

1.现代计算机系统需要存储层级的原因有以下几点:

可靠性和稳定性: 存储层级可以提供系统的可靠性和稳定性。通过将数据存储在不同的存储介质上, 可以确保数据的完整性和可靠性, 并减少数据丢失的风险。此外, 存储层级还可以提高系统的容错能力, 即在系统出现故障时, 能够自动切换到备用存储介质上, 从而确保系统的正常运行。

性能: 存储层级可以提高系统的性能。通过将数据分散存储在多个存储介质上, 可以提高数据的访问速度, 减少数据的传输时间。此外, 存储层级还可以提高系统的稳定性和可用性, 因为备用存储介质可以在系统出现故障时立即启用, 从而缩短系统停机时间。

成本: 存储层级可以降低系统的成本。通过使用多个存储介质, 可以分散数据存储的风险, 并降低单个存储介质的成本。此外, 存储层级还可以提高系统的可扩展性, 因为可以方便地添加或删除存储介质以适应系统的需求。

多样性: 存储层级可以提供系统的多样性。不同的存储介质具有不同的存储能力和成本, 存储层级可以使系统选择适合自己的存储介质, 从而提高系统的性价比。

综上所述, 存储层级是现代计算机系统的重要组成部分, 它可以提高系统的可靠性、稳定性、性能和成本, 同时提供系统的多样性。

2.在页式虚拟存储中, 过大或过小的页会引起以下问题:

过大的页: 当页的大小过大时, 可能会导致页内存储的数据过多, 从而使页的访问速度变慢。这是因为计算机在访问页时需要读取整个页的内容, 如果页太大, 就需要读取更多的数据, 从而提高了访问时间。此外, 过大的页还可能会导致内存空间的浪费, 因为页的大小通常是固定的, 而应用程序中的数据大小可能是不同的。

过小的页: 当页的大小过小时, 可能会导致页内存储的数据过少, 从而不能满足访问需求。这是因为计算机在访问页时需要读取整个页的内容, 如果页太小, 就可能无法存储需要访问的数据, 从而导致访问失败。此外, 过小的页也可能会导致内存空间的浪费, 因为页的大小通常是固定的, 而应用程序中的数据大小可能是不同的。

综上所述, 过大或过小的页都可能导致访问速度变慢或内存空间的浪费, 因此应该选择合适的页大小来优化虚拟存储的性能。通常, 页的大小应该根据系统的需求和硬件资源的实际情况进行选择, 以确保系统的稳定性和性能。

3. 1) D: Dirty, D 位为 1 时, 表明该页是否被改写, D 位为 0 时, 对此页面进行写操作会触发 Page Fault (Store) 异常, 通过软件在异常服务程序中操控 D 位来维护 D 位满足是否被改写/可写的定义。

A: Accessed, A 位为 1 时, 表明该页可访问。为 0 时不可访问, 否则会触发 Page Fault (对应访问类型) 异常。

G: Global, 全局页面标识, 当前页可供多个进程共享。

U: User, 用户模式可访问。

X: 可执行

W: 可写

R: 可读

V: Valid, 表明物理页在内存中是否分配好, 访问一个 V=0 的页面, 将触发 Page Fault 异常。

2) 如果用户进程可以自由修改页表, 易造成 Page Fault 异常。这种跨权限的行为不利于对系统内核起到隔离作用。

3) 一个 X/W/R 位全部为 0 的有效页表条目表示指针指向下一级页表。

4. 1) PMP 单元负责对物理地址的访问权限进行检查, 判定当前工作模式下 CPU 是否具备对该地址的读/写/执行权限。PMP 从更高层次对读/写/执行权限进行管理, 是对页表条目的读/写/执行权限的更高层次的监视和控制。

2) L: Lock 使能位:

0: 机器模式的访问都将成功, 系统模式/用户模式的访问根据 R/W/X 判定是否成功

1: 表项被锁住, 无法对相关表项进行修改。当配置 TOR 模式, 其前一个表项的地址寄存器也无法被修改。所有模式都需要根据 R/W/X 判定是否访问成功

A: 地址匹配模式:

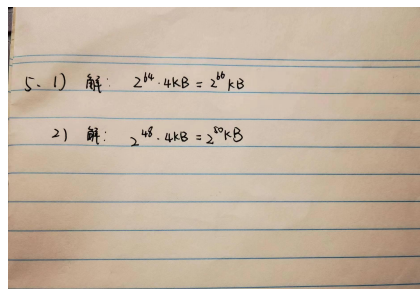
00: OFF, 无效表项

01: TOR (Top of range), 使用相邻表项的地址作为匹配区间的模式

10: NA4 (Naturally aligned four-byte region), 区间大小为 4 字节的匹配模式, 该模式不支持

11: NAPOT (Naturally aligned power-of-2 regions), 区间大小为 2 的幂次方的匹配模式, 至少为 4KB

5.



3) 多级页表和二级页表类似。多级页表和二级页表是为了节省物理内存空间。使得页表可以在内存中离散存储。(单级页表为了随机访问必须连续存储, 如果虚拟内存空间很大, 就需要很多页表项, 就需要很大的连续内存空间, 但是多级页表不需要。)