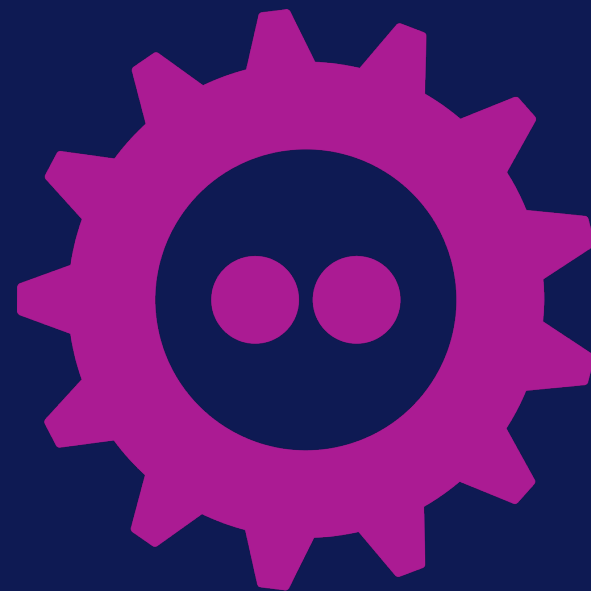




Deep dive into MySQL Query Performance

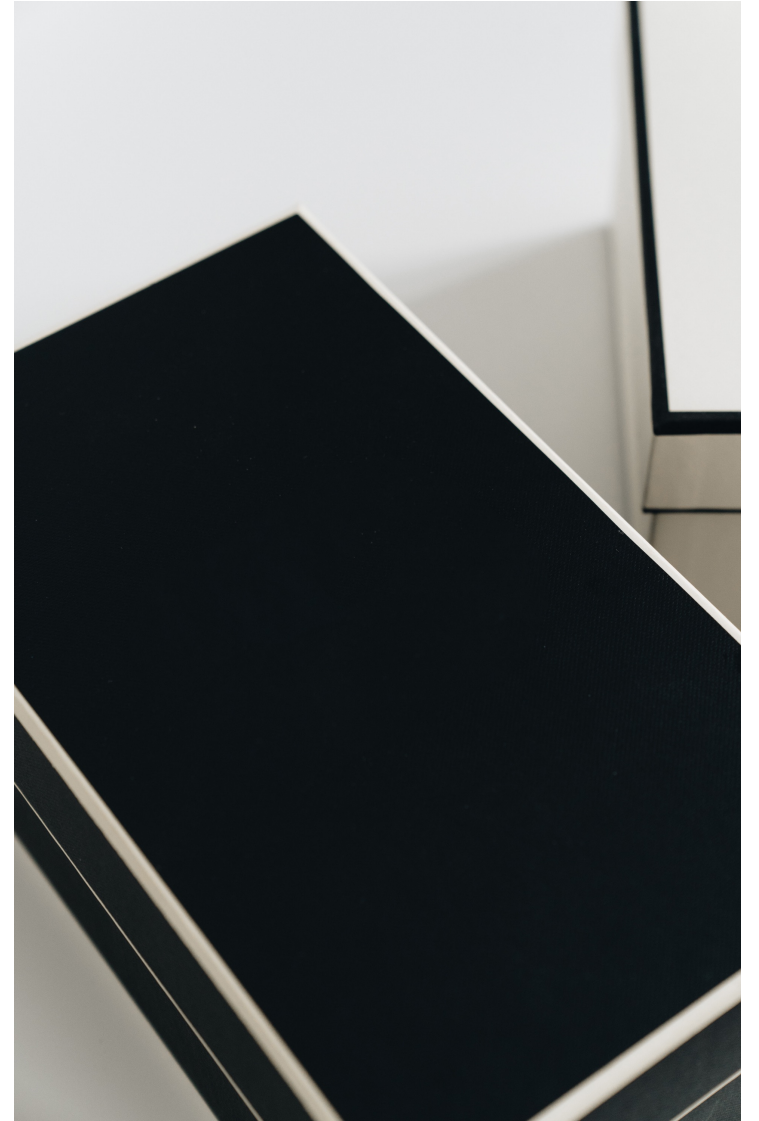
Peter Zaitsev

Founder at Percona
5 Feb 2023





Database is a Black Box





You can connect to the Database Service Point, Quickly



Run Queries you need to run

Meaning

Queries

1

**Run them
without
errors**

2

**Run them
with correct
results**

3

**Run them
with required
response
time**

The background of the slide features four Blue Angels fighter jets in a staggered formation, flying from the bottom left towards the top right. The jets are dark blue with yellow accents on the wings and tails. The sky is a deep blue with scattered white clouds. On the far left, there is a solid orange vertical bar. On the far right, there is a light blue vertical bar.

Performance

**Performance is about Response Time
you get for your Queries**

Response Time - Database View

**"I see database responds to
queries in 5ms in average"**



Response Time – Business View

**All Users have outstanding
performance experience with all their
application interactions**



Downtime

**Very Bad Performance is
indistinguishable from downtime**






An aerial photograph of a river meandering through a dense, lush green forest. The river is dark and reflects the surrounding trees. The forest is thick with various shades of green, indicating a healthy ecosystem.

Forget averages

There once lived a man who tried to cross a river,
in average one meter deep

PMM Query Analytics ▾

Reset All


#	Query ▾	Search by... 🔍	Load	Query Time
	TOTAL			7.44 ms
