ORACLE®

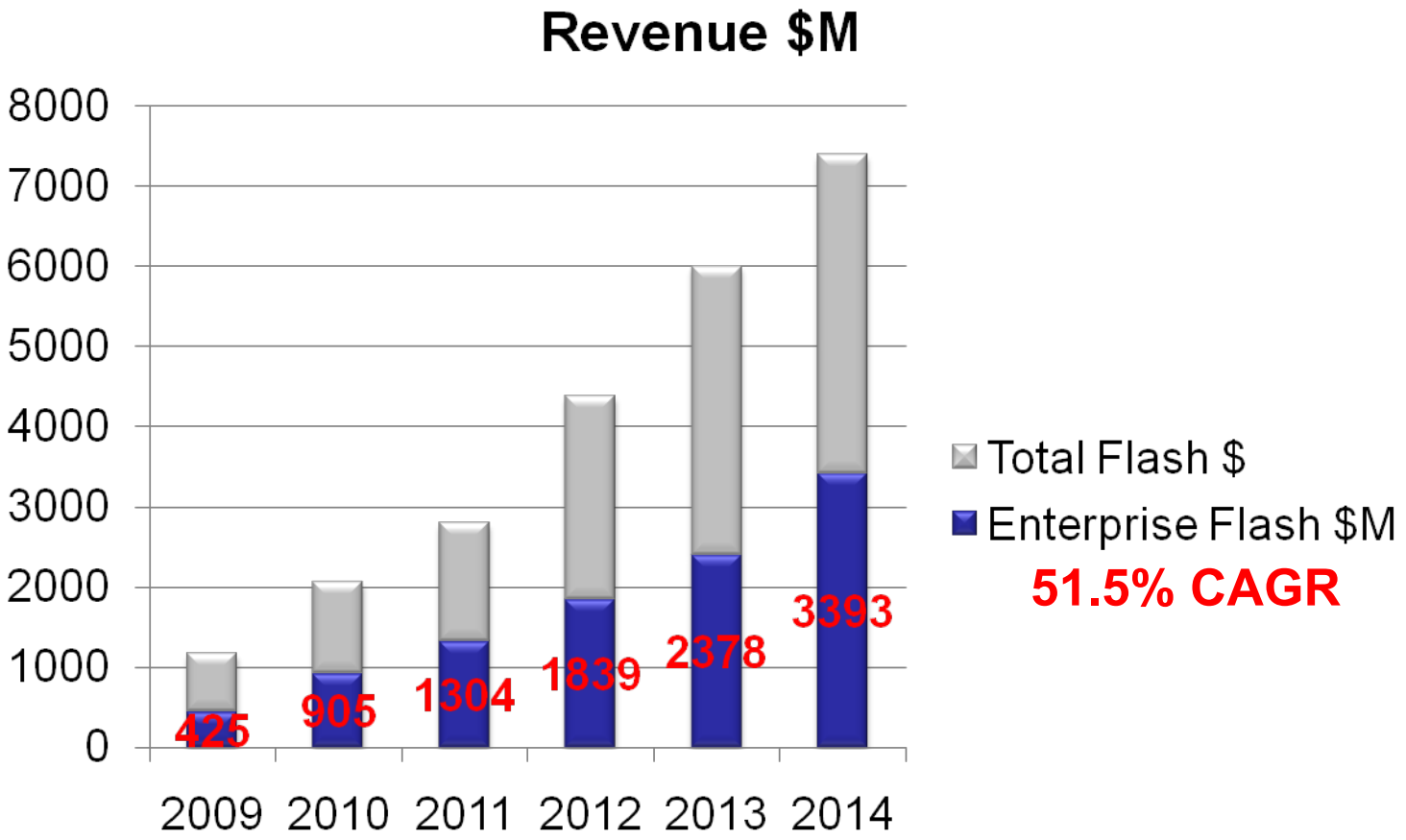
Mythbusting Flash Performance

Bill Nesheim
VP of Solaris Platform Engineering

Agenda

- Why Oracle Cares About Flash
- Oracle Flash Integration
- Flash Mythbusting
- What we need from you

