

19

MySQL System Variables

A variety of settings can be used to customize the operation of MySQL. Settings fall into two main categories:

- Operating System Environment variables
- MySQL variables, which can be set via the command line or in a configuration file.

Variables defined in multiple places follow obey the following order of precedence:

1. Items defined on the command line take precedence over configuration files and environment variables.
2. Items defined in configuration files take precedence over environment variables.

For more information on the use of configuration files to set system variables and command line options, refer to Chapter 5, “Database Administration.”

Environment Variables

The following variables are specific to MySQL programs. They may be defined in the current shell or as part of a shell script. To set a variable for the MySQL daemon (*mysqld*), define the variable in the *safe_mysqld* script that is used to start the daemon or define the variables in the MySQL configuration file.

MYSQL_DEBUG

The debug trace level for the program. This option can be used with any MySQL program. There are a wide variety of debug trace options which are documented in the MySQL documentation at <http://www.mysql.com/documentation> in the section on “Debugging a MySQL server” and “Debugging a MySQL client.”

MYSQL_HOST

The hostname used to connect to a remote MySQL database server. This option can be used with any of the MySQL client programs (*mysql*, *mysqlshow*, *mysqladmin*, etc.). This is equivalent to the `--host` command line option.

MYSQL_HISTFILE

The location of the MySQL history file, used by the *mysql* client. By default this is `“$HOME/.mysql_history”`. There is no equivalent command line options.

MYSQL_PWD

The password used to connect to the MySQL database server. This option can be used with any of the MySQL client programs, and is equivalent to the `--password` command line option.

Be careful where you put your passwords. A common use for environment variables is to set them within scripts. Of course, setting this particular variable in a script would make your password visible to anyone who can run the script. Even setting the variable manually on the command line exposes it to the superuser and any else who has the ability to examine the system memory.

MYSQL_TCP_PORT

When used with a client program, this is the TCP port on a remote machine used to connect to the MySQL database server. When used with *mysqld*, this is the port used to listen for incoming connections. This is equivalent to the `--port` command line option.

MYSQL_UNIX_PORT

When used with a client program, this is the Unix socket file used to connect to the MySQL database server. When used with *mysqld*, this is the name of the Unix socket file created that allows local connections. This is equivalent to the `--socket` command line option.

In addition, the MySQL programs use the following environment variables that are routinely set as part of the Unix environment.

EDITOR

VISUAL

The path of the default editor. The *mysql* program uses this program to edit SQL statements if a `\e` or `edit` command is encountered.

HOME

The home directory of the current user.

LOGIN

LOGNAME

USER

The username of the current user. On Windows, you can use the `USER` environment variable to indicate the default user when connecting to *mysqld*.

PATH

The list of directories used to find programs.

POSIXLY_CORRECT

If this variable is defined, no special processing is done on command line options. Otherwise, command line options are reordered so that extended options can be used. This variable can be used with any MySQL program.

TMP

TMPDIR

The directory in which temporary files are kept. If this variable is not defined ‘/tmp’ is used.

TZ

The time zone of the local machine.

UMASK

The umask used when creating new files.

UMASK_DIR

The umask used with creating new directories. This is ANDed with UMASK.

The following environment variables influence the configuration of your MySQL build. Use these when building MySQL from source.

CCX

Indicates which C++ compiler to use. This is used by the configure script.

CC

Indicates which C compiler to use. Used by the configure script.

CFLAGS

Indicates which C compiler flags to use.

CCXXFLAGS

Indicates which C++ compiler flags to use.

MySQL System Variables

In this section we document the system variables for the MySQL server mysqld. The client programs also have variables and command line options that control their operation. These are documented with the individual clients and utilities in Chapter 20, “MySQL Programs and Utilities”. You also may refer to the command line reference for mysqld in Chapter 20

Many of these options are supplied via the `-O` or `-set-variable` command line option to mysqld. Some are specified using specific command line options. Unless otherwise noted these are specified using the `-O` or `-set-variable`.

The current values of these options can be determined by using the `show variables SQL` command or with the `mysqladmin variables` command.

ansi_mode

This indicates whether the ANSI mode is on or off. This is ON if mysqld is started with the `--ansi` option.

back_log

The number of TCP connections that can be queued at once. When MySQL receives a connection request, it starts a new thread to handle that connection. There is a short

lag between the time the request is received and the session thread is created. If a large number of connection requests hit the MySQL server at the same time, they are queued up. This should only be increased if you expect a high-number of simultaneous connection requests. Another thing to note is that the operating system typically imposes a cap on `back_log`. Setting `back_log` higher than the operating system limit will be ineffective.

