

HP OpenSource ブループリント

MySQL Server 5.0 設定手順書

Red Hat Enterprise Linux 4

ver 2.0



はじめに

MySQL サーバには、様々なオプションを設定することが可能です。これらのオプションは MySQL サーバを起動するときに引数として指定することも出来ませんが、通常は MySQL 設定ファイルに設定して利用します。しかし、現実には多くのオプションがあるために、それぞれのオプションに適切な値を設定することは大変な作業となります。

本書では、Red Hat Enterprise Linux 4 上でオープンソースデータベースである MySQL Server 5.0 を使用する典型的な環境や使用条件を想定し、それぞれの環境や条件に合った設定ファイルを示すと共に、設定ファイルの利用方法についても説明します。また、本書で仮定している環境とは異なる環境で動作させる場合のために、設定ファイルの修正方法についても一部説明しています。

付録には設定ファイルの代表的な属性の意味についてもまとめました。

本書の範囲

本書では、Red Hat Enterprise Linux 4 x86 版(32bit)と AMD64/Intel64 版(64bit)上で MySQL Server 5.0 を動作させる場合について説明します。

また、本書で説明する MySQL 設定ファイルは、1 台のサーバマシン上で1つの MySQL サーバを実行する場合の設定ファイルです。1 台のサーバマシン上で複数の MySQL サーバを利用する場合や、レプリケーション構成、MySQL クラスタ構成の設定ファイルは範囲外です。

また本書では MySQL サーバのパフォーマンスチューニングを取り扱いません。

本書の構成

目次

第 1 章 MySQL 設定ファイル

本書で説明する設定ファイルの概要と、利用時の注意点、利用方法について説明します。

第 2 章 MySQL 設定ファイル例

利用環境と利用目的に応じて、3 つの設定ファイル例について説明します。

第 3 章 MySQL サーバ設定ファイル修正方法

本書で紹介する MySQL 設定ファイルを異なる環境で利用する際の修正方法について説明します。

第 4 章 MySQL サーバの動作確認

MySQL サーバが意図したメモリサイズ内で動作するか確認する方法について説明します。

付録 重要な MySQL 設定オプション

MySQL 設定ファイルのオプションのうち、重要なオプションについて説明します。

目次

はじめに	2
本書の範囲	2
本書の構成	2
目次	3
1. MySQL設定ファイル	5
1.1. 設定ファイル例	5
1.2. 設定ファイル例の利用	5
1.3. 設定ファイル例を利用する時の注意事項	6
1.4. MySQLサーバ付属サンプル設定ファイル	7
2. MySQL設定ファイル例	8
2.1. my-myisam.cnf	8
2.1.1. オプション設定	8
2.1.2. 設定ファイル	9
2.2. my-x86-innodb.cnf	9
2.2.1. オプション設定	10
2.2.2. 設定ファイル	11
2.3. my-x64-innodb.cnf	12
2.3.1. オプション設定	12
2.3.2. 設定ファイル	13
3. MySQLサーバ設定ファイル修正方法	15
3.1. 設定ファイルの構成	15
3.2. 各設定ファイルの修正方法	15
4. MySQLサーバの動作確認	17
4.1. 想定利用方法に関する動作確認試験の実施	17
4.2. MySQLサーバプロセスのメモリ使用量の確認	17
4.3. トラブルシューティング	17
付録A. 重要なMySQL設定オプション	18
A.1. クライアント用オプション	18
port	18
socket	18
default-character-set	18
A.2. サーバ用オプション	18
binlog_cache_size	18
datadir	18
default-character-set	18
default_table_type	19
innodb_additional_mem_pool_size	19
innodb_buffer_pool_size	19
innodb_data_file_path	19
innodb_log_buffer_size	19
innodb_log_file_size	19
innodb_log_files_in_group	19
join_buffer_size	20
key_buffer_size	20
log-bin	20
log-slow-queries	20
long-query-time	20
max_allowed_packet	20
max_connections	20
max_heap_table_size	20
myisam_sort_buffer_size	21
port	21
query_cache_limit	21
query_cache_size	21
read_buffer_size	21

read_rnd_buffer_size	21
skip_innodb	21
socket	21
sort_buffer_size	22
sync_binlog	22
table_cache	22
thread_cache_size	22
tmp_table_size	22

1. MySQL 設定ファイル

MySQL サーバのオプションは以下 3 つの方法で設定することが可能です。

- MySQL サーバを起動するコマンドラインオプションでの指定
- 設定ファイルでのオプション指定
- MySQL サーバを起動中に mysql コマンドを使用して動的に指定

本書では、主に「設定ファイル」での設定方法について記述しますが、設定ファイル中で指定したオプションとその値は他の方法で設定することも可能です。

以下本書では、特に断りの無い限り MySQL サーバ設定ファイルを単に「設定ファイル」と記述します。

1.1. 設定ファイル例

本書では、MySQL サーバの典型的な実行環境や利用条件を想定し「表 1 本書で説明する設定ファイル例」にある 3 つの設定ファイル例を示します。

本書で紹介する設定ファイル例は、利用 OS の bit 数(32bit 環境または 64bit 環境)および利用サーバの搭載メモリ、利用ストレージエンジンの 3 つの条件を元に分類しています。

表 1 で示す条件と同一条件では、本書の MySQL 設定ファイルをそのまま利用することが可能です。またサーバマシンの搭載メモリサイズや、MySQL サーバの利用可能メモリサイズが異なる場合や、特定用途向けに特化した設定を行う場合も、本書の設定ファイルを元にオプションの調整、追加を行うことで対応可能です。

表 1 本書で説明する設定ファイル例

設定ファイル	OS	必要メモリ	利用ストレージエンジン	説明
my-myisam.cnf	RHEL4 x86(32bit) RHEL4 x64(64bit)	4GB (約 2GB)	MyISAM	多数のコネクションによるデータ参照を主に行う場合の設定ファイルです。
my-x86-innodb.cnf	RHEL4 x86(32bit)	4GB (約 3.5GB)	MyISAM InnoDB	データ更新と参照の両方を行い、トランザクション処理が必要とされる場合の設定ファイルです。
my-x64-innodb.cnf	RHEL4 x64(64bit)	12GB (約 10GB)	MyISAM InnoDB	データ更新と参照の両方を行い、トランザクション処理が必要とされる場合の設定ファイルです。

「必要メモリ」欄は、本書の設定ファイルを利用して MySQL サーバを起動する際に、サーバマシンに必要とされる搭載メモリ量です。カッコ内は MySQL サーバが利用する想定最大メモリ量です。この値は利用状況によって増減します。本書の設定ファイル利用時には必ず実際の利用パターンを踏まえた動作試験を行ってください。

「利用ストレージエンジン」は、MySQL サーバのデータ処理コンポーネントです。MySQL サーバはいくつかのストレージエンジンを切り替えまたは併用することができ、データベースの利用目的によって最も適したストレージエンジンを選択して利用することができます。本書で設定する MyISAM と InnoDB は以下の特徴を持ちます。

