

4-1: 解: 存储器往往不能同时满足存取速度快、存储容量大、成本低的要求。为了满足大容量存储、CPU快速访问的要求, 现代计算机系统设置了存储层级, 最高层容量小速度快, 记录CPU最可能用到的数据, 满足CPU快速访问的需求, 后面的层级逐渐速度慢, 容量变大, 满足存储大量数据的需求。

4-2: 页过小: 可以减少内存碎片提高页空间利用率, 但会导致页表过长占用内存大、降低页面换进换出效率。

页过大: 缩短页表, 提高页换进换出速度, 但会使内存碎片增多降低了空间利用率。

4-3: (1) V: 是否有效, 1表示该页有效, 0表示无效。

R: 读权限, 1表示有权读取该页, 0表示无权。

W: 写权限, 1表示允许写入该页, 0表示不允许。

X: 执行权限, 1表示允许执行该页代码, 0表示不允许。

U: 用户/内核模式权限, 1表示用户、内核模式均可访问, 0表示仅内核模式可访问。

G: 全局页面标志, 1表示该全局页面可在所有地址空间共享。

No.

Date

A: 访问位, 1表示该页已被访问, 0表示未被访问.

D: 脏页标志, 1表示该页已被写入过, 0表示未被写入.

(4) ①可能会修改其他进程页表使程序异常.

②可能会修改操作系统页表, 使系统产生漏洞甚至崩溃.

③可能会修改权限标记位, 造成一系列问题.

(5) 表示不可读写/运行, 通常是为了保留一些虚拟地址(此时映射的物理空间不存在), 当需要时再把各个标记位更新为合适的值.

4-4: (1) 二者作用不同. 页表图中 X/W/R 用于控制虚拟内存到物理内存中的映射以及页面是否可读/写/执行. PMP中的则用于控制物理内存中实际数据的访问权限, 以及是否允许页面被缓存等. 通常PMP的控制位优先级更高.

(2) L: 内存区是否可被写入或配置是否可更改, L为1时内存区被锁定, PMP区访问权限不可通过M模式修改, 可以保护系统和关键代码区.

A: 指定地址区配模式. A=0使用地址掩码匹配模式,

A=1使用精确地址匹配模式. A为1时PMP地址区满范围小到页表中设置的虚拟地址范围.

$$4-5: (1) \frac{2^{64}}{4 \times 2^{10}} \times 8 = 2^{55} \quad B = 2^{45} \text{KB}$$

$$(2) \frac{2^{48}}{4 \times 2^{10}} \times 8 = 2^{39} \quad B = 2^{29} \text{KB}$$

(3) 单级页表要占用很多内存, 而多级页表还要在单级页表基础上加页表, 总体来说占用内存反而更大. 但值得注意的是二级页表

是可以不存页表的
从而省下来许多

No.

Date . . .

是可以不存在的,因为真实情况中大多数程序用不到所有的物理空间,从而省略下来许多二级页表,降低了存储开销.