

MySQL Performance Troubleshooting Best Practices

Francisco Bordenave - Architect, Percona

Agenda

- Who am I?
- Introduction
- Identifying the source of problem
- We know where the problem is, now what?
- Best practices on troubleshooting.

Who am I?

- Percona Architect
- DBA and MySQL guy for over 10 years.
- Oriented to simplify things - KISS approached.
- Homebrewer and football fan in my free time.

Introduction

- Defining performance issues.
 - My app is slow

Introduction

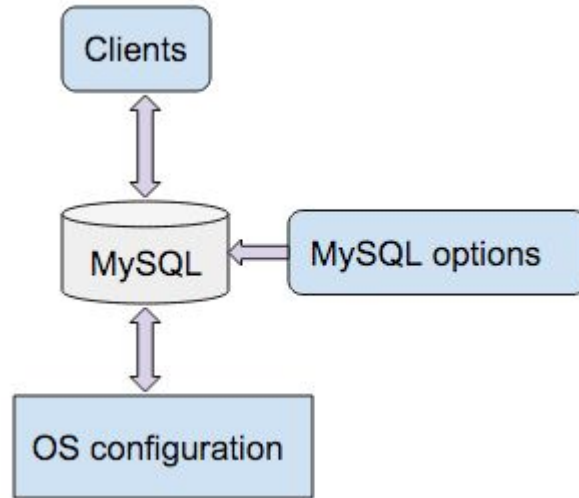
- Defining performance issues.
 - My app is slow
 - *Response time is not enough?*

Introduction

- Defining performance issues.
 - My app is slow
 - *Response time is not enough?*
 - *Try to identify a pattern (i.e. high load, concurrency)*