表 2 MySQL のストレージエンジン

ストレージエンジン	トランザクション	説明
MyISAM	×	データ更新が少なく、主にデータ参照を行うシステムに適しています。トランザクションをサポートしませんが、参照は高速に行うことができます。また GIS(地理情報システム)データを利用できます。
InnoDB		データ更新が頻繁に発生するシステムを想定しています。ACID 準拠のトランザクションやクラッシュリカバリをサポートします。

1.2. 設定ファイル例の利用

MySQL サーバ設定ファイルは、通常 /etc ディレクトリ下に my.cnf として配置されます。本書の設定ファイル例を利用する場合も、同様に /etc/my.cnf として配置してください。

以下に設定ファイル例の作成から利用までの手順を示します。

- (1) 「表 1 本書で説明する設定ファイル例」から利用する設定ファイル例を選択します。
- (2) 選択した設定ファイル例を本書からコピーし、ファイルを作成します。
- (3) 設定ファイルを配置します。

```
# 設定ファイル例として my-myisam.cnf を選択した場合。  
$ whoami # MySQL サーバ実行ユーザを mysql としています
```

```
mysql
$ ls /tmp/my-myisam.cnf # 作成した設定ファイルを/tmp に置いているとします
/tmp/my-myisam.cnf
$ su -
# cp /tmp/my-myisam.cnf /etc/my.cnf
# chown mysql:mysql /etc/my.cnf # my.cnf の所有者を MySQL サーバ実行ユーザに変更
# chmod 644 /etc/my.cnf # アクセス権の修正
```

- (4) MySQL 設定ファイルの利用文字コードを確認します。
日本語利用時の文字化けを防止するため、MySQL 設定ファイル内の文字コード設定を確認し、必要に応じて修正してください。デフォルトでは UTF-8 を利用するようになっています。
UTF-8 利用時には“utf8”を設定し、Shift-JIS 利用時には“cp932”を設定します。日本語 EUC を利用する場合は“eucjms”を設定します。

```
$ whoami
mysql
$ vi /etc/my.cnf
[client]
port          = 3306
socket        = /tmp/mysql.sock
default-character-set = utf8          utf8, cp932, eucjms の何れかを指定

[mysqld]
datadir       = /usr/local/mysql/data
...省略...
default-character-set = utf8          utf8, cp932, eucjms の何れかを指定
...省略...
```

- (5) データディレクトリ未作成の場合 mysql_install_db を実行します。
以下の例ではデータディレクトリを /usr/local/mysql/data としています。

```
$ whoami
mysql
$ su -
# mkdir -p /usr/local/mysql/data
# chown mysql:mysql /usr/local/mysql/data
# chmod 755 /usr/local/mysql/data
# exit
$ cd /usr/local/mysql
$ ./scripts/mysql_install_db --datadir=/usr/local/mysql/data
```

- (6) mysqld_safe コマンドを実行し、MySQL サーバを起動します。

```
$ whoami
mysql
$ /usr/local/mysql/bin/mysqld_safe &
```

MySQL サーバ設定ファイルを /etc/my.cnf 以外の場所およびファイル名で配置する場合、mysqld_safe コマンドに対するオプションで指定することができます。以下に例を示します。

```
$ whoami
mysql
$ /usr/local/mysql/bin/mysqld_safe --defaults-file=/MySQL-DATA/my.cnf
```

また起動オプションを指定しなかった場合、MySQL サーバ(mysqld)は、以下の順序で MySQL 設定ファイルを検索し、存在する MySQL 設定ファイルを順に読み込みます。重複するオプションは上書きされていきます。

```
/etc/my.cnf
$MYSQL_HOME/my.cnf ($MYSQL_HOME は環境変数)
--defaults-extra-files オプションで指定した MySQL 設定ファイル
~/my.cnf (MySQL サーバ実行ユーザのホームディレクトリ内の my.cnf)
```

1.3. 設定ファイル例を利用する時の注意事項

「表 1 本書で説明する設定ファイル例」で示した設定ファイルを利用する際は以下の点に注意してください。

■ 利用 OS の種類

利用する RHEL4 が x86 版(32bit OS)である場合、1つのプロセスが利用可能なメモリの上限は約 4GB です。また利用状況によっては 4GB を超えてしまう可能性があります。必ず実運用を前提にした動作確認を行ってください

い。RHEL4 AMD64/Intel64 版(64bit OS)を利用する場合でも、スワップアウトが発生しないよう MySQL サーバが必要とするメモリ量に注意してください。

■ **サーバの利用可能メモリ量**

ご利用のサーバマシンの搭載メモリが表 1 で示したメモリ量と異なる場合、「3.MySQL サーバ設定ファイル修正方法」で簡単な調整方法について説明しています。そちらを参照ください。

■ **動作試験の実施**

本書の設定ファイルおよび、これを改変した設定ファイルを利用する場合は、必ず実際の利用パターンを踏まえた動作試験を行ってください。RHEL4 x86 版利用時に MySQL サーバの利用メモリ量が 4GB を超えたり、スワップアウトが発生する場合は、設定ファイルのオプション設定値を少なくするなど対処が必要となります。

1.4. MySQLサーバ付属サンプル設定ファイル

MySQL をインストールすると設定ファイルのサンプルもインストールされます。このサンプル設定ファイルにはコメントが記載されており、MySQL 設定ファイルを作成する上で参考にすることができます。本書で説明する設定ファイル例は、このサンプル設定ファイルをベースに作成します。

MySQL 設定ファイルのサンプルは、インストールパッケージ毎に異なります。

表 3 サンプル設定ファイルの格納場所

パッケージ	サンプル設定ファイルの格納場所
non RPM パッケージ	/usr/local/mysql/support-files/
RPM パッケージ	/usr/share/mysql/
ソースパッケージ	/usr/local/mysql/share/mysql/

以下の MySQL 設定ファイルのサンプルが提供されています。

表 4 サンプル設定ファイルのリスト

ファイル名	概要説明
my-small.cnf	メインメモリが 64MB 以下のサーバで利用する場合の MySQL 設定ファイル
my-medium.cnf	メインメモリが 32MB から 64MB のサーバで利用する場合の MySQL 設定ファイル
my-large.cnf	メインメモリが 512MB のサーバで利用する場合の MySQL 設定ファイル
my-huge.cnf	メインメモリが 1GB から 2GB のサーバで利用する場合の MySQL 設定ファイル 本書の my-mysam.cnf のベース設定ファイルです。
my-innodb-heavy-4G.cnf	メインメモリが 4GB のサーバ上で InnoDB を主に利用する場合の MySQL 設定ファイル 本書の my-x86-innodb.cnf および my-x64-innodb.cnf のベース設定ファイルです。

