

10 Things Developers should know about Databases

Peter Zaitsev
CEO, Percona

1 July 2019



Thank you

The logo features the text '@mail.ru' in a large, rounded font. The '@' symbol and '.ru' are orange, while 'mail' is white. Below this, the word 'group' is written in a smaller, white, rounded font. The entire logo is set against a dark blue rectangular background.

@mail.ru
group

Who Are you ?

**More
Developer ?**

More OPS ?

Ops

Focused on Database Only

Generalist

Programming Language

**What Programming
Languages does your
team use ?**

There is often Tension Devs vs Ops

DevOps suppose to have solved it but tension is still common between Devs and Ops

Especially with Databases which are often special snowflake

Especially with larger organizations

Large Organizations

**Ops vs Ops have conflict
too**

Devs vs Ops Conflict

Devs

- Why is this stupid database always the problem.
- Why can't it just work and work fast

Ops

- Why do not learn schema design
- Why do not you write optimized queries
- Why do not you think about capacity planning

Database Responsibility

**Shared Responsibility for
Ultimate Success**

Top Recommendations for Developers

Learn Database Basics

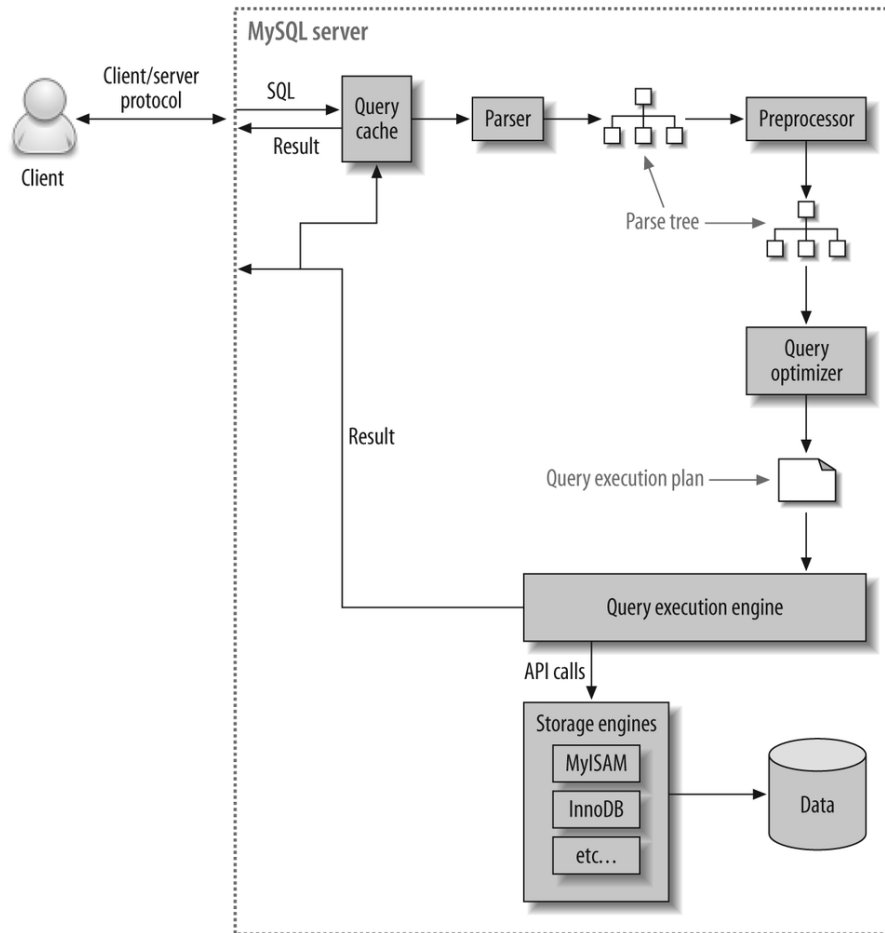
You can't build great database powered applications if you do not understand how databases work

