# 前言：

我们究竟应该如何对mysql数据库进行优化？下面就从mysql对硬件的选择、mysql的安装、my.cnf的优化、mysql如何进行架构设计及数据切片等方面来说明这个问题。

# 1-服务器物理硬件的优化

在挑选硬件服务器时，我们应该从下面几个方面着重对mysql服务器的硬件配置进行优化，也就是说将项目中的资金着重投入如下几项：

1、磁盘寻道能力（即磁盘I/O），我们现在通常使用的的都是SAS 15000转的硬盘，用6块这样的硬盘做RAID1+0。mysql每一秒都在进行大量、复杂的查询操作，对磁盘的读写量可想而知，所以，通常认为磁盘I/O是限制mysql性能的最大因素之一。对于日均访问量在100万PV以上的论坛（Discuz）、博客（Wordpress），如果性能不好，造成的直接后果就是mysql的性能会非常低下。解决这一制约因素可以考虑解决方法是：使用RAID1+0磁盘阵列，注意不要尝试RAID5，mysql在RAID5磁盘阵列上的效率不会像你期待的那样快，如果资金允许，可以选择固态硬盘SSD来替代SAS硬盘做RAID1+0。

2、CPU对于mysql的影响也是不容忽视的，建议选择运算能力强悍的CPU。推荐使用DELL R710（双四核），商家的卖点也是其强大的虚拟化和数据库能力。

3、对于一台使用mysql的Database Server来说，建议服务器的内存不要小于2GB，推荐使用4GB以上的物理内存，不过内存对于现在的服务器而言可以说是一个可以忽略的问题，如果是高端服务器，基本上内存都超过了32GB，我们的数据流服务器都是32GB内存。

我们工作中用的比较多的数据库服务器是DELL R710/R720，其稳定性和性能都不错，同时我也发现许多同行都是采用它作为数据库的服务器，所以在这里也向大家推荐一下。

# 2-Mysql安装方式

关于mysql数据库的线上环境安装，建议采取编译安装的方式，这样性能会有较大的提升。服务器系统则建议使用CentOS 6.7 X86\_64.，源码包的编译参数会默认以Debug模式生成二进制代码，而Debug模式给mysql带来的性能损失是比较大的，所以当我们编译准备安装的产品代码时，一定不要忘记使用-without-debug参数禁止Debug模式。如果把-with-mysqld-ldflags和-with-client-ld-flags两个编译参数设置为-all-static的话，可以告诉编译器以静态的方式编译，编译结果将得到最高的性能。使用静态编译和使用动态编译的代码相比，性能差距可能会达到5%至10%之多。后面会分享线上mysql数据库的编译参数，可以参考，然后根据自己的线上环境自行修改内容。

# 3-mysql配置文件优化

[client]

#password = [your\_password]

port = 3306

socket = /application/mysql-5.5.32/tmp/mysql.sock

default-character-set = utf8 # 客户端字符集 (控制character\_set\_client character\_set\_connection character\_set\_results)

[client]

port = 3306

socket = /application/mysql-5.5.32/tmp/mysql.sock

[mysqld] # 组包括了mysqld服务启动的参数，它涉及的方面很多，其中mysql的目录和文件，通信、网络、信息安全，内存管理、优化、 查询缓存区，还有mysql日志设置等

user = mysql # mysql\_safe脚本使用mysql运行用户（编译时user=mysql指定），推荐使用mysql用户

port = 3306 # mysql服务运行时的端口号，建议更改默认端口号，默认端口易被攻击

socket = /application/mysql-5.5.32/tmp/mysql.sock

# socket文件是在Linux/Unix环境下特有的，用户在Linux/Unix环境下客户端连接可以不通过TCP/IP网络而直接使用socket连接MySQL。

basedir = /application/mysql # mysql程序所存放路径,常用于存放mysql启动、配置文件、日志等

datadir = /data/3306/data  # mysql数据存放文件(极其重要)

character-set-server = utf8  # 数据库和数据库表的默认字符集。(推荐utf8,以免导致乱码)

back\_log = 50

max\_connections = 100

max\_connect\_errors = 10

table\_open\_cache = 2048

max\_allowed\_packet = 16M

binlog\_cache\_size = 1M

max\_heap\_table\_size = 64M

read\_buffer\_size = 2M

read\_rnd\_buffer\_size = 16M

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-storage-engine = MYISAM

thread\_stack = 192K

transaction\_isolation = REPEATABLE-READ

tmp\_table\_size = 64M

log-bin=mysql-bin

binlog\_format=mixed

slow\_query\_log

long\_query\_time = 2

server-id = 1

key\_buffer\_size = 32M

bulk\_insert\_buffer\_size = 64M

myisam\_sort\_buffer\_size = 128M

myisam\_max\_sort\_file\_size = 10G

myisam\_repair\_threads = 1

myisam\_recover

innodb\_additional\_mem\_pool\_size = 16M

innodb\_buffer\_pool\_size = 2G

innodb\_data\_file\_path = ibdata1:10M:autoextend

innodb\_write\_io\_threads = 8

innodb\_read\_io\_threads = 8

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

[mysqldump]

quick

max\_allowed\_packet = 16M

[mysql]

no-auto-rehash # 仅仅允许使用键值的updates和deletes

[myisamchk]

key\_buffer\_size = 512M

sort\_buffer\_size = 512M

read\_buffer = 8M

write\_buffer = 8M

[mysqlhotcopy]

interactive-timeout

[mysqld\_safe]

open-files-limit = 8192

2-mysql采用编译方式安装