上記から実行環境およびデータベースの利用目的に適したものを選択し、MySQL 設定ファイルのベースとします。

2. MySQL 設定ファイル例

2.1. my-myisam.cnf

対象環境および利用目的は以下の通りです。

表 5 対象環境と利用目的

OS	Red Hat Enterprise Linux 4 x86 版(32bit) Red Hat Enterprise Linux 4 x64 版(64bit)
搭載メモリ	4GB 以上 (約 2GB を MySQL サーバが利用と想定)
利用目的	トランザクションは不要。 データ更新が少なく、多数のコネクションからの参照が多い。 GIS(地理情報システム)データを利用したい。
データディレクトリ	/usr/local/mysql/data (datadir オプションで指定)

2.1.1. オプション設定

本設定ファイルは、MySQL サーバ付属サンプル設定ファイル my-huge.cnf をベースとして利用しています。以下に my-huge.cnf からの変更点を示します。

表 6 my-myisam.cnf のオプション

オプション	変更前	変更後	コメント
[client]			
port	3306		サーバへの接続ポート番号
socket	/tmp/mysql.sock		サーバへの接続用 UNIX ドメインソケットファイル
default-character-set	設定なし	utf8	利用文字コードに応じて設定
[mysqld]			
datadir	設定なし	(*1)	データディレクトリ指定
port	3306		クライアント接続用ポート番号
socket	/tmp/mysql.sock		クライアント接続用 UNIX ドメインソケットファイル
skip-locking	-		
key_buffer_size	384M		MyISAM 用 INDEX 用バッファサイズ
max_allowed_packet	1M	2M	クライアントとの通信用バッファの最大サイズ
table_cache	512	2048	テーブルキャッシュにキャッシュ可能なテーブル数
sort_buffer_size	2M	8M	ソート処理用バッファサイズ
read_buffer_size	2M		シーケンシャルなデータ読み込みバッファサイズ
read_rnd_buffer_size	8M	16M	ランダムなデータ読み込みバッファサイズ
myisam_sort_buffer_size	64M	128M	MyISAM テーブルの INDEX 作成用バッファサイズ
thread_cache_size	8		コネクション用スレッドのキャッシュ
query_cache_size	32M		クエリ結果をキャッシュするクエリキャッシュサイズ
thread_concurrency	8		
log-bin	mysql-bin		バイナリログのベースファイル名
server-id	1		レプリケーション構成時のサーバ ID
skip-innodb	設定なし	-	InnoDB 利用を禁止する
default-character-set	設定なし	utf8	利用文字コードに応じて設定
max_connections	設定なし(100)	100	接続可能な最大クライアント数
sync_binlog	設定なし(1)	1	バイナリログの書き込み方法の指定
max_heap_table_size	設定なし(16M)	64M	テンポラリ・テーブルの最大サイズを規定
tmp_table_size	設定なし(32M)	64M	テンポラリ・テーブルの最大サイズを規定
binlog_cache_size	設定なし(32K)	1M	バイナリログ用キャッシュサイズ
join_buffer_size	設定なし(128K)	8M	フルジョイン処理用バッファサイズ
log_slow_queries	設定なし	-	スロークエリーログ出力指定
query_cache_limit	設定なし(1M)	2M	キャッシュ可能なクエリ実施結果最大サイズ

(*1) /usr/local/mysql/data

オプション

MySQL 設定ファイルのオプション名を記載しています。大括弧でくくられている[client]と[mysqld]はセクションをしめして

います。my-huge.cnf には上記以外のセクションも存在しますが、MySQL サーバ自体の動作に影響はないため変更をしていません。このため省略しています。

変更前

my-huge.cnf での各パラメータの設定値です。「設定なし」とかかかれているオプションは my-huge.cnf に記載されていないオプションです。カッコ内はデフォルトの値です。MySQL 設定ファイルに追記する際は各セクション内に追記してください。ハイフン(-)が記載されているオプションは設定値が指定されていないか、不要なオプションです。

変更後

本設定ファイルでの設定値を記述しています。矢印()が記載されている欄は左欄の値と同一の値を設定しています。

コメント

オプションの簡単な説明を記述しています。

2.1.2. 設定ファイル

設定ファイルを以下に示します。

```
[client]
port                = 3306
socket              = /tmp/mysql.sock
default-character-set = utf8

[mysqld]
datadir             = /usr/local/mysql/data
port                = 3306
socket              = /tmp/mysql.sock
skip-locking
key_buffer          = 384M
max_allowed_packet = 2M
table_cache         = 2048
sort_buffer_size    = 8M
read_buffer_size    = 2M
read_rnd_buffer_size = 16M
myisam_sort_buffer_size = 128M
thread_cache_size   = 8
query_cache_size    = 32M
thread_concurrency  = 8
log-bin             = mysql-bin
server-id           = 1
skip_innodb
default-character-set = utf8
max_connection      = 100
sync_binlog         = 1
max_heap_table_size = 64M
tmp_table_size      = 64M
binlog_cache_size   = 1M
join_buffer_size    = 8M
log_slow_queries
query_cache_limit   = 2M
```

2.2. my-x86-innodb.cnf

対象環境および利用目的は以下の通りです。

表 7 対象環境と利用目的

OS	Red Hat Enterprise Linux 4 x86 版(32bit)
搭載メモリ	4GB 以上 (約 3.5GB を MySQL サーバが利用と想定)
利用目的	トランザクションが必要。 データ更新頻度が多く、参照も必要。複雑なクエリを実施。
データディレクトリ	/usr/local/mysql/data (datadir オプションで指定)

InnoDB ストレージエンジンを主に利用しますが、MyISAM ストレージエンジンも利用することが可能です。MyISAM ストレージエンジンの利用停止はできません。

2.2.1. オプション設定

本設定ファイルは、MySQL サーバ付属サンプル設定ファイル my-innodb-heavy-4G.cnf をベースとして利用しています。以下に my-innodb-heavy-4G.cnf からの変更点を示します。

