



PERCONA
Performance Consulting Experts

Wonderful world of MySQL Storage Engines

July 24, 2008

OSCON

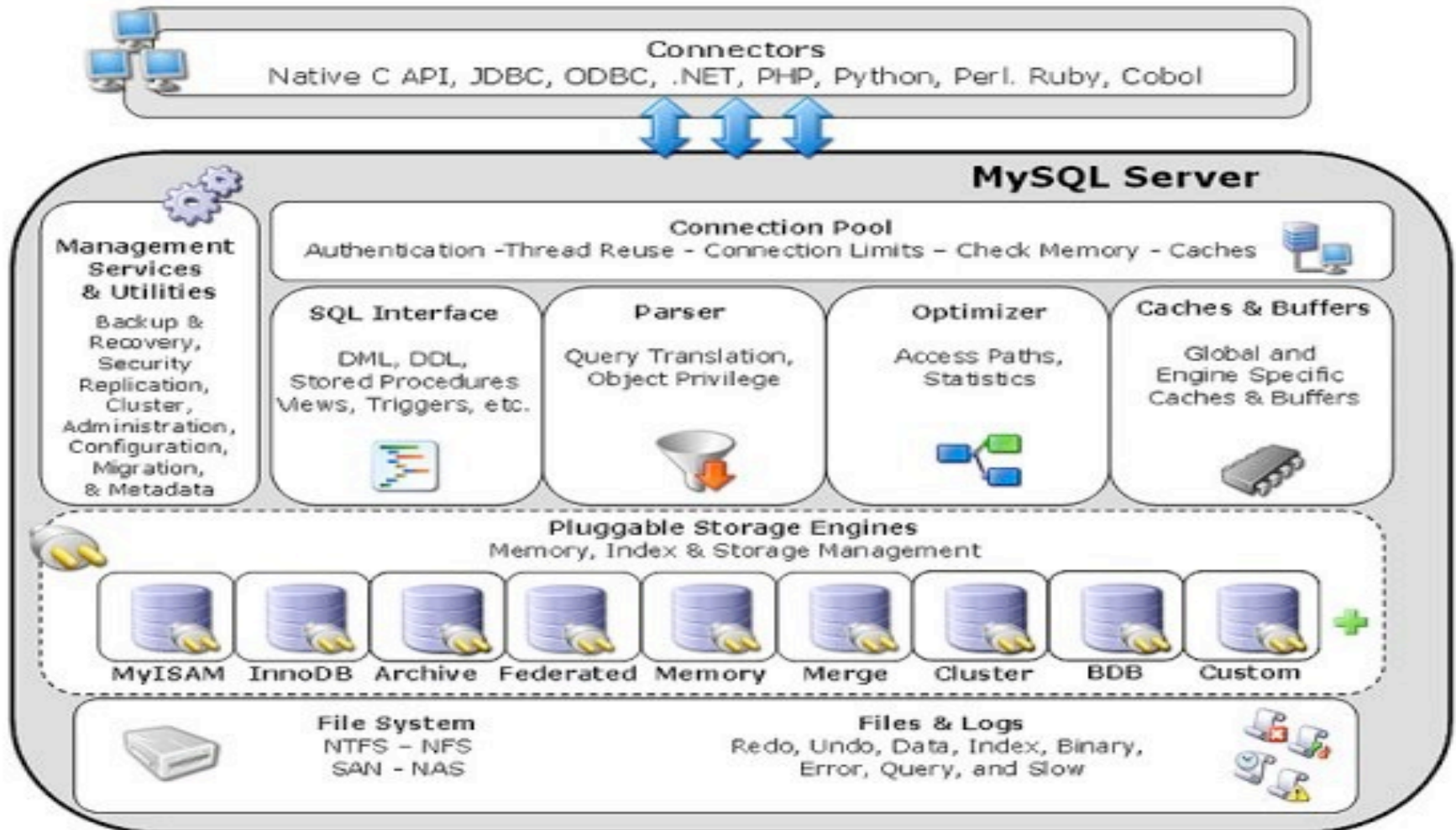
Portland, OR

by Peter Zaitsev, Percona Inc

What are Storage Engines

- MySQL Server was for a while consisting of 2 layers
 - “SQL Layer” - Responsible for all high level stuff
 - “Storage Layer” - Storage, Transactions etc
- MySQL 5.1 changes
 - Storage Engine interface is modular
 - Can compile storage engine separately and load it in the server
 - MySQL Actively cultivated both External and Internal storage engine development
 - So there are many of them
 - Partners come up with many (often close source) engines too.

Aproximate MySQL Architecture



What Storage Engines Are ?

