Achieving Flash Memory’s Full Potential

Darpan Dinker

VP Engineering

Schooner Information Technology

Forward Looking Statement

During our meeting today we will be making forward-looking statements.

Any statement that refers to expectations, projections or other characterizations of future events or circumstances is a forward-looking statement, including those relating to revenue, pricing, market share, market growth, product sales, industry trends, expenses, gross margin, future memory technology, production capacity and technology transitions and future products.

Actual results may differ materially from those expressed in these forward-looking statements due to the factors detailed under the caption "Risk Factors" and elsewhere in the documents we file from time-to-time with the SEC, including our annual and quarterly reports.

We undertake no obligation to update these forward-looking statements, which speak only as of the date hereof.

Agenda

- Flash memory application in the data-center
- Why focus on balanced systems
- Architectural approaches to exploit Flash
- Scenarios to leverage Flash
- Lessons learned

Flash Memory Application in Data Centers

Cloud

VDI

Databases

Virtualization

Hadoop

Big Data

SharePoint

NoSQL

Multi-tier storage

SAN NAS

Memcached

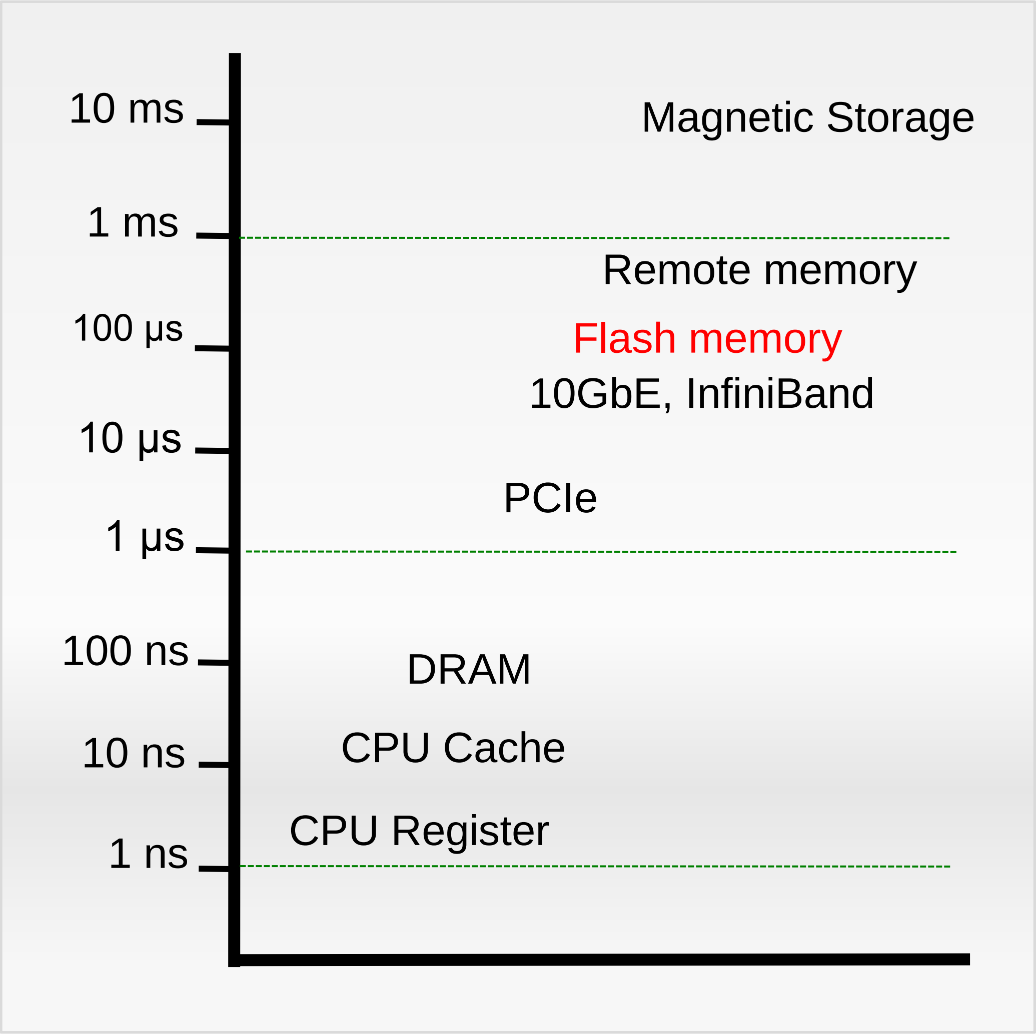
MongoDB

MySQL

Exchange

Why Focus on Balanced Systems

- CPU
- Main-Memory
- **Storage**
- Network



Architectural Approaches to Exploiting Flash Memory

1. Augment main-memory
   - Directly (mmap, system paging, custom kernel)
   - Indirectly (object cache)

Memcached

Architectural Approaches to Exploiting Flash Memory

1. Augment main-memory
   - Directly (mmap, system paging, custom kernel)
   - Indirectly (object cache)
2. Cache blocks of magnetic storage
   - Disjoint from storage (WB or WT)
   - Integrated with storage

Architectural Approaches to Exploiting Flash Memory

1. Augment main-memory
   - Directly (mmap, system paging, custom kernel)
   - Indirectly (object cache)
2. Cache blocks of magnetic storage
   - Disjoint from storage (WB or WT)
   - Integrated with storage
