

1. 成本和运算速度的平衡, 提供性价比. 一方面我们知道CPU的读取速度快, 但存储容量小.

2. 过小: 导致进程占用较多的页面, 导致页表过长, 占用大量内存;

过大: 减少了页表的长度, 提高换进换出的速度, 降低了利用率.

4.3 " V: 页表项的合法性

" 引发信息安全问题, 无权限的地址空间可能被访问

R, W, X: 读, 写, 执行权限

U: 可否由U使用

B: PTE为一个指向下一级页表的指针

G: 全局映射

A: 状态位

D: 脏位.

4 " P<sup>M</sup>中的XWR可以对原XWR进行覆写.

" L: 锁定P<sup>M</sup>配置寄存器中的设置. A: 匹配地址模式, 以确定访问权限.

4.5 "  $\frac{2^{64}}{(2^{18} \times 8)} \times 8 = 2^{52} B$  大小用于存储页表

" 同理  $\frac{2^{64}}{2^{18} \times 8} \times 8 = 2^{46} B$  大小.

" ① 节省内存空间: 使用多级结构, 分解为多级小页表

② 减少单页表的访问时间