



PERCONA
Performance Consulting Experts

An Overview of Flash Storage for Databases

Vadim Tkachenko
Morgan Tocker
<http://percona.com>

MySQL CE
Apr 2010

Introduction

- Vadim Tkachenko
 - Percona Inc, CTO and Lead of Development
- Morgan Tocker
 - Percona Inc, Director of Training

What is talk about

- Flash technologies
 - Server usage
 - not USB/digital camera flash cards
- FusionIO and Intel SSD
- Database (MySQL) application
- Flash changes performance landscape
 - Talk gives basic background what to look into

Revolutionary

- Change in technology
 - From spinning to solid state
 - No mechanical moving parts
 - Jump in performance
 - Requires changes in applications
 - My prediction: in 5-10 years it will replace hard disks totally

Physics behind

- “floating gate transistors”
 - Non-volatile memory
 - (more details)
- One state – Single Level Cell (SLC)
 - Faster, more reliable, more expensive
- Many states – Multi Level Cell (MLC)
 - Usually 4 states
 - Slower, less reliable, cheaper

Classification

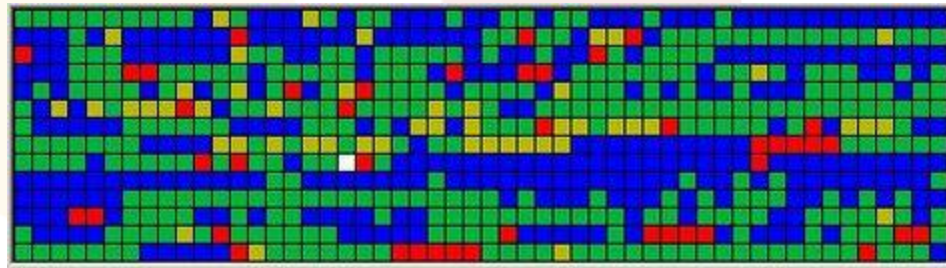
- NOR
 - Random read access (bit granularity)
 - Speed compared with DRAM
 - Slow write and erase
 - Firmware storage
- NAND (this talk about)
 - Faster writes
 - Only block-level read access (4K)
 - Idea is to compact many cells in limited space
 - Make competition with Hard Disk Drives

Erasing (NAND)

- Erase is to set all bits to “1111...”
 - Erasing process is similar to “flash” in photocameras – there where name **FLASH** comes from
 - Erase is slow, done in batch operation (up to 1MB)
- Change “1”->”0” is fast
- Change “0”->”1” is possible only by erasing
 - 1st write: “1111” -> “1110” . Block marked as “written”
 - 2nd write: even “1110” -> “1010” is not possible
 - Smart software could detect it

Erase challenges

- Erase is slow
 - You want to erase many blocks in single flash
 - Block management
- When you write – card never writes the same block
- Background process to run garbage collector