表 8 my-x86-innodb.cnf のオプション

オプション	変更前	変更後	コメント
[client]			
port		3306	サーバへの接続ポート番号
socket	/tmp/mysql.sock		サーバへの接続用 UNIX ドメインソケットファイル
default-character-set	設定なし	utf8	利用文字コードに応じて設定
[mysqld]			
datadir	設定なし	(*1)	
port		3306	クライアント接続用ポート番号
socket	/tmp/mysql.sock		クライアント接続用 UNIX ドメインソケットファイル
back_log		50	接続待ちキューの長さ
max_connections		100	クライアントの最大接続数
table_cache		2048	テーブルキャッシュにキャッシュするテーブル数
max_allowed_packet		16M	クライアントとの通信用バッファの最大サイズ
binlog_cache_size		1M	バイナリログのキャッシュサイズ
max_heap_table_size		64M	テンポラリ・テーブルの最大サイズを規定
sort_buffer_size		8M	ソート用バッファサイズ
join_buffer_size		8M	JOIN 用バッファサイズ
thread_cache_size		8	クライアント接続用スレッドのキャッシュサイズ
thread_concurrency		8	
query_cache_size		64M	クエリ結果をキャッシュするクエリキャッシュサイズ
query_cache_limit		2M	キャッシュ可能なクエリ実施結果最大サイズ
ft_min_word_len		4	
default_table_type	MYISAM	INNODB	テーブル作成時のデフォルト・ストレージエンジン
thread_stack		192K	
tmp_table_size		64M	テンポラリ・テーブルの最大サイズを規定
log-bin	mysql-bin		バイナリログファイルのベースファイル名
log_slow_queries		-	スロークエリログ出力指定
long_query_time		2	スロークエリ用の閾値(秒)
log_long_format		-	
server-id		1	レプリケーション用サーバ識別子
key_buffer_size		32M	MyISAM 用 INDEX バッファサイズ
read_buffer_size		2M	シーケンシャルなデータ読み込みバッファサイズ
read_rnd_buffer_size		16M	ランダムなデータ読み込みバッファサイズ
bulk_insert_buffer_size		64M	バルクインサート用バッファ
myisam_sort_buffer_size		128M	MyISAM テーブルの INDEX 作成用バッファサイズ
myisam_max_sort_file_size		10G	
myisam_max_extra_sort_file_size		10G	
myisam_repair_threads		1	
myisam_recover		-	
skip-bdb		-	
innodb_additional_mem_pool_size		16M	InnoDB 用メモリプールサイズ
innodb_buffer_pool_size		2G	InnoDB 用バッファサイズ
innodb_data_file_path		(*2)	InnoDB 用データファイルパス
innodb_file_io_threads		4	
innodb_thread_concurrency		16	
innodb_flush_log_at_trx_commit		1	InnoDB 用ログ書き込みタイミング設定
innodb_log_buffer_size		8M	InnoDB 用ログバッファサイズ
innodb_log_file_size		256M	InnoDB 用ログファイルサイズ
innodb_log_files_in_group		3	InnoDB 用ログファイルの数
innodb_max_dirty_pages_pct		90	

innodb_lock_wait_timeout	120		
default-character-set	設定なし	utf8	利用文字コードに応じて設定
sync_binlog	設定なし	1	バイナリログの同期書き込みを指定

(* 1) /usr/local/mysql/data

(* 2) ibdata1:10M:autoextend

オプション

MySQL 設定ファイルのオプション名を記載しています。大括弧でくくられている[client]と[mysqld]はセクションをしめています。my-innodb-heavy-4G.cnf には上記以外のセクションも存在しますが、MySQL サーバ自体の動作に影響はないため変更をしていません。このため省略しています。

変更前

my-innodb-heavy-4G.cnf での各パラメータの設定値です。「設定なし」とかかかれているオプションは my-innodb-heavy-4G.cnf に記載されていないオプションです。各セクション内に追記してください。ハイフン(-)が記載されているオプションは設定値が指定されていないか、不要なオプションです。

変更後

本設定ファイルでの設定値を記述しています。矢印()が記載されている欄は左欄の値と同一の値を設定しています。

コメント

オプションの簡単な説明を記述しています。

2.2.2. 設定ファイル

設定ファイルを以下に示します。

```
[client]
port                = 3306
socket              = /tmp/mysql.sock
default-character-set = utf8

[mysqld]
datadir             = /usr/local/mysql/data
port                = 3306
socket              = /tmp/mysql.sock
back_log            = 50
max_connections     = 100
max_connect_errors = 10
table_cache         = 2048
max_allowed_packet = 16M
binlog_cache_size  = 1M
max_heap_table_size = 64M
sort_buffer_size    = 8M
join_buffer_size    = 8M
thread_cache_size   = 8
thread_concurrency  = 8
query_cache_size    = 64M
query_cache_limit   = 2M
ft_min_word_len     = 4
default_table_type  = INNOSB
thread_stack        = 192K
transaction_isolation = REPEATABLE-READ
tmp_table_size      = 64M
log-bin             = mysql-bin
log_slow_queries    =
long_query_time     = 2
log_long_format     =
server-id           = 1
key_buffer_size     = 32M
read_buffer_size    = 2M
read_rnd_buffer_size = 16M
bulk_insert_buffer_size = 64M
myisam_sort_buffer_size = 128M
myisam_max_sort_file_size = 10G
```

```

mysam_max_extra_sort_file_size = 10G
mysam_repair_threads    = 1
mysam_recover
skip-bdb
innodb_additional_mem_pool_size = 16M
innodb_buffer_pool_size = 8G
innodb_data_file_path    = ibdata1:10M:autoextend
innodb_file_io_threads  = 4
innodb_thread_concurrency    = 16
innodb_flush_log_at_trx_commit = 1
innodb_log_buffer_size    = 8M
innodb_log_file_size     = 256M
innodb_log_files_in_group = 3
innodb_max_dirty_pages_pct = 90
innodb_lock_wait_timeout  = 120
default-character-set    = utf8
sync_binlog              = 1

```

2.3. my-x64-innodb.cnf

対象環境および利用目的は以下の通りです。

表 9 対象環境と利用目的

OS	Red Hat Enterprise Linux 4 AMD64/Intel64 版(64bit)
搭載メモリ	12GB 以上(約 10GB を MySQL サーバが利用と想定)
利用目的	トランザクションが必要。 データ更新頻度が多く、参照も必要。複雑なクエリを実施。
データディレクトリ	/usr/local/mysql/data (datadir オプションで指定)

InnoDB ストレージエンジンを主に利用しますが、MyISAM ストレージエンジンも利用することが可能です。MyISAM ストレージエンジンの利用停止はできません。

2.3.1. オプション設定

本設定ファイルは、MySQL サーバ付属サンプル設定ファイル my-innodb-heavy-4G.cnf をベースとして利用しています。以下に my-innodb-heavy-4G.cnf からの変更点を示します。

