Tuning For Speed: Percona Server and Fusion-io

Torben Mathiasen – Percona Live 2011 London
tmathiasen@fusionio.com
Overview

• XtraDB performance tuning for Fusion-io

• Sharding MySQL instances locally

• Fusion-io Atomic write feature for InnoDB

• ioDrive2/VSL 3 and MySQL expectations
Basic Tuning

• I/O Schedulers
  – Default with Fusion-io is NOOP
  – Merges requests but does not optimize for rotational devices
  – For MySQL, submitting requests directly to the device have shown performance improvements. use_workqueue=0

• File systems
  – EXT4, trim/discard
  – XFS, online discard with latest kernels
  – Write barriers
    • No cache on ioMemory modules. Acknowledged writes are guaranteed to be persistent. Even during power failure, crashes.

• Outstanding I/O
  – Default is 128 reads and 128 writes per queue. Batching can increase this further. Can make a difference with benchmarking. No clear advantage for MySQL

• Distribute Interrupts
  – Distribute interrupts among multiple cores for multi card setups to avoid soft lockups -> irqbalance
MySQL Tuning

• How to test? Which workloads?
  – Settle on a few, concentrate on those and then verify in production. Production data is great, but may not be available
  – We have been using Percona tpcc-mysql and tpce-mysql. Good workloads, but may not match yours

• Direct I/O vs Buffered
  – Use direct-I/O for data. Logs are usually buffered
  – Buffered I/O introduces an extra data copy. Memory for page cache better spent in buffer pool. Less complex write path
  – When does buffered I/O makes sense?

• Block sizes
  – Linux MD on RHEL6 seems to favor 4K sectors and blocks
  – Marginal better efficiency of the Virtual Storage Layer with 4K sectors
  – For direct-io, Linux requires all I/O to be hardware sector aligned
MySQL Tuning Cont.

- **Check pointing** -> heavy writes
  - Adaptive, push I/O constantly. `Innodb_adaptive_checkpoint=keep_average, keep short loop cycle`
  - Buffer Pool vs Dirty pages
    - Large buffer pool can mean a lot of dirty pages. A LOT!

- **Mount points**
  - Sequential vs Random I/O
    - Spindles are ok at sequential platter access. Flash is better
    - A 500GB database is a big database. But a 640GB io-duo can fit it all
    - If possible, put everything on flash. If needed, put logs on hard drives
  - Spindle hosted database logs
    - Make sure that the spindle does not receive other I/O. Multiple database logs may be sequential individually, but still cause the heads to seek
What simple optimization can do
MySQL Tuning Cont.

- `innodb_thread_concurrency=0` (limit if CPU bound)
- `innodb_read_ahead=0`
- `innodb_read_io_threads=8` (16 for writes)
- `innodb_adaptive_checkpoint=keep_average`
- `innodb_flush_method=O_DIRECT`
- `innodb_io_capacity=10000`
- `Innodb_flush_neighbor_pages=0`
Atomic write or Double Write buffer

- Why is the double write buffer needed?
- Introducing the Fusion-io Atomic Write feature
  - Moves atomicity into the storage device
  - Multiple page writes will either occur in its entirety or not all
  - Less writes, less complexity, better scalability
  - Improves performance considerably compared to using the double write buffer
Atomic Write Performance

WRITES REDUCED BY

2x

LATENCY REDUCED BY

26%

95%

LATENCY REDUCED BY

2.5x

Data Written (GB) | Average Latency (ms) | 95% percentile Latency (ms)
--- | --- | ---
Double-write | 24.3 | 12.15 | 2.73 | 2.02 | 18.24 | 7.21
Atomic-write | 12.15 | 2.73 | 2.02 | 18.24 | 7.21

Double-write
Atomic-write
MySQL Multi-instance

• Improving performance with local shards
  – Split a 200GB DB/64GB Buffer pool setup into 4 instances
  – Get 2.4x the performance on the same system

• Multiple independent MySQL instances can better utilize the fast storage device

• Less software locking, better CPU utilization

• Gets more important as core count increases
Multi-instance - TPC-C

Testing by Percona with Fusion-io:

The ioDrive2 and MySQL

- Virtual Storage Layer 3 improvements
  - Adaptive Flashback
  - Low queue depth performance improvements
  - Synchronous I/O enhancements
  - Better performance on large boxes
  - Improved I/O submission

- Performance consistency

- Low-queue depth performance that matters

- 16K synchronous read/write performance improvements for MySQL
ioDdrive2 and MySQL cont.

**16k synchronous write**
Sysbench - 16 threads

**16k synchronous writes**
Sysbench - 4 threads
Questions? - Thank You!

Torben Mathiasen – Percona Live 2011 London
tmathiasen@fusionio.com