

第十二周作业

1. 答: 现代计算机系统需要存储层级的原因是为了更有效地管理存储器资源, 提高系统性能和可靠性。

计算机的处理速度通常比内存存取速度快得多, 若CPU需要访问内存中的每个字节, 那么它'll需要等待数百个时钟周期, 将导致严重的性能问题。因此现代计算机使用了存储层级的概念, 通过使用多层缓存和虚拟内存等技术来加速内存访问, 并有效利用存储器资源。

计算机中的存储器有着不同的访问方式, 因此程序中的数据访问模式对存储器层级的性能影响很大, 为了更好地利用不同层级的存储介质, 需要对程序进行存储层级优化。

2. 答:

在页式虚拟存储中, 过大的页会浪费内存空间, 因为每个页大小固定, 若页的大小设置得太大, 可能导致程序中只使用了部分页中的数据而其余部分没有被使用, 造成内存空间的浪费。同时过大的页可能导致内存碎片问题, 若一个进程需要分配的虚拟地址空间不是页的整数倍, 则最后一页可能只有部分被利用, 导致内部碎片。

相反, 过小的页会导致内存中需要存储更多的页表项, 因为每个页表项只能映射一个固定大小的页, 因此需要更多的页表项来覆盖整个地址空间。这会增加页表的大小, 导致更多的内存开销和更长的访问时间, 同时也可能导致内存空间的浪费。

3. (1) 答: 位0 (V): 有效位, 为1表示该页表有效, 0表示无效

位1 (R): 读权限, 为1表示允许读取该页, 0表示不允许读取

位2 (W): 写权限, 为1表示允许写入该页, 0表示不允许写入

位3 (X): 执行权限, 为1表示允许执行该页的代码, 0表示不允许执行

位4 (U): 用户/内核模式权限, 为1表示该页可以在用户模式下访问, 0表示只能在内核模式下访问。

位5 (G): 全局页面标志, 为1表示该页是全局页面, 可在所有地址空间中共享。

位6(A):访问位,为1表示该页已被访问,0表示未被访问。

位7(D):脏页标志,为1表示该页已被写入过,0表示未被写入过。

(2)答:如果用户进程能够自由修改自己的页表,可能会导致以下问题:用户进程可能会修改其它进程的页表,导致进程间相互影响;可能会修改操作系统的页表,导致系统崩溃或安全漏洞;可能会修改权限标记位,从而获取未授权的访问权限,例如将只读的页表转变为可写的,将内核模式的转变为为用户模式的,造成一系列问题。

(3)答:如果一个有效页表条目的X/W/R位全部为0,则表示该页不可读、不可写、不可执行。若此时访问该页将导致一个访问异常。这样的页表不允许被访问,通常用于占位,即为了保留一些虚拟地址空间而映射到了物理内存中的一个不存在的页。当该页需要被映射到物理内存时,该页表条目的各个标记位将被更新为合适的值。

4.(1)答:在页表条目中已经存在X/W/R位的情况下,PMP控制寄存器中的X/W/R位与页表中的作用不同。页表条目中的X/W/R位用于控制虚拟地址到物理地址的映射,以及页面是否可读、写、执行;而PMP控制寄存器中的X/W/R位则用于控制对物理内存区域的访问权限,以及是否允许页面被缓存等。如果PMP的控制位和页表条目中的控制位同时存在,则PMP的控制位~~和~~优先级更高。

(2)答:L位控制了内存区域是否可以被写入或者配置是否可以被修改。L位置为1时内存区域将被锁定,PMP区域的访问权限将不能通过M模式修改。可以保护操作系统和虚拟化软件的关键代码区域,防止应用程序通过修改PMP配置来绕过访问控制。

A位:用于指定地址匹配模式。A=0时使用地址掩码匹配模式,A=1时使用精确地址匹配模式。A置为1时,PMP控制寄存器的地址匹配范围将缩小到页表中设置的虚拟地址范围。

5. 解: (1) 一页可存储 $\frac{4KB}{8B} = 512$ 个页表条目

- 对一个单级页表系统, 虚拟地址中页号部分的位数为 $\log_2(4KB) = 12$

$$\therefore 2^{64-12} \times 8B = 2^{52} \times 8B = 32TB$$

(2) 在 Sv48 中, 虚拟地址只使用 48 位,

则一个单级页表系统需要的空间用于存储页表为:

$$2^{48-12} \times 8B = 2^{36} \times 8B = 256GB$$

(3) 多级页表可以通过将一个大的虚拟地址空间分成多个较小的部分来降低实际页表存储的开销。在多级页表中, 虚拟地址空间被划分成多个级别, 从而避免了存储一个单一的大型页表。需要访问虚拟地址时, 多级页表系统将虚拟地址划分成页目录、页表和页内偏移量, 并使用这些部分查找正确的物理地址。这种分层结构可以使每个页目录和页表更小, 从而降低了总的页表存储开销。