- Storage Engine responsible for **storage**
 - Can implement different storage concepts, file format, remote storage, efficient scans.
- Can't handle upper level functions – sorting, group by, limit clause
 - Future MySQL versions are expected to raise this limits
- Some vendors (like Kickfire) implement hacks to fully intercept query and handle it via their own processing engine

Extending by Storage Engines

- MySQL took unusual choice for extension for DBMS
 - Practically no Pluggable index types, language features
 - Though wide choice of storage engines
- Benefits
 - Different applications may need different storage properties
 - Persistence, transactions, lock granularity, compression etc
- Drawbacks
 - Performance Overhead
 - 2 phase commit on transaction commit, separate logs.
 - Complexity
 - Development and testing (all these interactions)
 - Choosing storage engines
 - Operational challenges – backup, balancing etc

Types of Storage Engines

- **General Purpose Storage Engines**
 - Transactional
 - InnoDB, Falcon, PBXT, Maria (future)
 - Non-Transactional
 - MyISAM, ISAM (dead), Maria(current)
- **Clustered Storage Engines**
 - NDB, ScaleDB (CloseSource)
- **Special Purpose Storage Engines**
 - Memory, Federated, Archive, Blackhole, CSV, NitroDB(CS), InfoBright(CS), Queue, Graph(CS), SphinxSE ...

Storage engines in Practice

- Chose one main storage engine for the application
 - InnoDB is de-facto standard at this point
- User other storage engines for what they are good
 - MyISAM – compact, non transactonal, temporary storage
 - MEMORY – temporary tables
 - Federated – Light duty remote data access etc

Storage Engines

General Purpose Storage Engines Overview

MyISAM

- Traditional MySQL Storage Engine
- First appeared in 3.23.x
 - How many of you remember this version ?
- Based on ISAM going back to UNIREG 15+ years ago
- Table Locks, No Transactions, Non Crash Safe
- Pretty compact, Fast Updates, Can be read only

When to use MyISAM

- If you have Read Only or Read Mostly data which you want to be compact
 - Note: MyISAM is **not** always faster than Innodb for reads.
- When you need fast write performance
 - But not mixed reads/writes for the same table
 - Logging, temporary tables, data crunching
- When recovery time is not critical
 - Large MyISAM table can take many hours to repair after crash.

InnoDB

- Originated by Heikki Tuuri
- Now owned by Oracle Corp
- Was “dormant” for years but new release came out on MySQL Users Conference
 - “Plugin” features compression, fast index creation
- Advanced transactional storage engine
 - MVCC, Row Level Locks, Clustered Keys
- Automatic crash recovery
- Support for foreign keys

When to use Innodb

- Good as default storage engine in many cases
- When you need transactions or foreign keys
- When you need high concurrency
 - So readers do not block writers
- If you do not want corrupted tables on power crash
- Tables (especially indexes) are larger than MyISAM
 - 2-5 times larger in majority of cases
- If table gets corrupted recovery is complicated
- Import/Index built can be very slow before Plugin release.

Falcon

- Storage engine designed by Jim Starkey
- MySQL's prime planned transactional storage engine
 - Though told not to be Innodb Competitor
- Focused on working on systems with many CPUs and large amount of memory
- Has a lot of Innovative (different?unproved?) decisions
- Not Clustered, Can't use covered Indexes
- Still Unstable

When would you use Falcon ?

- Very limited production use to tell
 - Yes we've seen people running Slaves on Falcon in production to test it out
- Will be hard to replace Innodb for many applications
- May be more scalable if Innodb does not gets its bottlenecks fixed
- Promises to be faster for small transactions

Maria

- Design and Development lead by Michael Widenious
 - With few MySQL old timers
- The ideological breed between MyISAM and Innodb
 - MyISAM storage efficiency, base structure
 - Innodb's Multi Versioning, Transactions, Recoverability
- Can use Page level and Storage storage format
- Transaction support planned to be Optional
- Currently Crash Safe version available
- Not optimized for performance yet.
- Alpha stage

What Maria will be good for ?

- No production use feedback yet
- Good replacement for MyISAM
 - Crash safe system tables, cachable large temporary tables
- Will likely be good alternative for Innodb for many cases
 - When clustering by Primary key is not critical for application
- Operational ease of use of MyISAM
 - Repairing tables, moving tables between servers with some preparation

PBXT

- Developed by Paul McCullagh from PrimeBase
- Developed specially for MySQL
- Has its already 3rd or 4th rewrite
 - The last version is now ACID by design
- A lot of innovative ideas.
 - “Write Once”, Log per transaction etc
- Designed to deal with blobs very efficiently
- Rather unstable still

What PBXT may be used for ?

- To work together with MyBS – blob streaming
- Storing lots of log data
- May work well with SSD Drives
 - Mainly bulky sequential drives
- Should know better as we see more production use
- Generally showed good performance for some queries.