We Focus On Enterprise



Source: IDC, Dec 2010

Flash Characteristics Our Customers Demand

Uptime/Servicability

Performance

Reliability

Data Integrity

Visibility

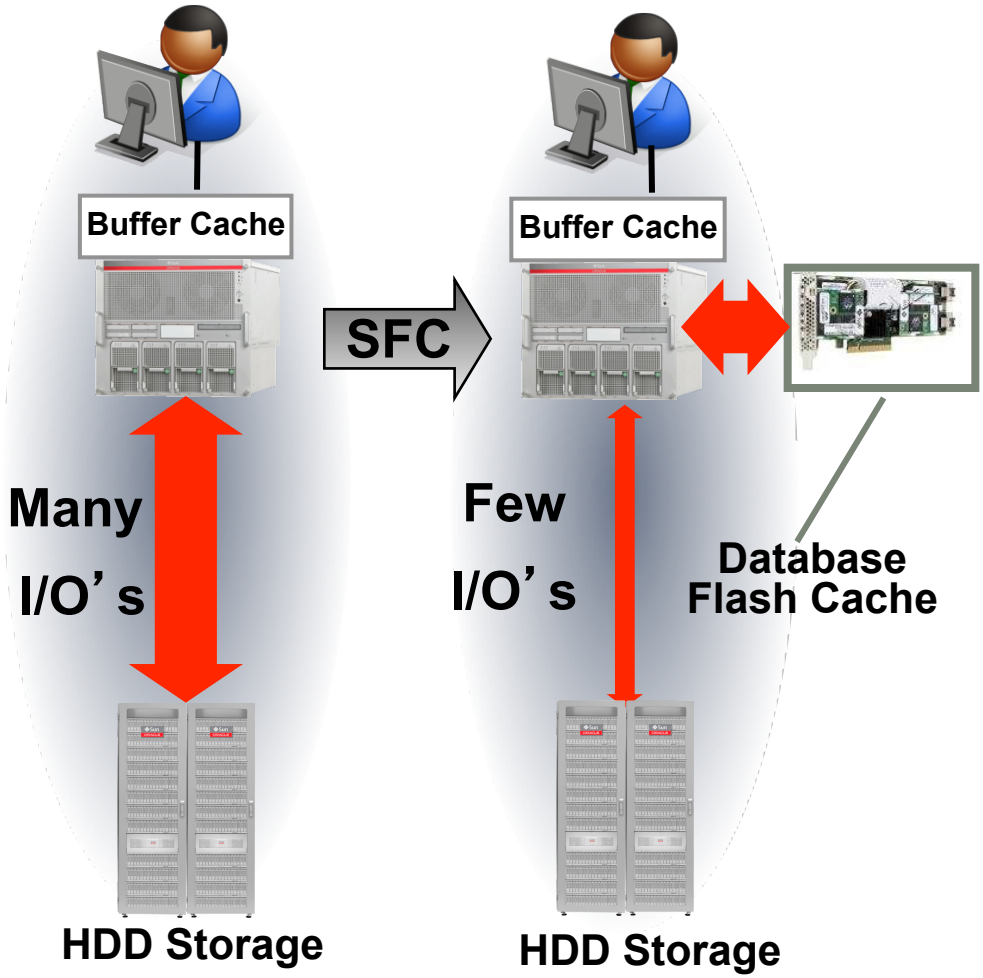
Flash at Oracle

- Flash increases the performance of our storage products
  - ZFS filer uses flash for the file system intent log and L2 ARC read cache
- OLTP and Database run faster with Flash
  - SGA Extension (DB Smart Flash Cache), table space acceleration, indexes, logs
- Customers looking for SSDs (easier to service, hot swappable), PCIe cards, and flash arrays
  - Oracle supports all of these form factors
- Flash is used effectively in all Oracle Engineered Systems
  - Exadata, Exalogic and the ZFS Storage Appliance all have designed-in flash