3. Replace magnetic storage
   - DAS
   - NAS or SAN

Scenarios for Leveraging Flash Memory

1. Applications are IO bound
2. Lots of random IO
3. Prepare for data growth
4. Heavily virtualized environment
5. Prepare for unpredictable load
6. Improved response time / SLAs
7. Improved reliability
8. Save on space, power, cooling

Scenarios for Leveraging Flash Memory

1. Applications are IO bound

- Common symptoms
  - Low CPU usage
  - High io-wait
  - Sluggish response time (are locks to blame?)

Scenarios for Leveraging Flash Memory

2. Lots of random IO

Databases

MongoDB

MySQL

- Magnetic drive heads cannot keep up with random IO (~200 IOPS/drive)
- Common symptoms observed with 'iostat'
  - High IO service time, queue size, wait time
  - Low IO bandwidth

Scenarios for Leveraging Flash Memory

3. Prepare for data growth

Big Data

- Data growth can increase demand for IO
- Common symptoms
  - Database queries slow down as table size increases
  - Wrong trends observed with 'iostat' as data grows

Scenarios for Leveraging Flash Memory

4. Heavily virtualized environment

Cloud

VDI

Virtualization

- Individual VMs with low IO requirements build up a heavy IO workload for a shared storage system
- Many individual sequential accesses create a random IO workload
- IO path is not particularly optimized in VMs

Scenarios for Leveraging Flash Memory

5. Prepare for unpredictable load

- DAS with magnetic disks have limited upper bound for IOPS. In order to size systems for peak load, the “work” per server may require substantial throttling → resulting in underperforming servers

Scenarios for Leveraging Flash Memory

6. Improved response time / SLAs

- Many applications require disk access. Disk access can be responsible for significant portion of the response time
- Sequential and random access times are asymmetric for magnetic drives. Defining SLAs is thus non trivial

SharePoint

Exchange

Scenarios for Leveraging Flash Memory

7. Improved reliability

- Moving mechanical parts make magnetic drives quite susceptible to failure. RAID is a common practice even if data is made redundant on another server

