SRCMap: Energy Proportional Storage using Dynamic Consolidation

Akshat Verma[1] Ricardo Koller[2] **Luis Useche**[2]
Raju Rangaswami[2]

[1]IBM Research, India

[2]School of Computing and Information Sciences
College of Engineering and Computing
FLORIDA INTERNATIONAL UNIVERSITY

FAST Conference, 2010

- Current power density of data centers is 100 W/sq.ft & increasing 15-20% per year.
- Storage consume 10-25% of computing equipment.
- Storage load low (10-30%), but still peak power consumed.
- CPUs are more energy proportionality than storage.
- Consolidation is a well known technique for energy proportionality in virtualized servers.

Storage Consolidation?

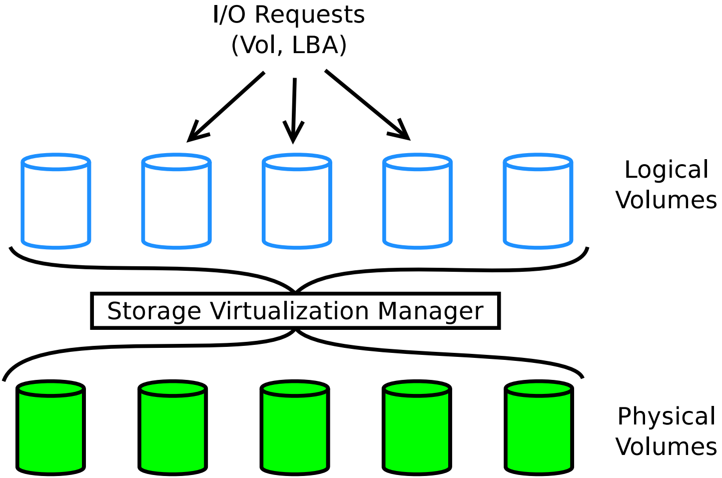
Can we use a storage virtualization layer to design a practical energy proportional storage system?

- Storage virtualization I/O indirection useful for consolidation.

Challenge

Moving logical volumes from one device to another is prohibitively expensive.

Background: Storage Virtualization



Outline

Workloads

- **mail** Our department mail server.
- **web-vm** Virtual machine hosting two web-servers: CS web-mail & online course management.
- **homes** NFS server that serves the home directories for our research group.

Block traces collected downstream of an active page cache for three weeks.

Observations

Observation 1

The active data set is only a small fraction of total storage used. (about 1.5-6.5%)

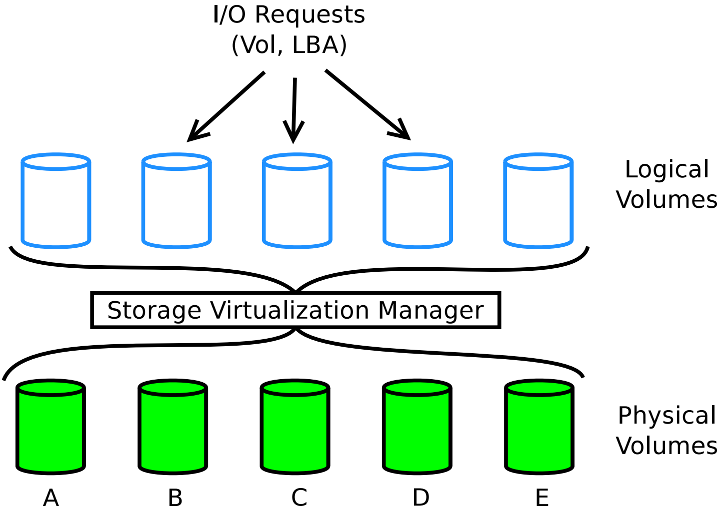
Observation 2

There is a significant variability in I/O load. (5-6 orders of magnitude)

