What’s New in MySQL and MongoDB Ecosystem

Year 2017

Peter Zaitsev
CEO
Percona University, Budapest
May 11th, 2017
In This Presentation

- Few Words about Percona
- Few Words about Percona University Program
- Most Interesting Developments in MySQL and MongoDB ecosystems
Thank you!

Graphisoft Park

Linux Akademia

Hatékonyság

Önállóság

Szakértelmem

Egyszerűség
Thank You for Feeding us!

Arco Services

PartyCo
Few Words about Percona
Percona’s Purpose

To Champion Unbiased Open Source Database Solutions
We Do

Support, Managed Services for MySQL and MongoDB

Also Consulting and Training

Helping companies to migrate to Open Source Database

Develop Open Source Software

Solutions to maximize your success
Broad Software Ecosystem Support

MySQL
Percona Server
MariaDB
Percona XtraDB Cluster
Galera Cluster for MySQL
MariaDB Galera Cluster
MongoDB
Percona Server for MongoDB
Amazon RDS for MySQL/MariaDB/Aurora
Google CloudSQL
100% Free and Open Source Software

- Percona Server for MySQL
- Percona Server for MongoDB
- Percona XtraDB Cluster
- Percona Xtrabackup
- Percona Toolkit
- Percona Monitoring and Management
Percona University

- Educational Technical Presentations
- Multiple Locations in the World
- Partnering with Local Companies
- Affordable to Attend (Free)
Percona University – What to Expect

- Several presentations on different topics
- Feel free to only attend those you’re interested in
- Keep it Interactive! Ask Questions
- Breaks
- Prize Give away in the end
Whats new in MySQL and MongoDB

Top Highlights
## Innovations worth Noticing

<table>
<thead>
<tr>
<th>MySQL 5.7</th>
<th>MySQL 8</th>
<th>MariaDB 10.1</th>
<th>Amazon Aurora</th>
</tr>
</thead>
<tbody>
<tr>
<td>Percona XtraDB Cluster 5.7</td>
<td>Percona Monitoring and Management</td>
<td>MyRocks</td>
<td>ProxySQL</td>
</tr>
<tr>
<td>Orchestrator</td>
<td>Gh-ost</td>
<td>MongoDB 3.4</td>
<td>Percona Server for MongoDB 3.4</td>
</tr>
</tbody>
</table>
MySQL 5.7 – Current GA
MySQL 5.7 - Security

“Secure by Default”

Password validation (no weak passwords by default)

Automatic SSL certificate creation

Simple SSL setup

“root” user created with password by default
MySQL 5.7 Encryption

Can encrypt Innodb tables on disk

Only data is encrypted at this point

Innodb log files, binary log files are not encrypted

MySQL 8 does Innodb log file encryption
Native JSON data type

Can index fields in JSON documents

CRUD access through Protocol X
MySQL 5.7 - Replication

- Parallel Replication
- Multi-Source Replication
- Can enable GTID online
- MySQL Group Replication
- MySQL InnoDB Cluster
MySQL 5.7 – Performance Schema

- Automatic configuration for Performance Schema
- Reduced overhead (especially memory overhead)
- Memory Usage Instrumentation
- Instrumentation of Storage Procedures
- Instrumentation of Transactions
- Sys_schema included for simple Performance Schema access
MySQL 5.7 - Performance

Further improved Multi-core scalability

Optimizations for Innodb Temporary Tables

New Compression for Innodb Tables

Optimizer Improvements
MySQL 5.7 Benchmarking: Sysbench OLTP Read Write

1.5x Faster than MySQL 5.6
2.5x Faster than MySQL 5.5

17,000 TPS

*Information from Oracle OpenWorld presentation by Geir Hoydalsvik*
Complete list of MySQL 5.7 Improvements

http://www.thecompletelistoffeatures.com/
MySQL 8 – Currently in Development
MySQL 8

What was previously referred as MySQL 5.8

Native Data Dictionary (no more .frm files)

Roles

Further Performance Schema Improvements

Common Table Expressions (CTE) and Windowing Functions

Replication Performance Improvements

Invisible Indexes

More to come
MariaDB 10.1 - Current GA
About MariaDB

“fork” by MySQL Founder Michael Widenious,

Replaced MySQL in many Linux Distributions

Has number of interesting features MySQL does not

More and More Becomes separate Database

Not every MySQL feature ported to MariaDB

Differences in Replication, Optimizer, JSON, Protocol X, GIS, Encryption
New in MariaDB 10.1

