两级储存器的性能特征

CPU高速缓存作为内存和处理器间的缓冲器，为一个两级内部储存器。与一级存储器相比，两级结构通过开发局部性提供了更高的性能。虚拟存储器和磁盘高速缓存也利用了局部性原理，并且有一部分是有操作系统实现的。

|  |  |  |  |
| --- | --- | --- | --- |
|  | 内存高速缓存 | 虚拟存储器(分页) | 磁盘高速缓存 |
| 典型的访问时间比 | 5:1 |  |  |
| 内存管理系统 | 由特殊硬件实现 | 硬件和系统软件结合 | 系统软件 |
| 典型的块大小 | 4~128字节 | 64~4096字节 | 64~4096字节 |
| 处理器访问的第二级 | 直接访问 | 间接访问 | 间接访问 |

局部性原理

在执行程序期间，处理器的指令访存和数据访存呈“簇”(一组数据集合)状。典型的程序包含许多迭代循环和子程序，一旦程序进入一个循环或子程序执行，就会重复访问一个小范围的指令集合。同理，对表和数组的操作涉及存取“一簇”数据。经过很长的一段时间，程序访问的“簇”会改变，但在较短的时间内，处理器主要访问存储器中的固定“簇”。即无论是存取指令还是存取数据，所访问的存储单元都趋于聚集在一个较小的连续区域中。

局部性原理有效的原因如下：

1. 除了分支和调用指令，程序执行都是顺序的，而这两类指令在所有程序指令中只占一小部分。因此，大多数情况下，要取的下一条指令都是紧跟在取到的上一条指令之后的。
2. 很少出现很长且连续的过程调用序列及相应的返回序列。相反，程序中过程调用的深度窗口限制在一个很小的范围内，因此在较短的时间内，指令的引用局限在很少的几个过程中。
3. 大多数循环结构都由相对较少的几个指令重复若干次组成。在循环过程中，计算被限制在程序内一个很小的相邻部分中。
4. 在许多程序中，很多计算都涉及处理诸如数组、记录序列之类的数据结构。在大多数情况下，对这类数据结构的连续引用都是对位置相邻的数据项进行操作

时间局部性和空间局部性

空间局部性(spatial locality)指涉及多簇存储器单元的执行趋势，这反映了处理器顺序访问指令的倾向，同时也反映了程序顺序访问数据单元的倾向。

时间局部性(temporal locality)指处理器访问最近使用过的存储器单元的趋势

两级存储器的操作

两级存储器结构也采用了局部性原理。上层存储器()比下层存储器()更小、更快、成本更高(每bit),用于临时存储空间中的上下文。访问存储器时，首先试图访问中对应的内容，若成功，就可进行快速访问；若不成功，则把一块储存器单元从复制到中，再通过进行访问。由于局部性，当一个块被取到中时，将会有很多对块中单元的访问，从而加快整个服务。

要说明访问一项的平均时间，不仅要考虑两级存储器的速度，而且要考虑能在中找到给定引用的概率。为此有

表示(系统)平均访问时间，表示(如高速缓存、磁盘高速缓存)的访问时间，表示(如内存、磁盘)的访问时间，表示命中率(访问可在中找到的次数比)。