Atitit insert delayed mysql8

由于您撰写的文章多于阅读的文章，因此我建议以下内容

# **体面的InnoDB调整将是关键**

## 缓冲池（由**[innodb\_buffer\_pool\_size调整大小](http://dev.mysql.com/doc/refman/5.5/en/innodb-parameters.html" \l "sysvar_innodb_buffer_pool_size)**）

由于**[InnoDB不支持INSERT DELAYED](http://bugs.mysql.com/bug.php?id=5777)**，因此使用大型InnoDB缓冲池是最接近INSERT DELAYED的方法。所有DML（INSERT，UPDATE和DELETE）都将被缓存在InnoDB缓冲池中。Writes的事务性信息将立即写入重做日志（ib\_logfile0，ib\_logfile1）。通过ibdata1（二级索引的InsertBuffer，双写缓冲区）定期将缓冲区池中发布的写入从内存刷新到磁盘。缓冲池越大，可以缓存的INSERT数量就越大。在具有8GB或更多RAM的系统中，将75-80％的RAM用作innodb\_buffer\_pool\_size。在RAM很少的系统中，为25％（以容纳OS）。

CAVEAT：您可以将**[innodb\_doublewrite](http://dev.mysql.com/doc/refman/5.5/en/innodb-parameters.html" \l "sysvar_innodb_doublewrite)**设置为0来加快写入速度，但是有数据完整性的风险。您还可以通过将[innodb\_flush\_method](http://dev.mysql.com/doc/refman/5.5/en/innodb-parameters.html" \l "sysvar_innodb_flush_method)设置为O\_DIRECT来加快处理速度，以防止将InnoDB缓存到操作系统。

## 重做日志（大小为**[innodb\_log\_file\_size](http://dev.mysql.com/doc/refman/5.5/en/innodb-parameters.html" \l "sysvar_innodb_log_file_size)**）

缺省情况下，重做日志名为ib\_logfile0和ib\_logfile1，每个重做日志为5MB。该大小应为innodb\_buffer\_pool\_size的25％。*[如果重做日志已经存在，则在my.cnf中添加新设置，关闭mysql，删除它们，然后重新启动mysql](https://dba.stackexchange.com/a/1265/877)*。

## 日志缓冲区（由**[innodb\_log\_buffer\_size调整大小](http://dev.mysql.com/doc/refman/5.5/en/innodb-parameters.html" \l "sysvar_innodb_log_buffer_size)**）

日志缓冲区将更改保存到RAM中，然后再将其刷新到重做日志中。默认值为8M。日志缓冲区越大，磁盘I / O越少。对于非常大的事务要小心，因为这可能会使COMMIT的速度降低几毫秒。

## 访问多个CPU

MySQL 5.5和MySQL 5.1 InnoDB插件具有可让InnoDB存储引擎访问多个CPU的设置。这是您需要设置的选项：

* [innodb\_thread\_concurrency](http://dev.mysql.com/doc/refman/5.5/en/innodb-parameters.html" \l "sysvar_innodb_thread_concurrency)设置InnoDB可以保持打开状态的并发线程数的上限。通常建议为此设置为（2 X CPU数量）+磁盘数量。去年，我从Percona NYC大会上第一手了解到，您应该将其设置为0，以提醒InnoDB Storage Engine为正在运行的环境找到最佳线程数。
* [innodb\_concurrency\_tickets](http://dev.mysql.com/doc/refman/5.5/en/innodb-parameters.html" \l "sysvar_innodb_concurrency_tickets)设置可以绕过并发检查而不受惩罚的线程数。达到该限制之后，线程并发检查再次成为常态。
* [innodb\_commit\_concurrency](http://dev.mysql.com/doc/refman/5.5/en/innodb-parameters.html" \l "sysvar_innodb_commit_concurrency)设置可以提交的并发事务数。由于默认值为0，因此未设置此选项将允许任意数量的事务同时提交。
* [innodb\_thread\_sleep\_delay](http://dev.mysql.com/doc/refman/5.5/en/innodb-parameters.html" \l "sysvar_innodb_thread_sleep_delay)设置在重新输入InnoDB队列之前InnoDB线程可以处于休眠状态的毫秒数。默认值为10000（10秒）。
* [innodb\_read\_io\_threads](http://dev.mysql.com/doc/refman/5.5/en/innodb-parameters.html" \l "sysvar_innodb_read_io_threads)（将其设置为3000）和[innodb\_write\_io\_threads](http://dev.mysql.com/doc/refman/5.5/en/innodb-parameters.html" \l "sysvar_innodb_write_io_threads)（将其设置为7000）（均从MySQL 5.1.38开始）为读取和写入分配指定数量的线程。默认值为4，最大值为64。将其设置为64。此外，将[innodb\_io\_capacity](http://dev.mysql.com/doc/refman/5.5/en/innodb-parameters.html" \l "sysvar_innodb_io_capacity)设置为10000。