**Linux系统内存管理总结**

1152703 方志晗 软件二班

（参考书籍《Linux操作系统教程》刘胤杰编著）

为了使存储管理尽量简化，Linux避开了复杂的分段存储管理机制，而采用分页存储管理方式。在虚拟内存管理上采用“按需调页”的存储管理方式。

1. **Linux中的页表**

Linux操作系统可以运行在多种计算机平台上。标准Linux总是假定处理器由三级页表，这三级页表分别成为页目录表（PGD）；中间页目录表（PMD）；根据不同平台，Linux操作系统适当合并页表，比如说在X86平台上，将PGD和PMD合二为一。

值得注意的是，我发现在Linux中每一个用户进程可访问4GB的线性逻辑地址空间。其中0到3GB是用户空间，可以直接访问；另外1GB是用来存放内核代码和数据，用户进程需要通过系统调用来访问。这样实现了让内核态进程共享代码段和数据段。这也和张老师在讲嵌入式Linux时描述一致。

**2．Linux内存分配与释放**

Linux采用分页管理方式，对内存的分配以2的幂次方页为单位。用数据结构mem\_map,free\_area来记录内存块。我觉得这种内存分配的方式比较巧妙，较好的解决了因未使用分段存储的弊端。因为是以2的幂次方页为单位分配，分配时采用最先适配法。首先搜寻大小相同的内存块，若失败则向上一数量级搜寻，及大小为其2倍的内存块。找到后对其进行平分，一份给需要的进程，另一半划给下一级的空闲内存。内存释放则刚好相反，小内存首先融合成其2倍，4倍…的大内存块。排入到free\_area数组的中。

1. **Linux虚拟内存的管理**

在Linux中，每一个进程的虚拟内存空间由数据结构mm\_struct描述，在进程控制块task\_struct中有指针指向它。mm\_struct中包含当前进程已经加载可执行部分的信息和指向进程也目录表的指针，还包括一个指向vm\_area\_struct（虚拟内存）链表的指针。Vm\_area\_struct里面的每一个节点描述了虚拟内存空间的一个区域，包括对该虚拟内存区域的起始和终止地址的描述。当进程的可执行部分连接到进程虚拟地址空间时，vm\_area\_struct数据结构生成新节点。里面包含进程信息。

1. **Linux缺页中断处理**

Linux对缺页中断处理时，首先通过搜索vm\_area\_struct链表确定发生页面失败的虚拟内存区域。Linux的搜索方法值得我们注意，它采用了高效率的AVL树进行搜索，就像数据结构课上的二叉树一样，它有两个指针，左指针vm\_avl\_left, 它指向vm\_area\_struct结构描述的虚拟内存区域的相邻低地址虚拟内存区域的vm\_area\_struct结构，右指针vm\_avl\_right指向与该虚拟内存区域相邻的高地址区域的vm\_area\_struct结构。

**5．Linux的交换空间和页面换出**

Linux采用“按需调页”方法实施虚拟内存管理，内核态内存空间是常驻内存不允许换出，因为这是对系统的管理，被系统调用的频率很高。Linux采用两种方式保存从内存换出的页面。一种是整个块设备，如硬盘的一个分区，称为交换设备；另一种是文件系统中固定长度的文件，称为交换文件。它们统称为交换空间。在Linux中，这一任务由处于内核态的交换进程kswapd来完成，称为内核态线程。Kswapd是用两个变量free\_pages\_high和free\_pages\_low, 当系统空闲内存块数小于free\_pages\_high或free\_pages\_low时，都需要淘汰在内存的页面，只不过如果系统中空闲块数小于free\_pages\_low，核心交换后台进程需要在下次运行之前释放6个页面，否则它只释放3个。Kswapd按照三种方法减少系统使用的物理内存数目：首先，减少缓冲区缓存和页面缓存的数目；然后，将Symtem V共享内存页交换出物理内存；如果经过前面两步，kswapd仍未找到足够数量的内存块，则试图将一般页面交换出物理内存。这里的换出采用“最近最少使用算法”。在描述物理内存块的mem\_map结构的每一个mem\_map\_t元素中有一个数据成员age，它保存着每个页的寿命信息。最初将物理内存开分配给进程的某个页的时候，页的寿命为3，每次该页被访问，其寿命增加3，直到20为止。而每当核心交换进程kswapd运行一次，页寿命减1。而唤出的候选页面选择age较小的页面。当有足够数量的页面换出后，kswapd休眠到再次定时器到时。

1. **Linux存储管理系统的缓存机制**

为了加快对磁盘文件的访问，Linux分别设置了页面缓存和交换缓存。页面缓存用Hash表page\_hash\_table来描述，而page\_hash\_table由指向mem\_map\_t数据结构的指针数组组成。在Linux中，每个文件通过一个VFS inode数据结构来描述，且是唯一的，它仅且可以描述一个文件。因此，可以惟一地描述页面信息所在文件极其所在文件的位置。Linux可以根据物理内存的大小来调整页面缓存的尺寸。换出时也是先换出页面缓存的页面。Linux采用物理内存的swap\_cache表来描述交换缓存。交换缓存描述的是换出物理内存的页面的情况，swap\_cache表目的总数就是物理内存块的总数。当Linux就爱那个一个在内存的页调出内存时，先查询交换缓存，如果其中有与该页对应的有效页表表目，则不需要将该页重新写到交换空间，因为原来处在交换空间的内容与待换出的内容是一致的。