What is about SolidDB

- Last year we looked into SolidDB
 - And this year we removed it from consideration
- SolidDB for MySQL project stopped after Solid Technologies was bought by IBM
 - So it never become real alternative
- The code is available on SourceForge
 - But does not have user community to move it forward.

Storage Engines

Using Special Purpose MySQL Storage Engines

MEMORY

- Stores Data in Memory
 - Contents lost on restart, though table remains
- Used internally for temporary tables
 - To resolve certain queries
- Very fast for storing temporary results sets
- Watch out ! Fixed size rows only.
- Pre-Loading data for fast access
- Be careful with replication !

ARCHIVE

- Fast and efficient compressed storage
- No Indexes – Full table scans only
- Non blocking inserts
- Can take space considerably less than MyISAM
- Helpful for storing logs
 - Keep one table per day or use Partitions for efficiency

FEDERATED

- Access data stored on remote server
- Good when you just need few rows from remote server
- Pretty bad with JOINS – requires many round trips
- Beware of large result sets retrieved
 - Server can run out of memory

BLACKHOLE

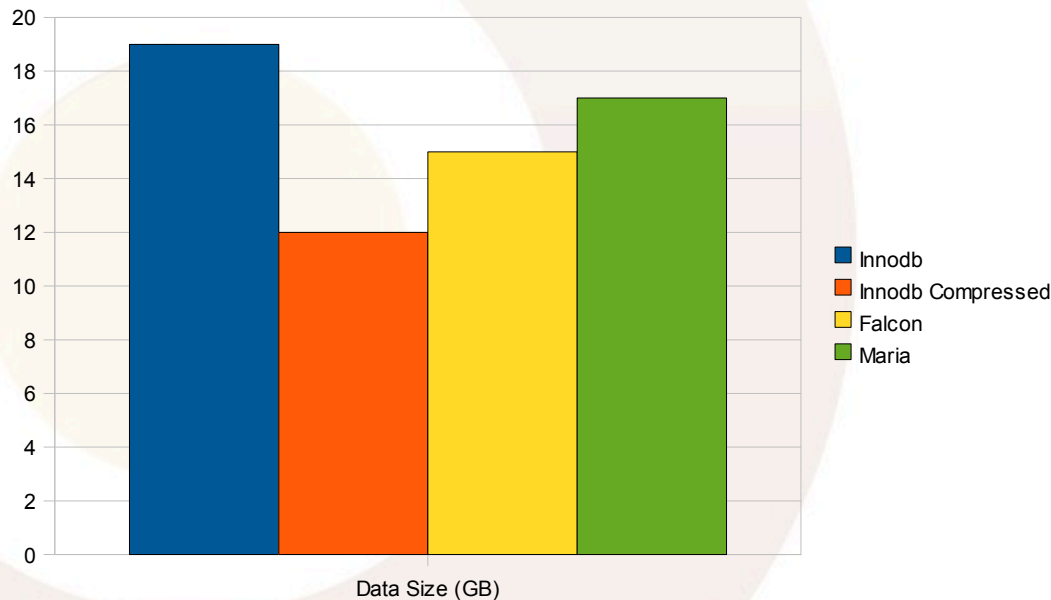
- Initially created for benchmarking and example purposes
- Though found to be very helpful for filtered replication
 - Though some gotchas remain
 - **ALTER TABLE TBL ... ENGINE=MYISAM**
 - Will convert blackhole engine to MyISAM

Benchmarks

Want some Benchmarks ?