Schema Design

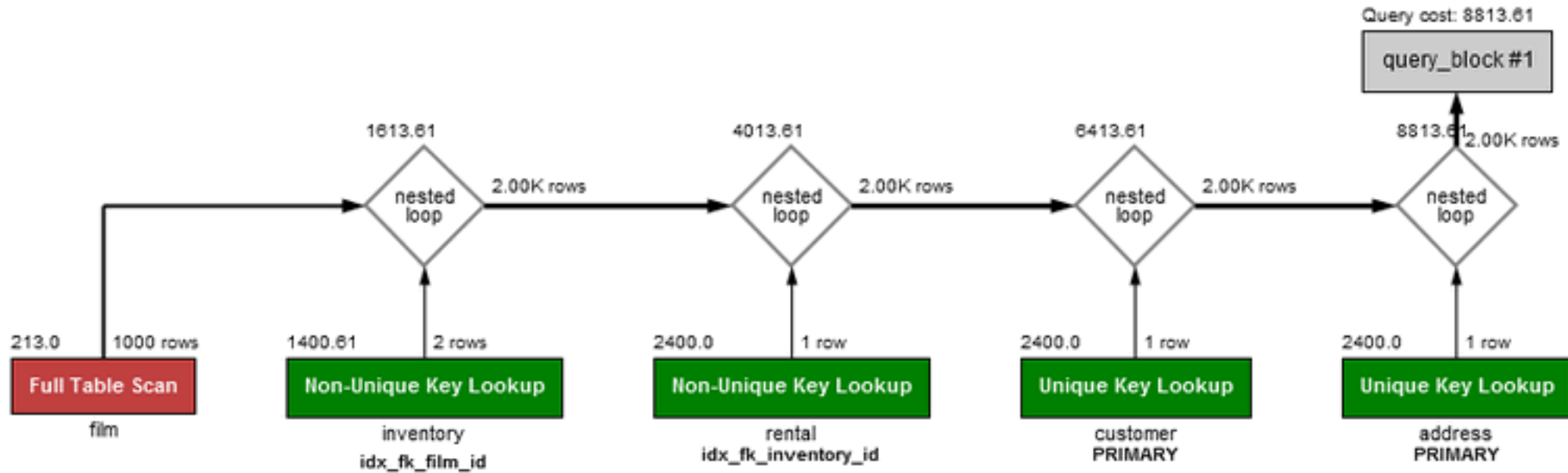
Power of the Database Language

How Database Executes the Query

Query Execution Diagram



EXPLAIN



<https://dev.mysql.com/doc/refman/8.0/en/execution-plan-information.html>

How are Queries Executed ?

Single Threaded

Single Node

Distributed

Indexes

**Indexes are
Must**

**Indexes are
Expensive**

Capacity Planning

No Database can handle “unlimited scale”

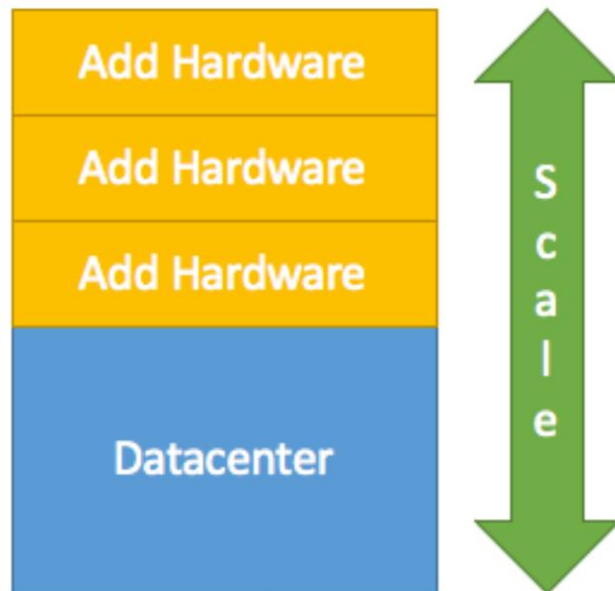
Scalability is very application dependent