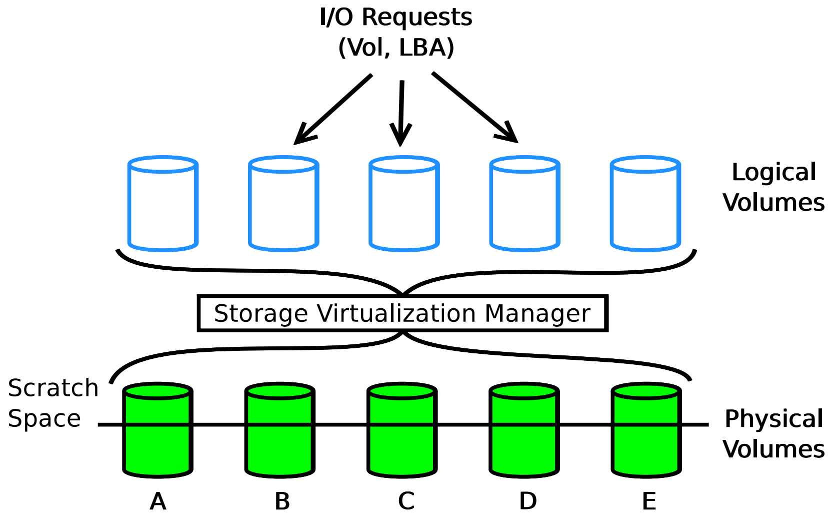
Observation 3

More that 99% of the working set consist of *really popular & recently accessed* data.

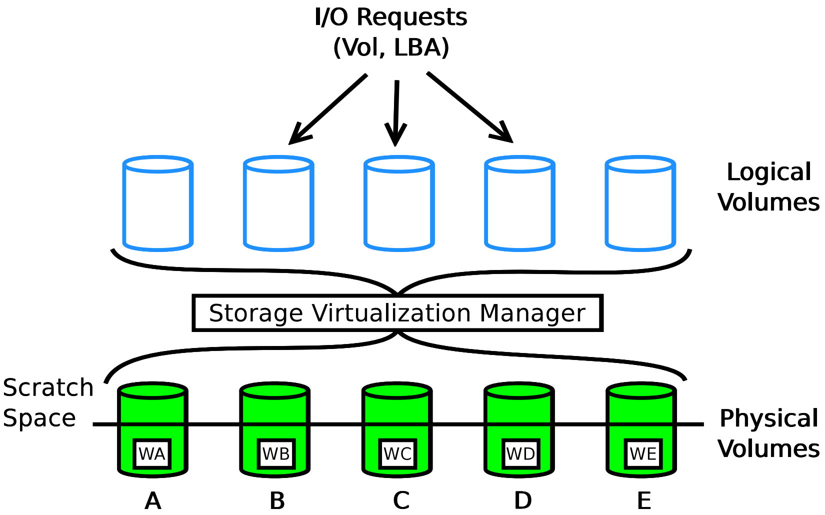
Overview


I/O Requests
(Vol, LBA)
Logical
Volumes
Storage Virtualization Manager
Physical
Volumes
A
B
C
D
E

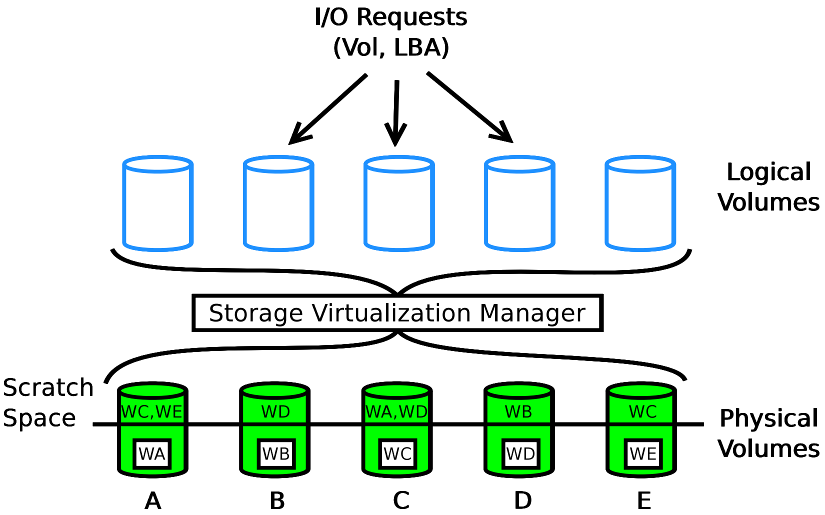
Overview


I/O Requests
(Vol, LBA)
Logical
Volumes
Storage Virtualization Manager
Scratch
Space
Physical
Volumes
A
B
C
D
E