`basedir`

The location of your MySQL installation. All other paths are usually resolved relative to this. Set with the `--basedir` command line option.

`bdb_cache_size`

The size of the buffer used to cache BDB rows and indexes. If you are not using BDB tables, use the `--skip-bdb` option so that memory is not wasted on this cache.

`bdb_log_buffer_size`

???

`bdb_home`

The location of your BDB tables. Set with the `--bdb-home` command line option.

`bdb_max_lock`

The maximum number of locks allowed per BDB table. By default this is set to 1000. This should be increased if you get errors like “bdb: Lock table is out of available locks” or “Got error 12 from ...”.

`bdb_logdir`

The location of your BDB log files. Set with the `--bdb_logdir` command line option.

`bdb_shared_data`

Indicates whether you are using shared data for BDB tables. This is ON if `mysqld` is started with the `--bdb_shared_data` option.

`bdb_tmpdir`

The location of your BDB temporary files. Set with the `--bdb-tmpdir` command line option.

`binlog_cache_size`

The size of the cache for storing binary log transactions. If you use large, multi-statement transactions, this can be increased to improve performance.

`character_set`

The default character set. This is set with the `--default-character-set` command line option.

`character_sets`

The supported character sets. The available character sets are determined at compile time. The `--with-charset` and `--with-extra-charsets` options to the configure script. See Chapter 3 for more information on installing character sets.

`concurrent_inserts`

If this is ON, MySQL will allow INSERTs on MyISAM tables concurrently with SELECTs. The default is ON. This is turned off with the `--safe` command line option or the `--skip-new` command line option.

`connect_timeout`

The number of seconds the *mysqld* server waits for a connect packet before responding with “Bad handshake”.

`datadir`

The location of the database files. Set with the `--datadir` command line option. By default this is `<basedir>/data`.

`delay_key_write`

If this is ON, MySQL will not flush the key buffer on every index update. It will only flush this buffer on table close. This only applies to tables created with the `delay_key_write` CREATE TABLE option. By default this is ON. You can disable this with the `--skip-new` or `--safe-mode` command line options.

This increases the performance of key updates considerably, however we recommend automatic checking of all tables with `myisamchk --fast --force`.

If you use the `--delay-key-write-for-all-tables` command line option, MySQL will treat all tables as if they had been created with the `delay_key_write` option.

`delayed_insert_limit`

Causes the INSERT DELAYED handler to check whether there are any SELECT statements pending after inserting *delayed_insert_limit* rows. If so, the handler allows the statements to execute before continuing.

`delayed_insert_timeout`

How long an INSERT DELAYED thread should wait for INSERT statements to finish before terminating.

`delayed_queue_size`

How big a queue (in rows) should be allocated for handling INSERT DELAYED. If the queue becomes full, any client that does an INSERT DELAYED must wait until there is room in the queue again.

`flush`

If this is ON, MySQL will flush all changes disk after every SQL command. Normally MySQL writes the data and lets the operating system handle flushing. This is enabled with `--flush` command line option.

`flush_time`

If set, all tables are closed then every `flush_time` seconds to free resources and synchronize changes to disk. This is only recommended for use on Windows 95/98 or on systems with very little resources.

`have_bdb`

If this is YES, *mysqld* supports Berkeley DB tables (which is determined at compile time with the `--with-dbd` configure option). NO means that MySQL was not compiled with support for this option. DISABLED means the server was started with the `--skip-bdb` command line option.

`have_innodb`

If this is YES, *mysqld* supports Berkeley DB tables (which is determined at compile time with the `--with-dbd` configure option). NO means that MySQL was not compiled with support for this option. DISABLED means the server was started with the `--skip-bdb` command line option.

`have_raid`

If this is YES, mysqld has support for RAID compiled in. NO means that the server was compiled without support for RAID.

`have_ssl`

If this is YES, mysqld supports SSL encryption on the client/server protocol. This is enabled at compile time with the `--with-ssl` configure option. NO means that MySQL was compiled without support for this option.

`init_file`

The name of the file specified with the `--init-file` command line option. mysqld will execute the SQL commands included in this file during startup.