Trust Measurements more than Promises

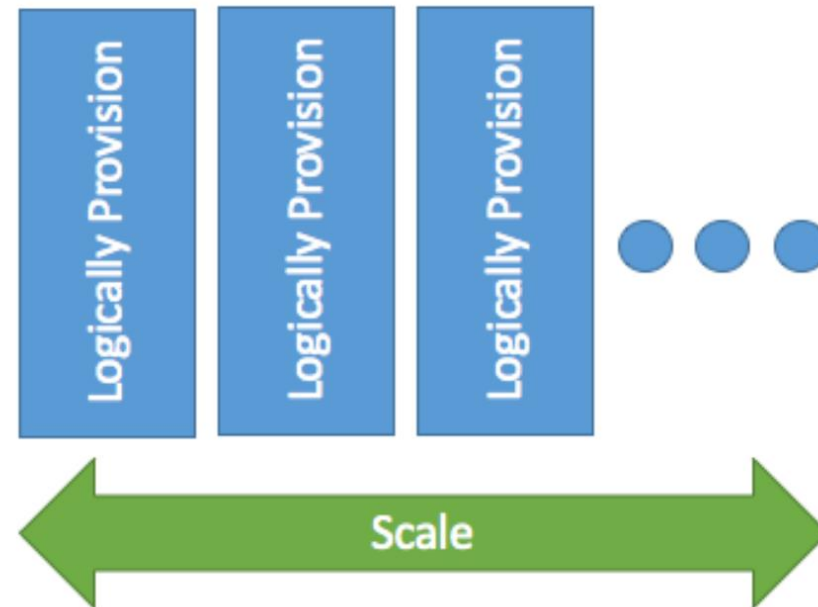
Can be done or can be done Efficiently ?