Oracle Optimized Solution for Oracle Database

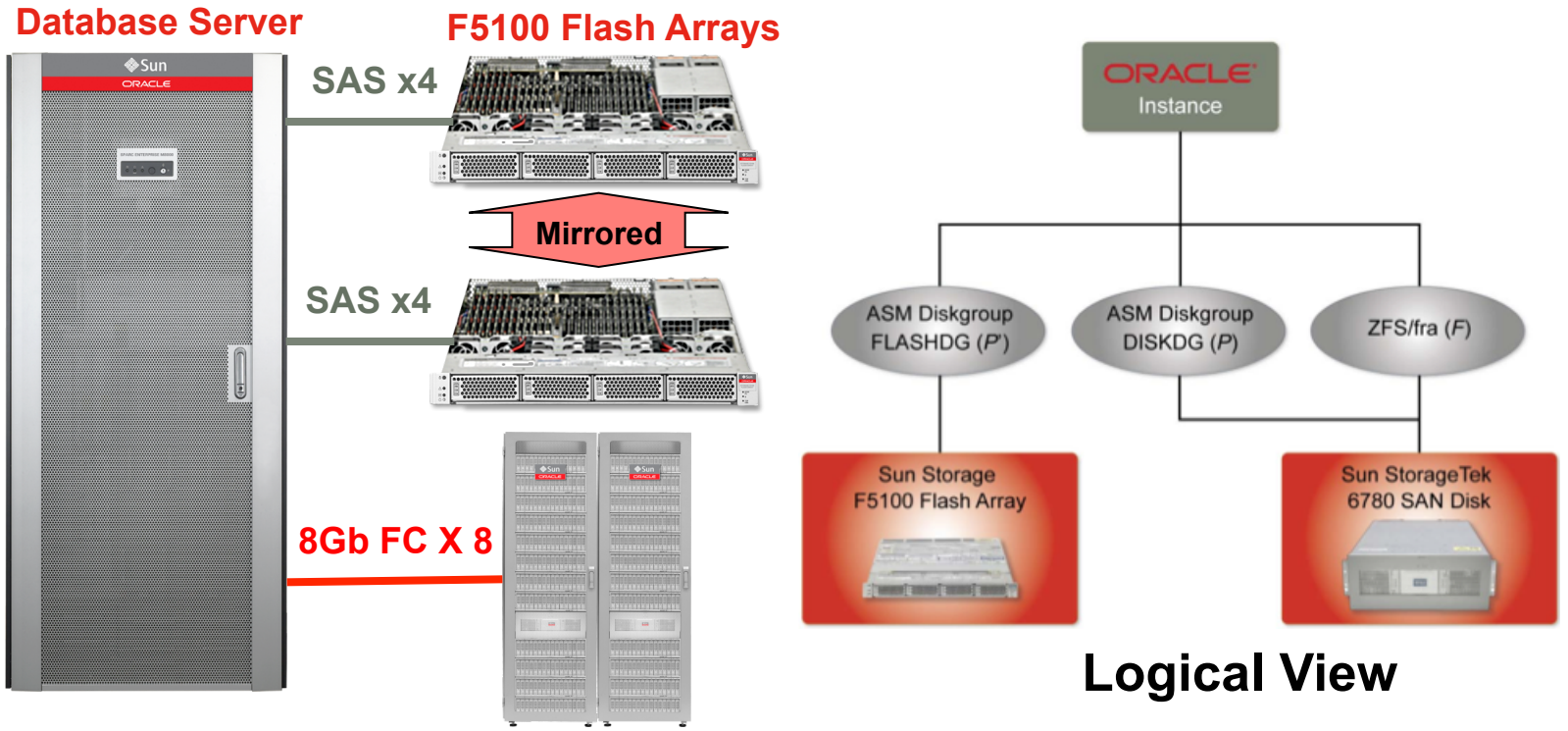
Balancing I/O with 11gR2 Database Smart Flash Cache

- Acts as extension of database buffer cache
  - 'Clean Cache'
- Reduces physical buffer read I/Os
  - Converts to logical I/O in database
- Principally accelerates read intensive workloads
  - But writes improved by HDD offload.