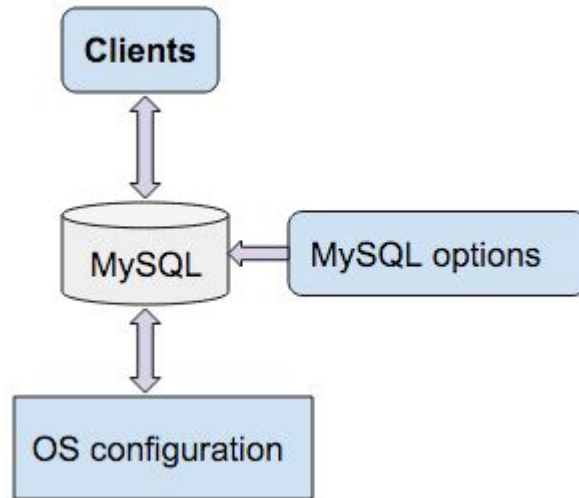
Identifying the source of slowness

- Check what is being slow



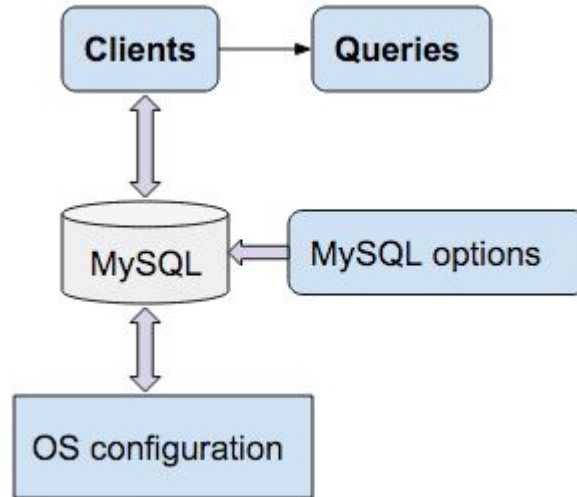
Identifying the source of slowness

- Check what is being slow



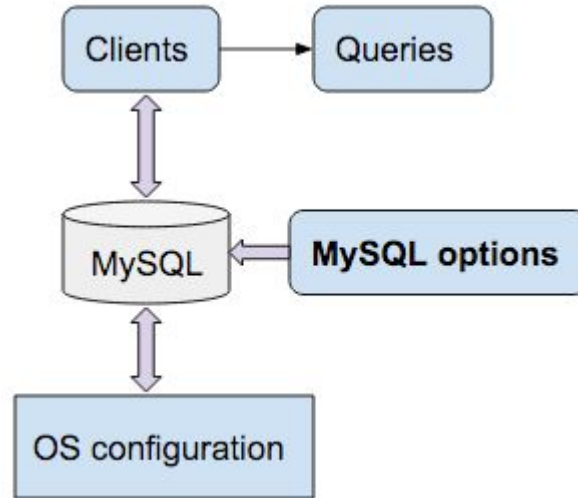
Identifying the source of slowness

- Check what is being slow



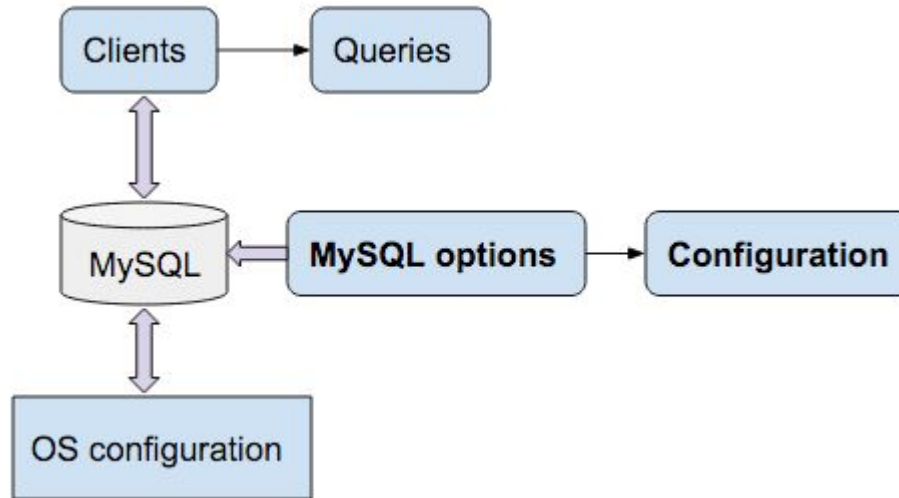
Identifying the source of slowness

- Check what is being slow



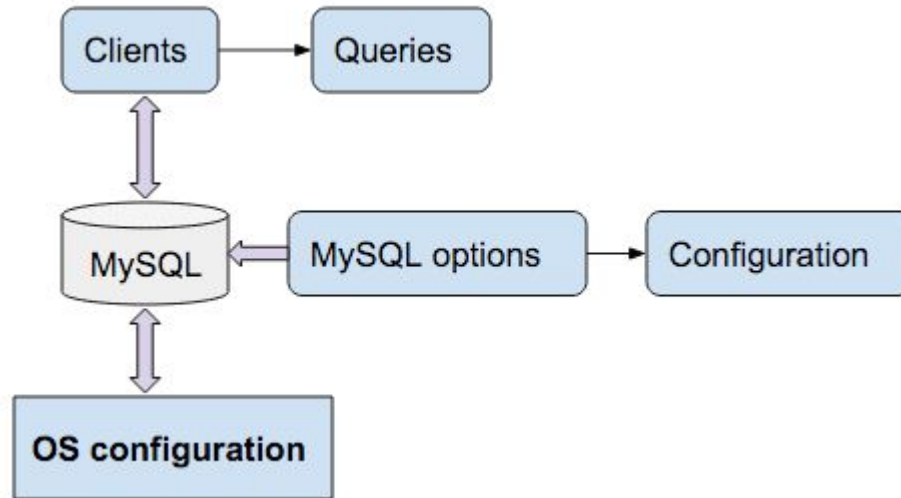
Identifying the source of slowness

- Check what is being slow



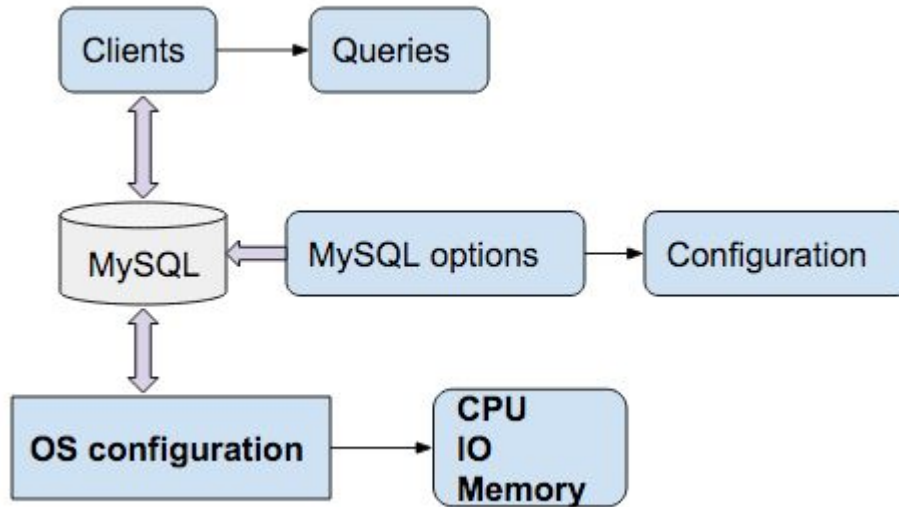
Identifying the source of slowness

- Check what is being slow



Identifying the source of slowness

- Check what is being slow

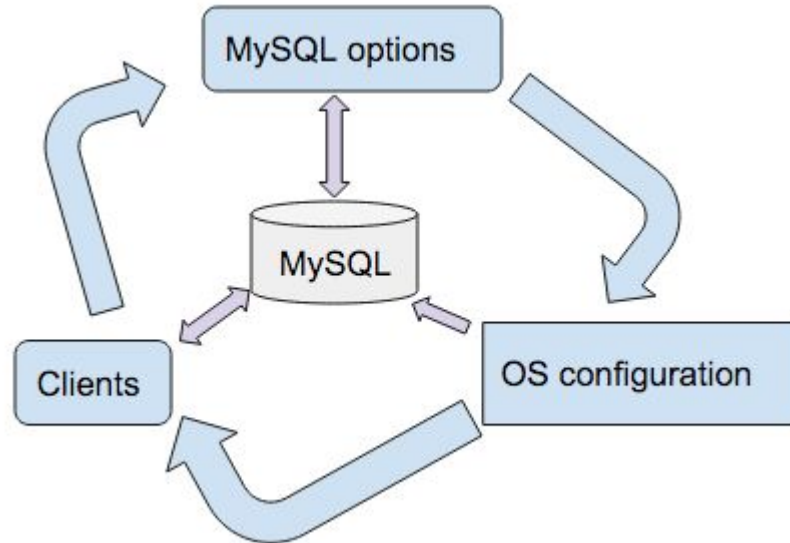


Identifying the source of slowness

- Iterate - No predefined order

Identifying the source of slowness

- Iterate - No predefined order



Identifying the source of slowness

- Check what's going on inside MySQL



Identifying the source of slowness

- Check what's going on inside MySQL
 - show processlist
 - *Check for long running queries*



Identifying the source of slowness

- Check what's going on inside MySQL
 - show processlist
 - *Check for long running queries*
 - show engine innodb status
 - *Check for open transactions*
 - *Check for semaphore contention*



Isolating and profiling problems

- Get response time of queries by scripting problems

Isolating and profiling problems

- Get response time of queries by scripting problems

```
<some code>  
get start time  
mysql_real_query (execute the query)  
get finish time
```

Isolating and profiling problems

- Get response time of queries by scripting problems

```
<some code>  
get start time  
mysql_real_query (execute the query)  
get finish time
```

- Capture queries

Isolating and profiling problems

- Get response time of queries by scripting problems

```
<some code>
get start time
mysql_real_query (execute the query)
get finish time
```

