

## (一)连接

连接通常来自Web服务器，下面列出了一些与连接有关的参数，以及该如何设置它们。

### 1、max\_connections

这是Web服务器允许的最大连接数，记住每个连接都要使用会话内存(关于会话内存，文章后面有涉及)。

### 2、max\_packet\_allowed

最大数据包大小，通常等于你需要在一个大块中返回的最大数据集的大小，如果你在使用远程mysqldump，那它的值需要更大。

### 3、aborted\_connects

检查系统状态的计数器，确定其没有增长，如果数量增长说明客户端连接时遇到了错误。

### 4、thread\_cache\_size

进站连接会在MySQL中创建一个新的线程，因为MySQL中打开和关闭连接都很廉价，速度也快，它就没有象其它数据库，

如Oracle那么多持续连接了，但线程预先创建并不会节约时间，这就是为什么要MySQL线程缓存的原因了。

如果在增长请密切注意创建的线程，让你的线程缓存更大，对于2550或100的thread\_cache\_size，内存占用也不多。

## (二)查询缓存

## (三)临时表

内存速度是相当快的，因此我们希望所有的排序操作都在内存中进行，我们可以通过调整查询让结果集更小以实现内存排序，或将变量设置得更大。

tmp\_table\_size

max\_heap\_table\_size

无论何时在MySQL中创建临时表，它都会使用这两个变量的最小值作为临界值，除了在磁盘上构建临时表外，还会创建许多会话，这些会话会抢占有 限制的资源，

因此最好是调整查询而不是将这些参数设置得更高，同时，需要注意的是有BLOB或TEXT字段类型的表将直接写入磁盘。

## (四)会话内存

MySQL中每个会话都有其自己的内存，这个内存就是分配给SQL查询的内存，因此你想让它变得尽可能大以满足需要。但你不得不平衡同一时间数据库内一致性会话的数量。

这里显得有点黑色艺术的是MySQL是按需分配缓存的，因此，你不能只添加它们并乘以会话的数量，这样估算下来比MySQL典型的使用要大得多。

最佳做法是启动MySQL，连接所有会话，然后继续关注顶级会话的VIRT列，mysqld行的数目通常保持相对稳定，这就是实际的内存总用量，

减去所有的静态MySQL内存区域，就得到了实际的所有会话内存，然后除以会话的数量就得到平均值。

#### 1、read\_buffer\_size

缓存连续扫描的块，这个缓存是跨存储引擎的，不只是MyISAM表。

#### 2、sort\_buffer\_size

执行排序缓存区的大小，最好将其设置为1M-2M，然后在会话中设置，为一个特定的查询设置更高的值。

#### 3、join\_buffer\_size

执行联合查询分配的缓存区大小，将其设置为1M-2M大小，然后在每个会话中再单独按需设置。

#### 4、read\_rnd\_buffer\_size

用于排序和order by操作，最好将其设置为1M，然后在会话中可以将其作为一个会话变量设置为更大的值。

## (五)慢查询日志

慢速查询日志是MySQL很有用的一个特性。

#### 1、log\_slow\_queries

MySQL参数中log\_slow\_queries参数在my.cnf文件中设置它，将其设置为on，默认情况下，MySQL会将文件放到数据目录，文件以“主机名-slow.log”的形式命名，但你在设置这个选项的时候也可以为其指定一个名字。

#### 2、long\_query\_time

默认值是10秒，你可以动态设置它，值从1到将其设置为on，如果数据库启动了，默认情况下，日志将关闭。

截至5.1.21和安装了Google补丁的版本，这个选项可以以微秒设置，这是一个了不起的功能，因为一旦你消除了所有查询时间超过1秒的查询，说明调整非常成功，这样可以帮助你在问题变大之前消除问题SQL。

#### 3、log\_queries\_not\_using\_indexes

开启这个选项是个不错的主意，它真实地记录了返回所有行的查询。

## 小结

我们介绍了MySQL参数的五大类设置，平时我们一般都很少碰它们，在进行MySQL性能调优和故障诊断时这些参数还是非常有用的。

MySQL中的缓存查询包括两个解析查询计划，以及返回的数据集，如果基础表数据或结构有变化，将会使查询缓存中的项目无效。

#### 1、query\_cache\_min\_res\_unit

MySQL参数中query\_cache\_min\_res\_unit查询缓存中的块是以这个大小进行分配的，使用下面的公式计算查询缓存的平均大小，根据计算结果设置这个变量，MySQL就会更有效地使用查询缓存，缓存更多的查询，减少内存的浪费。

#### 2、query\_cache\_size

这个参数设置查询缓存的总大小。

#### 3、query\_cache\_limit

这个参数告诉MySQL丢掉大于这个大小的查询，一般大型查询还是比较少见的，如运行一个批处理执行一个大型报表的统计，因此那些大型结果集不应该填满查询缓存。

view plain copy

```
qcache hit ratio = qcache_hits / (qcache_hits + com_select)
```

view plain copy

```
SQL> show status like 'qcache%';
```

```
SQL> show status like 'com_%';
```

view plain copy

```
average query size = (query_cache_size -  
qcache_free_memory)/qcache_queries_in_cache
```

view plain copy

```
SQL> show variables like 'query%';
```

qcache\_\* status variables you can get with:

```
SQL> show status like 'qcache%';
```