表 10 my-x64-innodb.cnf のオプション

オプション	変更前	変更後	コメント
[client]			
port	3306		サーバへの接続ポート番号
socket	/tmp/mysql.sock		サーバへの接続用 UNIX ドメインソケットファイル
default-character-set	設定なし	utf8	利用文字コードに応じて設定
[mysqld]			
datadir	設定なし	(*1)	
port	3306		クライアント接続用ポート番号
socket	/tmp/mysql.sock		クライアント接続用 UNIX ドメインソケットファイル
back_log	50		接続待ちキューの長さ
max_connections	100		クライアントの最大接続数
table_cache	2048		テーブルキャッシュにキャッシュするテーブル数
max_allowed_packet	16M		クライアントとの通信用バッファの最大サイズ
binlog_cache_size	1M		バイナリログのキャッシュサイズ
max_heap_table_size	64M		テンポラリ・テーブルの最大サイズを規定
sort_buffer_size	8M		ソート用バッファサイズ
join_buffer_size	8M		JOIN 用バッファサイズ
thread_cache_size	8		クライアント接続用スレッドのキャッシュサイズ
thread_concurrency	8		
query_cache_size	64M		クエリ結果をキャッシュするクエリキャッシュサイズ
query_cache_limit	2M		キャッシュ可能なクエリ実施結果最大サイズ
ft_min_word_len	4		

default_table_type	MYISAM	INNODB	テーブル作成時のデフォルト・ストレージエンジン
thread_stack	192K		
tmp_table_size	64M		テンポラリ・テーブルの最大サイズを規定
log-bin	mysql-bin		バイナリログファイルのベースファイル名
log_slow_queries	-		スロークエリログ出力指定
long_query_time	2		スロークエリ用の閾値(秒)
log_long_format	-		
server-id	1		レプリケーション用サーバ識別子
key_buffer_size	32M		MyISAM 用 INDEX バッファサイズ
read_buffer_size	2M		シーケンシャルなデータ読み込みバッファサイズ
read_rnd_buffer_size	16M		ランダムなデータ読み込みバッファサイズ
bulk_insert_buffer_size	64M		バルクインサート用バッファ
myisam_sort_buffer_size	128M		MyISAM テーブルの INDEX 作成用バッファサイズ
myisam_max_sort_file_size	10G		
myisam_max_extra_sort_file_size	10G		
myisam_repair_threads	1		
myisam_recover	-		
skip-bdb	-		
innodb_additional_mem_pool_size	16M		InnoDB 用メモリアルサイズ
innodb_buffer_pool_size	2G	8G	InnoDB 用バッファサイズ
innodb_data_file_path	(*2)		InnoDB 用データファイルパス
innodb_file_io_threads	4		
innodb_thread_concurrency	16		
innodb_flush_log_at_trx_commit	1		InnoDB 用ログ書き込みタイミング設定
innodb_log_buffer_size	8M		InnoDB 用ログバッファサイズ
innodb_log_file_size	256M		InnoDB 用ログファイルサイズ
innodb_log_files_in_group	3		InnoDB 用ログファイルの数
innodb_max_dirty_pages_pct	90		
innodb_lock_wait_timeout	120		
default-character-set	設定なし	utf8	利用文字コードに応じて設定
sync_binlog	設定なし	1	バイナリログの同期書き込みを指定

(*1) /usr/local/mysql/data

(*2) ibdata1:10M:autoextend

オプション

MySQL 設定ファイルのオプション名を記載しています。大括弧でくくられている[client]と[mysqld]はセクションをしめています。my-innodb-heavy-4G.cnf には上記以外のセクションも存在しますが、MySQL サーバ自体の動作に影響はないため変更をしていません。このため省略しています。

変更前

my-innodb-heavy-4G.cnf での各パラメータの設定値です。「設定なし」とかかかれているオプションは my-innodb-heavy-4G.cnf に記載されていないオプションです。各セクション内に追記してください。ハイフン(-)が記載されているオプションは設定値が指定されていないか、不要なオプションです。

変更後

本設定ファイルでの設定値を記述しています。矢印()が記載されている欄は左欄の値と同一の値を設定しています。

コメント

オプションの簡単な説明を記述しています。

2.3.2. 設定ファイル

設定ファイルを以下に示します。

```
[client]
port          = 3306
socket        = /tmp/mysql.sock
default-character-set = utf8

[mysqld]
datadir       = /usr/local/mysql/data
```

```
port = 3306
socket = /tmp/mysql.sock
back_log = 50
max_connections = 100
max_connect_errors = 10
table_cache = 2048
max_allowed_packet = 16M
binlog_cache_size = 1M
max_heap_table_size = 64M
sort_buffer_size = 8M
join_buffer_size = 8M
thread_cache_size = 8
thread_concurrency = 8
query_cache_size = 64M
query_cache_limit = 2M
ft_min_word_len = 4
default_table_type = INNOSB
thread_stack = 192K
transaction_isolation = REPEATABLE-READ
tmp_table_size = 64M
log-bin = mysql-bin
log_slow_queries
long_query_time = 2
log_long_format
server-id = 1
key_buffer_size = 32M
read_buffer_size = 2M
read_rnd_buffer_size = 16M
bulk_insert_buffer_size = 64M
myisam_sort_buffer_size = 128M
myisam_max_sort_file_size = 10G
myisam_max_extra_sort_file_size = 10G
myisam_repair_threads = 1
myisam_recover
skip-bdb
innodb_additional_mem_pool_size = 16M
innodb_buffer_pool_size = 8G
innodb_data_file_path = ibdata1:10M:autoextend
innodb_file_io_threads = 4
innodb_thread_concurrency = 16
innodb_flush_log_at_trx_commit = 1
innodb_log_buffer_size = 8M
innodb_log_file_size = 256M
innodb_log_files_in_group = 3
innodb_max_dirty_pages_pct = 90
innodb_lock_wait_timeout = 120
default-character-set = utf8
sync_binlog = 1
```

3. MySQL サーバ設定ファイル修正方法

MySQL サーバ設定ファイルの構成について説明し、本書の設定ファイル例の修正方法について説明します。

3.1. 設定ファイルの構成

設定ファイルはテキストファイルとして記述されており、MySQL サーバ本体の設定だけでなく MySQL のクライアントやツール類の設定も記述することができます。これらの記述は設定対象毎にセクションで別れています。以下に設定ファイルの例を示します。

```
[client]
#password      = your_password
port           = 3306
socket         = /tmp/mysql.sock
default-character-set = utf-8

# Here follows entries for some specific programs

# The MySQL server
[mysqld]
port           = 3306
socket         = /tmp/mysql.sock
skip-locking
key_buffer    = 384M
#max_allowed_packet = 1M
max_allowed_packet = 16M
...省略...
```

■ セクション

セクションは、**[client]** のように大括弧で囲まれた行から開始し、次のセクションの開始位置またはファイル末尾までの範囲になります。この範囲に対象のオプションを記述します。以下のようなセクションを利用できます。他のセクションについては MySQL サーバ付属のサンプル設定ファイルを参照してください。

- **[client]** は MySQL サーバのクライアント全般に対するオプションです。
- **[mysqld]** は MySQL サーバに対するオプションです。

■ コメント

MySQL 設定ファイルのコメントはシャープ記号(#)以降の文字列です。

■ オプション

オプションは MySQL サーバの動作や内部処理用バッファのサイズなどを設定します。設定値が存在する場合はイコール記号(=)に続けて設定値を記述します。設定値の単位は各オプションによって異なります。メモリサイズを指定する場合には、K(キロバイト)、M(メガバイト)、および G(ギガバイト)を利用することができます。