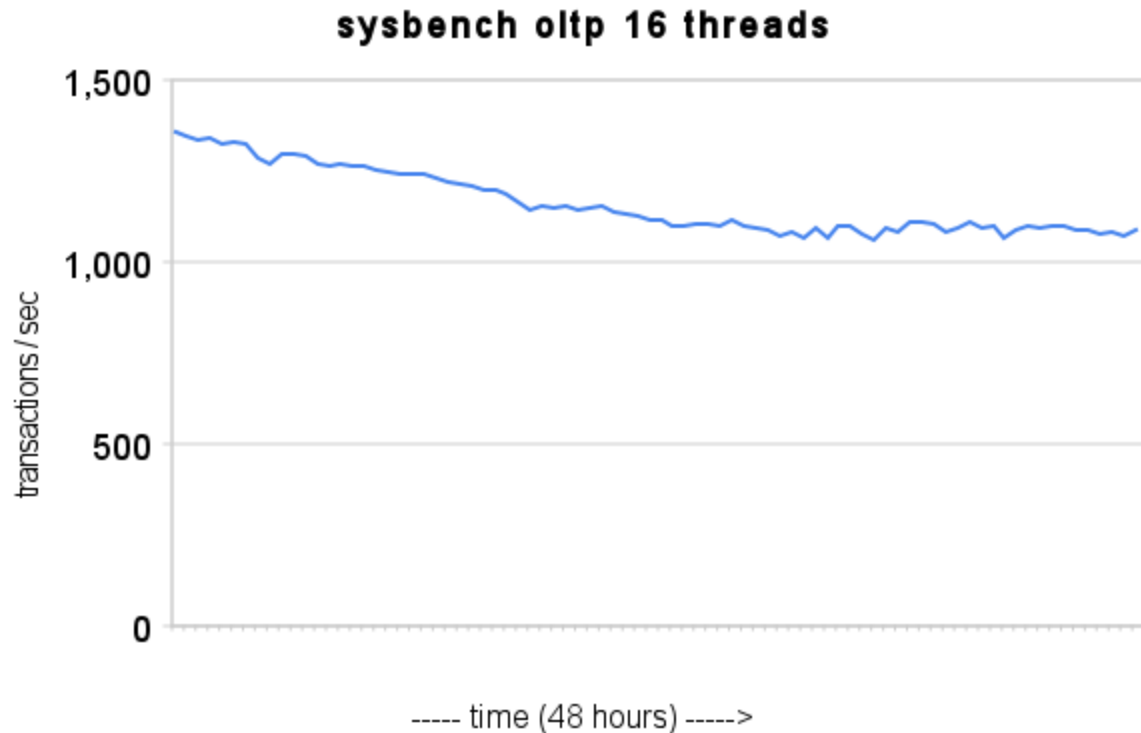


Erasing lifetime

- SLC
 - 100.000 times per cell (may vary)
- MLC
 - 10.000 times per cell (may vary)
- Many cell and even distribution (wear leveling) make it couple years under heavy write load

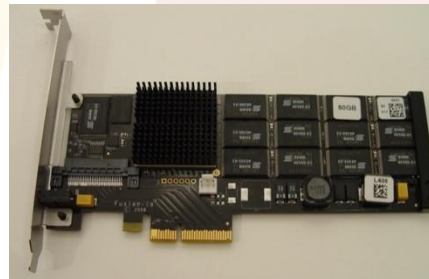
Write degradation

- Expected, steady state
 - Graph for FusionIO 320GB MLC card



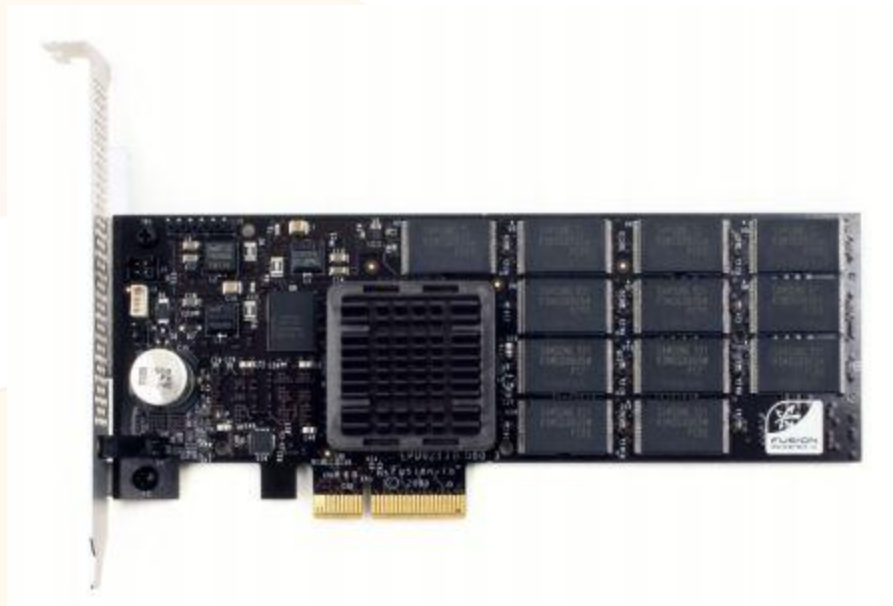
Soft(firm)ware matters

- Complexity of erasing process make software logic really important



FusionIO Intel SSD

FusionIO



FusionIO performance

- Data from specification:
- 160 GB SLC card
 - 116K read IOS (4K)
 - 26 μ s read latency
- 320 GB MLC card
 - 71K read IOS
 - 41 μ s read latency
- Lifetime:
 - SLC flash @ 40% write duty | 25 calendar years
 - MLC flash @ 20% write duty | 10 calendar years
 - MLC flash @ 40% write duty | 5 calendar years

FusionIO overview

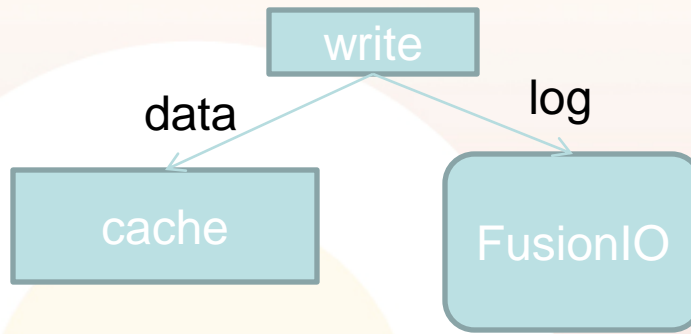
- Fast. Very fast.
- PCI-E, closest to CPU
- MLC / SLC / Duo Cards
- “Transactional” log – durability
- Shares host memory / CPU
- Most complex part – firmware
- Space reservation for heavy writes