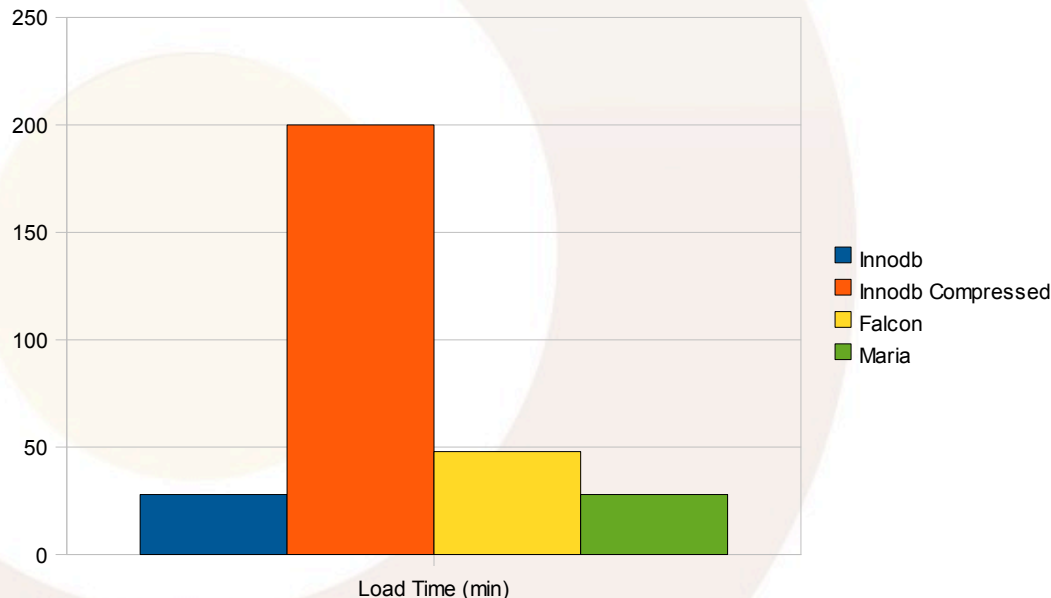
DBT2 Data Sizes

- Loading 200 warehouse DBT2 database
- PBXT Crashed during the load
- Maria footprint is surprisingly large



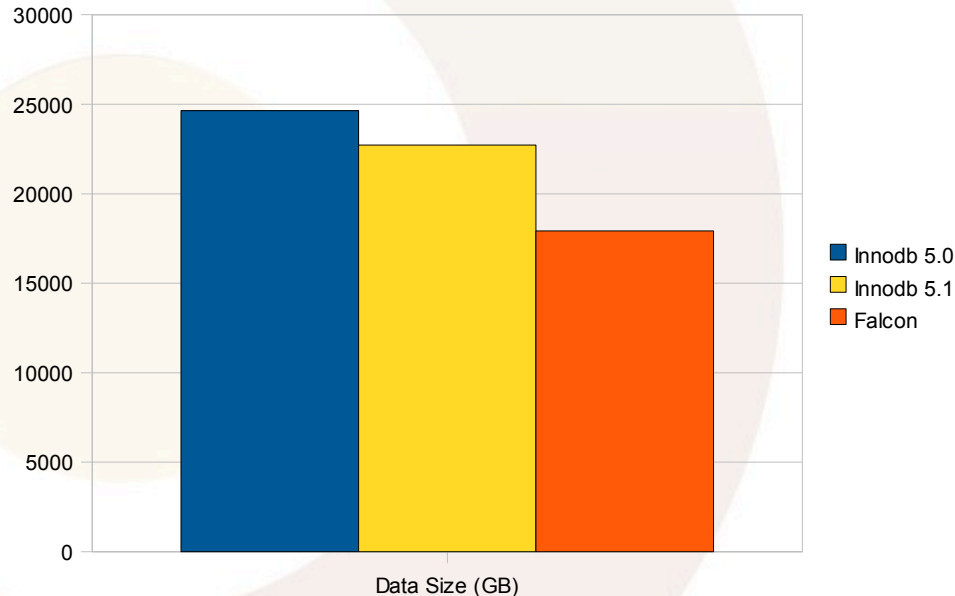
DBT2 Load Times

- 200W load Time
- 2*Dual Core Xeon 5148, 16G Ram
 - 8GB buffers allocated to the engines
- Fast index creation was **not** used for Innodb