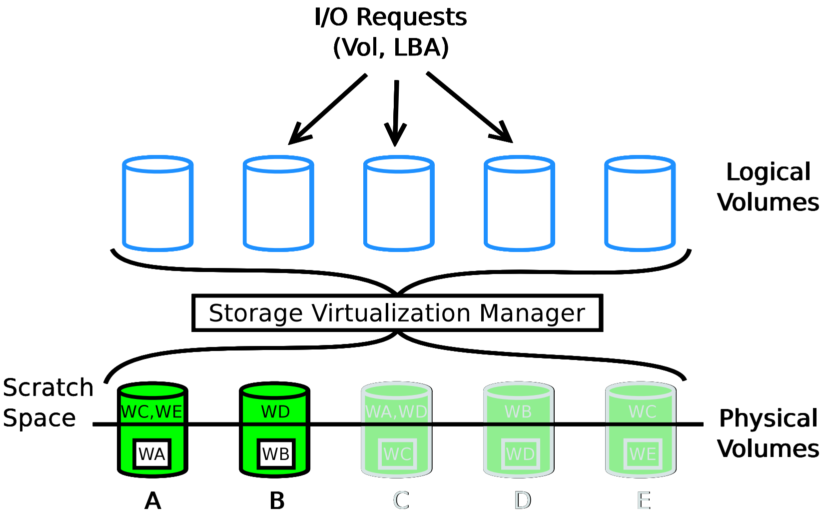
Overview


I/O Requests
(Vol, LBA)
Logical Volumes
Storage Virtualization Manager
Scratch Space
WA
WB
WC
WD
WE
Physical Volumes
A
B
C
D
E

Overview


I/O Requests
(Vol, LBA)
Logical Volumes
Storage Virtualization Manager
Scratch Space
WC,WE
WA
WD
WB
WA,WD
WC
WB
WD
WC
WE
Physical Volumes
A
B
C
D
E

Overview


I/O Requests
(Vol, LBA)
Logical
Volumes
Storage Virtualization Manager
Scratch
Space
WC,WE
WA
WD
WB
WA,WD
WC
WB
WD
WC
WE
Physical
Volumes
A
B
C
D
E

Our Approach

Sample Characterize the logical volume to find the working set.

Replicate Create multiple working-set replicas in various physical volumes' scratch space.

Consolidate Based on I/O workload intensity, activate a sub-set of physical volumes and serve workloads either from original copies or working set replicas on these active disks.

Our Approach

Initialization

Sample	Characterize the logical volume to find the working set.
Replicate	Create multiple working-set replicas in various physical volumes' scratch space.
Consolidate	Based on I/O workload intensity, activate a sub-set of physical volumes and serve workloads either from original copies or working set replicas on these active disks.

Every H hours

Goals → Solutions

Goal
Fine grained proportionality
Low space overhead
Reliability
Workload Adaptation
Heterogeneity support

Goals → Solutions

Goal	Solution
Fine grained proportionality	Multiple replica targets.
Low space overhead	Instead of entire volumes, only working-sets are replicated.
Reliability	Coarse-grained consolidation intervals. (hours)
Workload Adaptation	Update working set replicas with new data that lead to read misses.
Heterogeneity support	Performance-power ratio accounted for in the replica placement benefit function.

SRCMap work-flow

Event	**Response**
Initialization	Detect working-sets of logical volumes & create replicas.
Every H hours	Identify what volumes and replicas to activate the next H hours.
Change in workload	Same as initialization.