FusionIO drawbacks

- Expensive: 50\$/GB (effective space)
 - Requires 25% space reservation
 - Regular DRAM – 30-40\$/GB
 - 320 GB MLC PCIe ioDrive \$6,829.99 (dell.com)
- PCI-E : not “hot-swap”
 - PCI-E errors
 - FusionIO takes care about it

FusionIO - durability

- Cache is located in host system
- “transactional” log



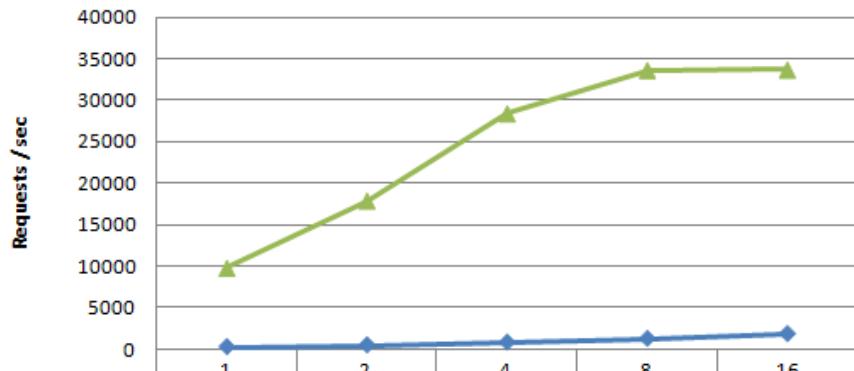
- Crash recovery
- No data loss in case power / system failure

FusionIO read performance

160GB SLC card

8 threads: 33K IOS (525MB/sec), 0.28 ms 95% response time

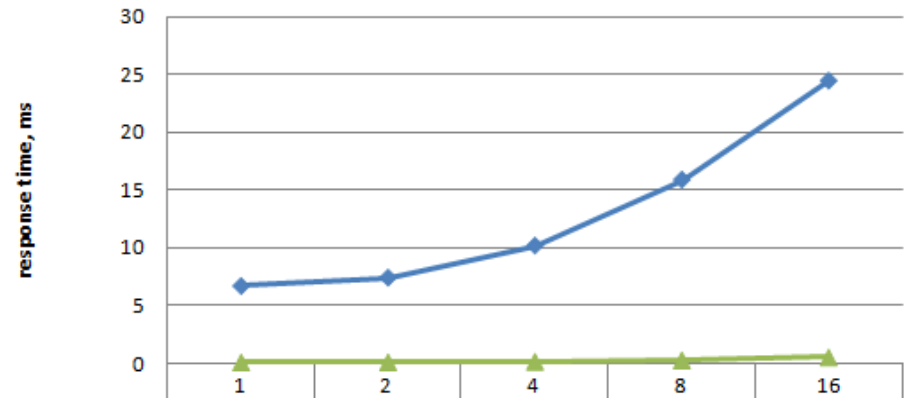
random read, 16K, throughput



RAID10	225.52	443.12	779.88	1229.87	1800.49
FusionIO	9722.24	17866.04	28397.41	33615.89	33672.27