3.2. 各設定ファイルの修正方法

利用サーバのメモリサイズまたは MySQL サーバが利用可能なメモリサイズが本書の想定する環境より大きい場合、あるいは小さい場合、本書の設定ファイルのオプションを修正します。

my-myisam.cnf も my-x86-innodb.cnf および my-x64-innodb.cnf も修正する場合の考え方はほぼ同じです。違いは InnoDB ストレージエンジン用のオプションの有無のみです。また x86 版と x64 版のどちらも同じ考え方で修正できます。ただし x86 環境では、MySQL サーバプロセスの最大メモリサイズが約 4GB までとなっている点に注意してください。

MySQL サーバが利用可能なメモリサイズが大きい場合

MySQL サーバが利用可能なメモリサイズが大きい場合、InnoDB 用バッファプールサイズを大きくします。具体的には、innodb_buffer_pool_size を利用可能なメモリサイズの 80%程度を目安に設定します。x86 環境では 2GB 以下に設定してください。

MySQL サーバが利用可能なメモリサイズが小さい場合

以下に、より小さなメモリサイズで MySQL サーバを動作させる場合に設定ファイルを修正するための考え方を示します。

- (1) InnoDB 用バッファプールを小さくする (my-x86-innodb.cnf および my-x64-innodb.cnf)
InnoDB ストレージエンジンでは innodb_buffer_pool_size で指定される InnoDB 用バッファプールサイズが重要となります。MySQL サーバがメモリ不足の場合は、このサイズの縮小を検討してください。ほとんどの場合、このオプションを変更することで対応可能です。
- (2) コネクション数が多いためにメモリ不足となる場合
MySQL サーバに接続するコネクション数が非常に多い場合、コネクション毎に設定されるオプションを変更します。以下のオプションのサイズを変更してください。

表 11 コネクション数が多いためにメモリ不足となる場合に検討するオプション

オプション	変更	説明
read_rnd_buffer_size	16M 8M またはそれ以下	行読み込み処理に利用するバッファサイズを指定します。コネクション毎に 1 つ作成されます。
read_buffer_size	2M 1M またはそれ以下	シーケンシャルなデータ読み込みが必要な場合に利用されます。アクセスするテーブルごとに作成されます。
tmp_table_size	64M 32M またはそれ以下	クエリ実施時に必要に応じて内部的に作成されるテンポラリテーブルの最大サイズです。このサイズを超える場合、ファイルベースのテンポラリテーブルが作成されます。

(3) その他の場合

その他の場合、以下のオプションの設定値の変更を検討してください。以下のオプションによってサイズ指定されるバッファは MySQL サーバ全体で共有されます。

表 12 その他の場合に検討するオプション

オプション	変更	説明
key_buffer	384M 128M またはそれ以下	MyISAM ストレージエンジン用のインデックスキーバッファのサイズを指定しています。
myisam_sort_buffer_size	128M 32M またはそれ以下	MyISAM ストレージエンジンでインデックス作成時に利用されるバッファです。
query_cache_size	64M 32M	クエリ実施結果をキャッシュするキャッシュのサイズを指定します。

(4) それでもメモリ不足が解消しない場合

上記のオプション変更でも問題が発生する場合は、MySQL サーバ付属のサンプル設定ファイルの利用を検討してください。my-small.cnf や my-huge.cnf などでも InnoDB ストレージエンジンを利用可能です。各サンプル設定ファイルの詳細は「表 4 サンプル設定ファイルのリスト」を参照してください。また、その他のオプションについては、付録を参照してください。

4. MySQL サーバの動作確認

4.1. 想定利用方法に関する動作確認試験の実施

本書で紹介している設定ファイル例は、想定した環境や使用条件での利用を前提としています。このため、異なる環境や使用条件では、MySQL サーバプロセスの利用メモリサイズが想定したメモリサイズ以上のメモリを必要とし、意図しない障害が発生する可能性があります。

実運用に利用する前に、想定する同時接続数および想定するクエリ実施パターンで試験を行い、問題が発生しないことを確認してください。

4.2. MySQLサーバプロセスのメモリ使用量の確認

動作確認時に MySQL サーバプロセスのメモリ使用量の確認を必ず行ってください。以下のコマンドを利用すると確認が可能です。

```
$ ps v -u mysql
  PID TTY          STAT TIME  MAJFL  TRS   DRS   RSS %MEM COMMAND
 23500 pts/3        S    0:00    0   682   4737  1388  0.1 -bash
 23541 pts/3        S    0:00    0   682   4693  1096  0.1 /bin/sh /usr/local/mysql/bin/mysq
 23556 pts/3        Sl   0:00   15  5074 111433 13800  1.7 /usr/local/mysql/bin/mysqld --ba
 23567 pts/3        R+   0:00    0    73   5078   908  0.1 ps v -u mysql
```

mysqld プロセスの RSS の項目を参照します。単位はキロバイト(KB)で出力されています。上記の例では 13800KB 利用していることとなります。

4.3. トラブルシューティング

設定した MySQL 設定ファイルを利用した MySQL サーバに問題が発生した場合、エラーログを参照します。エラーログは MySQL サーバのデータディレクトリに配置されています。デフォルトの状態では以下のファイルにエラーログが出力されます。

表 13 エラーログのファイル名

利用 MySQL パッケージ	ファイル名
non RPM/ソースパッケージ	/usr/local/mysql/data/ホスト名.err
RPM パッケージ	/var/lib/mysql/ホスト名.err

上記のエラーログファイルに障害原因に関する情報が記録されます。こちらを参照し、オプションの値を変更するなどをを行い対処してください。

付録 A. 重要な MySQL 設定オプション

MySQL 設定のうち重要なオプションについて説明します。

A.1. クライアント用オプション

クライアントセクションの設定オプションについて説明します。

port

MySQL サーバへの接続時にクライアントが利用する Port 番号を指定します。通常 3306 を指定しますが、2 つ以上の MySQL サーバを利用する場合は、それぞれ異なる Port を指定する必要があります。

```
例: port=3306
```

socket

MySQL サーバへの接続時に利用可能なソケットファイルを指定します。クライアントが MySQL サーバと同一マシン上で動作している場合に指定されているソケットファイルを利用して接続します。Linux で RPM 版をインストールした場合、デフォルト値は /var/lib/mysql/mysql.sock となります。その他の場合は /tmp/mysql.sock が利用されます。2 つ以上の MySQL サーバを利用する場合は、それぞれ異なるソケットファイル名を指定する必要があります。

```
例: socket=/var/lib/mysql/mysql.sock
```

default-character-set

mysql コマンドなどのクライアントが利用する文字コードセットを指定します。日本語では以下を利用できます。Shift-JIS を利用する場合には sjis ではなく cp932 を利用することが推奨されます。また同様に EUC-JP を利用する場合には ujis ではなく eucjms を利用することが推奨されます。

