深入理解 **InnoDB** 缓冲池：性能调优指南

Duthaho · 2026年9月28日

MYSQL **INNODB** **BUFFER-POOL** **PERFORMANCE**

InnoDB buffer pool
Cache pages in memory · watch misses, eviction and flushing
MEMORY
Young · frequently used
Old · recent scans
Red marker: modified page awaiting flush
flush
read
DISK

InnoDB 使用内存来避免每次访问都从磁盘读取数据页。缓冲池保存近期访问过的数据页和索引页；写入操作先修改内存中的页面，再将其刷回磁盘。弄清哪些页面进入缓冲池、哪些页面被淘汰，以及预热需要多长时间，有助于解释稳定

缓冲池中有什么

`innodb_buffer_pool_size` 决定这块共享内存的大小。InnoDB [illegible] 16 KiB，[illegible]数据页、索引页和内部结构共同占用缓冲池。已在内存中的页面无需再读数据文件；缺失的页面

专用数据库服务器可能适合较大的缓冲池，[illegible]固定的内存百分比当作通用配置。操作系统、连接、临时操作、其他存储引擎和同机服务都需要内存。64 GiB 内存的主机配置 48 GiB 缓冲池只是

应在相同时间区间内测量读取量，不要只看从服务器启动以来累计的百分比：

```
SHOW GLOBAL STATUS WHERE Variable_name IN (
  'Innodb_buffer_pool_read_requests',
  'Innodb_buffer_pool_reads',
  'Innodb_buffer_pool_pages_dirty',
  'Innodb_buffer_pool_wait_free'
);
```

区[illegible]的未命中率为 `buffer_pool_reads / ΔInnodb_buffer_pool_read_requests`。[illegible]果该比率、物理读[illegible]量和[illegible]询延迟同[illegible]上升，说明活跃[illegible]集可能[illegible]。命中

为什么大范围扫描未必会赶走所有热点页面

InnoDB 使用经过改进的 [illegible] 列表[illegible]插入方式可减轻一次性全表扫描对热点工作集的冲击。

`innodb_old_blocks_pct` 控制 old 区域在列表中的比例（MySQL 8.4 [illegible]认值[illegible]37）[illegible]s_time 控制新页面晋升前的等待时间（MySQL 8.4 默认 1000 毫秒）。延长等待

[illegible]

[illegible] MySQL [illegible] InnoDB [illegible] `innodb_buffer_pool_dump_at_shutdown` [illegible] `innodb_buffer_pool_load_at_startup` [illegible]

`innodb_buffer_pool_dump_pct` [illegible]

[illegible] I/O[illegible] 80% [illegible]

[illegible]redo [illegible]

[illegible] InnoDB [illegible] redo [illegible] `innodb_flush_log_at_trx_commit=1` [illegible] redo [illegible] I/O[illegible]

Change Buffer [illegible] [illegible] MySQL `innodb_buffer_pool_instances=8` [illegible] MariaDB 10.5

[illegible]

[illegible] I/O[illegible] SQL [illegible]

[illegible] MySQL 8.4 [illegible] GiB [illegible] 1 GiB [illegible] 99% [illegible]

[illegible]

[illegible]

[illegible] Medium [illegible] 2025 [illegible] 8.2 [illegible] MariaDB [illegible]