

mysql配置文件详解

以下是MySQL数据库服务器配置文件my.ini的详细配置。应用场合是InnoDB引擎，2核CPU， 32位SUSE。

```
[client]
#password = your_password
port = 3306
socket = /tmp/mysql.sock
# Here follows entries for some specific programs
# The MySQL server
[mysqld]
port = 3306
socket = /tmp/mysql.sock
#避免外部锁定(默认是ON).
skip-external-locking
#索引缓冲区(仅作用于MYISAM表和临时表)，决定了数据库索引处理的速度，改为10M
key_buffer_size = 16K
#通信缓冲区的最大长度。包或任何生成的/中间字符串的最大大小。可改为更大的比如32M
max_allowed_packet = 1M
#所有线程打开的表的数目(一个表使用2个文件描述符)，表数量多，就要大一些。增大该值可以增加mysqld需要的文件描述符的数量。可以检查
Opened_tables(已经打开的表的数量)状态变量来检查你是否需要增加表缓存。可改为2048
table_open_cache = 4
#每个排序线程分配的缓冲区的大小。增加该值可以加快ORDER BY或GROUP BY操作，可改为5M
sort_buffer_size = 64K
#每个线程连续扫描时为扫描的每个表分配的缓冲区的大小,如果进行多次连续扫描，可能需要增加该值，可改为1M
read_buffer_size = 256K
#当排序后按排序后的顺序读取行时，则通过该缓冲区读取行，避免搜索硬盘。将该变量设置为较大的值可以大大改进ORDER BY的性能。但是，这是为每个客户端分配的缓冲区，因此你不应将全局变量设置为较大的值。相反，只为需要运行大查询的客户端更改会话变量。可改为2M
read_rnd_buffer_size = 256K
#在查询之间将通信缓冲区重设为该值。如果语句超出该长度，缓冲区自动扩大，直到max_allowed_packet字节。
net_buffer_length = 2K
#每个线程的堆栈大小
#thread_stack = 128K      #thread_stack = 194K
#mysql服务器安装目录
basedir=f:/server/mysql
#mysql数据存储目录
datadir=f:/server/mysql/data
#mysql数据库编码（防止命令行下乱码）
character-set-server = gbk

#主从同步配置。主机是1。备机从2开始。
server-id = 1
#启动数据库更新二进制日志记录，日志文件名前缀为mysql-bin
#log-bin=mysql-bin
# binary logging format - mixed recommended
#binlog_format=mixed
# Causes updates to non-transactional engines using statement format to be
# written directly to binary log. Before using this option make sure that
# there are no dependencies between transactional and non-transactional
# tables such as in the statement INSERT INTO t_myisam SELECT * FROM
# t_innodb; otherwise, slaves may diverge from the master.
#binlog_direct_non_transactional_updates=TRUE
# Uncomment the following if you are using InnoDB tables
#InnoDB 不会自己建立目录，必须自己使用操作系统命令建立相应的目录。检查你的 MySQL 服务程序在datadir 目录里 有足够的权限建立文件(mysql
用户组拥有)。
#这是InnoDB表的目录共用设置。如果没有设置，InnoDB 将使用MySQL的 datadir 目录为缺省目录。如果设定一个空字符串,可以在 innodb_data_file_path
中设定绝对路径。
#innodb_data_home_dir = C:\\mysql\\data\\
```

#这里应当预先创建好10个2048M的文件，目前还没有创建 innodb_data_file_path =

```
ibdata1:2000M;ibdata2:2000M;ibdata3:2000M;ibdata4:2000M;ibdata5:2000M;ibdata6:2000M;ibdata7:2000M;ibdata8:2000M;ibdata9:2000M;ibdata10:2000M:autoext  
end
```

```
#innodb_data_file_path = ibdata1:10M:autoextend
```

#InnoDB 日志文件的路径。如果没有明确指定将默认在 MySQL 的 datadir 目录下建立两个 5 MB 大小的ib_logfile0和ib_logfile1文件。

#将日志文件与数据文件分别放在不同的物理硬盘中对提高性能通常是很有益的

```
#innodb_log_group_home_dir = C:\mysql\data\
```

#表和索引数据的内存缓冲区，越大性能越高。但不能超过物理内存的50%。若64位OS，该值可以更大。32位OS受2G内存的限制，不能将内存使用设置太高，glibc 会把进程堆增长到线程堆栈之上，这将会使服务器崩溃。下面的接近或超过于 2G 将会很危险：innodb_buffer_pool_size + key_buffer + max_connections * (sort_buffer_size + record_buffer_size + binlog_cache_size) + max_connections * 2 MB

#每个线程将使用 2MB(MySQL AB 二进制版本为 256 KB)的堆栈，在最坏的环境下还会使用 sort_buffer_size + record_buffer_size 的附加内存。可改为 512M

```
#innodb_buffer_pool_size = 16M
```

#主要用来存储表结构和数据字典，表越多要求内存就越大，可改为16M

```
#innodb_additional_mem_pool_size = 2M
```

#日志组中的每个日志文件的大小(单位 MB)。如果 n 是日志组(innodb_log_files_in_group)中日志文件的数目，那么理想的数值为 1M 至缓冲池

(innodb_log_buffer_size)大小的 1/n。较大的值，可以减少刷新缓冲池的次数，从而减少磁盘 I/O。但是大的日志文件意味着在崩溃时需要更长的时间来恢复数据。可改为256M

```
#innodb_log_file_size = 5M
```

#InnoDB 将日志写入日志磁盘文件前的缓冲大小。理想值为 1M 至 8M。大的日志缓冲允许事务运行时不需要将日志保存入磁盘而只到事务被提交(commit)。因此，如果有大的事务，设置大的日志缓冲可以减少磁盘I/O。

```
#innodb_log_buffer_size = 8M
```

#1表示每次事务结束都写日志并刷新磁盘；2表示每次事务写日志但不刷新磁盘(每秒刷新)；0(默认值)表示每秒写日志并刷新磁盘。0表示最多丢失1秒的数据，但性能最好。

```
#innodb_flush_log_at_trx_commit = 1
```

#InnoDB行锁导致的死锁等待时间(默认值是50S)，可改为30

```
#innodb_lock_wait_timeout = 50 [mysqldump]
```

```
quick
```

```
max_allowed_packet = 16M
```

```
[mysql]
```

```
no-auto-rehash
```

Remove the next comment character if you are not familiar with SQL

```
#safe-updates
```

```
[myisamchk]
```

```
key_buffer_size = 8M
```

```
sort_buffer_size = 8M
```

```
[mysqlhotcopy]
```

```
interactive-timeout
```

以下设置仅供参考：

```
[mysqldump]
```

```
quick
```

```
max_allowed_packet = 16M
```

```
[mysql]
```

```
no-auto-rehash
```

Remove the next comment character if you are not familiar with SQL

```
#safe-updates
```

```
[isamchk]
```

```
key_buffer = 20M
```

```
sort_buffer_size = 20M
```

```
read_buffer = 2M
```

```
write_buffer = 2M
```

```
[myisamchk]
```

```
key_buffer = 20M
```

sort_buffer_size = 20M

read_buffer = 2M

write_buffer = 2M

[mysqlhotcopy]

interactive_timeout

参考：<http://blog.csdn.net/zhu1289303556/article/details/50823100>