MySQL Performance by Mayur Purushvani

MySQL Configurations :

DISTROS :

MySQL

Percona (Beifit is : EXTENDED LOGS)

MariaDB

WeScaleSQL (Collaboration by Google, FB, LinkedIn, Twitter)

MySQL Configuration Tuning :

PRACTICAL OF SOME COMMANDS :

First Go to C:/Xampp, Open cmd, Connect your mysql server with 'mysql –u root –p'

- Select @@max_connections ; [This will show you a maximum connections supported in your mysql server]
- Set global max_connections = 500; [This will set your connections globaly in your mysql server]
- Again you can check 'select @@max_connections ;
- Show variales ; [This will show you all the variables which is declared I your mysql configuration file]
- Show vaiables like '%log%'; [This will show only the log variables from your configuration file]
- Show vaiables like '%version%';
- Show vaiables like '%max%';
- Show vaiables like '%error%';

System Variables :

- MySQL has lots of variables that you can consider to change.
- Some variables are dynamic which means they can be set using the SET statement.
- Others require a server restart, after they are set in the configuration file (e.g. /etc/my.cnf, etc/mysql/my.cnf).

- Sort_buffer_size :

  This variable controls how large your filesort buffer is.

  If this value is large, you should consider increasing the value of the sort_buffer_size system variable. Otherwise, take it to the moderate limit that you need. If you set this too low or if you have large queries to process, the effect of sorting your rows can be slower than expected because data is retrieved randomly doing disk dives.

- Read_buffer_size :
  it allocates a read buffer and it determine the size of buffer.
  table allocates a buffer of this size (in bytes) for each table it scans.

- Read_rnd_buffer_size :
  the rows are read through this buffer to avoid disk seeks.
  Setting the variable to a large value can improve ORDER BY performance by quite a lot.

- Join_buffer_size :

  By default value is 256K. The minimum size of the buffer that is used for plain index scans, range index scans, and joins that do not use indexes and thus perform full table scans.

- max_heap_table_size :

  This is the maximum size in bytes for user-created MEMORY tables are permitted to

  grow. This is helpful when your application is dealing with MEMORY storage engine

  tables.

- table_open_cache_instances:

  Setting this variable would help improve scalability, and of course, performance which

  would reduce contention among sessions.

- Table_definition_cache :
  The CREATE TABLE are cached to speed up opening of tables and only one entry per table.

- Max_allowed_packet :
  Pre-connection maximum size of an SQL query.

- Skip_name_resolve :
  You may optionally start mysqld daemon, passing –skip-name-resolve option to enable this.

- Max_connections :
  This is the number of permitted connections for your MySQL server.

- Thread_cache_size :
  This is the cache to prevent excessive thread creation. When a client disconnects, the client's threads are put in the cache if there are fewer than thread_cache_size threads there.

- Query_cache_size :
  This variable must be set to 0 along with query_cache_type = 0 as well to turn it off.

STORAGE ENGINE – InnoDB

- innodb_buffer_pool_size

  this can be monitored using our Dashboards -> InnoDB Metrics -> InnoDB Buffer Pool Pages graph. You may also monitor this with SHOW GLOBAL STATUS using the variables Innodb_buffer_pool_pages*.

- innodb_buffer_pool_instances

  For your concurrency workload, setting this variable can improve concurrency and reduce contention as different threads of read/write to cached pages. Minimum innodb_buffer_pool_instances should be lie between 1 (minimum) & 64 (maximum).

- innodb_log_file_size

  The combined size of log files (innodb_log_file_size * innodb_log_files_in_group) cannot exceed a maximum value that is slightly less than 512GB.

- innodb_log_buffer_size

- it uses the value of innodb_log_buffer_size having a default value of 8MiB. This is beneficial especially for large transactions as it does not need to write the log of changes to disk before transaction commit.

- innodb_flush_log_at_trx_commit

  innodb_flush_log_at_trx_commit is set to 1 the log buffer is flushed on every transaction commit to the log file on disk and provides maximum data integrity but it also has performance impact.

- innodb_thread_concurrency

  improvements to the InnoDB engine, it is recommended to allow the engine to control the concurrency by keeping it to default value (which is zero).

- innodb_file_per_table

  This is usually recommended as it avoids having a huge shared tablespace and as it allows you to reclaim space when you drop or truncate a table.

- innodb_stats_on_metadata

  This attempts to keep the percentage of dirty pages under control, and before the Innodb plugin, this was really the only way to tune dirty buffer flushing.

- innodb_io_capacity

  It can quickly starve data reads and writes to the transaction log if you set this too high.

- innodb_write_io_threads

  Controls how many threads will have writes in progress to the disk.

- innodb_adaptive_flushing

  Specifies whether to dynamically adjust the rate of flushing dirty pages in the InnoDB

  buffer pool based on the workload

- innodb_dedicated_server

The downside only is that you cannot have the feasibility to apply your desired values on the detected variables mentioned.

MyISAM :

- Key_buffer_size :

  The default for key_buffer_size can probably be decreased unless you are using MyISAM productively as part of your application

Others :

- Show_query_log :

  This variable can help you out analyze slow performing queries.

- Long_query_time :

  If the slow query log is enabled, the query is logged to the slow query log file. This value is measured in real time, not CPU time.

- Sync_binlog :

  By default (>=5.7.7), this is set to 1 which means it will sync to disk before transactions are committed.

- To Dump/Restore buffer pool :

  set both variables innodb_buffer_pool_dump_at_shutdown = ON and innodb_buffer_pool_load_at_startup = ON.

Reference :

MySQL Performance Cheat Sheet | Severalnines

MySQL Performance Tuning: Part 1. Configuration (Covers MySQL 5.7) - YouTube

5 Mysql Video Tutorial System Variables max connections - YouTube