RAID10 is
Dell Perc 6i RAID10 on
8 disks 2.5" 15K RPM SAS

random read, 16K, 95% response time, ms

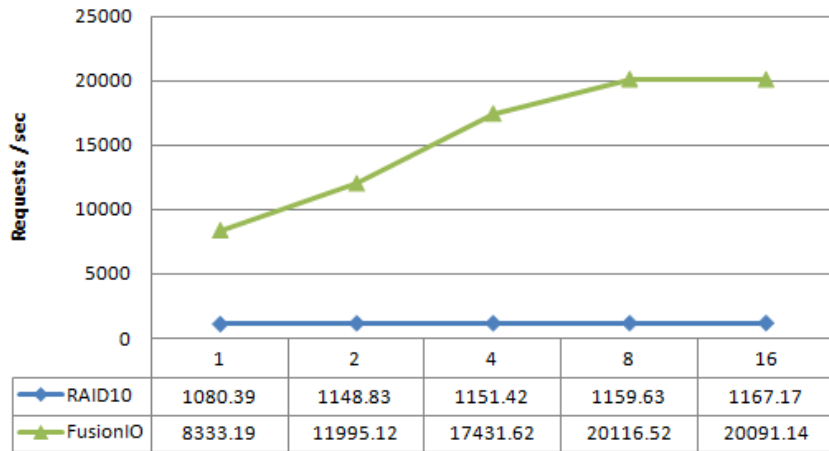


RAID10	6.68	7.4	10.17	15.86	24.39
FusionIO	0.1	0.12	0.17	0.28	0.53

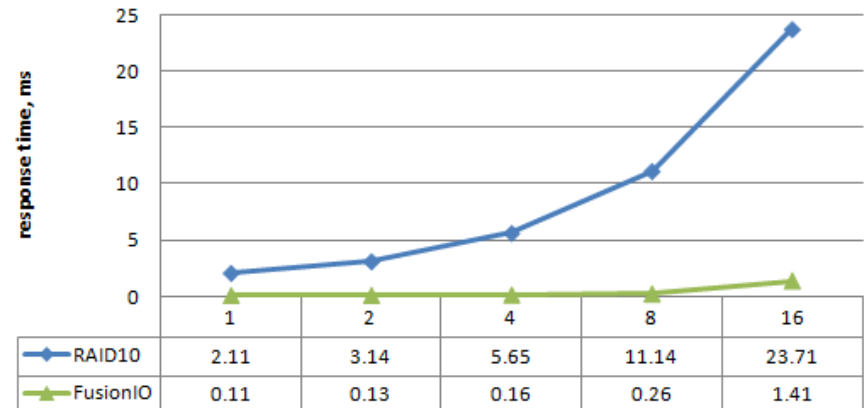
FusionIO write performance

8 threads: 20K IOS (314MB/sec), 0.26 ms 95% response time

random write, 16K, throughput

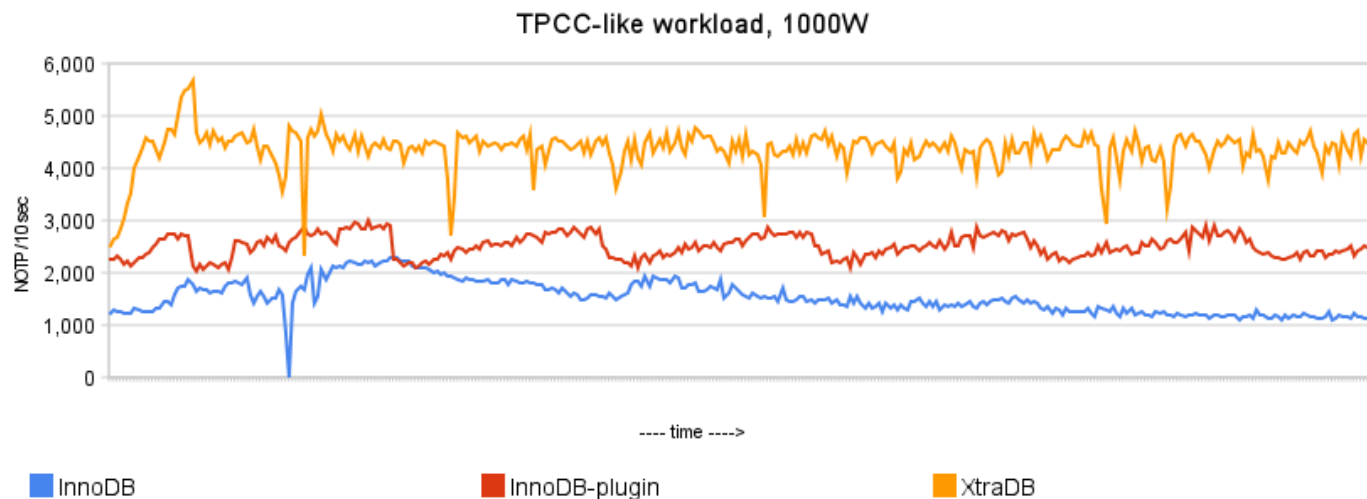


random write, 16K, 95% response time, ms

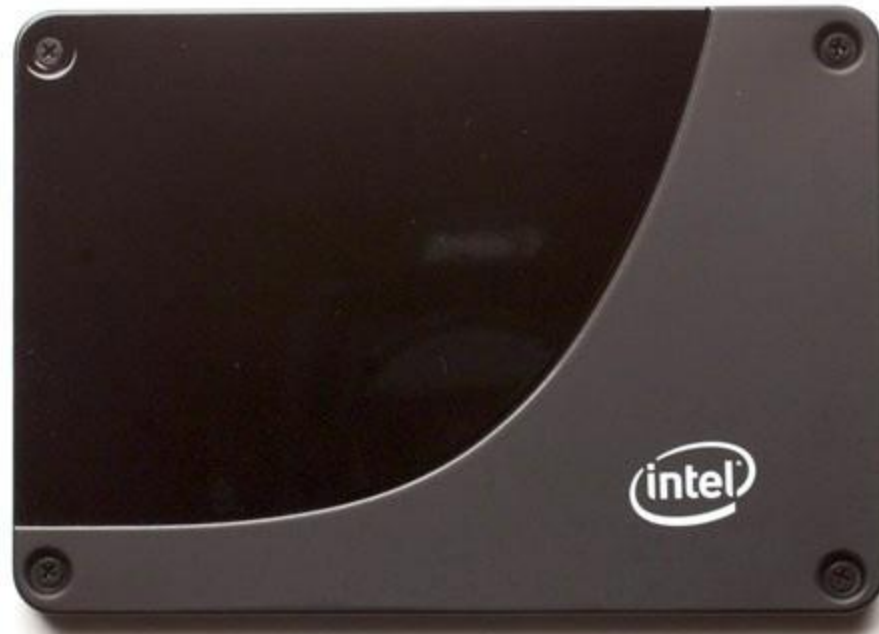


FusionIO – for database

- Many read / write threads to utilize full throughput
- MySQL is not able to load it fully
 - XtraDB / InnoDB-plugin has multi-io threads
- InnoDB IO path has to be re-implemented



Intel SSD



Intel SSD

- SATA form factor
- Intel X25-M Gen I (50nm) & Gen II (35nm)
 - MLC
 - “... High-performance storage for notebook and desktop PCs ...” - intel.com
