Tapestry: Wide-area Location and Routing

Ben Y. Zhao
John Kubiatowicz
Anthony D. Joseph
U. C. Berkeley
Why Tapestry?

- Distributed systems scaling to WAN
  - Larger scale $\Rightarrow$ frequent component faults
  - More data + centralization $\Rightarrow$ performance bottleneck
  - Dynamic environment $\Rightarrow$ manageability complexity
  - More principals $\Rightarrow$ attacks on system (e.g. DoS) more likely

- Tapestry:
  - Decentralized approach to location and routing focusing on fault-resilience and adaptability
  - Builds on previous work: Plaxton trees
Plaxton Trees

Wide-area naming
- Nodes/Objs named by hashed bit-sequence IDs

Incremental routing
- Route to root via local neighbor maps
- Incremental progress towards destination

Properties
- Exploits search locality
- Route around failures
- Decentralized scaling
- Log_b N hops to destination

Route(3B8C -> 203A)
Tapestry Improvements

- **Root nodes => single point of failure**
  - Soln: Root redundancy via hash salts

- **Topology changes => high cost**
  - Soln: Local heartbeats, alternate pointers, second chance invalidation

- **Dynamic system => error persistence**
  - Soln: Proactive node-integration, fault-detection, Self-optimization via query state

- **Vulnerable to DoS attack**
  - Soln: Approx. nodes for load diversion, online data verification, compromised node isolation
Project Status

- Providing location/routing support for the Oceanstore global storage project
  - http://oceanstore.cs.berkeley.edu
- Java-based prototype
- C-based simulation / measurements
- For more details, see Poster Session
- Contact
  - ravenben@cs.berkeley.edu