Design Considerations for Using Flash Memory for Caching

Edi Shmueli, IBM

XIV Storage Systems

edi@il.ibm.com

Solid-State Storage

- In a few decades solid-state storage will *replace* all spinning disks in enterprise data stores
  - We are not there yet…
- Prices are still relatively high
  - Not enough manufacturing capacity to satisfy storage needs
- Today it usually *complements* existing storage
  - High performance, low latency, low power, etc.
- Flash memory is the dominant technology
  - As primary store (persistent storage)
  - As cache in front of the spinning disks (buffer pool)

Flash as Primary Store

- Standard disk-drive form factor (SSD)
  - Compatible with current subsystems designs
- Good performance for relatively small # of SSDs
  - Up to 8x improvement for some workloads
- Main question: what data to put on the SSD?
  - Filesystem metadata, database indexes, logs, etc.
- Can be done manually or *semi*-automatically
  - Tiering software: LUN and sub-LUN levels
- Considered a disruptive process
  - Low frequency: at nights or during periods of low activity

Flash as a Cache

- Much less disruptive and more dynamic approach
  - Adapts quickly and with minimal interference to short-term conditions in the workload e.g., few seconds of locality
- No administration costs
  - Inherently transparent and fully-automated solution
- Much more difficult to implement
  - Leverage the Flash capacity for performance, while hiding the fact it is much slower than DRAM
- Various design options
  - Extension to DRAM cache (single LRU list)
  - Second-level cache, etc.

Considerations

- Reliability
- Read-only vs. write caching
- Caching algorithms
- Metadata
- Miscellaneous

Reliability

- Most disturbing issue with Flash technology
  - Not enough field statistics
- Device-level failures (SSD)
  - No moving parts: expected to be better than disks
- Flash medium failures (bit errors)
  - Quality deteriorates with usage (wear-out)
  - Quality deteriorates with time (retention)
- Caching workloads difficult to anticipate
  - Different IO patterns compared to disks

Read-Only vs. Write Caching

- Read-only is a simpler option
  - In presence of bit errors read data from the disks
- Write cache complicates things
  - Write-through (read cache extension)
  - Write-back exposes to potential data loss
- Good idea to consider redundancy
  - Inter-device redundancy e.g., RAID-like
  - Intra-device redundancy
- Are RAID schemes appropriate for Flash?
  - Are Flash failures correlated?

Caching algorithms

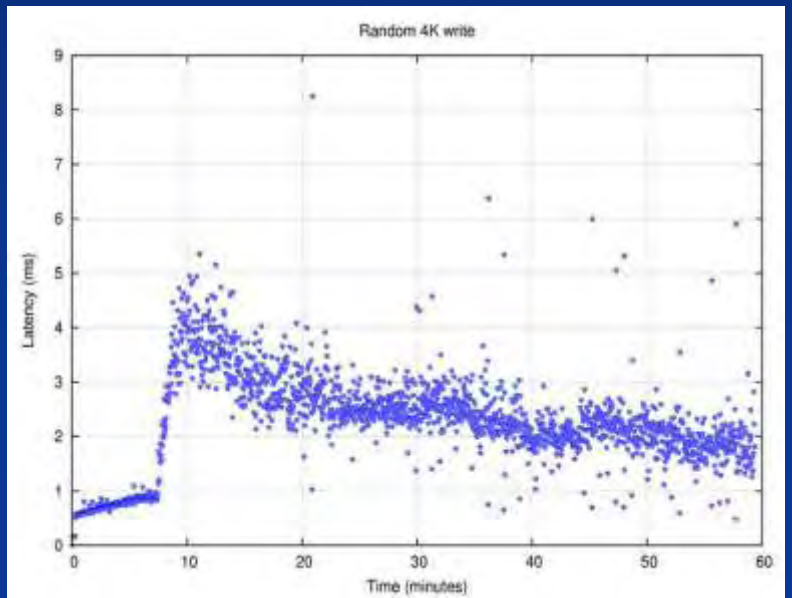
- Baseline algorithm is LRU
  - LFU (1970), FBR (1990), 2Q (1994), LRU-2 (1999), LRFU (2001), MQ (2001), LIRS (2002), ARC (2004)
- Rules of thumb for good cache performance
  - Five-minutes rule (Gray & Putzolu, 1987)
  - Empirical study (Bruce McNutt, 1998)
- 256GB SSD doing 100MB/s takes 45m to fill-up
  - Might not need sophisticated algorithms
- Uncontrolled caching leads to excessive wearing
  - Account for endurance in the algorithm

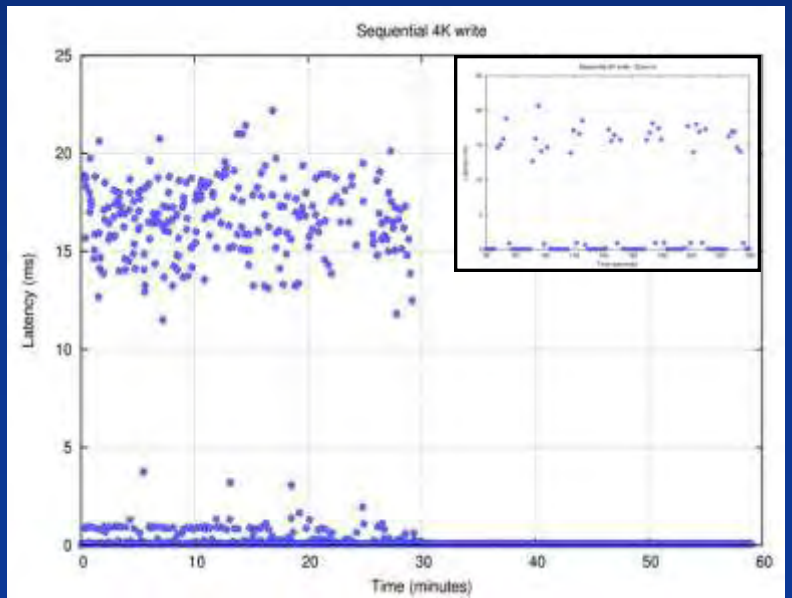
Metadata

- Data structures to help locate data in the cache
  - Typically a few tens of MB, DRAM resident
- Usually discarded on shutdown / reboot
  - Few minutes of warm-up penalty
- For 256GB SSD using 4KB pages
  - 64M entries x 8B (LPA + Bitmap) → 0.5GB
- Extremely long warm-up periods
  - Up to several hours of degraded performance

Miscellaneous

- Flash controllers are complicated entities
  - Proprietary algorithms for wear-leveling, GC, etc.
- Performance might not always be predicted
  - Random write latency bursts exhibiting high variance
  - Sequential latency drops to microseconds after 30 minutes





Summary

- Cache-to-storage ratios have dropped significantly
  - 1991: IBM 3990: 256MB cache / 20GB storage → 1.3%
  - Today: 64GB cache / tens of TB storage → less than 0.1%
- Flash memory is a great opportunity to close the gap
  - Potentially huge impact in performance
- New challenges requiring new ways of thinking
  - Less sophisticate algorithms but account for endurance
  - Huge metadata effect on warm-up, etc.

http://www-03.ibm.com/ibm/history/exhibits/storage/storage_3390.html

IBM *XIV* Storage Systems

http://www.xivstorage.com/

- Easy management
  - No more ILM
  - Less power higher density
- Superior performance
  - Optimal use of resources
  - Innovative cache architecture
- Superb reliability
  - 30-minutes rebuild time or less!
  - Innovative grid-based redundancy
- Powerful snapshotting
  - Instant snapshot creation
  - No performance overhead