Vertical and Horizontal Scaling

Vertical Scaling



Horizontal Scaling



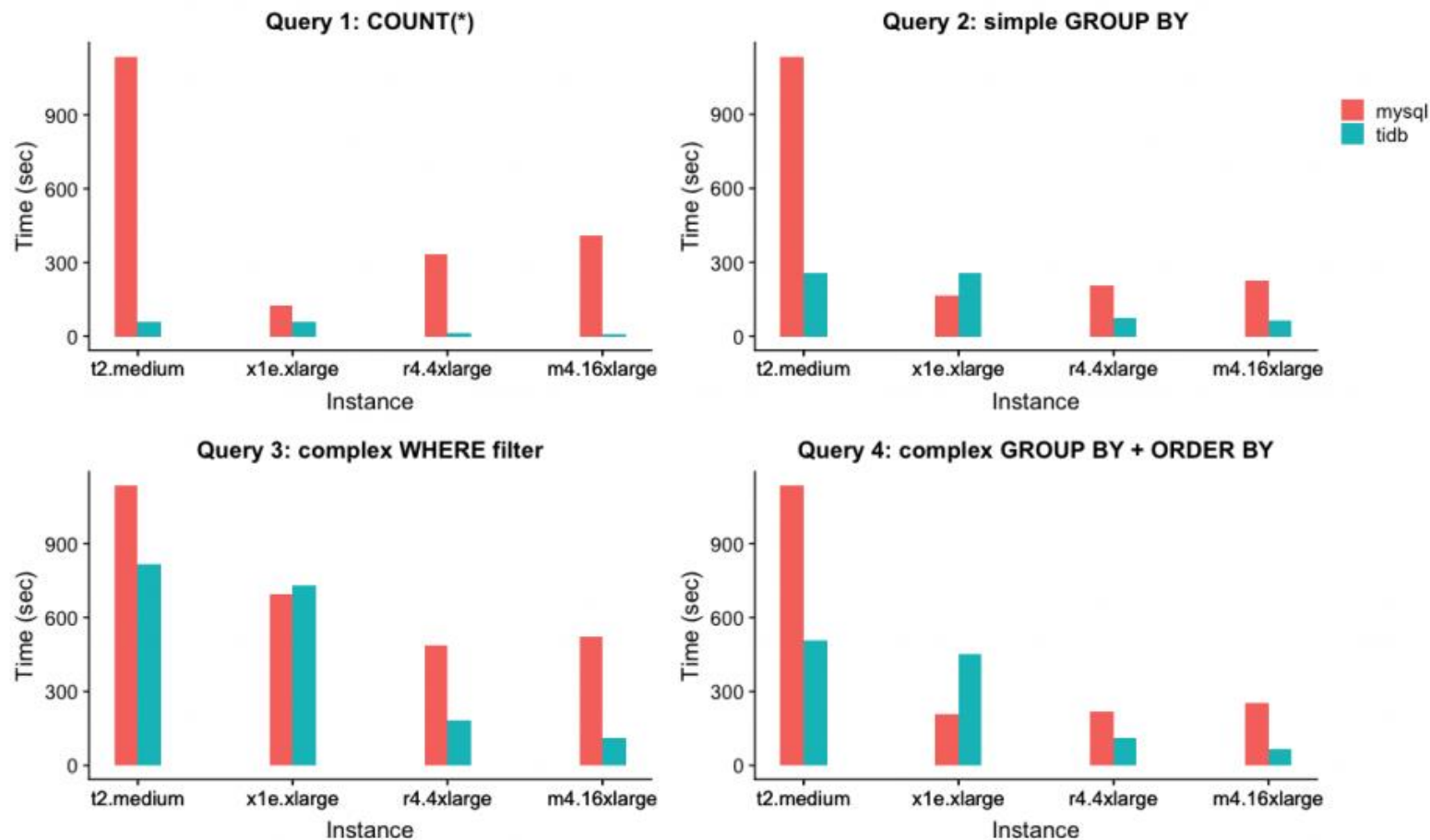
Scalable != Efficient

The Systems which promote a scalable can be less efficient

Hadoop, Cassandra, TiDB are great examples

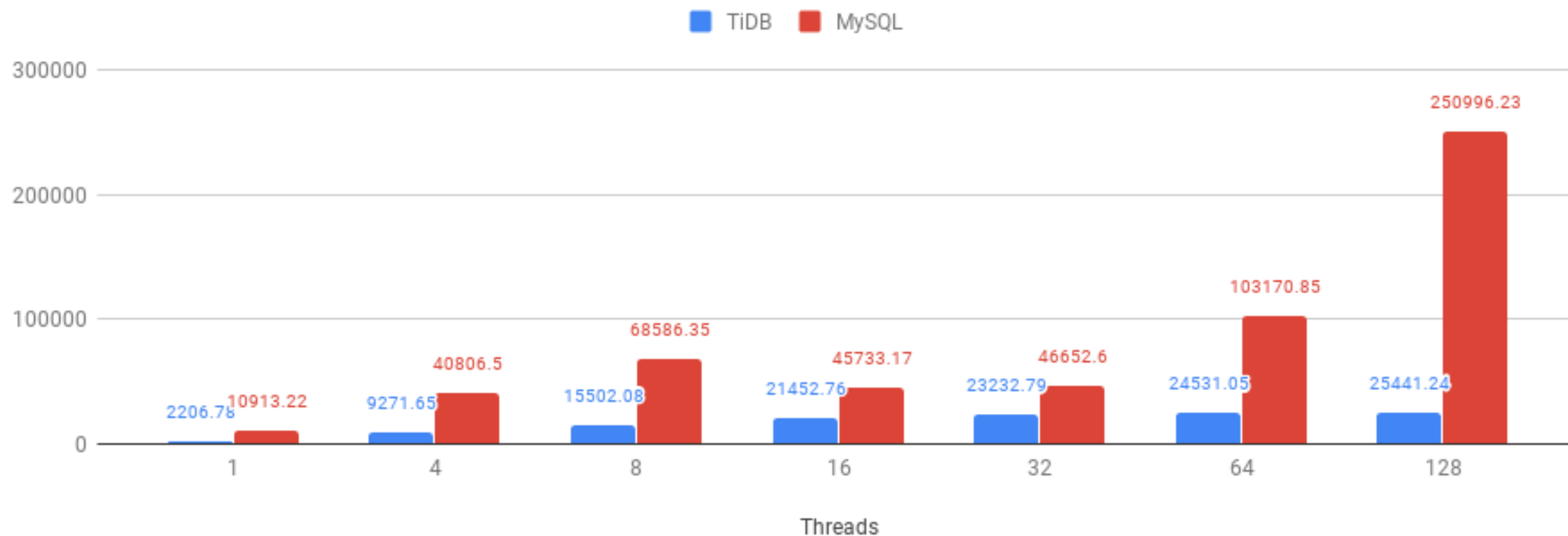
By only the wrong thing you can get in trouble

TiDB Scalability (Single Node)



TiDB Efficiency

TiDB and MySQL - point selects - sysbench



Throughput != Latency

If I tell you system can do
100.000 queries/sec
would you say it is fast ?

Speed of Light Limitations

High Availability Design Choices

You want instant durable replication over wide geography or Performance ?