- Intel X25-E (50nm)
 - SLC
 - “Enterprise”
 - “... Extreme performance and reliability for servers, storage, and workstations. ...” - intel.com

X25-E

- 32GB / 64GB
- Throughput: 35K IOS reads, 3.5K IOS writes
- Latency: 75 μ s reads, 85 μ s writes
- 64 GB - **\$725^{.00}**
 - 11\$/GB
- Write Endurance:
 - 1 petabyte of random writes (32 GB)
 - 2 petabyte of random writes (64 GB)
- Roadmap:
 - 128GB ? Replace SLC->MLC ?

X25-M Gen II

- 80 GB / 160 GB
- Throughput: 35K IOS reads, 6.5 / 8.5K IOS writes
- Latency: 65 μ s reads, 85 μ s writes
- 160GB – 500\$
 - 3.12\$ / GB
- Write Endurance
 - Not mentioned in official specification

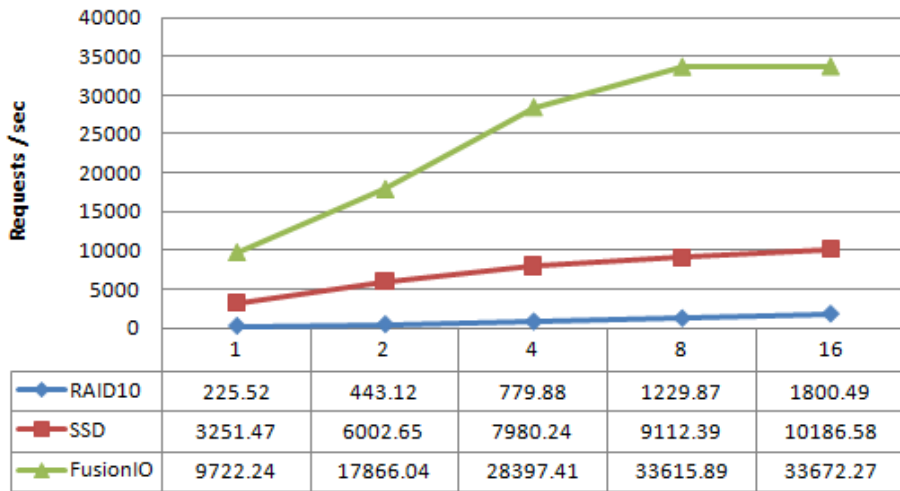
X25-E challenges

- Write cache is not battery backup
 - Loss of transactions
- Disabling write cache is performance hit
- No clear roadmap

