**Notes on Cache Memory**

Basic Ideas

The cache is a small mirror-image of a portion (several "lines") of main memory.

* + cache is faster than main memory ==> so we must maximize its utilization
  + cache is more expensive than main memory ==> so it is much smaller

How do we keep that portion of the current program in cache which maximizes cache utilization?

1. Direct Mapping

Each block is supposed to fit-in in the specific line of a cache.

Rigid Mapping

Inefficient utilization of cache memory.

1. Associative Mapping
2. Set Associative Mapping

====================================================================

**Calculating Internal Fragmentation**

Page Size = 2048 Bytes

Process Size = 72766 Bytes

35 Pages + 1086 Bytes

Internal Fragmentation = 2048 – 1086 = 962 Bytes

1. Worst Case Internal Fragmentation = 1 frame – 1 Byte = 2048 -1 = 2047 Bytes
2. So small frame size desirable?
   1. But each page table entry takes memory to track?
   2. Page sizes growing over time.
   3. Solaris support two values for page sizes – 8 KB and 4 MB