- Capture queries

- show full processlist

Isolating and profiling problems

- Get response time of queries by scripting problems

```
<some code>  
get start time  
mysql_real_query (execute the query)  
get finish time
```

- Capture queries

- show full processlist
- Parse slow query log with pt-query-digest

Isolating and profiling problems

- Analyze queries

Isolating and profiling problems

- Analyze queries

- explain extended <select....>
- explain format=JSON
- Execute queries and verify Handler% status counters

Isolating and profiling problems

- Analyze queries
 - explain extended <select....>
 - explain format=JSON
 - Execute queries and verify Handler% status counters
- Review schema

Isolating and profiling problems

- Analyze queries

- explain extended <select....>
- explain format=JSON
- Execute queries and verify Handler% status counters

- Review schema

- show create table: MyISAM or InnoDB

Isolating and profiling problems

- Analyze queries

- explain extended <select....>
- explain format=JSON
- Execute queries and verify Handler% status counters

- Review schema

- show create table: MyISAM or InnoDB
- show indexes from [table]

Identifying the source of slowness

- Check HW utilization - use `top` command

```
top - 08:23:20 up 3 days, 12:00, 5 users, load average: 9.32, 10.18, 11.06
Tasks: 270 total, 1 running, 269 sleeping, 0 stopped, 0 zombie
%Cpu(s): 7.9 us, 3.2 sy, 0.0 ni, 49.1 id, 39.5 wa, 0.0 hi, 0.3 si, 0.0 st
MiB Mem: 129160 total, 114496 used, 14663 free, 131 buffers
MiB Swap: 18484 total, 0 used, 18484 free, 8016 cached
```

```
  PID USER      PR  NI  VIRT  RES  SHR  S  %CPU  %MEM    TIME+  COMMAND
 6449 mysql    20   0  87.0g  85g  9.9m  S   104  67.7   7047:41  mysqld
25616 root     20   0  157m  34m 5636  S    31   0.0  312:46.84  percona-agent
   957 root     20   0     0     0     0  S     6   0.0   95:22.15  l2arc_feed
  3034 root      0 -20     0     0     0  S     6   0.0   10:32.63  z_rd_int_4
```

This may or may not be useful depending on the context

Identifying the source of slowness

- Check HW utilization - use `top` command

```
top - 08:23:20 up 3 days, 12:00, 5 users, load average: 9.32, 10.18, 11.06
Tasks: 270 total, 1 running, 269 sleeping, 0 stopped, 0 zombie
%Cpu(s): 7.9 us, 3.2 sy, 0.0 ni, 49.1 id, 39.5 wa, 0.0 hi, 0.3 si, 0.0 st
MiB Mem: 129160 total, 114496 used, 14663 free, 131 buffers
MiB Swap: 18484 total, 0 used, 18484 free, 8016 cached
```

```
  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
 6449 mysql    20   0  87.0g  85g  9.9m S 104 67.7 7047:41 mysqld
25616 root     20   0 157m  34m 5636 S  31  0.0 312:46.84 percona-agent
   957 root     20   0     0     0     0 S    6  0.0  95:22.15 l2arc_feed
 3034 root      0 -20     0     0     0 S    6  0.0 10:32.63 z_rd_int_4
```

This may or may not be useful depending on the context
104% CPU and 67.7% RAM - meh

Identifying the source of slowness

- Check HW utilization - use `top` command

```
top - 08:23:20 up 3 days, 12:00, 5 users, load average: 9.32, 10.18, 11.06
Tasks: 270 total, 1 running, 269 sleeping, 0 stopped, 0 zombie
%Cpu(s): 7.9 us, 3.2 sy, 0.0 ni, 49.1 id, 39.5 wa, 0.0 hi, 0.3 si, 0.0 st
MiB Mem: 129160 total, 114496 used, 14663 free, 131 buffers
MiB Swap: 18484 total, 0 used, 18484 free, 8016 cached
```