DBT2 Results

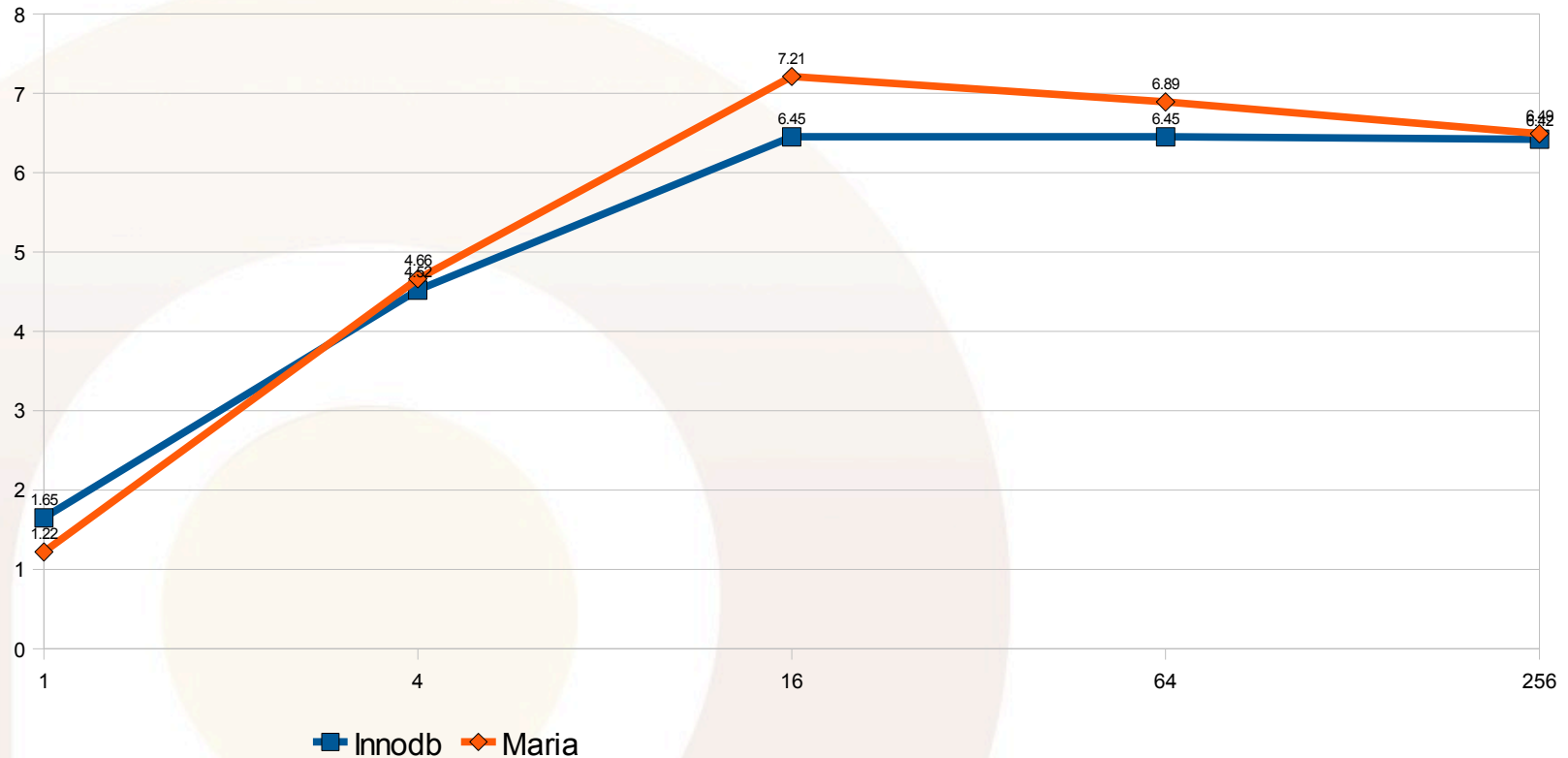
- DBT2 10W Results (CPU bound)
- Falcon significantly improved since last year run
- 5.1 is a bit slower than 5.0 with Innodb
- Maria results were too unstable to show