`join_buffer_size`

The size of a buffer used when performing full table joins (i.e. joins that do not use indexes). This buffer is allocated one time for each full join between two tables. Normally the best way to increase performance of full joins is to add indexes (see Chapter 6 for more details). However, if adding indexes is not possible, increasing this value will improve the performance of full joins.

`key_buffer_size`

The size of a buffer shared by all threads to store index blocks. Increasing this buffer size will improve performance of index reads and writes. Increase this to as much as you can possibly afford. For example, 64Mb on a dedicated MySQL server with 256 Mb of physical memory is common. Be warned, however, that setting this too high (i.e. greater than 50% of physical memory) can cause your operating system to start paging heavily, which will negatively impact the performance of your server down, sometimes severely.

`language`

The language used for error messages. This is specified with the `--language` command line option.

`large_file_support`

This is ON if mysqld was compiled support for big files.

`locked_in_memory`

This is ON if mysqld was locked in memory with the `--memlock` command line option.

`log`

This is ON if query logging is enabled. This is enabled with the `--log` command line option. See Chapter 5 for more information on the query log.

`log_update`

This is ON if update logging is enabled. This is enabled with the `--log-update` command line option. See Chapter 5 for more information on the update log.

`log_bin`

This is ON if binary logging is enabled. This is enabled with the `--log-bin` command line option.

`log_slave_updates`

Turn this ON if updates from the slave should be logged.

`long_query_time`

If set, the `slow_queries` counter is incremented each time a query takes more than *long_query_time* seconds. If you have enabled the slow query log with the `--log-slow-queries` command line option, each slow query will be logged in the slow query log. See Chapter 5 for more information about the slow query log.

`lower_case_table_names`

If set to 1, tables names are stored in lowercase on the file system. This will allow for case-insensitive table access on Unix.

`max_allowed_packet`

The maximum size of one packet. The message buffer is initialized to `net_buffer_length` bytes, but is allowed to grow up to `max_allowed_packet` bytes.

By default, this is small to catch big packets that are possibly erroneous. However, if you are using BLOB columns, you will need to increase this value to the size of the largest BLOB you use.

This is limited to 16M in the current protocol.

`max_binlog_cache_size`

Increase this if a multi-statement transaction returns the error “Multi-statement transaction required more than ‘`max_binlog_cache_size`’ bytes of storage”.

`max_binlog_size`

Rotate the the binary log when reaches the size specified by `max_binlog_size`. The value must be greater than 1024 bytes, but no more than 1Gb. The default is 1Gb.

`max_connections`

The number of simultaneous clients allowed by `mysqld`. Upping this value increases the number of file descriptors needed by `mysqld`. See the `open_files_limit` variable for more information.

`max_connect_errors`

If set, the server blocks further connections from a remote host when the number of interrupted connections from that host exceeds *max_connect_errors*. You can unblock a host with the command `FLUSH HOSTS`.

`max_delayed_threads`

Start no more than this number of threads to handle `INSERT DELAYED`. If a client tries to use `INSERT DATA` to insert new data after this limit is reached, the request is handled as if the `DELAYED` attribute was not specified.

`max_heap_table_size`

`mysqld` disallows creation of heap tables larger than this size.

`max_join_size`

When `mysqld` estimates a join will read more than `max_join_size` records, it returns an error. Use this variable if you need to protect against ill-formed queries.

`max_sort_length`

The maximum number of characters to examine when sorting a BLOB or VARCHAR field. Only the first `max_sort_length` bytes of each value are used for the sort operation.

`max_user_connections`

The maximum number of active connections for a single user. A value of 0 indicates no limit.

`max_tmp_tables`

Maximum number of temporary tables a client can keep open at the same time. This option doesn't do anything yet, but it is going to control the maximum number of temporary tables a client can keep open at the same time.

`max_write_lock_count`

After this many write locks, allow some read locks to run in between.

`myisam_recover_options`

This is used to control how MySQL handles MyISAM tables that are marked as crashed or weren't opened properly. This is specified with the `--myisam-recover` command line option. Refer to the `mysqld` documentation in Chapter 20 for more details on option.

`myisam_sort_buffer_size`

The size of a buffer that is allocated when sorting an index during a `REPAIR` operation or when creating indexes with `CREATE INDEX` or `ALTER TABLE`.

`myisam_max_extra_sort_file_size`

If the temporary file created for fast index creation would be `myisam_max_extra_sort_file_size` megabytes bigger than using the key cache, `mysqld` will use the key cache instead. This is primarily to force the use of the slower key cache method when creating long character keys in large tables.