Oracle Optimized Solution for Oracle Database

Balancing I/O with Flash Disk Group Configuration



Logical View

- ASM Normal Redundancy (Flash Module Extents Mirrored)
- Failure groups across SAS domains
  - Across chassis (shown) even better

Online Payment, Marketing and Sales Company

Before

RAC Flash Cache AWR Report: Flash Cache Off

Top 5 Timed Foreground Events

Event	Waits	Time (s)	Average Wait (ms)	%Total Call Time	Wait Class
db file sequential read	3,189,229	34,272	11	67.8	User I/O
CPU time		11,332		22.4	
log file sync	2,247,374	4,612	2	9.1	Commit
gc cr grant 2-way	1,365,247	793	1	1.6	Cluster
enq: TX – index contention	140,257	720	5	3.1	Concurrenc

- 15 minute workload snapshots ‘under load’
  - but 9.5 hours of storage I/O read waits!
- 13 minutes of commit time
  - Fraction of this is redo log write

Online Payment, Marketing and Sales Company

After

RAC Flash Cache AWR Report: Flash Cache On

Top 5 Timed Foreground Events

Event	Waits	Time (s)	Average Wait (ms)	%Total Call Time	Wait Class
CPU time		11,353		57.6	
log file sync	1,434,247	6,587	3	33.4	Commit
flash cache single block read	4,221,599	2,284	1	21.3	User I/O
Buffer busy waits	723,807	1,502	329	3.3	Concurrenc
db file sequential read	22,727	182	8	67.8	User I/O

- Average Flash Cache Read time 540 us using ASM flash cache file
  - Was 11 ms without flash cache
  - Total read wait time now 38 minutes vs. 9.5 hours before.
- Much higher throughput, much shorter DB/application transaction times.

中国石油

Chen Jian
IT Director, PetroChina
Changqing Oil Field Company

"The deployment of Oracle's SPARC Enterprise M5000 server and Sun Storage F5100 flash array enabled us to improve database and system response times. Our users can now receive timely responses to their ad hoc queries and generate oil production and distribution reports much faster."

PetroChina

Runs 60x Faster With Flash Storage

"Response time accelerated from 6 minutes to 4 seconds"

Oracle Sun 5100
Enterprise Flash Array

ORACLE

Great! But… it is often easier said than done!

- Customer trials often end with this question:
  - *Why is my workload slower on flash?*
- Obtaining optimal (or even good) performance from flash is complex.
- Available metrics and device benchmarks often not predictive of application benefits.
- Oracle addresses these challenges in *Engineered Systems*
  - Test, select, and optimize device performance
  - Tuning application and system software
  - Selecting appropriate DB, OS, and FW/HW versions
  - Profiling, testing and delivering complete database and target application workloads.

Application and System Integration is key to successful flash deployment

Oracle Exadata Database Machine

Database Grid

- Compute Nodes

InfiniBand Network

- Redundant 36 port 40Gb/s switches

Intelligent Storage Grid

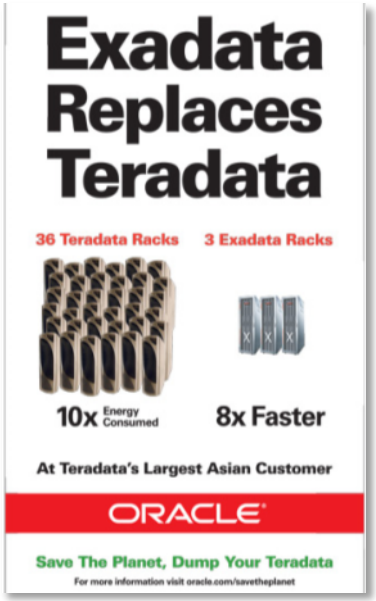
- 14 High-performance storage servers

- 100 TB **High Speed** disk, or 336 TB **High Capacity** disk
- **5.3 TB PCI Flash**
- Persistent Data mirrored across storage servers

- 5.3 TB Flash on 56 Flash Cards to avoid disk controller bottlenecks
- Automatically caches frequently-accessed 'hot' data in flash storage
- Balanced system (compute/network/storage) avoids IO waits
- Exadata Flash Cache achieves: **Over 1 million IO/sec from SQL (8K), Sub-millisecond response times with 50 GB/sec query throughput**

ORACLE

Engineered Systems Deliver *Spectacular* Results



"Softbank created a warehouse up to 8x faster while reducing costs 50%"
— *Keiichiro Shimizu, Softbank*

