

From: <https://www.linuxidc.com/Linux/2017-09/146671.htm>

`innodb_flush_log_at_trx_commit` 和 `sync_binlog` 是 MySQL 的两个配置参数。它们的配置对于 MySQL 的性能有很大影响（一般为了保证数据的不丢失，会设置为双 1，该情形下数据库的性能也是最低的）。

1、innodb_flush_log_at_trx_commit

`innodb_flush_log_at_trx_commit`：是 InnoDB 引擎特有的，`ib_logfile` 的刷新方式（`ib_logfile`：记录的是 redo log 和 undo log 的信息）

取值：**0/1/2**

`innodb_flush_log_at_trx_commit=0`，表示每隔一秒把 log buffer 刷到文件系统中(os buffer)去，并且调用文件系统的“flush”操作将缓存刷新到磁盘上去。也就是说一秒之前的日志都保存在日志缓冲区，也就是内存上，如果机器宕掉，可能丢失 1 秒的事务数据。

`innodb_flush_log_at_trx_commit=1`，表示在每次事务提交的时候，都把 log buffer 刷到文件系统中(os buffer)去，并且调用文件系统的“flush”操作将缓存刷新到磁盘上去。这样的话，数据库对 IO 的要求就非常高了，如果底层的硬件提供的 IOPS 比较差，那么 MySQL 数据库的并发很快就会由于硬件 IO 的问题而无法提升。

`innodb_flush_log_at_trx_commit=2`，表示在每次事务提交的时候会把 log buffer 刷到文件系统中去，但并不会立即刷写到磁盘。如果只是 MySQL 数据库挂掉了，由于文件系统没有问题，那么对应的事务数据并没有丢失。只有在数据库所在的主机操作系统损坏或者突然掉电的情况下，数据库的事务数据可能丢失 1 秒之类的事务数据。这样的好处，减少了事务数据丢失的概率，而对底层硬件的 IO 要求也没有那么高(log buffer 写到文件系统中，一般只是从 log buffer 的内存转移的文件系统的内存缓存中，对底层 IO 没有压力)。

2、sync_binlog

`sync_binlog`：是 MySQL 的二进制日志（binary log）同步到磁盘的频率。

取值：**0-N**

`sync_binlog=0`，当事务提交之后，MySQL 不做 `fsync` 之类的磁盘同步指令刷新 `binlog_cache` 中的信息到磁盘，而让 Filesystem 自行决定什么时候来做同步，或者 cache 满了之后才同步到磁盘。这个是性能最好的。

`sync_binlog=1`，当每进行 1 次事务提交之后，MySQL 将进行一次 `fsync` 之类的磁盘同步指令来将 `binlog_cache` 中的数据强制写入磁盘。

sync_binlog=n，当每进行 n 次事务提交之后，MySQL 将进行一次 fsync 之类的磁盘同步指令来将 binlog_cache 中的数据强制写入磁盘。

注：

大多数情况下，对数据的一致性并没有很严格的要求，所以并不会把 sync_binlog 配置成 1。为了追求高并发，提升性能，可以设置为 100 或直接用 0。

而和 innodb_flush_log_at_trx_commit 一样，对于支付服务这样的应用，还是比较推荐 sync_binlog = 1。