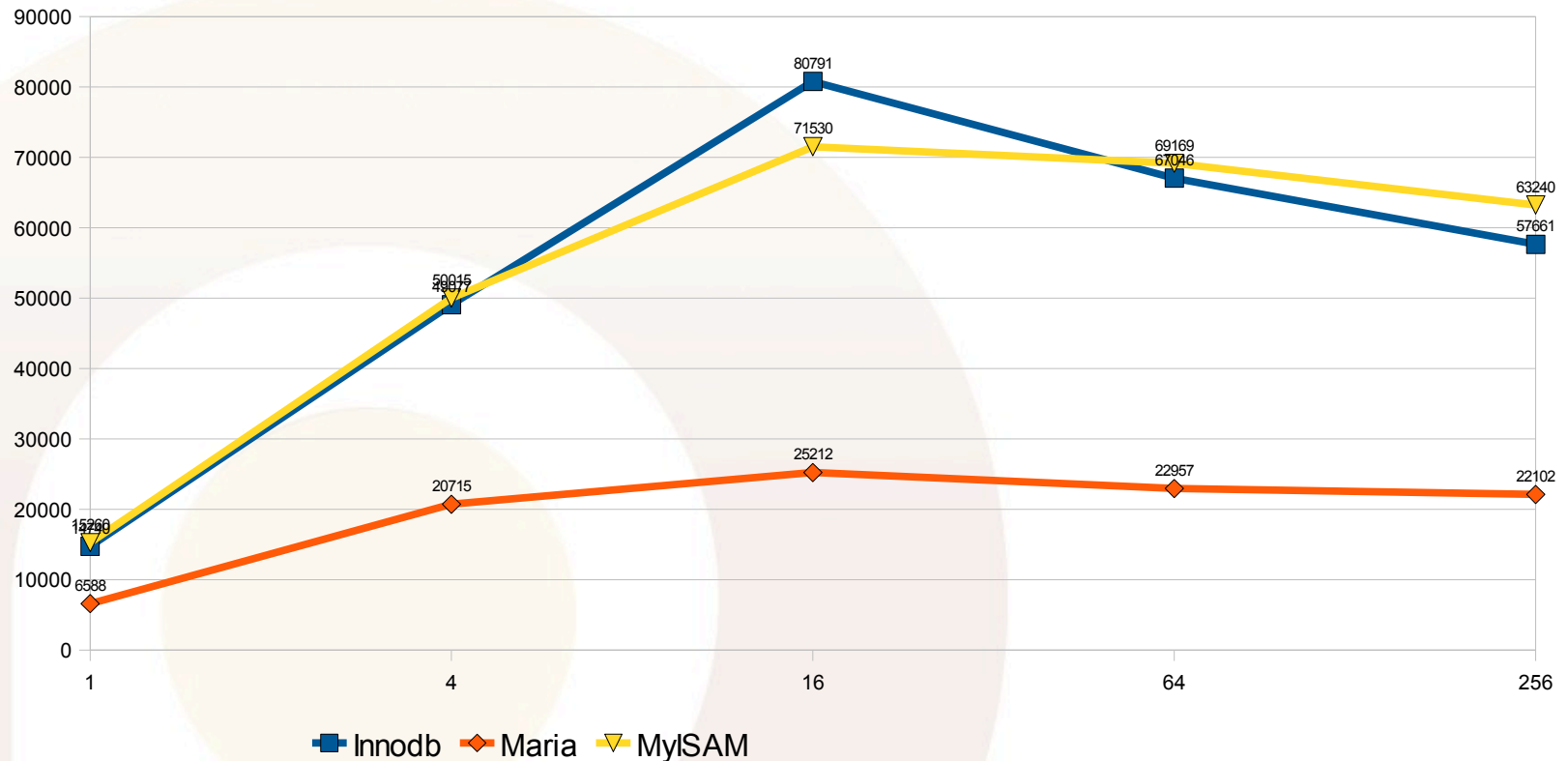
Primitives Benchmark

- CPU Bound, Results in Queries Per Sec
- 1.000.000 rows
- Dual Quad Core Xeon
- Focusing on the typical storage engine access paths for the data
- Details about queries and schema can be found
- <http://www.mysqlperformanceblog.com/files/benchma>
- Falcon excluded as crashing for some of the test
 - So not completing the run
- PBXT was hanging on data load

