测试一：数据插入性能测试，这里我分别对innodb\_flush\_log\_at\_trx\_commit参数打开和关闭都测了了一下，每次测试都是运行40s，表中数字都是实际插入条数。

|  |  |  |  |
| --- | --- | --- | --- |
|  | MyISAM | Innodb(打开) | Innodb(关闭) |
| 单线程，逐个插入 | 120000 | 60000 | 60000 |
| 4线程，逐个插入 | 40000\*4 | 20000\*4 | 40000\*4 |
| 单线程，批量100条/次插入 | 3600\*100 | 800\*100 | 3000\*100 |
| 单线程，批量200条/次插入 | 1800\*200 | 400\*200 | 1600\*200 |

可以发现批量插入的性能远高于单条插入，但是一次批量的大小对性能影响不大。每条记录是否都刷新日志的参数对innodb性能的影响巨大。总体上来说，MyISAM性能更优一点。这里有一点需要注意，在插入测试过程中，我对系统资源进行了监控，发现MyISAM对系统资源占用很低，但是Innodb对磁盘占用却很高，应该是对事务控制多了很多需要记录的日志。

测试二：数据读取性能测试。每次随机读取1000条记录，反复进行读取。

|  |  |  |
| --- | --- | --- |
|  | MyISAM | Innodb |
| 单线程，200次读取 | 5.7s | 16.7s |
| 4线程，200次读取 | 12s | 40.8s |

可以看出MyISAM的读取性能非常恐怖，性能差距在3倍的样子。

以上两个测试发现MyISAM在无事务的需求下几乎完胜，但是要知道它是表锁，Innodb是行锁，那么在并发读写同时存在的情况下，那结果会是怎么样呢？！

测试三：两个线程并发写入，2个线程并发读取。

|  |  |  |
| --- | --- | --- |
|  | MyISAM | Innodb |
| 逐个插入 | 写入40s：10000\*2 读取200次\*2：14s | 写入40s：60000\*2 读取200次\*2：50s |
| 批量100条/次插入 | 写入40s：1000\*100\*2 读取200次\*2：10s | 写入40s：1500\*100\*2 读取200次\*2：50s |

这下立刻显示出Innodb在并发情况下强劲的性能，几乎没有什么性能衰减。而MyISAM单条插入速度变得非常慢，批量插入也下降了40%性能。

    总结一下，在写多读少的应用中还是Innodb插入性能更稳定，在并发情况下也能基本，如果是对读取速度要求比较快的应用还是选MyISAM。

    另外提一下，这里需要用到对Innodb的热备份，除了用Master-Slave的方式，还可以选用XtraBackup这个开源软件。