But 39.5% IO wait looks like too much

```

PID USER      PR  NI  VIRT  RES  SHR  S  %CPU  %MEM     TIME+  COMMAND
 6449 mysql    20   0  87.0g  85g  9.9m  S   104  67.7   70:47.41  mysqld
25616 root      20   0  157m  34m  5636  S    31  0.0   312:46.84  percona-agent
   957 root      20   0     0     0     0  S     6  0.0    95:22.15  l2arc_feed
 3034 root       0 -20     0     0     0  S     6  0.0   10:32.63  z_rd_int_4
```

This may or may not be useful depending on the context
104% CPU and 67.7% RAM - meh

Identifying the source of slowness

- Check HW utilization - use `top` command

```
top - 08:23:20 up 3 days, 12:00, 5 users, load average: 9.32, 10.18, 11.06
Tasks: 270 total, 1 running, 269 sleeping, 0 stopped, 0 zombie
%Cpu(s): 7.9 us, 3.2 sy, 0.0 ni, 49.1 id, 39.5 wa, 0.0 hi, 0.3 si, 0.0 st
MiB Mem: 129160 total, 114496 used, 14663 free, 131 buffers
MiB Swap: 18484 total, 0 used, 18484 free, 8016 cached
```

Check swap activity (good in this case)

```
  PID USER      PR  NI  VIRT  RES  SHR  S    %CPU  %MEM    TIME+  COMMAND
 6449 mysql    20   0  87.0g  85g  9.9m  S    104  67.7   7047:41  mysqld
25616 root     20   0   157m  34m  5636  S    31   0.0  312:46.84  percona-agent
   957 root     20   0     0     0     0  S     6   0.0   95:22.15  l2arc_feed
  3034 root      0 -20     0     0     0  S     6   0.0   10:32.63  z_rd_int_4
```


Identifying the source of slowness

- Check configurations

Identifying the source of slowness

- Check configurations
 - OS

Identifying the source of slowness

- Check configurations

- OS
 - *CPU*
 - *Memory*
 - *IO*

Identifying the source of slowness

- Check configurations

- OS
 - *CPU*
 - *Memory*
 - *IO*
- MySQL

Identifying the source of slowness

- Check configurations

- OS
 - *CPU*
 - *Memory*
 - *IO*
- MySQL
 - *my.cnf*

Identifying the source of slowness

- Check configurations
 - OS
 - *CPU*
 - *Memory*
 - *IO*
 - MySQL
 - *my.cnf*
- Percona toolkit to the rescue

Identifying the source of slowness

- Check configurations
 - OS
 - *CPU*
 - *Memory*
 - *IO*
 - MySQL
 - *my.cnf*
- Percona toolkit to the rescue
 - Use `pt-summary`

Identifying the source of slowness

- Check configurations
 - OS/Hardware
 - *CPU*
 - *Memory*
 - *IO*
 - MySQL
 - *my.cnf*
- Percona toolkit to the rescue
 - Use `pt-summary`
 - Use `pt-mysql-summary`

Isolating and profiling problems

- Check OS/hardware configuration
 - swappiness - by default in 60

Isolating and profiling problems

- Check OS/hardware configuration
 - swappiness - by default in 60
 - io scheduler - by default cfq

Isolating and profiling problems

- Check OS/hardware configuration
 - swappiness - by default in 60
 - io scheduler - by default cfq
 - cpu governors - by default power saving

Isolating and profiling problems

- Check OS/hardware configuration

- swappiness - by default in 60
- io scheduler - by default cfq
- cpu governors - by default power saving

None of these are good for DB workload

Isolating and profiling problems

- Check OS/hardware configuration

- swappiness - by default in 60
- io scheduler - by default cfq
- cpu governors - by default power saving

None of these are good for DB workload

And all of this is reported by pt-summary in one shot!

Isolating and profiling problems

- Check OS/hardware configuration

- swappiness - by default in 60
- io scheduler - by default cfq
- cpu governors - by default power saving

None of these are good for DB workload

And all of this is reported by pt-summary in one shot!