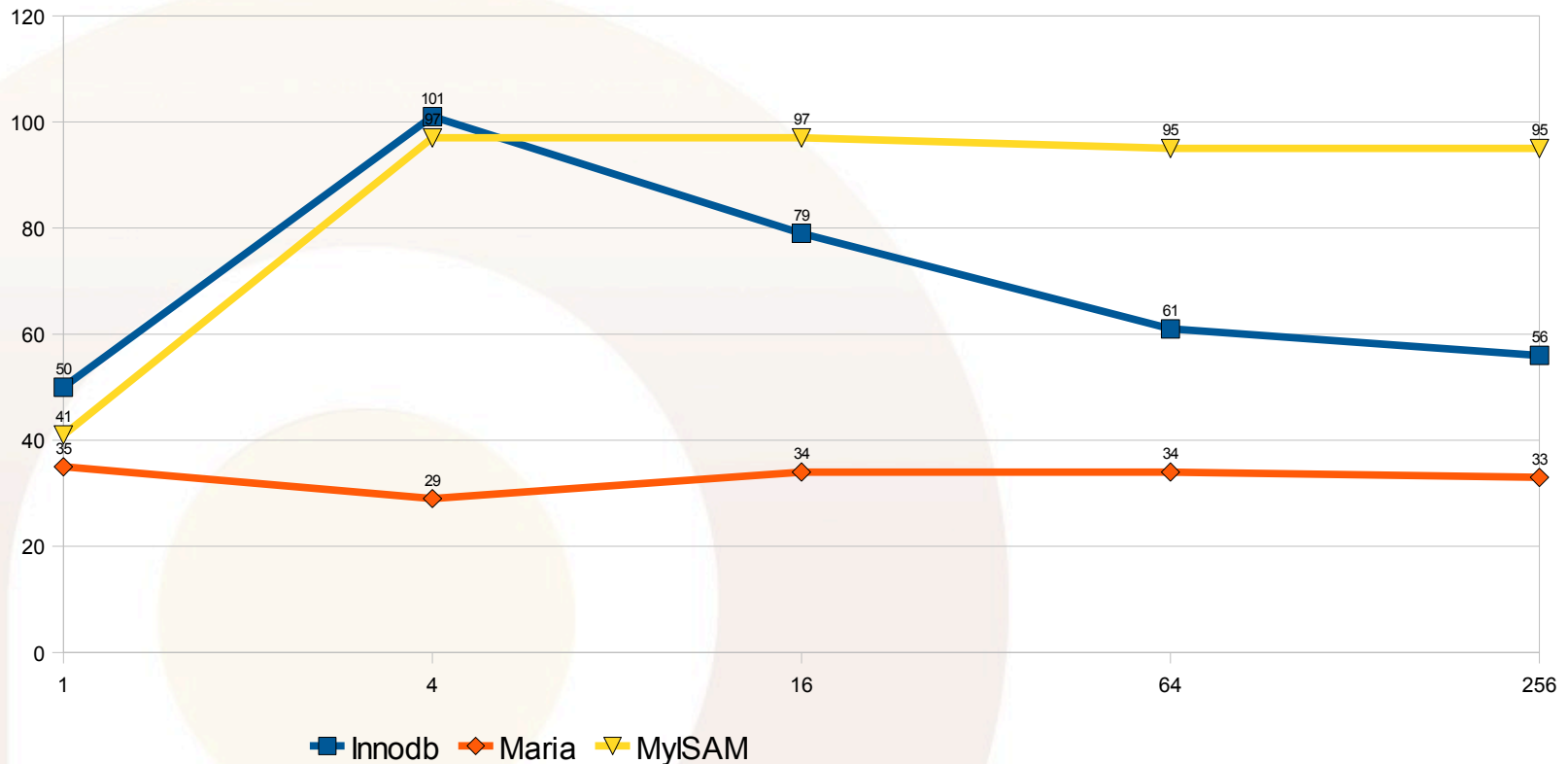
Full Table Scan



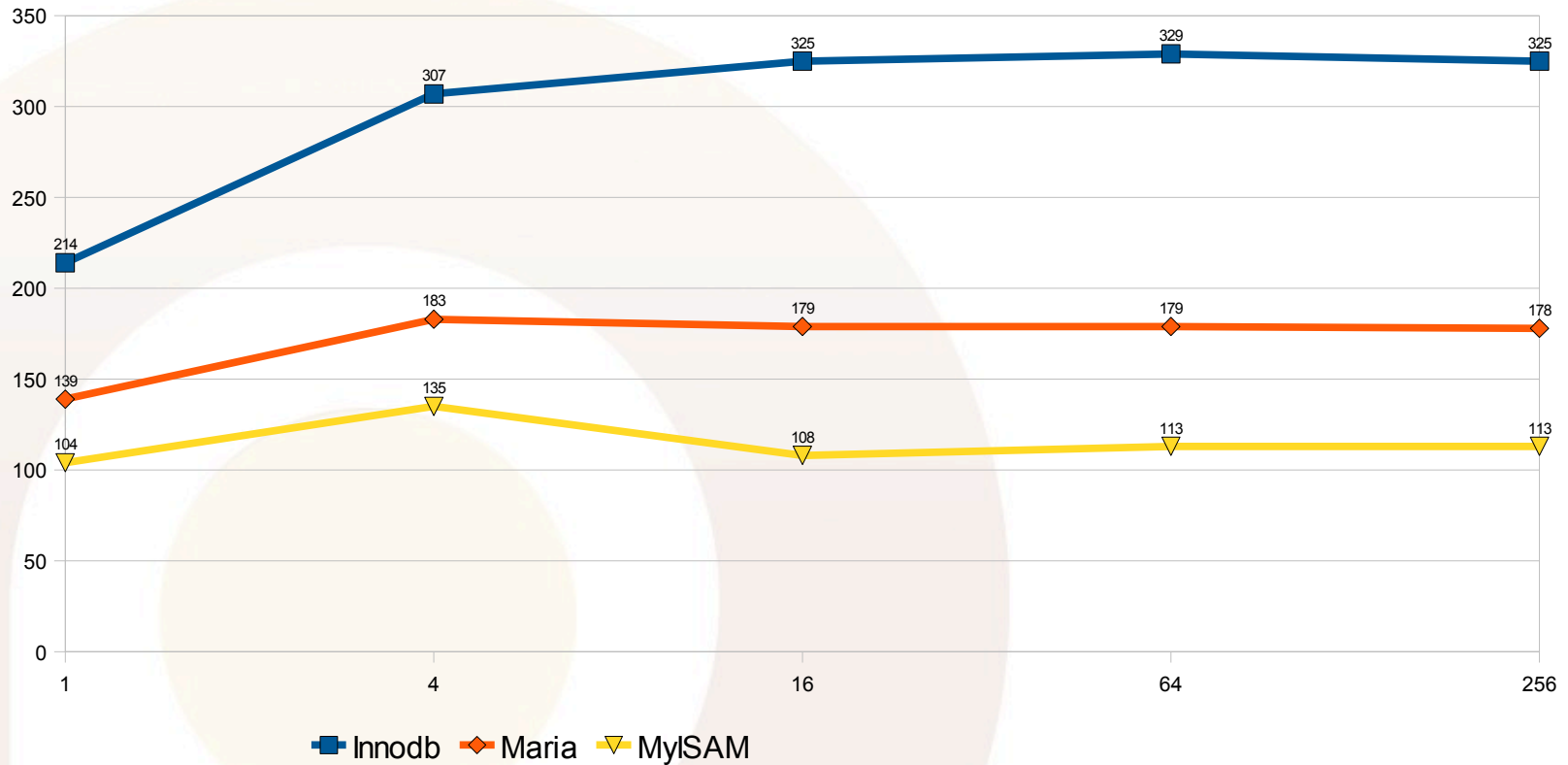
Single row access by PK



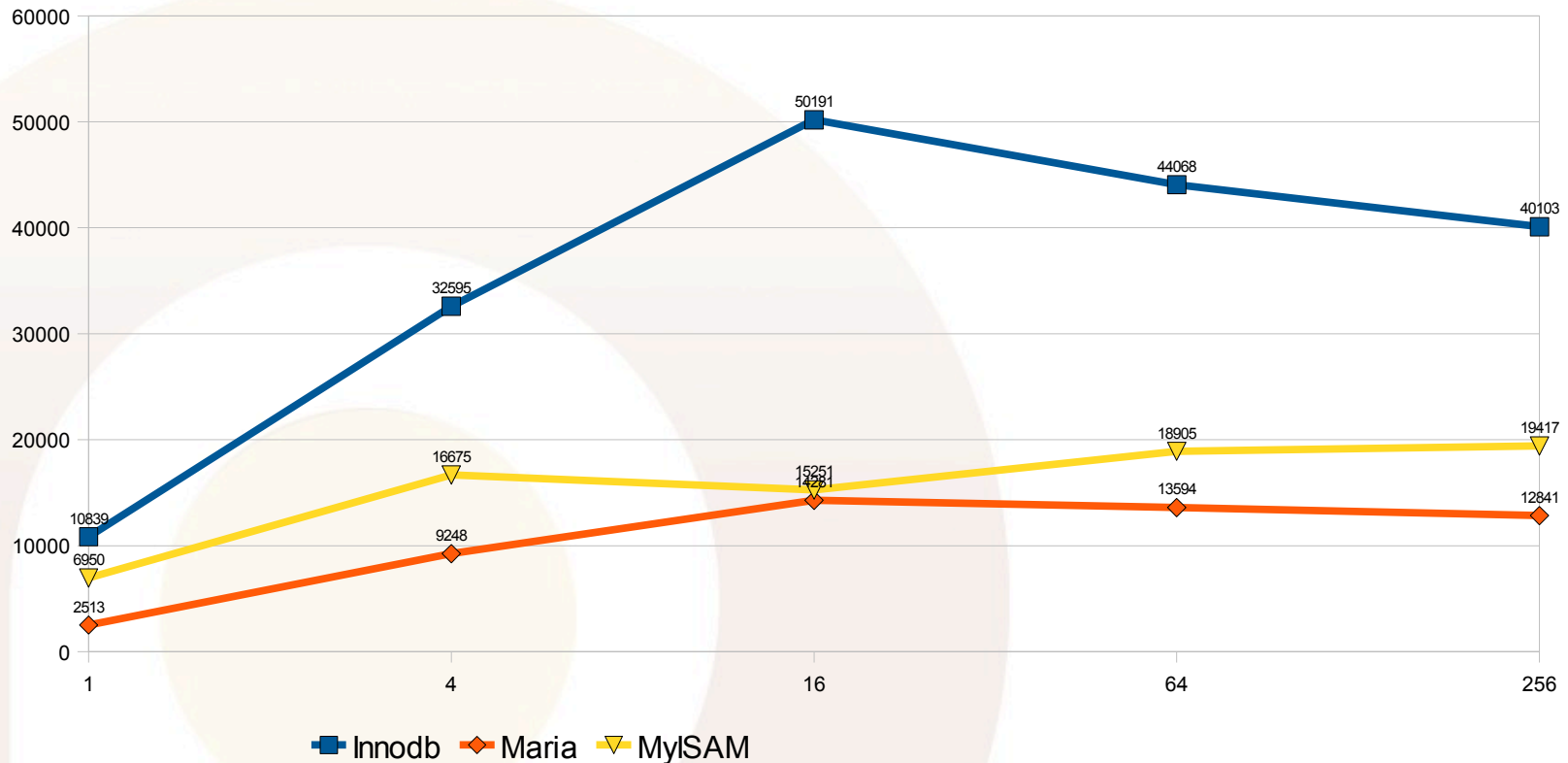
Access by Index



Access by Covering Index



Access by Key with LIMIT



Thanks for Coming

- Questions ? Followup ?
 - pz@percona.com
- Yes, we do **MySQL and Web Scaling Consulting**
 - <http://www.percona.com>
- Check out our book
 - Complete rewrite of 1st edition