Note that this variable is specified in megabytes.

`myisam_max_sort_file_size`

If the temporary file created for recreating an index is greater than `myisam_max_sort_file_size` megabytes, `mysqld` will use the key cache instead. Note that this variable is specified in megabytes.

`net_buffer_length`

The client/server communication buffer is reset to this size between queries. If you have very little memory, you can set this to the expected length of SQL statements from clients. If statements exceed this length, this buffer is automatically enlarged, up to `max_allowed_packet` length.

`net_read_timeout`

The number of seconds to wait for more data from a connection before aborting the read operation. This only applies when we expect data from a connection, otherwise the timeout is specified by `net_write_timeout`.

`net_retry_count`

Retry `net_retry_count` times if a read on a communication port is interrupted. On FreeBSD, this value should be high since internal interrupts are sent to all threads.

`net_write_timeout`

The number of seconds to wait for a block to be written to a connection before aborting the write.

`open_files_limit`

If set to a non-zero value, `mysqld` will reserve `open_files_limit` file descriptors with the `setrlimit()` system call.

If this is zero, then the greater of `max_connections*5` and `max_connections + table_cache*2` will be reserved.

Increase this if you receive a 'Too many open files' error from `mysqld`.

`pid_file`

The path to the pid file used by `safe_mysqld`. This is specified with the `--pid-file` command line option.

`port`

The port on which `mysqld` is listening for TCP/IP connections. This is specified using the `--port` command line option.

`protocol_version`

The protocol version used by the MySQL server. The `--old-protocol` command line option will enable the old MySQL protocol.

`query_buffer_size`

The initial size of the query buffer. If most of your queries are long, like when inserting blobs, increase this value.

`record_buffer`

The size of a buffer used to perform a sequential scan on tables (i.e. for non-index retrievals). Increase this if you perform lots of non-index reads from tables.

`record_rnd_buffer`

When reading rows in sorted order after a sort, they are read through this buffer to avoid disk reads. If not set, the size of the `record_rnd_buffer` is the same as `record_buffer`.

`safe_show_databases`

Don't show databases to a user unless they have been granted database or table privileges on that table. This can improve security, as it will prevent users from seeing databases they don't have access too.

`server_id`

The value of the `--server_id` command line option.

`skip_locking`

This is ON if system locking has been disabled. The `--skip-locking` command line option controls this. When this is OFF, you must shut down the server to run `isamchk` or `myisamchk`.

`skip_networking`

This is ON if `mysqld` only allows local (socket) connections. The `--skip-networking` command line option controls this.

`skip_show_databases`

If this is ON, `mysqld` will not allow users to run `SHOW DATABASES` if they don't have the `PROCESS_PRIV` privilege.

`slave_read_timeout`

The number of seconds to wait for more data from a master/slave connection before aborting the read.

`slow_launch_time`

If creating a thread takes longer than `slow_launch_time` seconds, the `slow_launch_threads` counter is incremented.

socket

The path to the UNIX socket used by mysqld. This is specified with the --socket command line option.

sort_buffer

The size of the buffer used when performing sorts on retrieved data. Increasing this can speed up performance for queries that use ORDER BY or GROUP BY statements.

table_cache

The maximum number of tables the database server can have open at once. Increasing this value increases the required number of file descriptors. See the max_open_files variable for more information. Make sure that your operating system can support the number of open file descriptors by the table_cache setting. If it can't MySQL may refuse connections, fail to perform queries, etc.

table_type

The default table type.

thread_cache_size

This specifies the number of threads to keep in a cache this is used to recycle client threads. Increasing this number can improve performance if you have a high number of connections.

thread_concurrency

On Solaris, mysqld will invoke the thr_setconcurrency() system call with the value of this variable. This will give the operating system a hint about the desired number of threads to be run concurrently.

thread_stack

The stack size for each thread. The default is large enough for most normal operation.

timezone

The timezone from the server.

tmp_table_size

The maximum size of temporary tables used by the database server. If an in-memory temporary table exceeds this size, MySQL will convert it to an on-disk MyISAM table. Increasing this value can improve your performance if you do lots of GROUP BY queries and have memory available.

tmpdir

The directory used for temporary files and tables. This specified using the --tmpdir command line option.

version

The version number of the mysqld server. This variable cannot be changed.

wait_timeout

The number of seconds the server waits for activity on a connection before closing