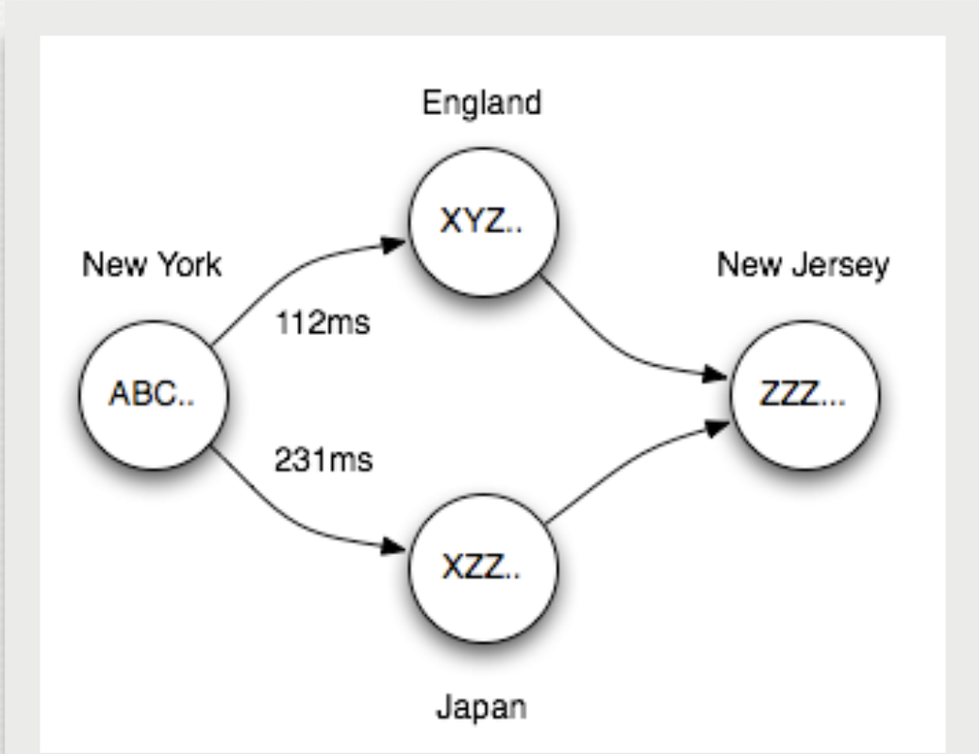


NODE JOINING

X is attempting to join the Pastry Network by sending a join request to A.

LOCALITY

- Each step in the routing moves the message closer to the destination.
- Pastry takes advantage of locality information to take the path of least latency. Uses a “levels” concept in the table.



