Reliability–Aware Energy Management for Hybrid Storage Systems

Wes Felter, Anthony Hylick, John Carter
IBM Research – Austin
Energy Saving using Hybrid Storage with Flash Caching

- **Goal:** Demonstrate significant disk energy savings for storage systems
- **Constraint:** Maintain performance and reliability
- **Target:** Medium-duty workloads  
  - can tolerate infrequent multi-second spin up delays, e.g., email, web, and file servers
- **How we do it:**  
  - Use flash SSD as a secondary cache behind DRAM  
  - Exploit (or create) opportunities to spin down idle disks  
  - Use token bucket to limit disk spinup wear
- **Why this saves energy:**  
  - Replaces high-energy disks (e.g., SAS/FC) with low-energy disks (e.g., SATA) and SSDs  
  - Spins down disks that are idle because I/O requests are serviced by the flash cache
Disk spindown background

- Disks are not very *energy-proportional* – idle uses nearly the same power as active
- Significant energy savings requires spindown
- Spinup takes time and consumes significant energy
  - Breakeven time is critical
- Plenty of work in this area
  - Extending battery life in laptops
  - Spindown timeout of 2x breakeven time shown to be *competitive*
  - Workload-adaptive timeouts
  - Servers – MAID, power-aware RAID and caching

- Most prior work treats disk reliability naively
Disk Energy Management

- Disks starting to support multiple idle states:
  - Idle_A: Everything on
  - Idle_B: A + some electronics off
  - Idle_C: B + Lower RPM, park head
  - Standby: C + Spindle motor off
- Trade off power and response time
- Most savings comes from Standby

<table>
<thead>
<tr>
<th>Mode</th>
<th>Power</th>
<th>Recovery Time</th>
<th>Breakeven Time</th>
<th>Max Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Idle_A</td>
<td>5.8 W</td>
<td>0 s</td>
<td>0 s</td>
<td>1 s</td>
</tr>
<tr>
<td>Idle_B</td>
<td>4.5 W</td>
<td>0 s</td>
<td>1 s</td>
<td>4 min</td>
</tr>
<tr>
<td>Idle_C</td>
<td>3.5 W</td>
<td>0.4 s</td>
<td>2.3 s</td>
<td>10 min</td>
</tr>
<tr>
<td>Standby</td>
<td>0.3 W</td>
<td>6 s</td>
<td>15.4 s</td>
<td>15 min</td>
</tr>
</tbody>
</table>

From Western Digital RE2–GP and Seagate Constellation 3.5” SATA disks

- **Observation**: Caching can increase idle intervals → enable more spindown
  - Non-linear relationship between I/O rate and power consumption

- **Constraint**: Each state has a reliability limit
Managing Reliability with Token Bucket Spindown

- Disks are rated for a limited number of lifetime spin-ups
  - Number varies depending on technology (e.g., SAS vs SATA)
  - Typical conservative default spindown policy: fixed timeout = lifetime / # of spin-ups

- Reliability dictates spindown frequency
  - Energy break-even point: 15 seconds (measured)
  - Reliability constraint: one spindown per 15 minutes (lifetime average)
  - Spindowns are a precious resource → do not waste opportunities
  - Fixed timeout policy wastes spindown opportunities during long idle/active phases (e.g., 10 hours of idle time overnight → 40 unused spindown opportunities)

- Key Idea: Use token bucket (from networking) to jointly manage energy & reliability
  - Add one “spindown token” to bucket as often as reliability allows (e.g., 15 mins)
  - Energy management policy can only spin down disk if token is available
  - Allows more aggressive spindown (e.g., after 1 idle minute)
  - Separate token bucket for each idle state

<table>
<thead>
<tr>
<th>Workload</th>
<th>Disk Lifetime</th>
</tr>
</thead>
<tbody>
<tr>
<td>proj_1</td>
<td>4 years</td>
</tr>
<tr>
<td>proj_2</td>
<td>14 years</td>
</tr>
<tr>
<td>prxy_1</td>
<td>1 year</td>
</tr>
<tr>
<td>usr_1</td>
<td>2 years</td>
</tr>
<tr>
<td>src1_1</td>
<td>5 years</td>
</tr>
</tbody>
</table>

Add token every 15 mins

Remove token before spinning down
— Can be combined with any spindown policy.

Accumulated spindown opportunities
Experimental Evaluation

- Used five MSR block I/O traces
  - proj_1, proj_2, prxy_1, usr_1, src1_1
- Two sets of experiments:
  - Simulation
  - Hardware testbed

- Baseline Configuration
  - 8 450 GB 3.5” SAS disks, RAID-6 (2.7 TB)
- Hybrid Storage Configuration
  - 8 750 GB 3.5” SATA disks, RAID-6 (4.5 TB)
  - 2 100 GB (128 GB raw) SandForce SF-1500 SSD cache (mirrored)

- Approximately equal-cost configurations
  - Note: SATA gives extra capacity (unused in our experiments)
TRAIDe Simulator

- Trace-driven RAID array energy Simulator
- Block trace-driven storage array software simulator
  - RAID-5 and RAID-6
  - Energy-aware LRU flash caching
  - Several disk spindown policies
  - Outputs the time and energy spent in each power state (reading, writing, seeking, spindown, idle, etc.) per disk