Understanding Difference between High Availability and Disaster Recovery protocols

Network Bandwidth is not the same as Latency

Also Understand

Connections to the database are expensive

Especially if doing TLS Handshake

Query Latency Tends to Add Up

Especially on real network and not your laptop

Law of Gravity

**Shitty Application at
scale will bring down any
Database**

Scale Matters

Developing and Testing with Toy Database is risky

Queries Do not slow down linearly

The slowest query may slow down most rapidly

Memory or Disk

Data Accessed in memory is much faster than on disk

It is true even with modern SSDs

SSD accesses data in large blocks, memory does not

Fitting data in Working Set

Newer is not Always Faster

Upgrading to the
new
Software/Hardware
is not always faster

Test it out

Defaults Change are
often to blame

Upgrades are needed but not seamless

Major Database Upgrades often require application changes

Having Conversation on Application Lifecycle is a key

Character Sets

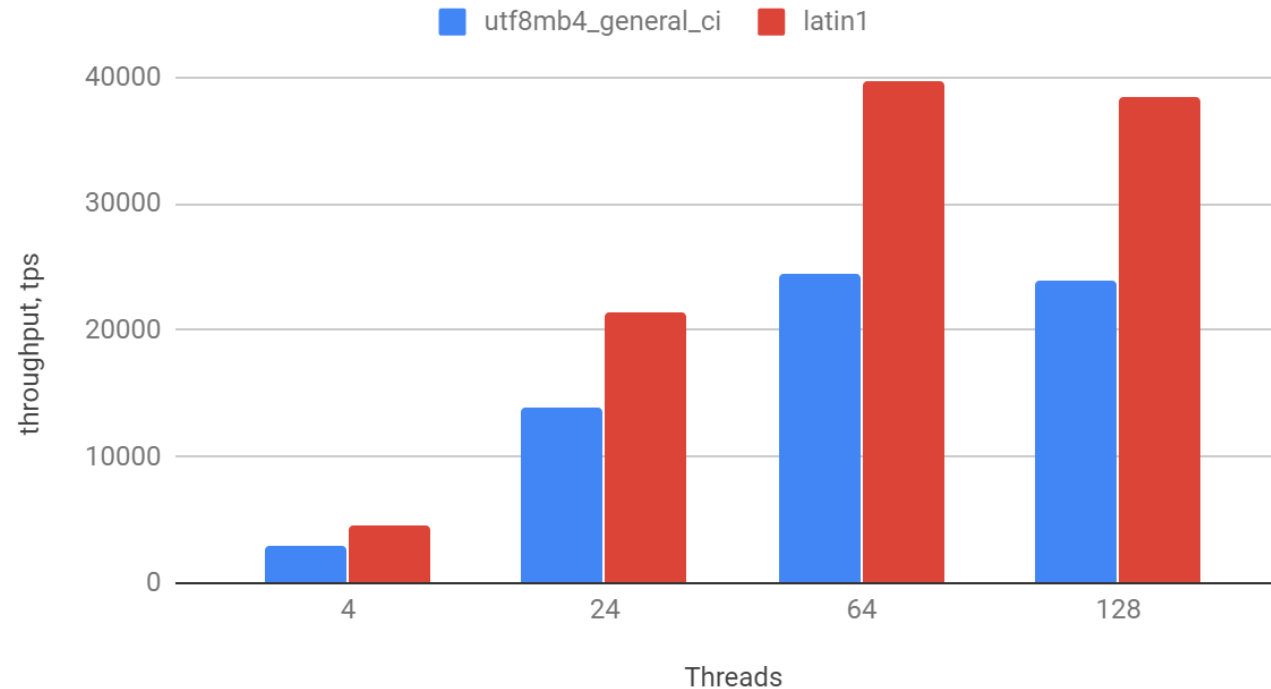
Performance Impact

Pain to Change

Wrong Character Set can cause Data Loss

Character Sets

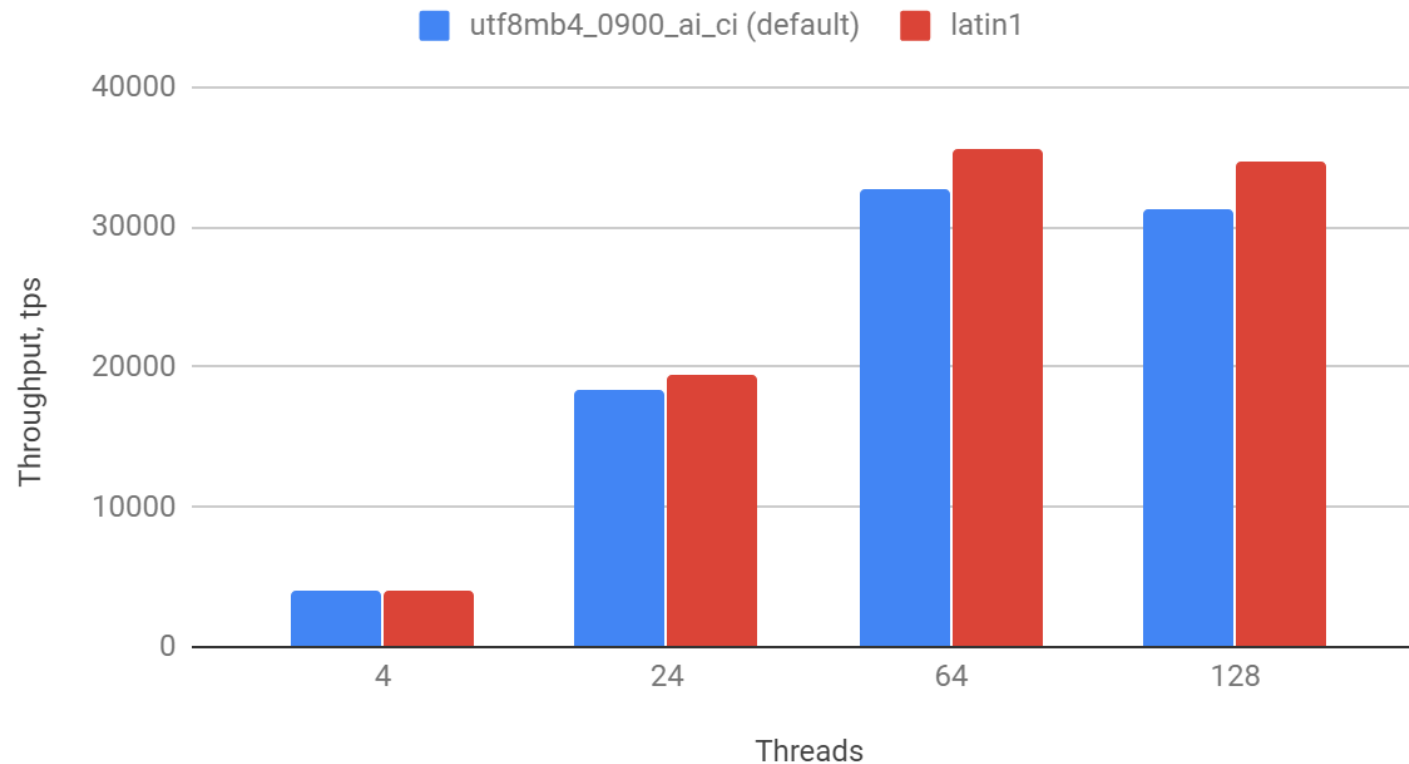
MySQL 5.7 utf8mb4_general_ci (default) and latin1



<https://per.co.na/MySQLCharsetImpact>

Less impact In MySQL 8

MySQL 8.0 utf8mb4_0900_ai_ci and latin1



Operational Overhead

Operations Take Time, Cost Money, Cause Overhead

10TB Database Backup ?

Adding The Index to Large Table ?

Distributed Systems

10x+ More Complicated

Better High Availability

Many Failure Scenarios

Test how application performs

Risks of Automation

**Automation is
Must**

**Mistakes can
destroy
database at scale**

What Else

What Would you Add ?

Check Out <http://per.co.na/careers>



**BECOME A
PERCONA SUPERHERO**

WE'RE HIRING

**CONTACT
careers@percona.com**

Call for Papers Now Open!

Submit by July 15
www.percona.com/cfp



PERCONA
LIVE EUROPE
AMSTERDAM

#PERCONALIVE

Thank You!

Twitter: @percona @peterzaitsev