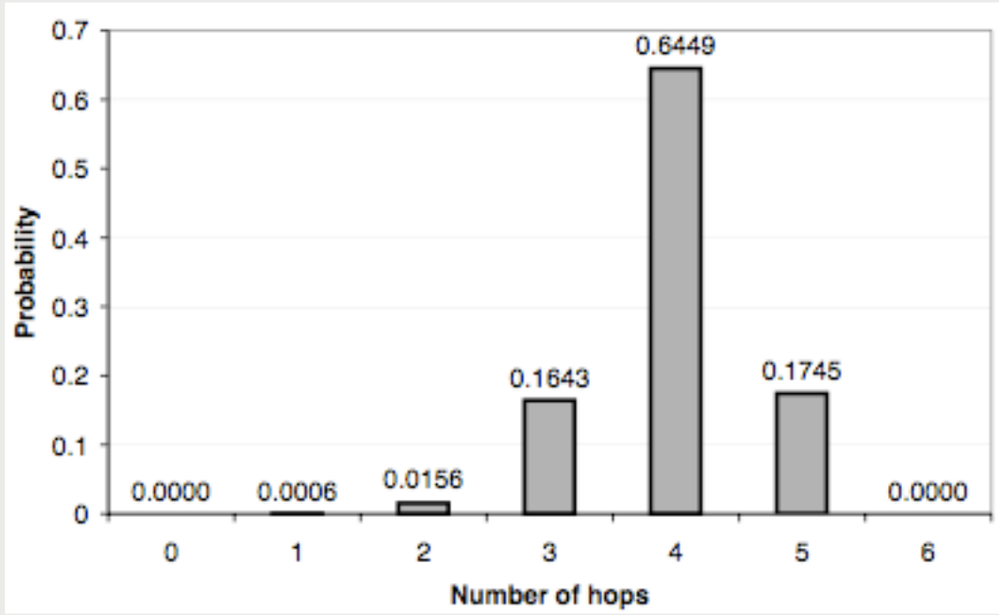
FAULT RESISTANCE

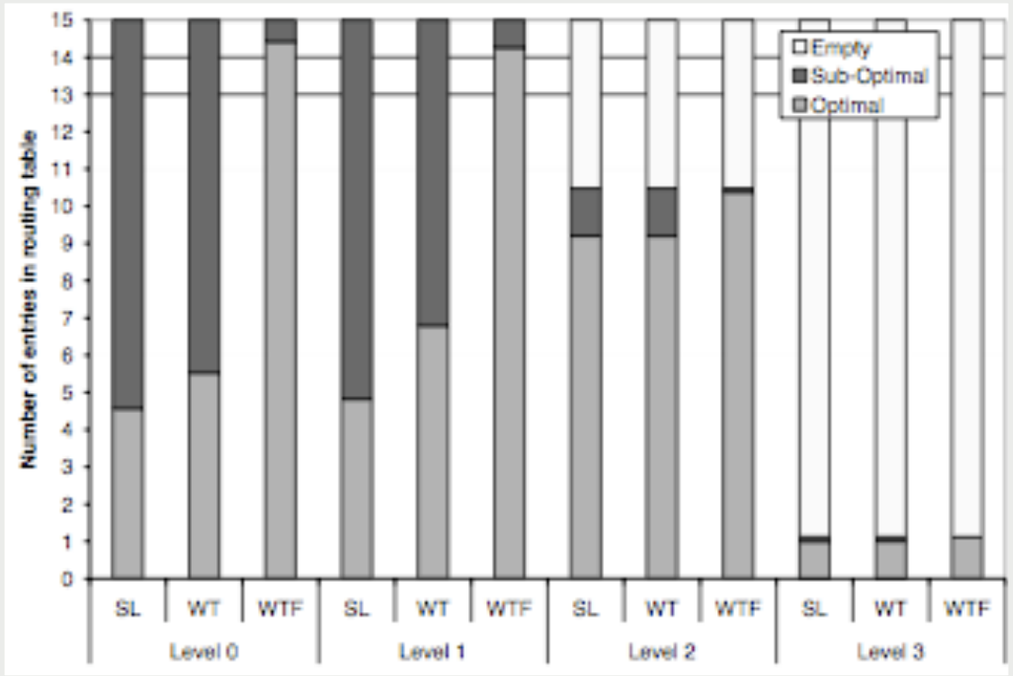
- IP Anomalies may happen with Router Failure. This results in Isolated Overlays.
  - Solved by periodic expanding ring multicasts to merge the rings.
- Malicious or Faulty Nodes
  - Routing randomization, instead of always taking the optimal path eventually finds a viable path.

EXPERIMENTAL RESULTS

- Experiment with 100,000 nodes, b=4, L=16, N=32, on 200,000 lookups:
  - Averaged 4 hops
  - Averaged 75 table entries
  - 3000 lookups per second

- Join algorithm negotiations in Pastry resulted in substantially more optimal table entries then existing basic join algorithms.





PAPER #2 - PAST

- P. Druschel and A. Rowstron, "*PAST: A large-scale, persistent peer-to-peer storage utility*", HotOS VIII, Schoss Elmau, Germany, May 2001.
- http://research.microsoft.com/en-us/um/people/antr/PAST/hotos.pdf
- Generic file replication and caching in Pastry.

