

T1

① 因为现代计算机处理数据的速度越来越快，所以对存储的访问成为了限制性能的重要因素，因此需要更快的存储器。

② 成本：更快的存储意味着更高的价格，所以可以将一部分经常用的数据放在更快的存储器中，而大多数数据放在更慢、更大的便宜存储器中。

T2.

① 页过大可能会使得进程无法完全使用完一页，而页已被占用，从而导致内存利用率降低，产生大量内存碎片。

② 页过小，可能导致一个进程需要好多页，就导致页表与TLB的命中率上升，且检索速度变慢。

T3.

1) 0. 有效位：用于表示该页是否有效。

1. 可读位：用于指示是否允许从该页读取数据。

2. 可写位：用于指示是否可向该页写入数据。

3. 可执行位：用于指示是否允许从该页执行指令。

4. 用户位：用于指示该页是否属于用户空间。

5. 全局位：用于指示该页是否是全局的，也就是说被多个进程共享。

6. 已访问位：用于指示该页面是否已被访问

7. 痞位：用于指示该页面是否已被修改

2) 会发现：

- (1) 安全问题是页：用户可以通过修改页表，将存储敏感数据的页映射至页表且设置可读，导致数据泄密

- (2) 系统稳定性问题：可能使得进程修改了不应修改的数据，执行了不应执行的指令，从而导致系统崩溃

- (3) 多任务问题：若修改了共用页，则可能导致其它进程出现问题。

3) 可能其上的数据遭到污染

② 可能其还未被分配物理页素

T4 1) 能够进一步控制对内存的权限，即 X/W/R 权限集才是最终的访问权限

2) L=0 时，对条目的目标地址的保护只对用户模式与系统模式有效

当 L=1 时，所有模式包括机器模式都需要权限

A=0 时，对任何权限等级不施加 DMP 保护

A=1, 2, 3 时，分别采用 T0P, NA4, NA10 策略进行保护。

T5

1) 每个页 4KB, 因此需 $\log_2(4KB) = 12$ 位用于标记页内偏移; 有 52 位用于检查页表项. : 页表共有 2^{52} 个表项. 每个表项 8KB 即 2^{16} 位. : 总共 $2^{16} \times 16 = 2^{18}$ Bit

2) 因此有 $48 - 12 = 36$ 位用于检查表项

$$\therefore \text{总共 } 2^{36} \times 2^{16} = 2^{52} \text{ Bit}$$

3) 有时进程占用很少, 所以, 只需每级一个页表即可.
从而减少了页表的开销