表 14 サポートする日本語用文字コード

文字コードセット	説明
sjis	Shift-JIS
cp932	Shift-JIS にベンダ拡張文字コード対応を追加したもの
ujis	日本語 EUC
eucjms	日本語 EUC にベンダ拡張文字コード対応を追加したもの
uft8	UTF-8
ucs2	UNICODE

```
例: default-character-set=sjis
```

A.2. サーバ用オプション

サーバセクションでのオプションについてアルファベット順に説明します。

binlog_cache_size

トランザクション実行中に、バイナリログのために SQL ステートメントを保持するキャッシュのサイズ(バイト)を指定します。このキャッシュはクライアント毎に作成されます。大きく、複雑なトランザクションを利用する場合にはこのキャッシュサイズを拡張することでパフォーマンスが向上します。

```
例: binlog_cache_size=1M
```

datadir

MySQL サーバのデータ格納ディレクトリを指定します。このディレクトリには各種ログファイルおよびテーブル定義ファイル、データが配置されます。

```
例: datadir=/usr/local/mysql/data
```

default-character-set

mysql コマンドなどのクライアントが利用する文字コードセットを指定します。「表 14 サポートする日本語用文字コード」を参照してください。

default_table_type

CREATE TABLE 文でテーブルを作成する際に利用するデフォルト・ストレージエンジンを指定します。デフォルトでは MYISAM となっています。InnoDB を主に利用し、またストレージエンジンを意識したくない場合に、この設定を行います。

```
例: default_table_type=INNODB
```

innodb_additional_mem_pool_size

InnoDB ストレージエンジン用内部データを保持するためのバッファプールサイズ(バイト)を指定します。16MB に設定すれば、ほとんどの状況で十分です。利用状況は SHOW INNODB STATUS 文で確認できます。

```
例: innodb_additional_mem_pool_size=16M
```

innodb_buffer_pool_size

InnoDB ストレージエンジン用バッファプールサイズ(バイト)を指定します。テーブルのデータおよび INDEX をキャッシュします。このサイズが大きいくほどディスク I/O が減るため、パフォーマンスが向上します。ただし大きすぎるとスワップが発生し、パフォーマンス低下を招く可能性があります。
x86 環境では 2GB までしか確保できません。x64 環境ではより大きなサイズを確保することができます。

```
例: innodb_buffer_pool_size=2G
```

innodb_data_file_path

InnoDB ストレージエンジン用データファイルのリストを指定します。ここにはファイル名とサイズそして自動拡張可能か否かの設定を記述します。MySQL サーバは起動時にこの設定を確認し、対応するデータファイルが存在しない場合、作成します。取り扱うデータ量が巨大で、またある程度サイズが明らかな場合は、予め innodb_data_file_path に指定した方が良いでしょう。またパフォーマンス向上のため InnoDB データファイルを任意の高速なディスクに配置することができます。

```
例: innodb_data_file_path=ibdata1:50M:autoextend
```

```
例: innodb_data_file_path=ibdata1:50M;ibdata2:50M;ibdata3:50M:autoextend
```

```
例: innodb_data_file_path=/data/ibdata1:1G;/data/ibdata2:1G:autoextend
```

innodb_log_buffer_size

InnoDB ログファイルへの書き出し時に利用するバッファサイズ(バイト)を指定します。1MB ~ 8MB の間で指定することができます。バッファを大きくすることで、大きなトランザクション実行時のログファイルへの書き出しを減少させることができます。

```
例: innodb_log_buffer_size=8M
```

innodb_log_file_size

InnoDB ログファイルのファイルサイズ(バイト)を指定します。InnoDB ログファイルはコミット同時書き込みの更新ログで、MySQL サーバがクラッシュした際に、データベースの状態をクラッシュ前の状態に戻すために利用されるファイルです。innodb_log_file_size のサイズの InnoDB ログファイルが innodb_log_files_in_group で指定する数だけ作成されます。通常、5MB から innodb_buffer_pool_size / innodb_log_files_in_group バイトの範囲で指定します。大きな値に設定することで、InnoDB データファイルへのディスク I/O が減ります。

```
例: innodb_log_file_size=256M
```

innodb_log_files_in_group

InnoDB ログファイルの数を指定します。デフォルトでは 2 に設定されています。InnoDB ログファイルは 1 つのログファイルが書き終わると、次のログファイルに書き込みます。すべてのログファイルへ書き込みが終了すると、最初のログファイルに上書きして記録していきます。

```
例: innodb_log_files_in_group=3
```

join_buffer_size

INDEX を利用せず、テーブルをフルスキャンする JOIN 処理を行うためのバッファサイズ(バイト)を指定します。通常は INDEX を追加するべきですが、なんらかの理由により INDEX の追加ができない場合にこのバッファサイズを大きくすることで、処理パフォーマンスが向上します。このバッファはフルスキャンを行う 2 つのテーブル毎に 1 つ作成されます。

```
例: join_buffer_size=8M
```

key_buffer_size

MyISAM 用の INDEX バッファサイズ(バイト)を指定します。MyISAM サーバを主に利用し、INDEX を多用する場合はサイズを大きくします。利用状況は SHOW STATUS 文を利用して Key_read_request, Key_reads, Key_write_request の各値から確認できます。通常 Key_reads/Key_read_request の割合が 0.01 になるように設定すべきです。

```
例: key_buffer_size=384M
```

log-bin

バイナリログの出力を指定します。値としてバイナリログのベースファイル名を指定することができます。ベースファイル名を指定しなかった場合「**ホスト名-bin**」をベースファイル名としてバイナリログが生成されます。バイナリログはトランザクション単位に更新情報を保持しており、ロールフォワード・リカバリなど障害対応に利用できます。log-bin を設定しなかった場合、バイナリログは出力されません。

```
例: log-bin
```

```
例: log-bin=mysql-bin
```

log-slow-queries

スロークエリーログの出力を指定します。値としてスロークエリーログファイル名を指定することができます。ファイル名を指定しなかった場合、「**ホスト名-slow.log**」をファイル名としてスロークエリーログが生成されます。スロークエリーログはクエリ実施時間が一定時間以上必要とするクエリを記録します。log-slow-queries を指定しなかった場合、スロークエリーログは出力されません。

```
例: log-slow-queries
```

```
例: log-slow-queries=mysql-slow.log
```

long-query-time

スロークエリーログに記録するクエリ実施時間の長さ(秒)を指定します。デフォルトは 10 秒で最小値は 1 秒です。

```
例: long-query-time=4
```

max_allowed_packet

クライアントとの通信時に利用するパケットメッセージバッファの最大サイズ(バイト)を指定します。巨大な BLOB や長大な文字列操作を必要とする場合に、大きくする必要があります。

```
例: max_allowed_packet=16M
```

max_connections

同時接続可能クライアント数の上限を指定します。デフォルトでは 100 となっています。実際のアプリケーションではコネクション・プールの利用などで 100 を超えることはないと思われませんが、接続が非常に多い場合にこの値を大きくします。MySQL サーバでは接続ごとにスレッドが生成され、またいくつかのバッファがスレッド毎に生成されるため、各メモリバッファサイズの見直しが必要になる可能性があります。

```
例: max_connections=150
```

max_heap_table_size

MEMORY(HEAP)ストレージエンジンで作成可能なテーブルの最大サイズ(バイト)を指定します。この設定は MySQL 内部で利用するテンポラリー・テーブルのサイズに影響します。tmp_table_size と同一か、より大きい値を設定すべきです。テンポラリー・テーブルはソート処理やその他クエリ実施時に利用されます。このサイズを超えるテンポラリー・テーブルはファイルに書き出されます。