1	update warehouse1 set w_ytd = w_ytd + ? where w_id = ?	①		231.41 ms
2	select i_price, i_name, i_data from item1 where i_id = ?	①		18.76 ms
3	select c from sbtest1 where id=?	①		1.87 ms
4	select d_next_o_id, d_tax from district1 where d_w_id = ? and d_id ...	①		69.65 ms
5	update district1 set d_ytd = d_ytd + ? where d_w_id = ? and d_id= ?	①		48.95 ms
6	insert into order_line1 (ol_o_id, ol_d_id, ol_w_id, ol_number, ol_i_id, ...	①		2.40 ms
7	update stock1 set s_quantity = ? where s_i_id = ? and s_w_id= ?	①		2.07 ms
8	select count(distinct (s_i_id)) from order_line1, stock1 where ol_w_...	①		132.84 ms
9	commit	①		4.74 ms
10	insert into new_orders1 (no_o_id, no_d_id, no_w_id) values(?,+)	①		10.50 ms
11	update order_line1 set ol_delivery_d = now() where ol_o_id = ? and ...	①		7.26 ms

Query Time

Per query : 7.44 ms

Sum : 10 days, 0:59:57

From total : 100.00 %



▮ Max : 32.19 sec

◦ Avg : 7.44 ms

• 99% : 83.47 ms



**99 percentile does not
translate in 99% users
having great performance**



**If every user interaction
has 10 database queries**



**User in average has 10
interactions**



**Roughly 50% of session
will have query with p99
response time**

Percentile

Errors

- Look at Response time of Successful Queries, do not let “fast errors” to screw up your data
- Measure response time of “slow errors” as it contributes to user experience

Over Time









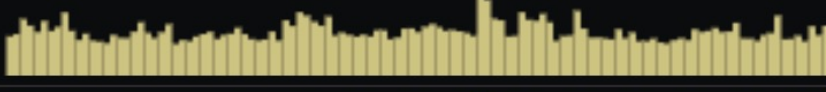
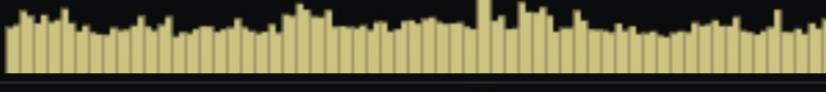
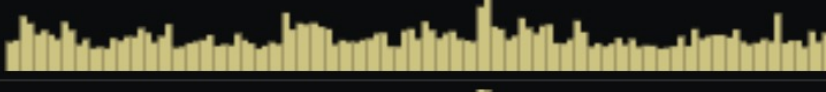
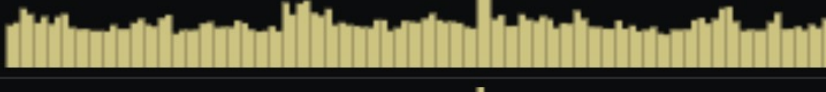