"Turkcell's largest 250 TB DB is now only 27 TB with Exadata Compression"
— *Ferhat Sengonul, Turkcell*

"Performance improved 17x with no changes to our application"
— *Jim Duffy, BNP Paribas*

Myths verses Reality

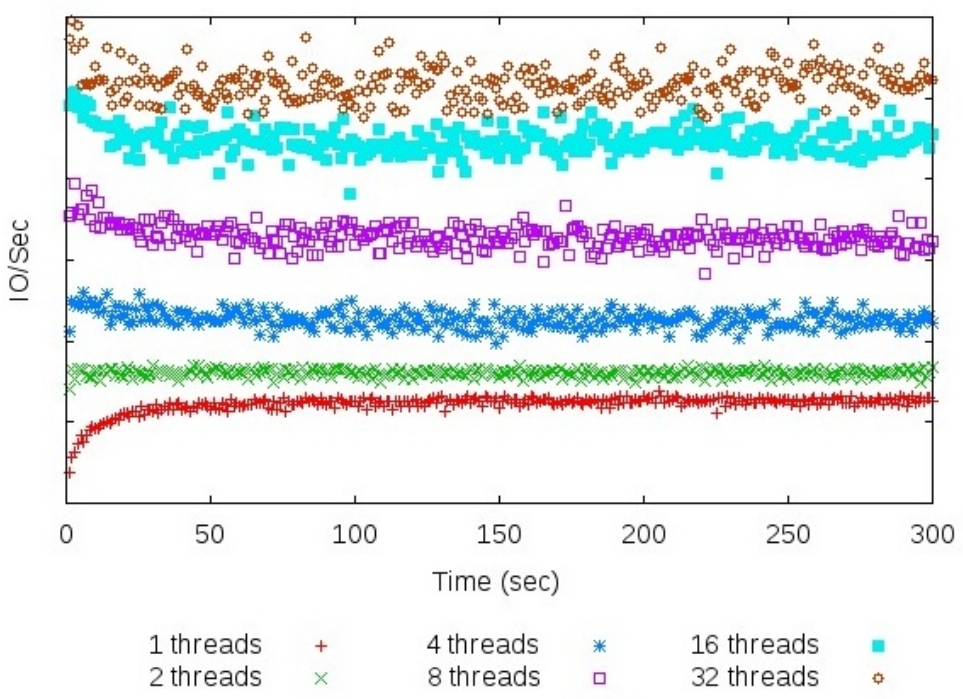
- Flash Claims to Fame:
  - High Performance
  - Small Footprint
  - Low Energy Consumption
  - More reliable than rotating magnetic media
- In practice:
  - *Substantial* performance variability
    - some cases can be even worse than disk
  - Density “opportunity” leads to power & thermal challenges
    - per-slot power limitations
    - Challenging environment for supercaps and batteries
  - Reliability realities:
    - Unpowered data retention constrains use as archival media
    - Varying and unique failure modes

Flash Performance Myths

- Myth : It's all about peak IOPs!
- Reality:
  - Applications primarily benefit from faster response times
  - Limited application threading results in lower IOP demand
  - Vendors quote IOPs for high queue depths (32) and theoretical workloads rarely seen in enterprise environments
  - Transactional application performance is dominated by turnaround
  - Performance outliers can have significant adverse impact
- What's Needed:
  - Predictable scaling & performance over time
  - Less asymmetry between reads/writes, random/sequential
  - Predictable response times