- Check numa - by default is not interleaved

Isolating and profiling problems

- Check MySQL configuration

Isolating and profiling problems

- Check MySQL configuration
 - InnoDB variables

Isolating and profiling problems

- Check MySQL configuration
 - InnoDB variables
 - *innodb_buffer_pool*

Isolating and profiling problems

- Check MySQL configuration
 - InnoDB variables
 - *innodb_buffer_pool*
 - *innodb_log_file_size*

Isolating and profiling problems

- Check MySQL configuration
 - InnoDB variables
 - *innodb_buffer_pool*
 - *innodb_log_file_size*
 - *innodb_flush_log_at_trx_commit*

Isolating and profiling problems

- Check MySQL configuration
 - InnoDB variables
 - *innodb_buffer_pool*
 - *innodb_log_file_size*
 - *innodb_flush_log_at_trx_commit*
 - *innodb_flush_method*

Isolating and profiling problems

- Check MySQL configuration

- InnoDB variables

- *innodb_buffer_pool*
 - *innodb_log_file_size*
 - *innodb_flush_log_at_trx_commit*
 - *innodb_flush_method*
 - *innodb_io_capacity/innodb_io_capacity_max*

Isolating and profiling problems

- Check MySQL configuration

- InnoDB variables

- *innodb_buffer_pool*
 - *innodb_log_file_size*
 - *innodb_flush_log_at_trx_commit*
 - *innodb_flush_method*
 - *innodb_io_capacity/innodb_io_capacity_max*
 - *Too fine tuning after this*

Isolating and profiling problems

- Check MySQL configuration
 - MyISAM variables

Isolating and profiling problems

- Check MySQL configuration
 - MyISAM variables (still using MyISAM? Shame on you!)

Isolating and profiling problems

- Check MySQL configuration
 - MyISAM variables (still using MyISAM? Shame on you!)
 - *key_buffer_size*

Isolating and profiling problems

- Check MySQL configuration
 - MyISAM variables (still using MyISAM? Shame on you!)
 - *key_buffer_size* - that's it

Isolating and profiling problems

- Check MySQL configuration
 - MyISAM variables (still using MyISAM? Shame on you!)
 - *key_buffer_size* - that's it
 - Query cache

Isolating and profiling problems

- Check MySQL configuration

- MyISAM variables (still using MyISAM? Shame on you!)
 - *key_buffer_size* - that's it
- Query cache - not even going to mention just turn it off

Isolating and profiling problems

- Check MySQL configuration

- MyISAM variables (still using MyISAM? Shame on you!)
 - *key_buffer_size* - that's it
- Query cache - not even going to mention just turn it off
- Per-thread buffers - main source of resource wasting

Isolating and profiling problems

- Check MySQL configuration

- MyISAM variables (still using MyISAM? Shame on you!)
 - *key_buffer_size* - that's it
- Query cache - not even going to mention just turn it off
- Per-thread buffers - main source of resource wasting
 - *temp_table_size|max_heap_table_size* - tune them defaults are conservative
 - *Few others* - keep defaults for them

Isolating and profiling problems

- Check MySQL configuration

- MyISAM variables (still using MyISAM? Shame on you!)
 - *key_buffer_size - that's it*
- Query cache - not even going to mention just turn it off
- Per-thread buffers - main source of resource wasting
 - *temp_table_size|max_heap_table_size - tune them defaults are conservative*
 - *Few others - keep defaults for them*

- All above is shown by pt-mysql-summary in one shoot!

