

分析SQLServer的内存为何不断增加 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/214/2021_2022__E5_88_86_E6_9E_90SQLS_c67_214503.htm 当SQL Server 数据库引擎在 Microsoftreg. 或 Windows®. 2000 上运行时，其默认内存管理行为并不是获取特定的内存量，而是在不产生多余换页 I/O 的情况下获取尽可能多的内存。为此，数据库引擎获取尽可能多的可用内存，同时保留足够的可用内存以防操作系统交换内存。SQL Server 实例在启动时通常获取8MB到12MB 的内存以完成初始化过程。当实例完成初始化后，就不会再获取更多的内存，直到用户连接到该实例并开始产生工作负荷。这时，该实例根据需要不停地获取内存以支持工作负荷。随着更多的用户连接并运行查询，SQL Server 将获取支持需求所需的额外内存。该实例将继续获取内存直到达到自身的内存分配目标，并且直到达到该目标的下限才会释放任何内存。为了在不产生多余换页 I/O 的情况下获取尽可能多的内存，SQL Server 的每个实例都设置一个内存获取目标，直到计算机的可用物理内存在4MB到10MB的范围内。之所以选择该范围是因为测试表明 Windows NT 和 Windows 2000 都有最小内存交换，直到内存分配等于可用物理内存减去4MB。工作负荷处理任务重的 SQL Server 实例保留的可用物理内存为范围的较低端 (4MB)；工作负荷处理任务轻的实例保留的可用物理内存为范围的较高端 (10MB)。SQL Server 实例的目标随工作负荷的改变而变化。当更多的用户连接并产生更多的工作时，该实例倾向于获取更多的内存以使可用的内存保持在4MB 的限制以下。当工作负荷减轻时，该实例将其目标调

整为10MB的可用空间，并释放内存给操作系统。将可用空间量保持在10MB与4MB之间可防止 Windows NT 或 Windows 2000 过多执行换页操作，同时使 SQL Server 得以获得尽可能最大的高速缓冲存储器而不至引起额外的交换。实例的目标内存设置与数据库缓冲池的页相对于可用池大小的需求有关。在任何即时点，缓冲区页的总需求取决于满足所有当前执行的查询所需的数据页数。如果相对于高速缓冲存储器内的页数，数据页的需求很大，则当前在缓冲区内的每一页很可能在相对较短的时间内由新页替换。这可由“缓冲区管理器”对象的“页生命期”性能计数器来度量。对于相对较小的缓冲区有较高需求的情况将生成短生命期，而纯粹的影响就是使 I/O 增加，因为在页可由多个逻辑读取引用之前往往要被重写。为减轻这个问题，数据库引擎可以获取更多的内存以增加高速缓冲存储器的大小。当页生命期长时，数据库引擎将可用内存定位于目标的高端 (10MB)；而当页生命期短时，数据库引擎定位于目标范围的低端 (4MB)。随着其它应用程序在运行 SQL Server 实例的计算机上启动，它们消耗内存致使可用物理内存量降到 SQL Server 的目标以下。SQL Server 实例于是从其地址空间释放足够内存，以使可用内存量回到 SQL Server 的目标。如果有其它应用程序停止运行而使可用内存增多，SQL Server 实例将增加其内存分配大小。SQL Server 可以每秒释放并获取几MB字节的内存，这使它得以根据内存分配变化作出快速调整。100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com