SRCMap work-flow

Event	Response
Initialization	Detect working-sets of logical volumes & **create replicas**.
Every H hours	**Identify what volumes and replicas to activate the next H hours**.
Change in workload	Same as initialization.

Replica Placement

- Replication benefit based on:
  1. Working set stability
  2. Average load
  3. Power efficiency of primary physical volume.
  4. Working set size
- Assign space with priorities based on benefit.
- Update replica creation benefit as additional replicas are created.
- Algorithm executes until scratch spaces are full.

Active Replica Identification

- Calculate predicted aggregate workload IOPS.
- Compute minimum number of volumes to serve the aggregate IOPS.
- Identify replicas for inactive volumes.
- The number of active disks is incremented by one in case no active replica has been identified for some inactive volume.

Workloads & Configuration

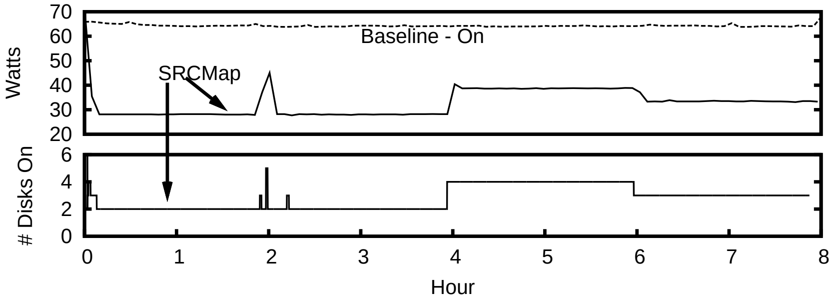
- 8 workloads to independent data volumes.
- Mix of web-servers of our CS department, and file server, SVN, and WiKi for our research group.

- $H = 2$. Change active replicas every 2 hours.
- Two minute disk time-outs.
- Working sets & replicas based on three week workload history.
- We report results of replaying the next 8 most active hours in the traces.
- We assume an oracle for estimation of load during each consolidation interval.

Storage test-bed

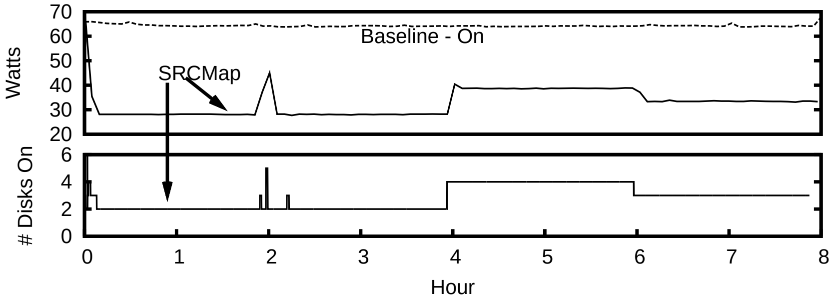
- One machine with 8 SATA ports.
- Intel P4 HT 3GHz, 1GB memory.
- Trace played back using *btreplay*.
- Dedicated power supply for disks connected to power meter.
- *Watts up? PRO* power meter: measures power every second with resolution of 0.1W.

Power



- Power consumption measured every second & active disks every 5 seconds.

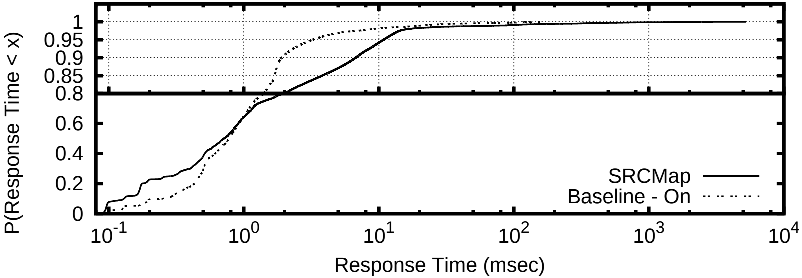
Power



- Power consumption measured every second & active disks every 5 seconds.