Device A

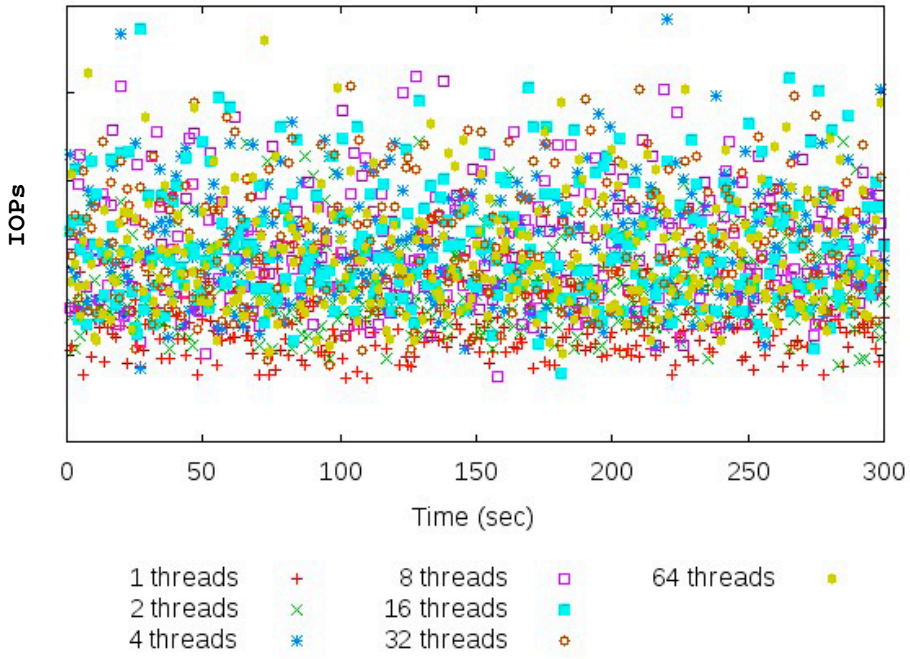
Predictable Performance vs Time And Predictable Scalability



- IO rate (generally) constant over time
- Performance increases with increasing thread count
- Typical behavior for random reads (scaling will vary)
- "Good" drives have random write and mixed r/w graphs that exhibit these characteristics (constant IO rate, performance increases with thread count)

