



Best Practices for MySQL Scalability

Peter Zaitsev,
CEO, Percona
Percona Technical Webinars
May 1, 2013

About the Presentation

- Look into what is MySQL Scalability
- Identify Areas which impact MySQL Scalability
- Provide brief overview of best practices in those areas
 - Conceptual Introduction

What is Scalability ?

- Very Multidimensional Term
 - Scaling number of users ?
 - Database size ?
 - Hardware size ?
 - Scaling up and down ?

Our Definition

- A very practical one
 - Scaling with system growth
 - Typically means
 - More data
 - More users
 - Higher concurrency
 - All at the same time

In Search of Scalability

- Architecture
- Hardware
- Operating System and Configuration
- MySQL Version
- MySQL Configuration
- Schema Design and Indexing
- Query Design

Architecture

- MySQL is not the only answer
 - Memcache, Sphinx, Hadoop, MongoDB etc
- The best way to optimize something is stop doing it
 - Eliminating Queries you do not need
 - Caching
- Latency Matters
 - Limit number of queries
 - Run queries in parallel
 - Know network topology

Architecture

- Any single node is limited
 - Ask the question whenever you can grow beyond single node early on
- Control MySQL Concurrency
 - None of MySQL versions will be happy with 50.000 of active transactions

Multi Node Architectures

- MySQL Replication
- MySQL Cluster
- Percona XtraDB Cluster
 - And other Galera based solutions
- Clustrix
- Sharding PODs etc
 - Parelastic
 - Shard-Query
 - Scalebase

Hardware

- Hardware can go a long way !
- Over 200K simple queries/sec on modern hardware.
 - With 100 queries/page = 120K page views/minute
 - Some 80M page views/day (considering daily spike)

Hardware

- CPU
 - Go for fast cores. All cores are rarely used these days
 - Cache and fast memory bus is important
- Memory
 - Often most important for performance
 - Your working set must fit in memory well.
 - Less memory = more pressure on IO

Storage

- Directly attached Flash is best for performance
 - PCI-E cards (FusionIO, Virident, Micron etc) are fastest
- SATA SSD are very cost effective
- Mind vendors and Monitor for uniform performance
- Ensure you have RAID w BBU on conventional drives
- RAID10 best for heavy load

Wait... What about SAN ?

- SAN/NAS is not for raw performance
- Flexibility of Sizing and Sharing Resources
- Additional Features
 - Storage Replication
 - Backups and CDP
 - Snapshots

Network

- Latency is king
- Minimize number of hops between Database and Web Server
- Ensure running 1Gbit link speed at least
- Monitor for packet loss and latency
- Network problems are often blamed on database

Virtualization and Cloud

- Virtualization has cost
- Cloud rarely provides highest performance hardware
- There is a lot of MySQL ran in the cloud and Virtualized Environments
- Know what is available to you

Operating System

- Linux is by far the most common choice
- Recent Kernel which supports your hardware well
- Server Side Linux distribution
 - RHEL, CentOS, Ubuntu, Debian

OS Configuration

- LVM if you can
 - Little performance overhead good flexibility
- SELinux disabled unless you really need it.
- XFS or EXT4 file systems
- `vm.swappiness=0`

MySQL Versions

- Newer versions scale better
- MySQL 5.6 is most scalable
 - Percona Server brings even more improvements 😊
- Start Development with MySQL 5.6 plan upgrade some 6 months after release
- Use official builds when possible
 - Bad builds frequent cause of problems

MySQL Configuration

- You need to tune MySQL configuration
 - Even newer MySQL 5.6 defaults are unlikely to be optimal
- Check out presentation which goes in depth into configuration tuning
 - <http://bit.ly/14p0q6O>
- Typically getting 5 variables right responsible for 90% performance gain

Top Variables

- **Innodb_buffer_pool_size**
 - Set 80% of memory sometimes more
- **Innodb_flush_method=O_DIRECT**
- **Innodb_log_file_size**
 - Set 256MB or more
 - Larger logs = longer recovery time

Schema Design

- Beware of “Textbook Schema”
- Think how your scheme will support your performance critical operations
- De-normalization, Summary tables are often needed
- Your Online Processing Schema might be different from reporting schema

Indexing

- Too many and too little indexes are both bad
- Check for duplicate indexes
 - **pt-duplicate-key-checker**
- Check for unused indexes
 - <http://bit.ly/MI0ILr>
- Check Indexing Best Practice presentation
 - <http://bit.ly/11Oid3e>

Queries

- Comes together with Schema and Index design, really
- Test your queries
 - On the real data
- Think how your queries will scale
- Establish query performance monitoring practices
 - **pt-query-digest**

Queries

- Learn to use EXPLAIN
 - <http://bit.ly/Zg61XS>
- Know MySQL Optimizer features and Limitations
 - Design queries appropriately
- Learn ways to control MySQL Optimizer
- **FORCE INDEX, STRAIGHT_JOIN** etc

Learn More

- Percona Training
 - <http://www.percona.com/training>
 - Now includes MySQL 5.6 !
- Percona Webinars
 - <http://www.percona.com/webinars>
 - May 15th – Performance Schema
 - May 31st – Query Optimization

Thank You!

Peter Zaitsev
pz@percona.com