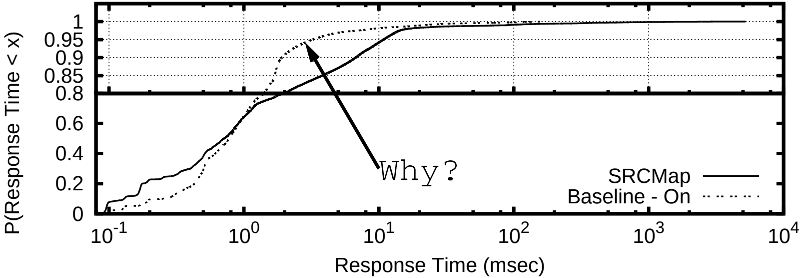
Disks off	Power Saved
4.33	23.5 (35.5%)

Response time



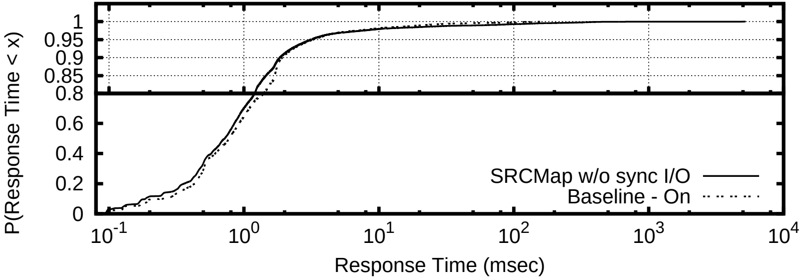
- After 1ms, Baseline - On demonstrate better performance.
- 8% of requests with latencies $\geq$ 10ms.
- 2% of requests with latencies $\geq$ 100ms.

Response time



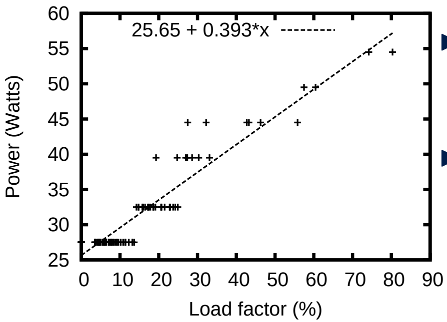
- After 1ms, Baseline - On demonstrate better performance.
- 8% of requests with latencies $\geq$ 10ms.
- 2% of requests with latencies $\geq$ 100ms.
- Synchronization I/Os issued at beginning of each interval.

Response time



- After 1ms, Baseline - On demonstrate better performance.
- 8% of requests with latencies $\geq$ 10ms.
- 2% of requests with latencies $\geq$ 100ms.
- Synchronization I/Os issued at beginning of each interval.
- Replaying without sync I/Os follows Baseline-On more closely.

Energy proportionality



- One point for each 2-hour interval in 24-hour duration.
- **Load Factor**: Load relative to the assumed volume maximum load capacity.

SRCMap is able to achieve close to N-level proportionality for a system with N physical volumes.

Conclusions

- We proposed and evaluate SRCMap, a storage virtualization solution for energy proportional storage.
- SRCMap establishes the feasibility of energy proportional storage systems.
- SRCMap meets all goals we set out to achieve energy proportional storage:
  - ✓ Low space overhead
  - ✓ Reliability
  - ✓ Workload adaptation
  - ✓ Heterogeneity support
  - ✓ Fine grain energy proportionality

Future Work

- Models to predict I/O workload intensity.
- Models that estimate the performance impact of storage consolidation.
- Investigate the presence of workload correlation for better workload estimation and consolidation decision.
- Optimizing the scheduling of synchronization I/Os to minimize impact on foreground requests.

http://dsrl.cs.fiu.edu/projects/srcmap/

Questions?

Related Work

- Singly redundant schemes: Spin down volumes with redundant data during low load.
- Geared RAIDs: Redundancy on several disks and each disk spun down represents a gear shift.
- Caching systems: Cache of popular data on additional storage.
- Write Offloading: Increase disk idle periods by redirecting writes to alternate locations.

Other Methods