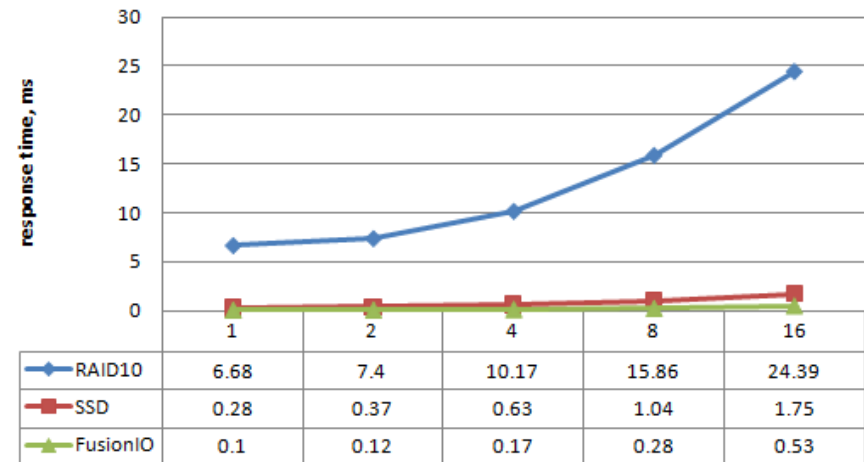
Benchmarks – random read

- X25-E, 8 threads: 9K IOS (140 MB/s), 1.04 ms

random read, 16K, throughput



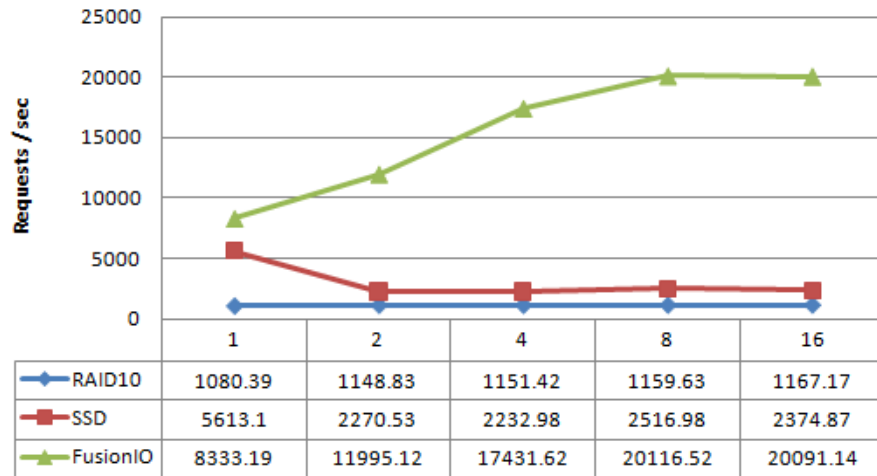
random read, 16K, 95% response time, ms



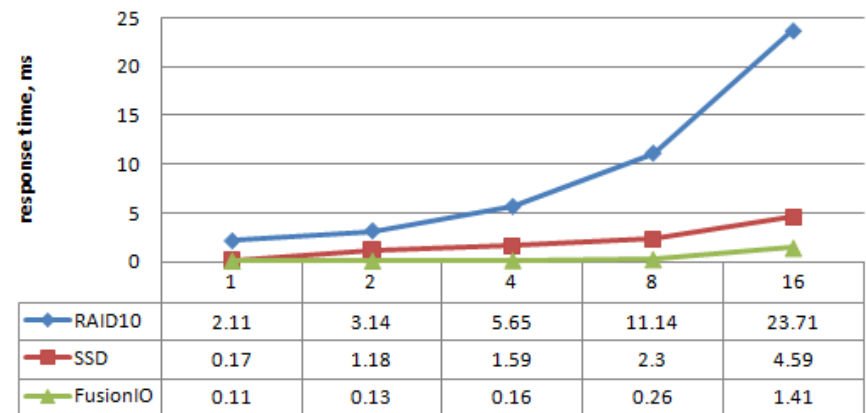
Random write

1 thread – 5.6K IOS, 0.17ms 8 threads – 2.5K IOS, 2.3ms

random write, 16K, throughput

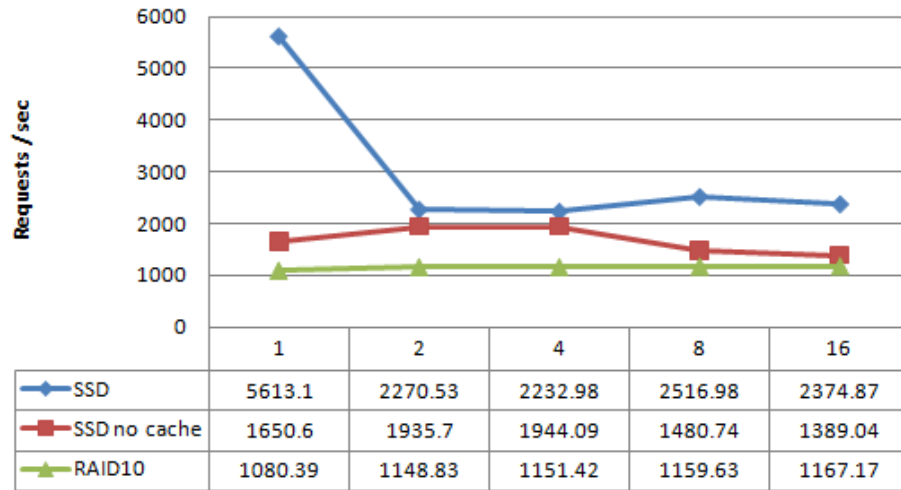


random write, 16K, 95% response time, ms