- *Also show global variables - just harder to read*

Sample pt-mysql-summary

```
# Percona Toolkit MySQL Summary Report #####
      System time | 2016-04-09 06:30:39 UTC (local TZ: CDT -0500)
# Instances #####
Port  Data Directory          Nice OOM Socket
=====
      /var/lib/mysql          0   0   /var/lib/mysql/mysql.sock
# MySQL Executable #####
      Path to executable | /usr/sbin/mysqld
      Has symbols | No
# Report On Port 3306 #####
      User | root@localhost
      Time | 2016-04-09 01:30:38 (CDT)
      Hostname | cur-master
      Version | 5.6.21-70.0-log Percona Server (GPL), Release 70.0, Revision 688
      Built On | Linux x86_64
      Started | 2016-04-08 23:31 (up 0+01:59:19)
      Databases | 6
      Datadir | /var/lib/mysql/
      Processes | 2 connected, 1 running
      Replication | Is a slave, has 0 slaves connected
      Pidfile | /var/lib/mysql/cur-master.pid (exists)
...

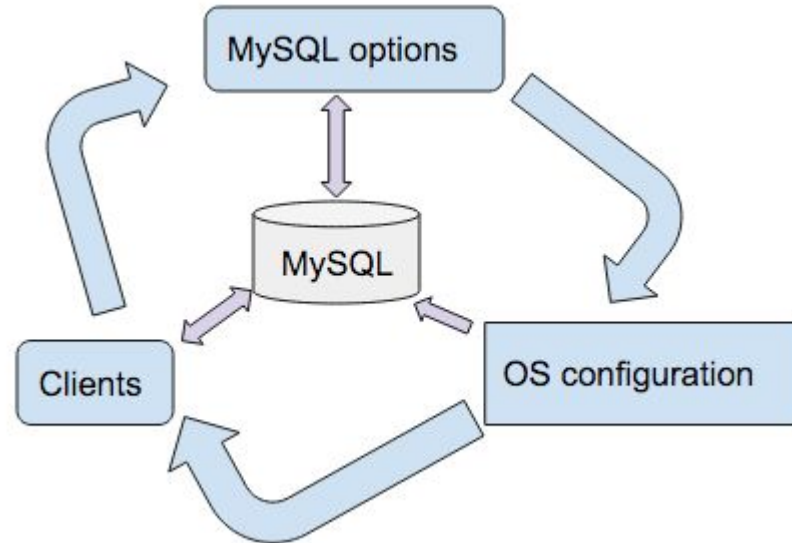
```


Sample pt-mysql-summary

```
...
# InnoDB #####
      Version | 5.6.21-70.0
      Buffer Pool Size | 128.0M
      Buffer Pool Fill | 90%
      Buffer Pool Dirty | 0%
      File Per Table | ON
      Page Size | 16k
      Log File Size | 2 * 48.0M = 96.0M
      Log Buffer Size | 8M
      Flush Method |
      Flush Log At Commit | 1
      XA Support | ON
      Checksums | ON
      Doublewrite | ON
      R/W I/O Threads | 4 4
      I/O Capacity | 200
      Thread Concurrency | 0
      Concurrency Tickets | 5000
      Commit Concurrency | 0
      Txn Isolation Level | REPEATABLE-READ
      Adaptive Flushing | ON
      Adaptive Checkpoint |
      Checkpoint Age | 0
      InnoDB Queue | 0 queries inside InnoDB, 0 queries in queue
      Oldest Transaction | 0 Seconds
      History List Len | 559
      Read Views | 0
      Undo Log Entries | 0 transactions, 0 total undo, 0 max undo
      Pending I/O Reads | 0 buf pool reads, 0 normal AIO, 0 ibuf AIO, 0 preads
      Pending I/O Writes | 0 buf pool (0 LRU, 0 flush list, 0 page); 0 AIO, 0 sync, 0 log IO (0 log, 0 chkp); 0 pwrites
      Pending I/O Flushes | 0 buf pool, 0 log
      Transaction States | 2xnot started
```

Identifying the source of slowness

- Iterate - Again and again and again



Best practices for troubleshooting

- Monitor your systems
 - PMM - VividCortex - DataDog just pick one
 - Read patterns continuously
- Use slow query log
 - `long_query_time`
 - `log_slow_verbosity`
 - Run periodic query reviews

Best practices for troubleshooting

- When in trouble
 - Use pt-stalk
 - Iterate through solutions

Best practices for troubleshooting

Questions?

Best practices for troubleshooting

Thanks!