OceanStore Status and Directions
ROC/OceanStore Retreat 6/10/02

John Kubiatowicz
University of California at Berkeley
Everyone’s Data, One Utility

- Millions of servers, billions of clients ....
  - 1000-YEAR durability (excepting fall of society)
  - Maintains Privacy, Access Control, Authenticity
  - Incrementally Scalable (“Evolvable”)
  - Self Maintaining!

- Not quite peer-to-peer:
  - Utilizing servers in infrastructure
  - Some computational nodes more equal than others
Big Push: OSDI

- We analyzed and tuned the write path
  - Many different bottlenecks and bugs found
  - Currently committing data and archiving it at about 3-5 Mb/sec
Big Push: OSDI

• Stabilized basic OceanStore code base

• Interesting issues:
  - Cryptography in critical path
    • Fragment generation/SHA-1 limiting archival throughput at the moment
    • Signatures are problem for inner ring
      - (although - Sean will tell you about cute batching trick)
  - Second-tier can shield inner ring
    • Actually shown this with Flash-crowd-like benchmark
  - Berkeley DB has max limit approx 10mb/sec
    • Buffer cache layer can’t meet that
OceanStore Goes Global!

- OceanStore components running “globally:”
  - Australia, Georgia, Washington, Texas, Boston
  - Able to run the Andrew File-System benchmark with inner ring spread throughout US
  - Interface: NFS on OceanStore

- Word on the street: it was easy to do
  - The components were debugged locally
  - Easily set up remotely

- I am currently talking with people in:
  - England, Maryland, Minnesota, ....
  - Intel P2P testbed will give us access to much more
Inner Ring

- Running Byzantine ring from Castro-Liskov
  - Elected “general” serializes requests
- Proactive Threshold signatures
  - Permits the generation of single signature from Byzantine agreement process
- Highly tuned cryptography (in C)
  - Batching of requests yields higher throughput
- Delayed updates to archive
  - Batches archival ops for somewhat quiet periods
- Currently getting approximately 5Mb/sec
We have Throughput Graphs! (Sean will discuss)
Self-Organizing second-tier

• Have simple algorithms for placing replicas on nodes in the interior
  – Intuition: locality properties of Tapestry help select positions for replicas
  – Tapestry helps associate parents and children to build multicast tree

• Preliminary results show that this is effective

• We have tentative writes!
  – Allows local clients to see data quickly
Effectiveness of second tier

![Bar chart showing the time (ms) for different operations with and without a multicast tree. The chart compares Disseminate Updates to Replicas, Process Update at Inner Ring, and Send Update Request to Inner Ring. The x-axis represents the number of replicas, and the y-axis represents time in milliseconds. The chart is divided into two sections: one without a multicast tree and one with a multicast tree. The bars indicate the time taken for each operation at different replica counts.]
**Archival Layer**

- Initial implementation needed lots of tuning
  - Was getting 1Mb/sec coding throughput
  - Still lots of room to go:
    - A "C" version of fragmentation could get 26MB/s
    - SHA-1 evaluation expensive

- Beginnings of online analysis of servers
  - Collection facility similar to web crawler
  - Exploring failure correlations for global web sites
  - Eventually used to help distribute fragments
New Metric: FBLPY

- No more discussion of $10^{34}$ years MTTF
- Easier to understand?
Basic Tapestry Mesh
Incremental suffix-based routing
Dynamic Adaptation in Tapestry

• New algorithms for nearest-neighbor acquisition [SPAA ’02]
• Massive parallel inserts with objects staying continuously available [SPAA ’02]
• Deletes (voluntary and involuntary): [SPAA ’02]
• Hierarchical objects search for mobility [MOBICOM submission]
• Continuous adjustment of neighbor links to adapt to failure [ICNP]
• Hierarchical routing (Brocade): [IPTPS’01]
Reality: Web Caching through OceanStore
Other Apps

• This summer: Email through OceanStore
  - IMAP and POP proxies
  - Let normal mail clients access mailboxes in OS

• Palm-pilot synchronization
  - Palm database as an OceanStore DB

• Better file system support
  - Windows IFS (Really!)
Summer Work

• Big push to get privacy aspects of OceanStore up and running
• Big push for more apps
• Big push for Introspective computing aspects
  - Continuous adaptation of network
  - Replica placement
  - Management/Recovery
  - Continuous Archival Repair
• Big push for stability
  - Getting stable OceanStore running continuously
  - Over big distances
  - ...

For more info:

- **OceanStore vision paper for ASPLOS 2000**
  “OceanStore: An Architecture for Global-Scale Persistent Storage”
- **OceanStore paper on Maintenance (IEEE IC):**
  “Maintenance-Free Global Data Storage”
- **SPAA paper on dynamic integration**
  “Distributed Object Location in a Dynamic Network”
- **Both available on OceanStore web site:**
  http://oceanstore.cs.berkeley.edu/