- Optimistic Parallel Replication
- Online Innodb Defragmentation (originally from Facebook)
- Encryption (originally from Google) with encryption of log files and temporary files
- Galera Replication is now included in MariaDB
- Roles
Amazon Aurora
Amazon Aurora

The fastest Growing Database Technology at AWS

High End of Amazon RDS MySQL

Integration with Amazon Cloud Storage for Improved Performance and Replication

Automatic Replication and Cluster Recovery

Improved Query Cache

Not always faster than MySQL on EC2
Percona Server 5.7

- Full compatibility with MySQL Community Edition
- Many equivalents to MySQL Enterprise features
- Improved InnoDB Performance on heavy load
- New design of DoubleWrite Buffer for better IO performance
- Column Compression (with custom dictionary support)
- Improved TokuDB Storage Engine
Percona XtraDB Cluster 5.7
Percona XtraDB Cluster 5.7

HA Solution based on Percona Server 5.7 and Galera Library

High Availability for MySQL without pains of Async replication

Automated Node Provisioning and Self Healing

Works great in the Cloud and with Containers

Integration with ProxySQL for traffic management

Improved Ease of Use, Security and Performance
PXC 5.7 Performance Improvements


Sysbench: dataset 100tables/4M rows(100GB)
innodb_buffer_pool=150GB,inndodb_doublewrite=1,
innodb_flush_log_at_trx_commit=1,sync_binlog=1
Box: 28 Cores+HT

<table>
<thead>
<tr>
<th>Transactions per second, tps</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
<tr>
<td>OLTP_RW</td>
</tr>
<tr>
<td>UPDATE_KEY</td>
</tr>
<tr>
<td>UPDATE_NO_KEY</td>
</tr>
</tbody>
</table>

Threads

Percona XtraDB Cluster: 5.7.16 5.7.17
MyRocks
MyRocks

MySQL Storage engine based on MyRocks

Used by Facebook for better efficiency and performance

Uses LSM trees as underlying data structure

Write Optimized Engine

Experimental integration available with Percona Server and MariaDB
## RocksDB Efficiency


### Small server: Linkbench, IO-bound

MyRocks: best throughput & QoS, most efficient  
MongoRocks: better than WiredTiger

<table>
<thead>
<tr>
<th></th>
<th>TPS</th>
<th>Iostat r/t</th>
<th>Iostat wKB/t</th>
<th>CPU usecs/t</th>
<th>Size (GB)</th>
<th>p99 update</th>
</tr>
</thead>
<tbody>
<tr>
<td>MongoRocks+zlib</td>
<td>1087</td>
<td>1.07</td>
<td>4.42</td>
<td>24656</td>
<td>23</td>
<td>2</td>
</tr>
<tr>
<td>WiredTiger+zlib</td>
<td>429</td>
<td>1.24</td>
<td>17.98</td>
<td>153763</td>
<td>33</td>
<td>22</td>
</tr>
<tr>
<td>MyRocks+zlib</td>
<td>2246</td>
<td>0.67</td>
<td>1.27</td>
<td>12688</td>
<td>18</td>
<td>1</td>
</tr>
<tr>
<td>InnoDB</td>
<td>1860</td>
<td>0.82</td>
<td>10.62</td>
<td>7991</td>
<td>63</td>
<td>14</td>
</tr>
<tr>
<td>InnoDB+zlib</td>
<td>1855</td>
<td>0.67</td>
<td>8.60</td>
<td>10431</td>
<td>40</td>
<td>8</td>
</tr>
</tbody>
</table>
Percona Monitoring and Management
## Percona Monitoring and Management

<table>
<thead>
<tr>
<th>Feature</th>
</tr>
</thead>
<tbody>
<tr>
<td>100% Free and Open Source</td>
</tr>
<tr>
<td>Comprehensive Database focused Monitoring</td>
</tr>
<tr>
<td>Supports MySQL and MongoDB (and variants)</td>
</tr>
<tr>
<td>Easy to Install and Use</td>
</tr>
<tr>
<td>Version 1.x focuses on Trending and Query Analyses</td>
</tr>
<tr>
<td>Management Features to come</td>
</tr>
</tbody>
</table>

© 2017 Percona
Why did we created PMM

No good database focused monitoring solution

Existing solutions Proprietary or Cloud Only

Do it yourself is possible but hard

Want everyone to be able to run monitoring in every environment

To fix problems before they appear
What Queries are causing the load?