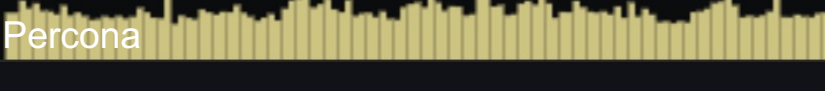
LOOK AT RESPONSE TIME
TRENDS OVER TIME



MINOR SLOWDOWN OFTEN
HAPPENS BEFORE POOR
PERFORMANCE
“DOWNTIME”



PERFORMANCE CAN BE
WORSE AT CERTAIN TIMES –
BACKUPS, BATCH JOBS,
MAINTENANCE

Search by...	Q	Query_Time		Query_Count	Lo
			7.44 ms	2.70k QPS	
w_ytd = w_ytd + ? where w_id = ?	ⓘ		231.31 ms	25.52 QPS	
data from item1 where i_id = ?	ⓘ		18.76 ms	255.92 QPS	
where id=?	ⓘ		1.87 ms	1.00k QPS	
x from district1 where d_w_id = ? and d_id ...	ⓘ		69.66 ms	25.61 QPS	
d = d_ytd + ? where d_w_id = ? and d_id= ?	ⓘ		48.95 ms	25.52 QPS	
l_o_id, ol_d_id, ol_w_id, ol_number, ol_i_id, ...	ⓘ		2.40 ms	255.67 QPS	
ntity = ? where s_i_id = ? and s_w_id= ?	ⓘ		2.07 ms	255.67 QPS	
i_id)) from order_line1, stock1 where ol_w_...	ⓘ		132.77 ms	2.56 QPS	
	ⓘ		4.73 ms	58.52 QPS	
(no_o_id, no_d_id, no_w_id) values(?+)	ⓘ		10.50 ms	25.61 QPS	
l_delivery_d = now() where ol_o_id = ? and ...	ⓘ		7.26 ms	25.38 QPS	15

Database or Application ?

- Instrument on Application side to understand what drives performance of user interactions

Instrument of Database Size to understand what causes queries to be slow and what can be done about it



Response Time – Business View

**All Users have outstanding performance
experience with all their application interactions**

Enhancing Query Meta Data

SQL Commenter project by Google

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

Query Meta Data Possibilities

Actual User/Tenant

Application/Functionality

Version Information (A/B Testing)

Responsible Engineer/Team

“Query”

- “Different Queries” have different performance profile
- They also correspond to different “user actions”
- And may have different acceptable level of Performance

Schema and Database

- Different Applications/Services may be using different ones
- In sharded environment can correspond to application "tenant"

#








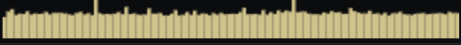
Database ▾

Search by...

Q

Copy Link

Add column

		Load	Query Count	Query Time
	TOTAL	 38.80 load	2.47k QPS	15.74 ms
1	tpcc1	 36.89 load	449.93 QPS	81.98 ms
2	sbtest	 1.28 load	1.56k QPS	815.27 μs
3	tpcc2	 0.58 load	437.09 QPS	1.34 ms
4	postgres	 0.06 load	11.01 QPS	5.13 ms
5	tpcc3	 <0.01 load	1.00 QPS	12.31 μs
6	tpcc4	 <0.01 load	0.99 QPS	11.17 μs
7	tpcc5	 <0.01 load	1.06 QPS	<9.80 μs

Database view in Percona Monitoring and Management

Table

- Can help identify “problematic data”
- Indexing changes impact queries hitting object
- Maintenance often impacts specific table



**IDENTIFY
SERVICE/APPLICATION**





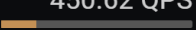



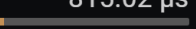

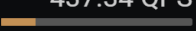
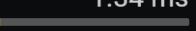