- Based upon prior research that accurately generates disk energy models from performance characteristics
  - Minimal disk profiling required
  - Seek time taken from disk data sheets
  - Does not model detailed timing for each request
  - Simulator output validated to be within 5% of measured energy
Policies Studied

- **SAS**: Conventional configuration w/ SAS disks, no flash, no spindown
- **SATA**: SATA disks, no flash cache, conservative fixed-timeout spindown
- **Write-back caching (WC)**: SATA + write-back mirrored energy-aware flash cache (Zhu et al.)
- **Token Bucket (TB)**: WC + competitive spindown algorithm moderated by token bucket — our contribution
  - Disk spins down when it is idle for twice the breakeven time and a token is available
- **Safe Oracle (SO)**: WC + reliability-aware oracle spindown
  - Disk spins down during the longest 672 intervals (avg. one spindown per 15 mins for one week)
  - Disk exactly meets its reliability target
  - Lowest possible energy while maintaining reliability
Time spent in each power state — Simulation

Breakdown (by trace) of the time spent in each state for each trace

Percentage of Time in Each State

proj_1          proj_2          prxy_1          usr_1          src1_1

(Microsoft Trace Used)

- IDLE_B
- IDLE_A
- Active Idle
- STANDBY
- IDLE_C

Page 9
Energy consumption — simulation

![Energy Consumption Chart](chart.png)

- **Energy Consumed (by policy) by each trace**

  - **proj_1**
  - **proj_2**
  - **prxy_1**
  - **usr_1**
  - **src1_1**
  - Averages

- **Energy Consumed in each state (Joules)**

- **Microsoft Trace Used**

- **Legend**:
  - IDLE_C
  - IDLE_B
  - IDLE_A
  - Active Idle
  - Active (R, W, S)
  - SSD
  - Spin-Up (all)
  - STANDBY

© 2011 IBM Corporation
Token accumulation over time

Some workloads never consume all their tokens

One week

Tokens accumulate over weekend

Bursty I/O consumes tokens
Performance — measured experimentally

- Heaviest hour of proj2
- Measured on real hardware
  - x86 server
  - Linux storage stack
  - Custom flash cache
- Equal-cost comparison:
  - 8 15K SAS disks vs
  - 8 7200 SATA disks + 100 GB SSD flash cache
- SATA capacity >> SAS

- System with caching is as fast or faster than without *(note log scale!)*
- proj2 representative of all runs (essentially identically-shaped CDF plots)
Related Work

- Making hard disks more energy efficient
  - DRPM (Gurumurthi 2003)
  - Intra-Disk Parallelism (Sankar 2008)

- Disk Spin-down Techniques
  - Laptops (Wilkes 1992, Douglis 1995)
  - Massive Array of Idle Disks (MAID) (Colerelli 2002)
  - Popular Data Concentration (Pinheiro 2004)
  - PARAID (Weddle 2007)
  - Write Off-Loading (Narayanan 2008)

- Flash Caching
  - SieveStore (Pritchett 2010)
  - FlashCache (Kgil 2006)

- Energy-Aware Caching
  - Power-Aware Cache Management (Zhu 2004)
  - NVCache (Bisson 2006)
  - Augmenting RAID with SSD (Lee 2008)
  - C-Burst (Chen 2008)

- Disk Reliability
  - Failure trends in a large disk drive population (Pinheiro 2007)
Conclusions

- 85% energy savings possible with spindown and hybrid storage
- Disk energy management must be reliability-aware
- Reliability management and energy management are separable concerns
- Token bucket reliability management is near-optimal
- Intermediate power states provide little benefit
Thank you!

Questions?
SAS vs. SATA disks

- "SAS": High RPM (10–15K), lower latency, lower capacity, higher power, higher MTBF*, fewer spinups, higher cost
- "SATA": Low RPM (5–7K), higher latency, higher capacity, lower power, lower MTBF, more spinups, lower cost

- Conventional wisdom: Only SAS drives can meet enterprise workload demands
  - E.g. Sub–10 ms latency

- Flash changes the situation
  - Sub–ms flash latency can offset slower SATA disks
Reliability-Aware (Safe) vs. Unsafe Oracle

Energy (Joules) Consumed (by policy) by each trace

Energy Consumed in each state

proj_1  proj_2  prxy_1  usr_1  src_1  averages

- IDLE_C
- IDLE_B
- IDLE_A
- Active Idle
- Active (R, W, S)
- SSD
- Spin-Up (all)
- STANDBY
Why time in each state

- Include hit rates here

<table>
<thead>
<tr>
<th>Workload</th>
<th>Cache Read Hit Rate (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>proj_1</td>
<td>39</td>
</tr>
<tr>
<td>proj_2</td>
<td>52</td>
</tr>
<tr>
<td>prxy_1</td>
<td>65</td>
</tr>
<tr>
<td>usr_1</td>
<td>67</td>
</tr>
<tr>
<td>src1_1</td>
<td>85</td>
</tr>
</tbody>
</table>
Performance (Simulated)