Device B

UNpredictable Performance vs Time And UNpredictable Scalability

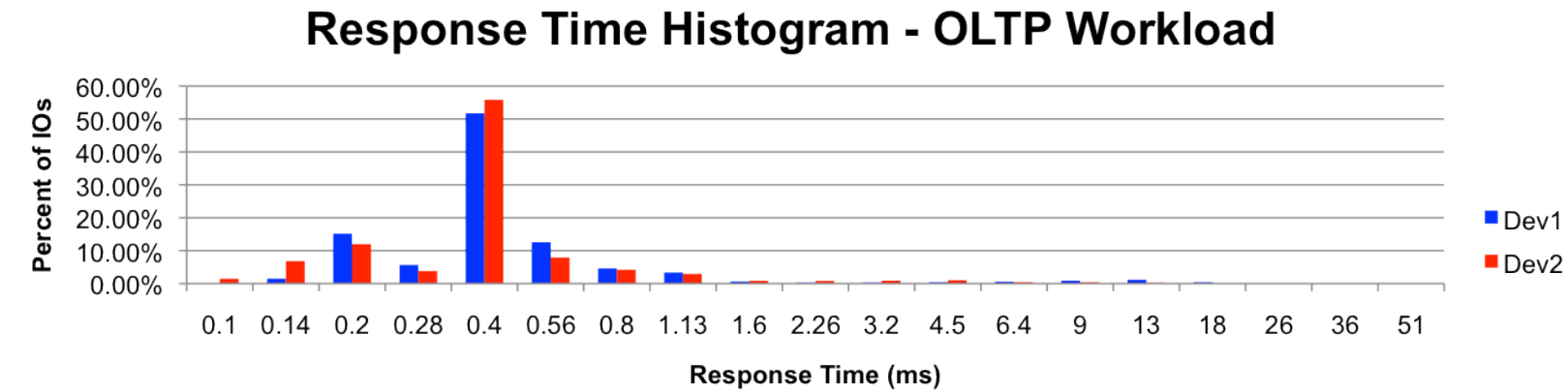


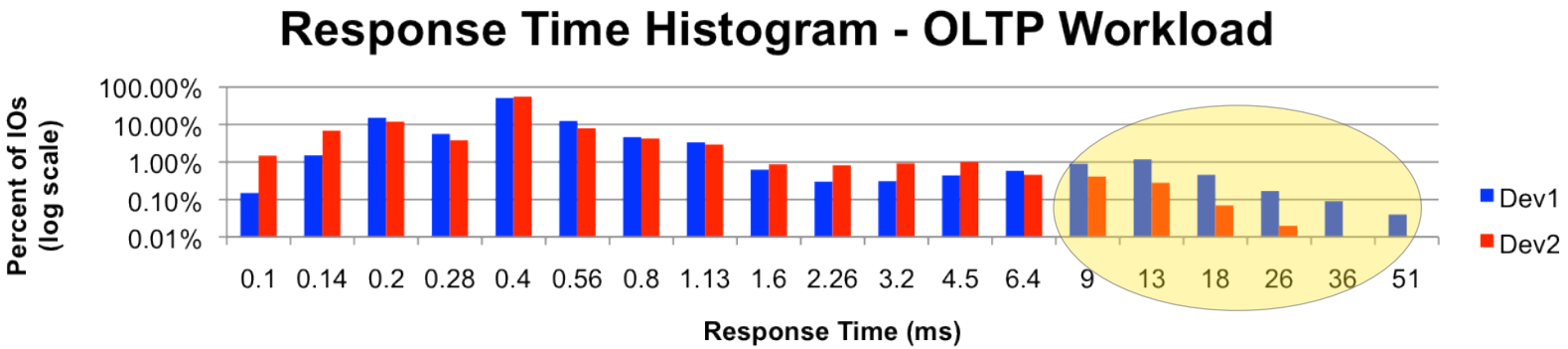
- IO rate not constant over time
- Average performance increases with thread count, but instantaneous performance is unpredictable
- We have seen similar performance on write and mixed workloads from multiple vendors, both SLC and MLC

Looking Deeper: Why Average isn't Good Enough

- Flash latencies generally quoted as averages
  - An average latency of < 1ms is irrelevant when the application is waiting for the 100+ms outlier to complete
  - Not to mention those annoying edge cases…
- Example: What happens to an OLTP workload when an IO is slow to return?
  - App keeps issuing IO, so IOs pile up in the device queues
  - By the time the "slow" IO returns, queue depths can easily reach depths of 25, 50, or more
  - After a "slow" IO returns, device has to work through its large queue of accumulated IO (higher queue depths mean higher latencies); recovery period can be substantial
  - DB workloads typically striped across many flash devs; some of them are likely to be in a "slow IO" or "recovery" state, and dragging down overall OLTP performance