Scenarios for Leveraging Flash Memory

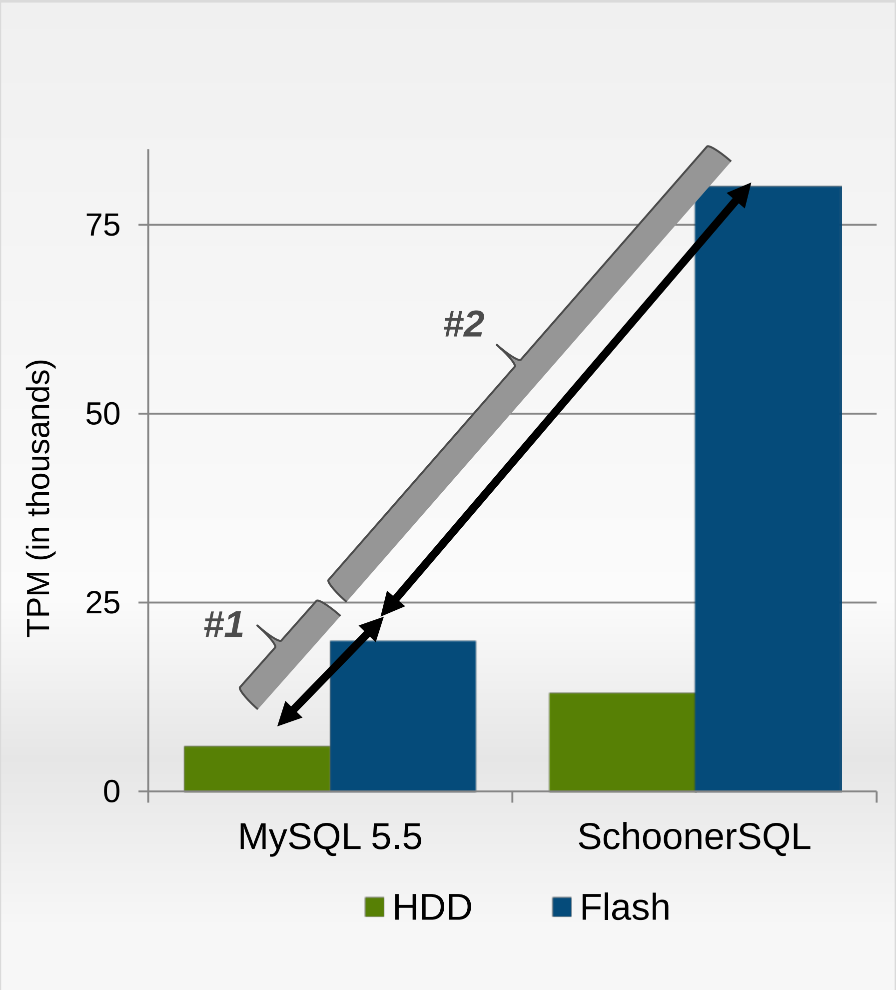
8. Save on space, power, cooling

- Magnetic drives take power and dissipate a lot of heat (>10X that of Flash memory)
- Random IOPS are expensive with magnetic drives

Lessons Learned: Don't Trust the Folk Lore

- Can observe response time and throughput improvements for most IO-bound workloads after exploiting Flash, but…
  - Nothing close to 100X theoretical improvement (5-10ms → 50-100µs)
  - Most existing software applications unable to push IOPS
  - May observe "write cliff" for workloads with heavy writes

Lessons Learned:
Your Mileage May Vary



- Workload: DBT2 open-source OLTP version of TPC-C
  - 1000 warehouses, 32 connections
  - 0 think-time
  - Result metric: TPM (new order)
- Measurement Configuration
  - 2 node Master-Slave configuration
  - 2 socket 6C Westmere, 72GB DRAM

#1 Replace HDD with Flash

#2 Understand application architecture & design. Re-implement portions and integrate with Flash.

Lessons Learned: Tweak or Rewrite Application SW

- Tips
  - Consider bypassing OS IO schedulers (noop)
  - When IO throughput is more important, employ larger block size
  - Target keeping high number of outstanding IOs
- To push 10k-50k IOPS, tweak storage access layer of application
- To push 50k+ IOPS, consider rewriting parallelization and storage access layer

Lessons Learned:
Know the Technology

- Consumer grade vs. enterprise grade device
- Random writes and non-uniform size writes
  - Algorithmic differences between vendors for space and wear management
- IOPS is not everything in production
  - Determine distribution and sustained latency
  - Does the device really offer "durability"?
- Compression in devices fare differently based on data (e.g. jpeg, mpeg)
- Monitor SMART metrics

Lessons Learned:
Know About the Form Factor

- 1 SAS with 1 PCIe Flash device is apples : oranges comparison for throughput
- SAS/SATA based Flash
  - Easier to RAID
  - Typically hot swappable and easy to replace
  - Not all HBAs are built to match Flash speeds
- PCIe based Flash
  - Typically offer IOs with lower latency
  - Convenient to add to existing systems for caching

SanDisk

acquires

SCHOONER

FREE YOUR DATA

June 27, 2012

SanDisk Flash-Accelerated Products

Schooner Membrain SchoonerSQL	Enterprise NoSQL Cache/Store Enterprise SQL Database
FlashSoft™ Caching software	Enterprise Storage Caching
Lightning® SAS Enterprise SSD Lightning® PCIe Enterprise SSA	Enterprise Flash Hardware