Engineering Issues in Preserving Large Digital Collections

- David S. H. Rosenthal
- LOCKSS Program
- Stanford University Libraries
- http://www.lockss.org/

Petabyte for a Century

- Suppose need to keep petabyte for century
  - With 50% chance of every bit surviving undamaged
  - Now that's *big*, in 100 years its 10^{-9} of a hard drive
- 0.8 exabit-year with 50% survival unimpaired
  - Consider possibility of *bit rot* affecting the system
  - Radioactivity analogy, small probability of bit flip
  - Bit half-life $0.8*10^{18}$ yr = ~100M times age of universe
- Imagine contracting for a box that does this
  - Give a test lab a year for black-box benchmark of bids
  - Demonstrated bit half-life $>0.8*10^{18}$ yr to qualify

Test Lab

- Build exabyte system, watch for year
  - That's $500M worth of disk
  - See ~5 bit flips
- Write exabyte once, read 9 times
  - 3 Tb/s sustained I/O bandwidth
- 64-bit arithmetic => 140 peta-comparisons
  - Say we need only 1% chance of mis-comparison
  - That's 18 nines reliability for comparison software
- Black-box approach isn't feasible
  - Reliability requirement beyond our ability to measure

Threat Model

- Media failure
- Hardware failure
- Software failure
- Network failure
- Obsolescence
- Natural Disaster

Threat Model

- Media failure
- Hardware failure
- Software failure
- Network failure
- Obsolescence
- Natural Disaster
- Operator error
- External Attack
- Insider Attack
- Economic Failure
- Organization Failure

Rules of Thumb

- Safer data but higher cost from:
  - More replicas (Lamport 1982)
    - BFT: $3f+1$ replicas survive f simultaneous faults
  - More independent replicas (Baker 2006)
    - Less correlation between faults, therefore
    - Fewer simultaneous faults
  - More frequent audits of replicas (Baker 2006)
    - Shorter lifetime of latent faults, therefore
    - Lower probability of coinciding faults

Implications

- #1 – make replicas cheaper
  - Nothing improves reliability more than replicas
- #2 – make replicas more independent
  - Correlations happen in unexpected ways
    - Tape & disk not tape vs. disk
- #3 – make auditing cheaper, less intrusive
  - Hardware & software need to work together
    - Auditing has to include the access copy

How Likely Are The Threats?

Examples:

- Hardware
  - Schroeder 2007
  - Pinheiro 2007
- Software
  - Prabhakaran 2005
  - Yang 2006
- Operator Error
  - "Most important cause of data loss"
  - Under-reported
- Internal Attack
  - Secret Service report
  - Under-reported
- External Attack
  - Software mono-culture
  - Staniford 2002

LOCKSS

- Lets libraries build collections from Web
  - In use at ~200 libraries worldwide for ~3 years
  - Preserving e-journals, e-books, ETDs, gov docs, ...
  - Collect by web crawl, disseminate by web proxy
- Preserve by P2P mutual audit/repair
  - natural overlap of collections => huge replication
  - highly independent replicas continually audited
  - if damage or loss detected, automatically repaired
- Independently managed local collections
  - cooperate *only* to reduce cost, raise reliability

LOCKSS Audit & Repair

- LOCKSS boxes continually call polls
  - Invite sample of voters to prove their copies the same
  - Voters compute hash of nonces + their copy
  - Poller computes hash of nonces + its copy, tallies votes
- Three possible poll results
  - Landslide agreement -> poller's copy good
  - Landslide disagreement -> poller's copy bad, get repair
  - Contested poll -> coherent damage -> attack
- Details complex, see ACM ToCS paper
  - Best paper SOSP'03, ACM research award

Credits

- LOCKSS Engineering Team (since 1998)
  - Tom Lipkis, Tom Robertson, Seth Morabito, Thib G-C.
- LOCKSS Research Team (since 2001)
  - Mary Baker, Mehul Shah & colleagues @ HP Labs
  - Mema Roussopoulos & students @ Harvard CS
  - Petros Maniatis & interns @ Intel Research Berkeley
- Vicky Reich, and funding from
  - NSF, Mellon, libraries, LoC, publishers, Sun, ...