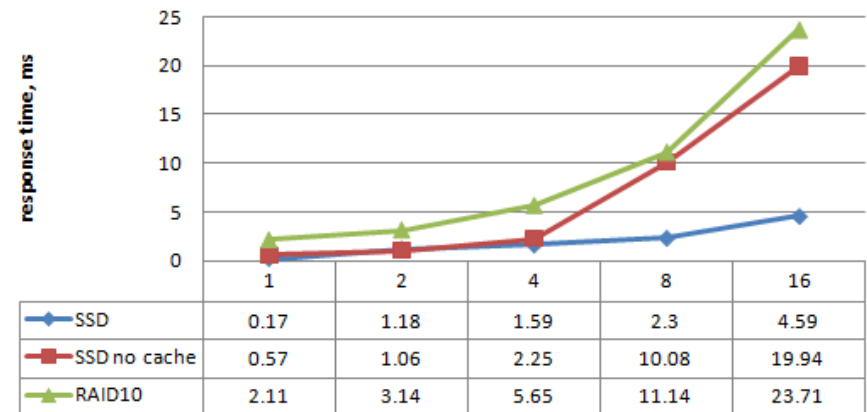


Write cache

random write, 16K, throughput



random write, 16K, 95% response time, ms



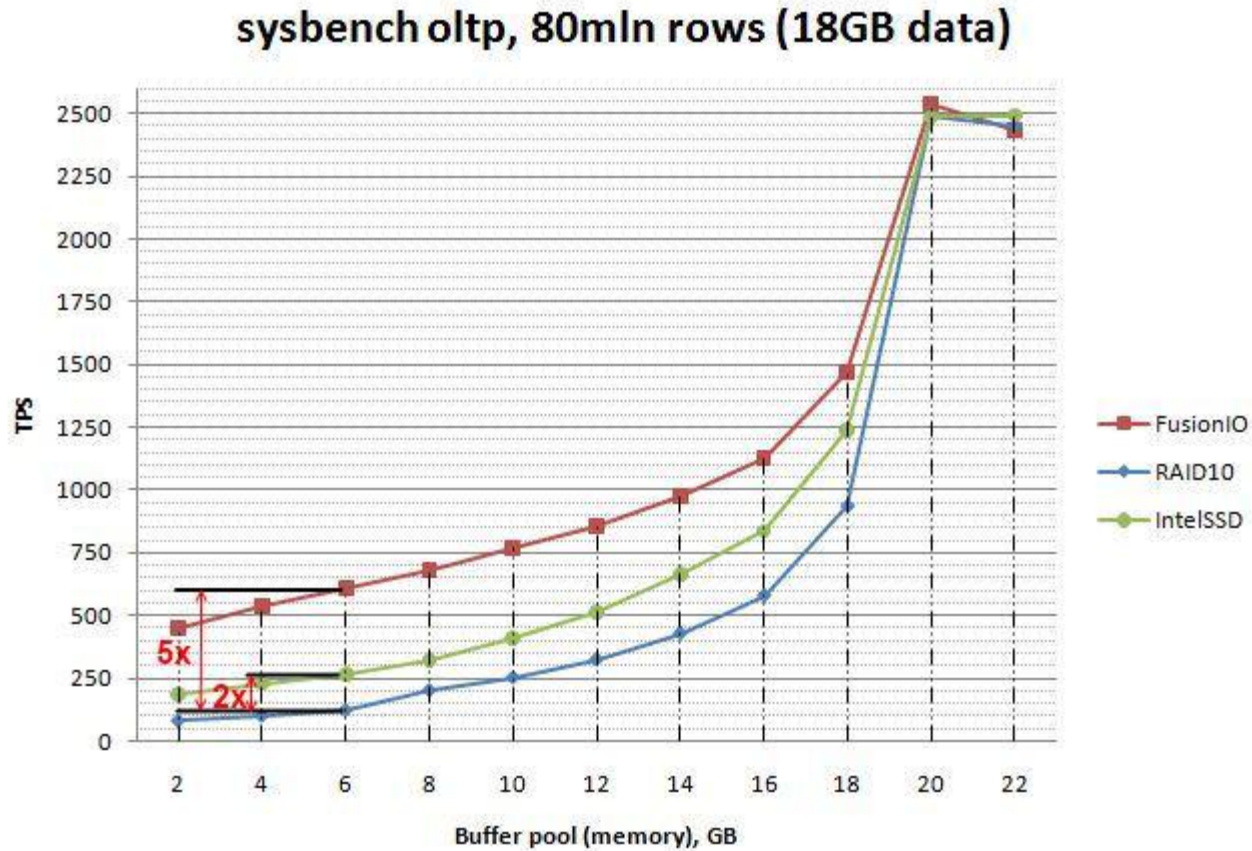
X25 deployment

- Couple cards are giving problem
- RAID
 - Software / hardware ?
 - Hardware throughput is limited to 4 cards
 - Level 0? 1 ? 10? 5? 50 ?
- Engineering process could be complex and expensive
 - Ready solutions: Schooner, Gear6, Cisco servers

