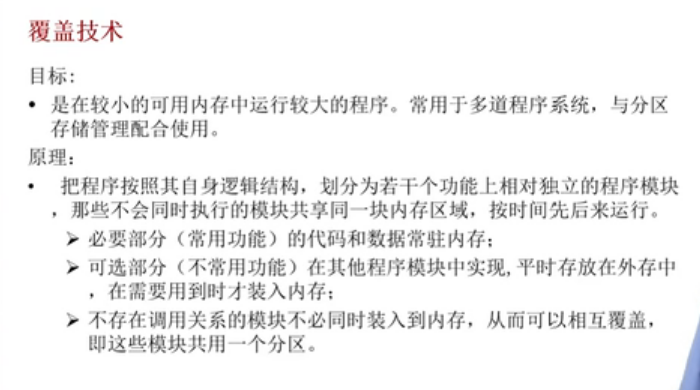
虚拟内存：

**内存不够用**

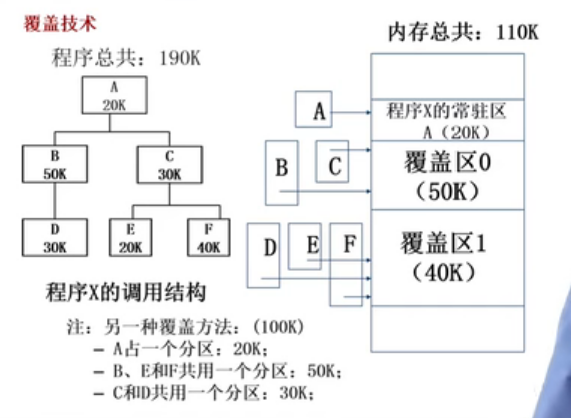
**不常用的数据放到硬盘上去**，**常用的**数据和代码**放到内存中**

**覆盖技术overlay：**



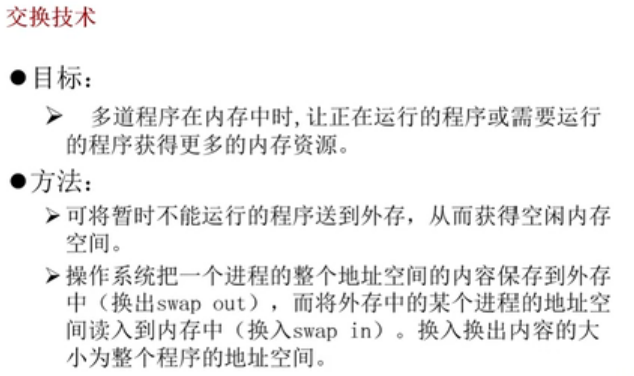
**分时共享同一块空间**

例子：



**开销是编程计算怎么划分模块**，**以及频繁的换入换出时间开销**

交换技术swapping：

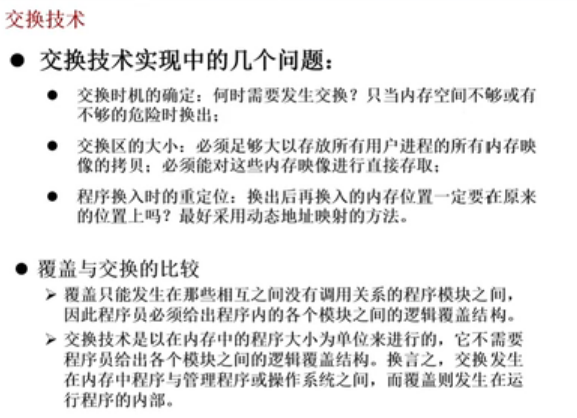


操作系统来做，**将某个时候将一个进程、程序整个地址空间的内容导入到硬盘。**

导入一般几百个页，开销很大

整个程序写到硬盘进去。

**硬盘操作和系统错很多数量级，硬盘操作是系统一直在等。**

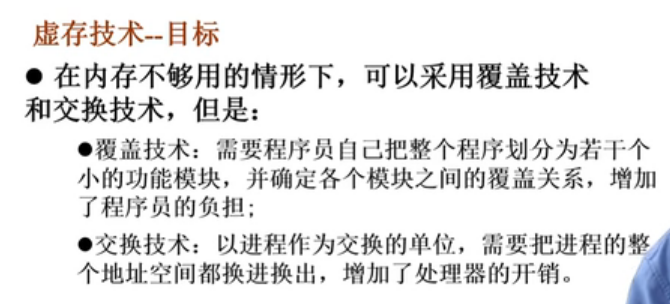


页表机制中虚拟地址和物理**没有准确的一一对应关系**。

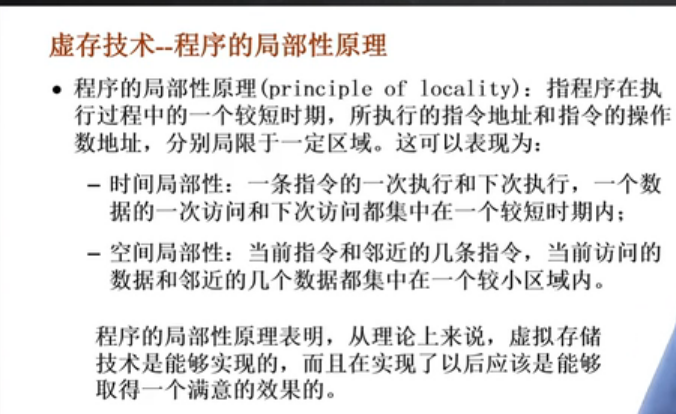
**覆盖发生在一个程序里面，是处理一些函数、模块，程序内**

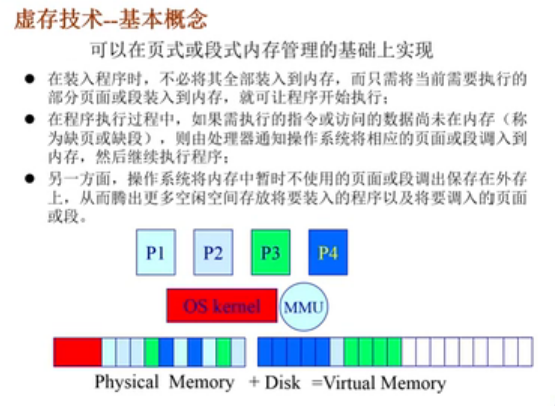
**而交换式发生在程序之间的**

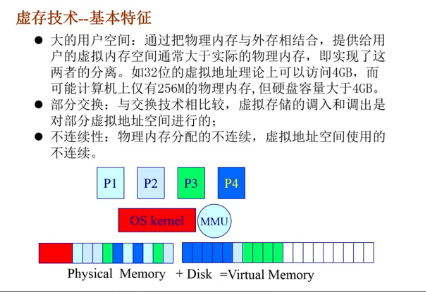
虚拟存储技术：



