

- 原因：随着现代处理器性能的不断提升，处理器与内存之间的速度差距不断扩大，形成“内存墙”的问题。为此，现代计算机使用多级分层存储结构，并引入缓存系统，利用数据的时间局部性和空间局部性来改善访存性能。
- 过大的页：①内部碎片，即每个页中可能存在未被使用的空间，浪费内存资源  
②页面置换开销：当需要替换一个页面时，如果页面大小过大，所需的磁盘I/O操作会更耗时  
过小的页：①外部碎片：进程在虚拟地址空间中分配的空间会更细粒度，产生更多外部碎片，无法利用。  
②页面表管理开销：过小的页大小意味着虚拟地址空间的划分更加细致，需要更多的页面表项来管理，这将增加页面表的大小和管理开销，包括页表的存储器空间和访问时间。
- 1) V位决定该页表项的其余部分是否有效 (V=1时有效)  
R, W, X位分别表示此页是否可以读取、写入和执行  
U位表示该页是否是用户页面  
G位表示这个映射是否对所有虚拟空间有效  
A位表示自从上次A位被清除以来，该页面是否被访问过  
D位表示自从上次清除D位以来页面是否被弄脏（例如被写入）
  - 可能发生的问题：安全性问题、内存访问错误、内存泄漏和资源浪费、多进程共享问题
  - 若X/W/R位全部为0，说明这个页表项是指向下一级页表的指针，否则它是页表树的一个叶节点
- PMP控制寄存器中的X/W/R位用于进行物理内存的保护控制，是硬件层面的保护机制
  - A域设置是否启用此PMP，L域锁定了PMP和对应的地址寄存器
- 64位虚拟地址空间为 $2^{64}$ 个地址，每个页的大小为 $2^{12}$ 字节(4KB)，所需页表条目数量：  
 $2^{64}/2^{12} = 2^{52}$   
每个页表条目占用8字节，所以一个单级页表系统用于存储页表的空间为：  
 $2^{52} \times 2^3 = 2^{55}$ 字节
  - 空间： $2^{48}/2^{12} \times 2^3 = 2^{39}$
  - 多级页表通过减少页表的总条目数，延迟分配页表，分级映射、共享页表，等方式降低虚拟内存系统的实际页表存储开销，提高内存利用率并减少存储资源的消耗。