<table>
<thead>
<tr>
<th>#</th>
<th>Query Abstract</th>
<th>ID</th>
<th>Load</th>
<th>Count</th>
<th>Latency</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>TOTAL</td>
<td></td>
<td>7.88</td>
<td>780.82 QPS</td>
<td>2.81 m (100%)</td>
</tr>
<tr>
<td>1</td>
<td>UPDATE sbtest</td>
<td>D30AD7E3079ABCE7</td>
<td>5.31</td>
<td>258.58 QPS</td>
<td>930.90 k (33.12%)</td>
</tr>
<tr>
<td>2</td>
<td>LOCK sbtest</td>
<td>0B759DF6D01BDB8F</td>
<td>1.20</td>
<td>2.54 QPS</td>
<td>9.15 k (0.33%)</td>
</tr>
<tr>
<td>3</td>
<td>SELECT sbtest</td>
<td>558CAEF5F387E929</td>
<td>0.42</td>
<td>170.65 QPS</td>
<td>614.35 k (21.68%)</td>
</tr>
<tr>
<td>4</td>
<td>COMMIT</td>
<td>813031B8B3C3B329</td>
<td>0.23</td>
<td>14.06 QPS</td>
<td>50.62 k (1.80%)</td>
</tr>
<tr>
<td>5</td>
<td>SELECT myisam.sbtest</td>
<td>C4832A98728C4424</td>
<td>0.18</td>
<td>&lt;0.01 QPS</td>
<td>4.00 (0.00%)</td>
</tr>
<tr>
<td>6</td>
<td>SELECT sbtest</td>
<td>87625C47A176EBDD</td>
<td>0.08</td>
<td>188.79 QPS</td>
<td>679.63 k (24.18%)</td>
</tr>
<tr>
<td>7</td>
<td>SELECT sbtest</td>
<td>6433B7802D745420</td>
<td>0.07</td>
<td>17.27 QPS</td>
<td>62.19 k (2.21%)</td>
</tr>
<tr>
<td>8</td>
<td>SELECT sbtest</td>
<td>9CD3EA5A1950648</td>
<td>0.07</td>
<td>16.88 QPS</td>
<td>60.75 k (2.16%)</td>
</tr>
<tr>
<td>9</td>
<td>SELECT sbtest</td>
<td>FE6F0A0683AC9BB4</td>
<td>0.07</td>
<td>17.09 QPS</td>
<td>61.53 k (2.19%)</td>
</tr>
<tr>
<td>10</td>
<td>SELECT sbtest</td>
<td>F54DBEF3D7AE474D</td>
<td>0.06</td>
<td>17.33 QPS</td>
<td>62.39 k (2.22%)</td>
</tr>
</tbody>
</table>
Why are they causing this load?

### UPDATE sbtest

<table>
<thead>
<tr>
<th>Metrics</th>
<th>Rate/Sec</th>
<th>Sum</th>
<th>Per Query Stats</th>
</tr>
</thead>
<tbody>
<tr>
<td>Query Count</td>
<td>259.12 (per sec)</td>
<td>932.85 k 32.40% of total</td>
<td></td>
</tr>
<tr>
<td>Query Time</td>
<td>5.31 load (67.16%)</td>
<td>19101.45 sec 87.16% of total</td>
<td>5.25 ms avg</td>
</tr>
<tr>
<td>Lock Time</td>
<td>1.38 (avg load)</td>
<td>4965.60 sec 51.50% of total 15.13% of query time</td>
<td>793.79 µs avg</td>
</tr>
<tr>
<td>Innodb Row Lock Wait</td>
<td>&lt;0.01 (avg load)</td>
<td>28.14 sec 42.03% of total 1.44% of query time</td>
<td>75.79 µs avg</td>
</tr>
<tr>
<td>Innodb IO Read Wait</td>
<td>&lt;0.01 (avg load)</td>
<td>35.85 sec 2.47% of total 10.93% of query time</td>
<td>573.60 µs avg</td>
</tr>
<tr>
<td>Innodb Read Ops</td>
<td>2.48 (per sec)</td>
<td>8.91 k 1.86% of total</td>
<td>0.00 avg</td>
</tr>
<tr>
<td>Innodb Read Bytes</td>
<td>39.61 KB (per sec)</td>
<td>139.25 MB 1.86% of total 16.00 KB avg io size</td>
<td>3.50 KB avg</td>
</tr>
<tr>
<td>Innodb Distinct Pages</td>
<td>-</td>
<td>-</td>
<td>6.03 avg</td>
</tr>
<tr>
<td>Bytes Sent</td>
<td>13.18 KB (per sec)</td>
<td>46.35 MB 1.72% of total</td>
<td>52.00 Bytes avg</td>
</tr>
<tr>
<td>Rows Examined</td>
<td>258.17 (per sec)</td>
<td>929.43 k 0.64% of total 0.00 per row sent</td>
<td>0.88 avg</td>
</tr>
</tbody>
</table>
How to fix them