MySQL specific

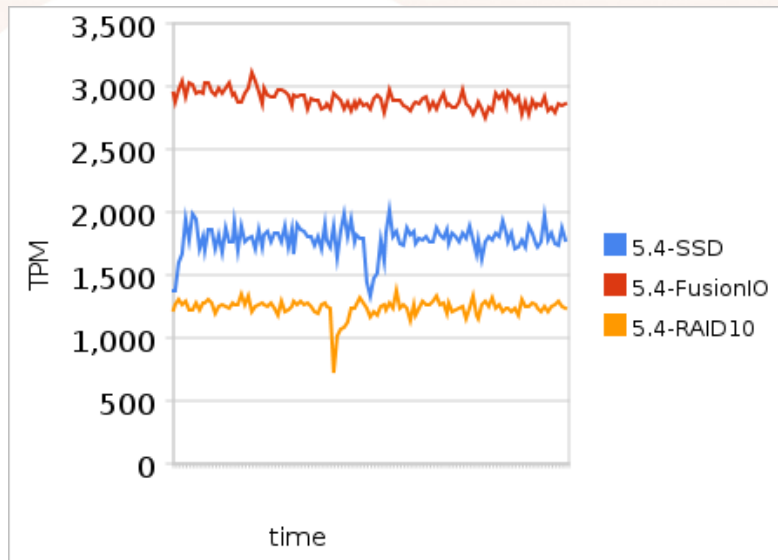
- SSD is very good at random reads, good at random writes, not so good at sequential writes, compared to HDD
 - <http://yoshinorimatsunobu.blogspot.com/2009/05/tables-on-ssd-redobinlogsystem.html>
- Data files – SSD
 - Table files (*.ibd)
 - UNDO segments (ibdata)
- Log files – RAID with BBU
 - REDO log files (ib_logfile*)
 - Binary log files (binlog.XXXXXX)
 - Doublewrite buffer (ibdata)
 - Insert buffer (ibdata)
 - Slow query logs, error logs, general query logs, etc
- [SSD Deployment Strategies for MySQL](#) , [2:00pm Thursday, 04/15/2010](#)
 - By [Yoshinori Matsunobu](#) (Sun Microsystems)

Performance overview



Tpcc-like benchmarks

- RAID10 – **7439.850 TPM** / 4.8 TPM / \$
SSD – **10681.050 TPM** / 27 TPM / \$
FusionIO – **17372.250 TPM** / 3.6 TPM / \$

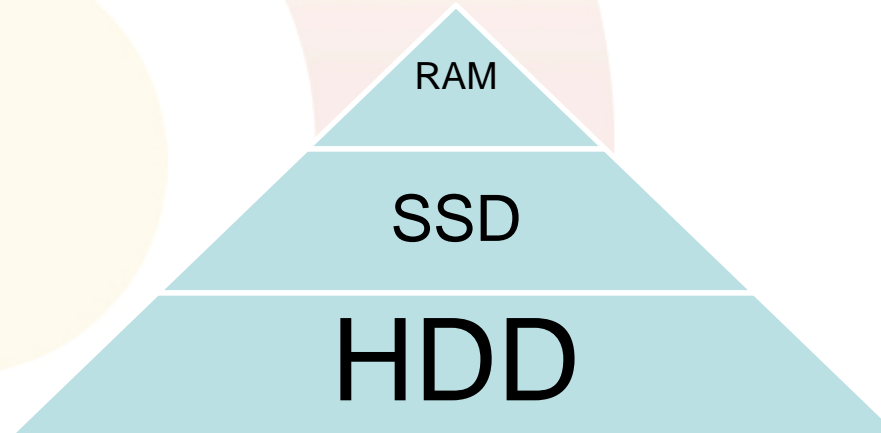


Others factors

- Consolidation factor
 - Replace 2x-10x servers by one
- Power consumption

Application directions

- Mutli-threaded IO
- Sequential / random separation
- Hierarchical (L2) cache
 - Already available in ZFS / Veritas
 - [http://blogs.sun.com/brendan/entry/test L2ARC](http://blogs.sun.com/brendan/entry/test_L2ARC)



Technologies to look

- FusionIO
- Seagate / LSI PCI card (end 2010 ?)
- Couple more PCI-E based
- Intel / Samsung SSD
- Schooner
 - MySQL appliance with performance customization for SSD
- Violin Memory
 - Flash as RAM

Thank you!

- Questions?
- vadim@percona.com
- morgan@percona.com