

## MySQL性能优化之参数配置

打开MySQL配置文件my.cnf

修改back\_log参数值:由默认的50修改为500 back\_log=500

back\_log值指出在MySQL暂时停止回答新请求之前的短时间内多少个请求可以被存在堆栈中。也就是说，如果MySQL的连接数达到max\_connections时，新来的请求将会被存在堆栈中，以等待某一连接释放资源，该堆栈的数量即back\_log，如果等待连接的数量超过back\_log，将不被授予连接资源。

查看mysql 当前系统默认back\_log值，命令：

show variables like 'back\_log'; 查看当前数量

**修改wait\_timeout参数值，由默认的8小时，修改为30分钟。**

wait\_timeout=1800（单位为秒）

show variables like 'wait\_timeout'; 查看值

**修改max\_connections参数值，由默认的151，修改为3000**

max\_connections=3000

**修改max\_user\_connections值，由默认的0，修改为800**

max\_user\_connections=800

针对某一个账号的所有客户端并行连接到MySQL服务的最大并行连接数。简单说是指同一个账号能够同时连接到mysql服务的最大连接数。设置为0表示不限制。

查看max\_user\_connections值

show variables like 'max\_user\_connections';

**修改thread\_concurrency值，由目前默认的8，修改为64 thread\_concurrency=64**

thread\_concurrency应设为CPU核数的2倍. 比如有一个双核的CPU, 那thread\_concurrency的应该为4; 2个双核的cpu, thread\_concurrency的值应为8.

show variables like 'thread\_concurrency';

default-storage-engine(设置MySQL的默认存储引擎)

default-storage-engine= InnoDB

创建表并指定存储类型

```
CREATE TABLE mytable (id int, title char(20)) ENGINE = INNODB;
```

```
show variables like '%max_connections%';
```