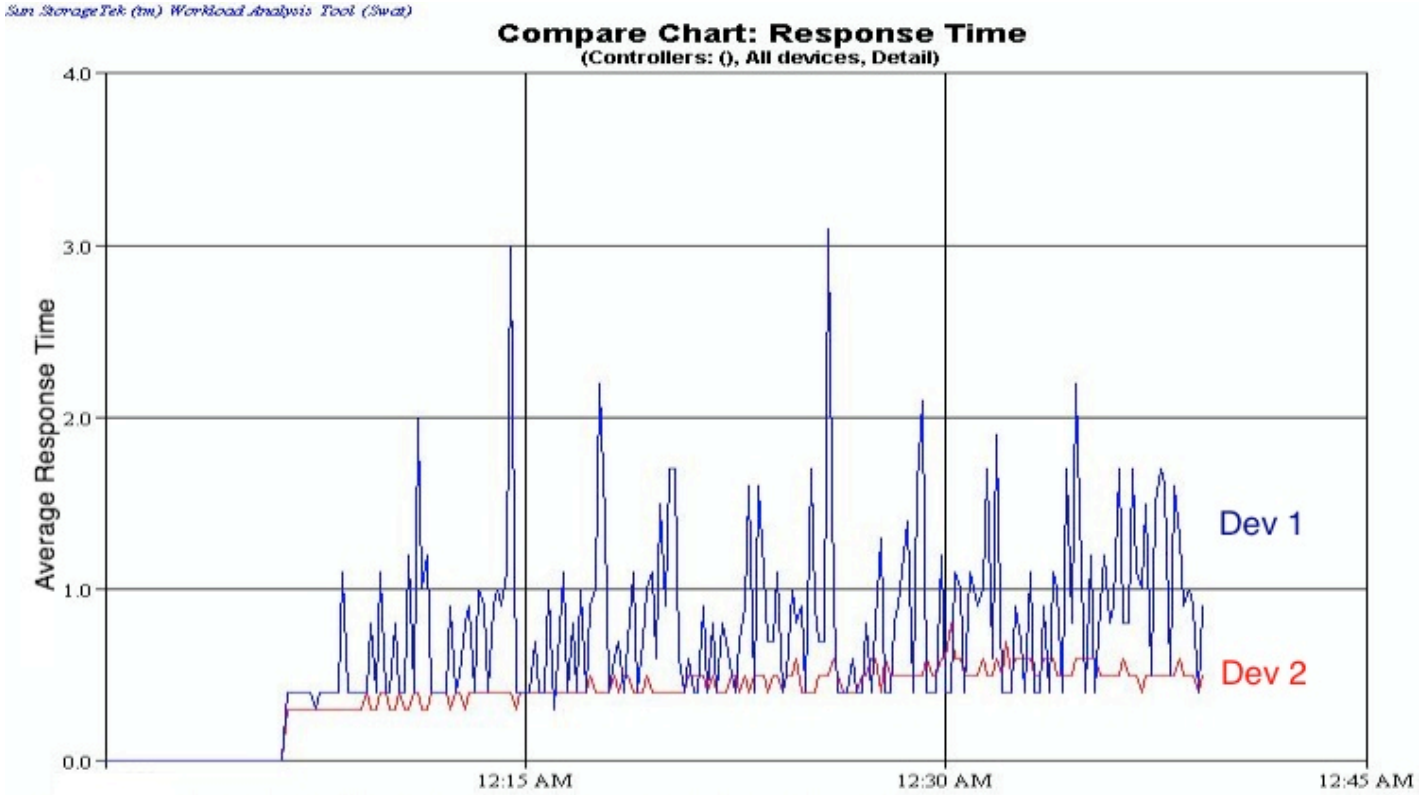
Latency Distribution: The Devil in the Details





- Response time histogram for two similarly performing devices
  - neither is in production, shipping, or used in any ORCL benchmarks
- Difference in device performance determined primarily by the slowest 1% of IOs
- Most vendors tend to disregard the < 1% data as irrelevant

Latency Distribution: The Devil in the Details



- 1 second average response times for single devices under an OLTP workload show stark differences between the two devices
- When striped over many devices, negative effects are multiplicative
- Dev2 provides 2.5x more sustained IOPs at a given Response time
  - Much less flash required to achieve target OLTP rates!

(un) reliability, (im) maturity

- 24x7 uptime: most enterprise systems never power off for > 3 years, need > 2M hours MTBF
- System availability impacted by service actions: PCIe flash cards requires cold service, supercapacitors or batteries require frequent service after prolonged thermal exposure
- Immature firmware presents new challenges to storage protocol stacks: hangs, timeouts, offlines…
- $-Per-Endurance increasing over time, MLC more expensive than SLC from a $-Per-Endurance view
- Devices often embed service policy assumptions at odds with enterprise deployment

Standards (or the lack thereof…)

- Vendor-unique PCIe flash interface protocols
  - Extremely challenging for operating system & driver support, high cost of adoption inhibits system vendors
  - Protocol convergence & ecosystem on the horizon?
- Lack of standardization on management interfaces
  - Statistics and error reporting on wear, ECC errors, other internal errors often vendor-specific or completely invisible to the host
  - Performance statistics vendor-unique or invisible to the host
  - Cache control and secure erase implementations inconsistent
- Compliance: Some devices don't
  - Example: Advertised logical and physical sector sizes
  - Flash generally runs at 512B logical size for compatibility with disk and OS's, actual physical sector size typically 4k or larger
  - Devices should report this correctly via READ CAPACITY(16) or ATA IDENTIFY DEVICE

Conclusions

- Flash offers unique benefits when used properly in enterprise systems
- Technology still immature compared to other storage options
  - Performance variability and predictability
  - Reliability and Availability
  - Standardization
- Application Integration is key
  - Flash performance much too variant to be a generic solution for all storage needs
  - Engineered systems allows for designs that exploit the benefits of flash while avoiding the pitfalls

Hardware and Software
Engineered to Work Together