例: `max_heap_table_size=64M`

myisam_sort_buffer_size

MyISAM ストレージエンジン用のバッファサイズ(バイト)を指定します。REPAIR TABLE、CREATE INDEX、ALTER TABLE で INDEX の変更・作成を行う際に、INDEX のソート処理に利用されます。INDEX 付加済みのテーブルへの大量データ投入時にも、バッファサイズ拡大は有効です。すべてのクライアントで共有されます。

例: `myisam_sort_buffer_size=128M`

port

MySQL サーバに接続するクライアント用の Port 番号を指定します。通常 3306 を指定しますが、2 つ以上の MySQL サーバを利用する場合は、それぞれ異なる Port を指定する必要があります。

例: `port=3306`

query_cache_limit

クエリキャッシュにキャッシュするクエリ実施結果の最大サイズ(バイト)を指定します。

例: `query_cache_limit=2M`

query_cache_size

クエリキャッシュのキャッシュサイズ(バッファ)を指定します。クエリキャッシュは `query_cache_limit` で指定されるサイズ以下のクエリ実施結果をキャッシュします。同一クエリ実施時にキャッシュからクエリ結果を取得します。サーバ上のデータが更新された場合、対象となるキャッシュは破棄されます。

クエリキャッシュの利用状況は、SHOW STATUS 文で `Qcache_free_blocks`、`Qcache_free_memory`、`Qcache_hits`、などの値から確認することができます。

例: `query_cache_size=64M`

read_buffer_size

テーブルをシーケンシャルに読み込む場合に作成・利用されるバッファサイズ(バイト)を指定します。このバッファは接続クライアント毎に、シーケンシャルに読み込むテーブルの数だけ作成されます。このため、利用状況によっては MySQL サーバプロセスのメモリが不足する可能性があります。

例: `read_buffer_size=8M`

read_rnd_buffer_size

行読み込み処理時に利用されるバッファサイズ(バイト)を指定します。このサイズを大きくすると ORDER BY によるソート処理パフォーマンスが向上します。このバッファはクライアント毎に作成されるため、大きな値を設定すると MySQL サーバプロセスのメモリが不足する可能性があります。必要なクライアントのみがセッション変数を変更することで、このバッファサイズを大きくするべきです。

例: `read_rnd_buffer_size=16M`

skip_innodb

InnoDB ストレージエンジンを利用しない場合に指定します。

例: `skip_innodb`

socket

MySQL サーバのソケットファイルを指定します。ソケットファイルは MySQL サーバ起動時に作成されます。クライアントが MySQL サーバと同一マシン上で動作している場合に指定されているソケットファイルを利用して接続します。Linux で RPM 版をインストールした場合、デフォルト値は `/var/lib/mysql/mysql.sock` となります。その他の場合は `/tmp/mysql.sock` が利用されます。2 つ以上の MySQL サーバを利用する場合は、それぞれ異なるソケットファイル名を指定する必要があります。

例: `socket=/var/lib/mysql/mysql.sock`

sort_buffer_size

ORDER BY と GROUP BY によるソート処理に利用するためのバッファサイズ(バイト)を指定します。バッファサイズを大きくすればより高速にソート処理を実行できます。しかし接続クライアント毎に作成されるため、クライアント数が多い場合に MySQL サーバプロセスのメモリが不足する可能性があります。

例: `sort_buffer_size=8M`

sync_binlog

バイナリログへの書き込み時に、ファイル書き込みの同期を行うか設定します。同期を有効(1 に設定)した場合、MySQL サーバはトランザクション毎にファイル書き込みの同期を行います。トランザクション情報を確実に記録する必要がある場合は、同期を有効にすべきです。

同期を無効(0 に設定)した場合、MySQL サーバは同期を行わず、オペレーティングシステムによってファイルの同期が行われることが期待されます。デフォルトは無効(0 に設定)されています。

例: `sync_binlog=1`

table_cache

MySQL サーバ全体でオープンすることのできるテーブルに対するファイル・ディスクリプタの最大数を指定します。簡単にはクライアントの数 × 一時にアクセスするテーブルの数以上の数値を設定する必要があります。

MySQL サーバ稼働時にテーブルキャッシュの状態を確認するには、SHOW STATUS 文を利用して得られる `Open_tables` の値で確認できます。

例: `table_cache=1024`

thread_cache_size

再利用するスレッドの数を指定します。MySQL サーバではクライアントとのコネクション毎にスレッドが 1 つ必要となります。MySQL サーバはコネクションが切断された時、`thread_cache_size` の数までスレッドを再利用するために保持します。この数を超えたスレッドは破棄されます。

クライアントと MySQL サーバ間のコネクションの接続・切断が多い場合にこの値を大きくすると、スレッドの再利用により、スレッド生成のオーバーヘッドを抑止することができます。

例: `thread_cache_size=8`

tmp_table_size

テンポラリ・テーブルの最大サイズ(バイト)を指定します。実際のテンポラリ・テーブルのサイズは `tmp_table_size` と `max_heap_table_size` の小さい方が採用されます。`max_heap_table_size` と同一の値を設定すべきでしょう。テンポラリ・テーブルはソート処理やその他クエリ実施時に利用されます。このサイズを超えるテンポラリ・テーブルはファイルに書き出されます。

例: `tmp_table_size=64M`

お問い合わせはカスタマー インフォメーションセンターへ

03-5304-6660 月～金9:00～19:00 土10:00～18:00(日、祝祭日、年末年始および5/1を除く)

Linux/オープンソース製品に関する情報は <http://www.hp.com/jp/linux/>

記載されている会社名および商品名は、各社の商標または登録商標です。

記載事項は 2007 年 4 月現在のものです。

本書に記載された内容は、予告なく変更されることがあります。

HP 製品とサービスに対する保証は、それらに付属する保証書に記載された事項に限られます。

ここに記載した内容は一切追加の保証を意味するものではありません。

本書中の技術的あるいは編集上の誤り、省略に対して、

いかなる責任も負いかねますのでご了承ください。

(c) Copyright 2007 Hewlett-Packard Development Company, L.P.

日本ヒューレット・パッカート株式会社

〒102-0076 東京都千代田区五番町 7