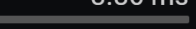


**FIND HUMAN TROUBLE
MAKERS WITH
INTERACTIVE ACCESS**

Database User

Copy Link

Add column

#	User Name▼	Search by... Q	Load	Query Count	Query Time
	TOTAL		 38.83 load	 2.47k QPS	 15.73 ms
1	app1		 36.92 load	 450.62 QPS	 81.92 ms
2	app3		 1.28 load	 1.57k QPS	 815.02 μs
3	app2		 0.59 load	 437.34 QPS	 1.34 ms
4	pmm		 0.06 load	 14.64 QPS	 3.86 ms












Database Host

Sharded environments often have multiple hosts handling the same traffic

Yet Problems often can be limited to some hosts

Data/Traffic Balance, configuration, invisible differences

Database Instances

						Copy Link	Add column
#	Service Name▼	Search by...	Q	Load	Query_Count	Query_Time	
	TOTAL			 10.78 load	3.60k QPS	3.00 ms	
1	ps-80-source-mysql			 6.65 load	2.52k QPS	2.63 ms	
2	pxc-80-1-mysql			 2.41 load	199.17 QPS	12.09 ms	
3	pxc-80-2-mysql			 0.74 load	339.64 QPS	2.19 ms	
4	mariadb-105-mysql			 0.61 load	408.85 QPS	1.49 ms	
5	AzureDB-mysql			 0.30 load	33.71 QPS	8.92 ms	
6	ps-80-gr-2-mysql			 0.02 load	12.67 QPS	1.31 ms	
7	ps-80-gr-1-mysql			 0.02 load	46.38 QPS	348.84 µs	
8	ps-80-gr-3-mysql			 0.01 load	12.67 QPS	1.11 ms	
9	pxc57-2-mysql			 0.01 load	9.64 QPS	1.31 ms	
10	pxc57-3-mysql			 <0.01 load	9.64 QPS	971.65 µs	