CREATE TABLE `sbtest1` (  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,  `k` int(10) unsigned NOT NULL DEFAULT '0',  `c` char(120) NOT NULL DEFAULT '',  `pad` char(60) NOT NULL DEFAULT '',  PRIMARY KEY (`id`),  KEY `k_1` (`k`) ) ENGINE=MyISAM AUTO_INCREMENT=100000001 DEFAULT CHARSET=latin1

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>sbtest1</td>
</tr>
<tr>
<td>Engine</td>
<td>MyISAM</td>
</tr>
<tr>
<td>Version</td>
<td>10</td>
</tr>
<tr>
<td>RowFormat</td>
<td>Fixed</td>
</tr>
<tr>
<td>Rows</td>
<td>100.00 m</td>
</tr>
<tr>
<td>AvgRowLength</td>
<td>189.00 Bytes</td>
</tr>
<tr>
<td>DataLength</td>
<td>17.60 GB</td>
</tr>
<tr>
<td>MaxDataLength</td>
<td>756.00 GB</td>
</tr>
<tr>
<td>IndexLength</td>
<td>1.70 GB</td>
</tr>
</tbody>
</table>
# System Information

## MySQL Summary

```
# Percona Toolkit MySQL Summary Report
System time | 2016-11-06 08:27:41 UTC (local TZ: CET +0100)
# Instances
<table>
<thead>
<tr>
<th>Port</th>
<th>Data Directory</th>
<th>Nice OOM Socket</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# MySQL Executable
Path to executable | /usr/sbin/mysqld
Has symbols | No
# Report On Port 3306
| User | localhost      |
| Time | 2016-11-06 09:27:41 (CEST) |
| Hostname | ps57 |
| Version | 5.7.14-8-log Percona Server (GPL), Release 8, Revision 1f84ccd |
| Built On | Linux x86_64 |
| Started | 2016-10-06 16:33 (up 30+16:54:37) |
| Databases | 8 |
| Datadir | /var/lib/mysql/ |
| Processes | 250 connected, 2 running |
| Replication | Is not a slave, has 1 slaves connected |
| Pidfile | /var/run/mysqld/mysqld.pid (exists) |
```
What happens on OS and Hardware Level
As well as Database Level
In-Depth MongoDB Dashboards
Best to check out Demo

http://pmmdemo.percona.com
ProxySQL
ProxySQL

Traffic Management Solution for MySQL (MySQL Proxy)

100% Free ad Open Source

Multiplexing

Query Routing

Sharding and Read Write Splitting

Query Caching

Integrates with Percona XtraDB Cluster and MySQL Group Replication

http://www.proxysql.com/
Orchestrator

- Visualize MySQL Replication
- Replication Monitoring
- Failover
- Change to Replication Topology
- Now Maintained by GitHub
- https://github.com/github/orchestrator
Orchestrator Screenshot
Gh-ost
GitHub’s Online Schema Migrations for MySQL

Pt-online-schema-change on steroids

Does not use Triggers (less overhead and limitations)

Requires ROW binary log enabled

Tracks table changes through Binary Log

https://github.com/github/gh-ost
MongoDB 3.4
MongoDB 3.4 – New version most popular NoSQL Database

- Collation support for more than 100 languages and locales
- Decimal128
- $graphLookup operator for Graph Processing
- Faceted navigation
- Zones for Geographically Distributed Clusters
- Faster initial sync and better shard balancing
- Views
Percona Server for MongoDB 3.4
Percona Server for MongoDB 3.4

100% Compatible with MongoDB 3.4 Community Edition

Open Source with Alternatives to many MongoDB Enterprise Features

MongoRocks (RocksDB) and Percona Memory Engine

New: Sensitive Data Masking

New: Query Sampling

New: Hot Backup for WiredTiger and MongoRocks
Percona Memory Engine for MongoDB Benchmarks
WiredTiger vs MongoRocks – write intensive
Before we take a Break...
BECOME A PERCONA SUPERHERO

WE’RE HIRING

CONTACT

careers@percona.com
Have a Friend?

Refer a friend and get $1000 if one is hired by Percona

... and eternal gratitude for helping to get the most amazing job
Save the Date!

Percona Live Europe Open Source Database Conference

- September 25-27, 2017
- Radisson Blu Royal Hotel
- Dublin, Ireland

Call for Proposals and Registration
- Opening Soon!