

steal:

是否允许一个 uncommitted 的事务将修改更新到磁盘，如果是 steal 策略，那么此时磁盘上就可能包含 uncommitted 的数据，因此系统需要记录 undo log，以防事务 abort 时进行回滚（roll-back）。如果是 no steal 策略，就表示磁盘上不会存在 uncommitted 数据，因此无需回滚操作，也就无需记录 undo log。

force:

force 策略表示事务在 committed 之后必须将所有更新立刻持久化到磁盘，这样会导致磁盘发生很多小的写操作（更可能是随机写）。no-force 表示事务在 committed 之后可以不立即持久化到磁盘，这样可以缓存很多的更新批量持久化到磁盘，这样可以降低磁盘操作次数（提升顺序写），但是如果 committed 之后发生 crash，那么此时已经 committed 的事务数据将会丢失（因为还没有持久化到磁盘），因此系统需要记录 redo log，在系统重启时候进行前滚（roll-forward）操作。

总结

steal/no-steal 主要决定了磁盘上是否会包含 uncommitted 的数据。force/no-force 主要决定了磁盘上是否会不包含已经 committed 的数据。在 DBMS 常用的是 steal/no-force 策略，因此一般都需要记录 redo log 和 undo log。这样可以获得较快的运行时性能，代价就是在数据库恢复（recovery）的时候需要做很多的事情，增大了系统重启的时间。

下图概括了各种策略搭配下的性能和要记录的 log 类型。

	No Steal	Steal
No Force		Fastest
Force	Slowest	

	No Steal	Steal
No Force	No UNDO REDO	UNDO REDO
Force	No UNDO No REDO	UNDO No REDO