对于**虚存对进程的部分内容**进行**内存和硬盘之间的导出导入**。（可能是以页来操作）







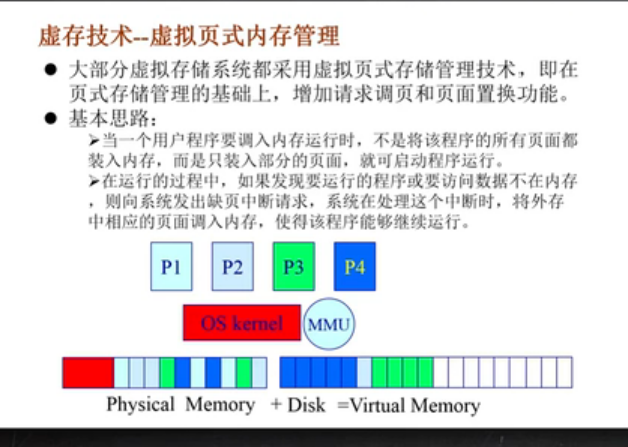
缺页异常、换入页面，重新执行指令，如果**空间不够则将内存中一段时间内不会被访问到的页进行换出**。页面置换选择算法。

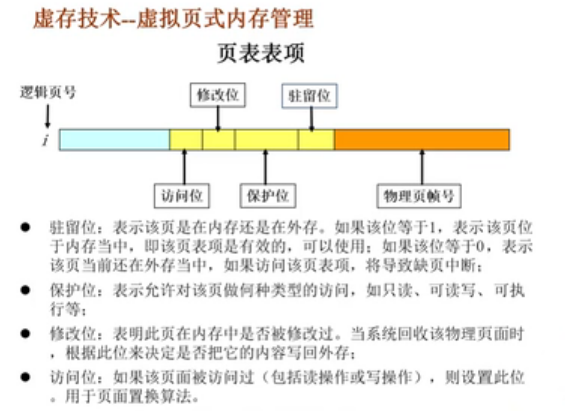
**把硬盘加到物理内存空间，形成一个虚拟空间**。（32位的虚拟地址理论上有4G）

**物理内存分配是不连续的，虚拟地址空间的使用也是不连续的。**

**虚存中以页来管理。**

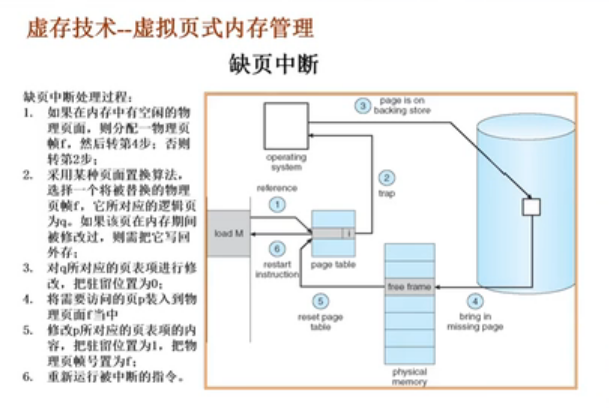
**请求调页，页面置换：**





**修改位表示最近有修改，内存中的数据和硬盘不一样了**，如果要换出这个页需要将该页**写回到硬盘中**，而不是直接丢弃。（没有修改直接释放就行）

缺页中断的流程：



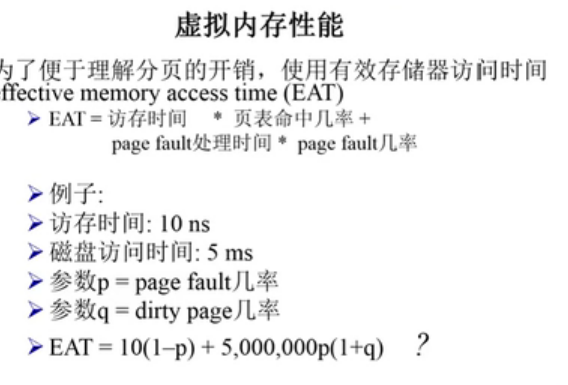
Backing store后备存储



数据、代码/执行（执行程序里面）、动态链接库 这些都在硬盘中 都是以文件的形式存在

其他数据，硬盘上划分一个swap分区（换入换出分区）->不是以文件的形式存在，而是以分区形式存在。

缺页异常处理，硬盘的读写



p是缺页的概率

q是触发写操作的概率

写操作时还要向硬盘写操作

性能取决于p，程序具有局部性特点，p会很小，缺页很少