**App Server/
Web Server/
Service
Instance**

**You may expect all instances of
the same type causing same
even load**



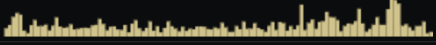


It may not be the case

**Code versions, configuration,
load balancer behavior,
security incidents**

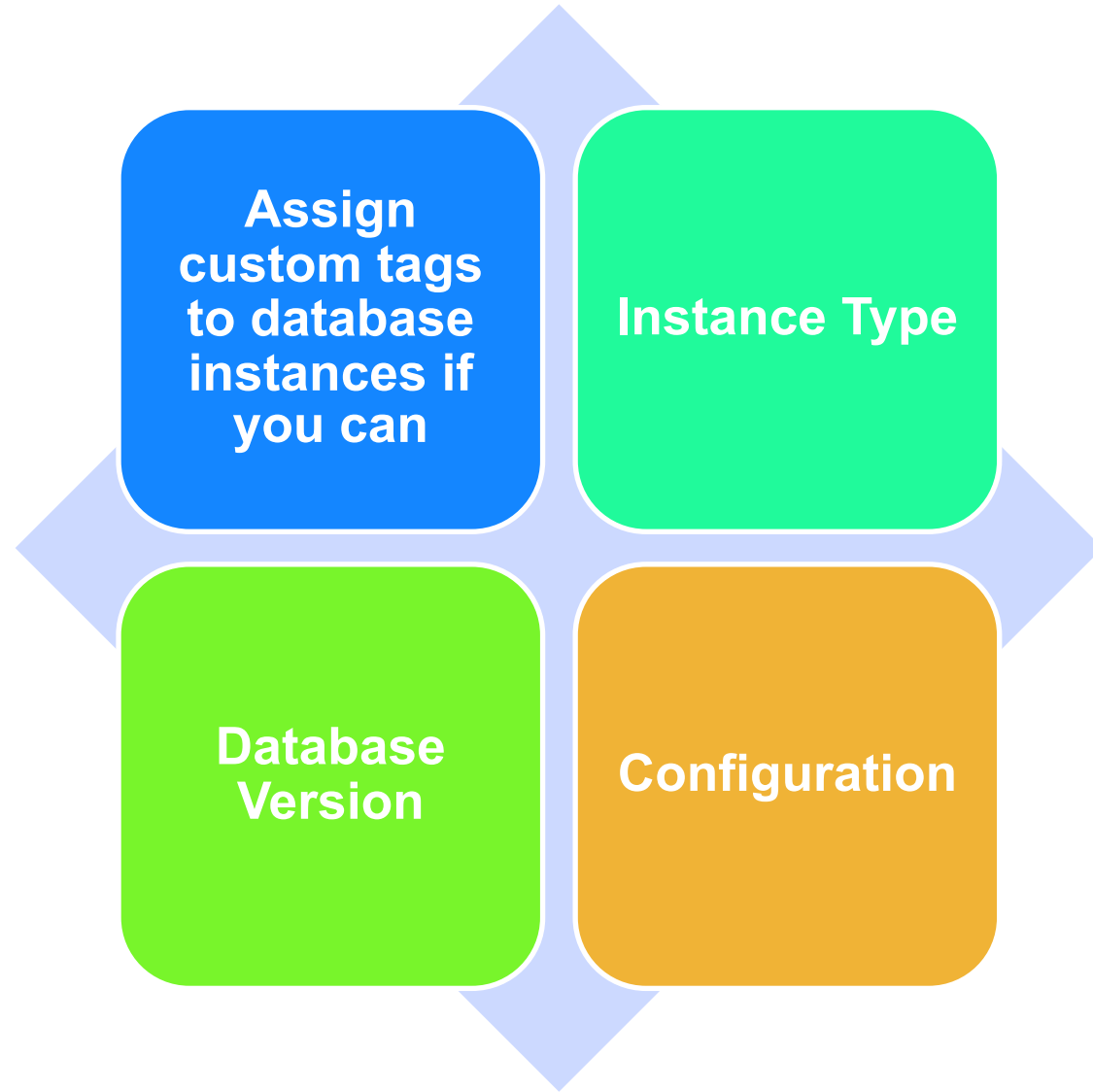
Client Hosts

Copy Link

Add column

#	Client Host	Search by...	Q	Load	Query Count	Query Time
	TOTAL			 38.97 load	2.47k QPS	15.75 ms
1	139.144.169.65			 14.36 load	623.02 QPS	23.05 ms
2	139.144.169.80			 13.26 load	627.49 QPS	21.13 ms
3	139.144.169.84			 11.29 load	1.21k QPS	9.33 ms
4	127.0.0.1			 0.06 load	14.65 QPS	3.86 ms

Custom Tags



Query Plan

- One Query Can have Multiple Different Query Plans
- Sometimes it is good, in other cases it is a problem
- Measure Query Performance by Query Plan
- Can take action to correct query plan if this is the issue

Where Response Time Comes From ?



**Data
Crunching/CPU**



**Waits on CPU
Availability**



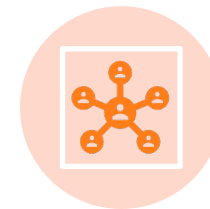
Disk IO



Row Locks



Contention



Network

Other Things to Consider





“Bad Queries” vs Victims

- Query might be slow because it is heavy on its own
- Or it might be victim of other queries or their volume

**Do not
forget
currently
running
queries**

- **Response time is measured when query completes**
- **You can write queries which “never” complete**
- **Consider killing runaway queries and whitelisting queries which need to run long**

Do not Ignore “Invisible”

- **Database Background Activities**
- **Maintenance Operations**
- **Cloud Noise**

Avoid Biased Sampling

“Let’s Look only on slow queries”

Focus on Outliers

Likely to ignore queries causing most load, typical impact

Suggested Improvements

- Better Support for Prepared Statements
- Consider grouping data by time
- Provide List of Tables Query Touches
- Plan ID Information
- Top Waits Summary

Good Luck

Get your query performance under control
Do not over-do scaling by Credit Card

Let's Connect!

<https://www.linkedin.com/in/peterzaitsev/>

<https://twitter.com/PeterZaitsev>

<http://www.peterzaitsev.com>



THANK YOU!

percona.com