PROBLEMS ADDRESSED

- Distributed peer-to-peer file storage
- Ensure reliability of information storage/retrieval
- Ensure fairness in storage responsibilities/load
- Ensure confidentiality, security of information stored in the system

PAPER'S NOVEL CONTRIBUTIONS

- Implemented a Distributed Hash Table (DHT) over PASTRY
- Designed PAST: a PASTRY application that uses this DHT for file-storage and replication
- PASTRY's overlay network properties are exploited for global distribution, ensures diversity of storage nodes, and efficient routing of storage/retrieval requests

PAPER'S NOVEL CONTRIBUTIONS

- PAST's DHT allows for simple lookup and efficient replication
- Optional smartcards/brokers allows for storage quotas and control of space supply/demand

PAST AND PASTRY

- System is composed of nodes connected to the Internet
- Nodes act as storage nodes and user access points
- Nodes/Users are identified by a cryptographic hash of the Node's public key (optionally provided via Smartcard/broker system)

PAST OPERATION

- File storage
  - Files have a “unique” 160-bit fileId created from the cryptographic hash of the file’s name and the owner’s public key
  - File certificates are generated for each file stored
    - Contain replication factor, insertion date, crypto hash of file’s contents, authorized accessors, etc.
    - Signed by owner
- File reclaim receipts are created for successful file stored
- Files are stored on the k nodes whose nodeIds are numerically closest to the 128 most significant bits of the fileId
- Files are requested from the live node with a nodeId numerically closest to the request fileId

PAST MEETS ITS OBJECTIVES

- Information reliability
  - File remains available so long as 1 of k nodes is alive
  - Stored across a diverse node-set: geographically, administratively, legally, ownership, etc. (guaranteed by virtue of nodeId and fileId generation)
  - Files are randomly requested and checked against their hash

PAST MEETS ITS OBJECTIVES

- Ensure fairness of storage responsibilities
  - Smartcard/brokers are responsible for ensuring storage-quotas, monitor receipts given and keep running total of amount stored/requested for storage
  - Randomized routing protocol distributes load across many nodes
  - Use of hashes on fileIds ensures a uniform distribution of nodes selected for storage across the network

PAST MEETS ITS OBJECTIVES

- Ensure confidentiality, security of information stored in the system
  - Public-key cryptosystems and cryptographic hashes are computationally infeasible to break
  - Nodes are not trusted, however, it is likely that most nodes are not malicious
  - Attacker's cannot control the behavior of smartcards/broker system
  - Signed file certificates restrict those that can store, delete, retrieve files

UTILIZING PAST'S RESULTS

- PASTRY and PAST's DHT will be leveraged to store information for users and nodes
  - User's personal file list
  - List of master nodes which understand where the file is stored
- Cryptographic hashes will serve as a node "heartbeat" to check file chunk integrity
- Public-key encryption may be leveraged for confidentiality of data

PAPER #3 - ERASURE

- Goodson, G.R.; Wylie, J.J.; Ganger, G.R.; Reiter, M.K., "*Efficient Byzantine-tolerant erasure-coded storage*," Dependable Systems and Networks, 2004 International Conference on , vol., no., pp. 135-144, 28 June-1 July 2004

- http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=1311884&isnumber=29105

- Decentralized consistency protocol for survivable storage.

PAPER #3 - ERASURE

- Overview
- Erasure Codes
- Write Operation Integrity
- Read Operation Integrity

PROGRESS

- Pastry peer-to-peer networks have been successfully established:
  - on the Virtual Machines in RIT's cluster.
  - multiple nodes per JVM on a single machine.
- Raid-5 like fault tolerance
  - We have split and reassembled files
  - We have recovered from information loss
- We have successfully stored our data structures in PAST's Distributed hash table.
  - the Master List and Personal File List

DEMO

LINKS

- Team Website:
  http://www.cs.rit.edu/~jjp1820/distributed/

- FreePastry:
  http://www.freepastry.org/