



虚拟Cache没有流行起来的一个原因是操作系统和用户程序对于同一个物理地址可能采用两种以上不同形式的虚拟地址来访问，这些地址称为**同义**或**别名**。它们可能会导致同一个数据在虚拟Cache中存在两个副本。如果其中一个被修改，那么再使用另一个的数据就是错误的。这种情况在物理Cache中是不会发生的，因为这些访问首先会把虚拟地址转换到同一物理地址，从而找到同一个物理Cache块。有一种用硬件解决这个问题方法，叫做**反别名法**，它保证每一个Cache块对应于唯一的一个物理地址。

如果强行要求别名的某些地址位相同，就可以用软件很容易地解决这一问题。例如，SUN公司的UNIX要求所有使用别名的地址最后18位都相同。这种限制被称为**页着色**。这一限制使得容量不超过 2^{18} 字节（256KB）的直接映象Cache中不可能出现Cache块有重复物理地址的情况。所有别名将被映象到同一Cache块位置。

另一种实现快速命中的技术是把地址转换和访问Cache这两个过程分别安排到流水的不同级中，从而使时钟周期加快，但这同时也增加了命中所需的时间。这种方法增加了访存的流水线级数，增加了分支预测错误时的开销，而且使得从Load指令流出到数据可用之间所需的时钟周期数增加。

还有一种方法，既能得到虚拟Cache的好处，又能得到物理Cache的优点。它直接用虚地址中的页内位移（页内位移在虚→实地址的变换中保持不变）作为访问Cache的索引，但标识却是物理地址。CPU发出访存请求后，在进行虚→实地址变换的同时，可并行进行标识的读取。在完成地址变换之后，再把得到的物理地址与标识进行比较。

这种虚拟索引、物理标识方法的局限性，在于直接映象Cache的容量不能超过页的大小。Alpha AXP 21064采用了这种方法，其Cache容量为8KB，最小页大小为8KB，所以可以直接从虚地址的页内位移部分中得到8位的索引（块大小为32字节）。

为了既能实现大容量的Cache，又能使索引位数比较少，以便能直接从虚拟地址的页内位移部分得到，我们可以采用提高相联度的办法。这一点可以从下面的公式中看出：

$$2^{\text{index}} = \frac{\text{Cache容量}}{\text{块大小} \times \text{相联度}}$$



Copyright 2005-2007 晨曦工作室