Cache & Virtual Memory

2110352 Comp. Sys. Arch.

Krerk Piromsopa, Ph.D.
Department of Computer Engineering
Chulalongkorn University

Cache

- Reduce time to access memory
- Principle of Locality
  - Temporal Locality
  - Spatial Locality

Performance

- CPU Time=IC x CPI x Cycle time
- Access Time=
  Hit Time + Miss Rate * Miss Penalty
- or (In your slide)
  Hit Time * Hit Rate + Miss Rate * Miss Penalty

Cache Parameters

- Size
  - Enough to fit working set (temporal)
  - Big = slow ?
- Associativity
  - Large to avoid conflicts
  - Big = slow?
- Block
  - Large to exploit spatial
  - Too Large = Higher Misses & Penalty
- Replacement Algorithm

Hit Time

- How to reduce cache hit time?
  - smaller cache?
  - lower associativity
  - wide interfaces

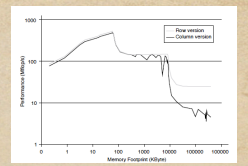
Miss Rate

- How to reduce miss rate?
  - Large cache
  - higher associativity
  - replacement algorithm

Miss Penalty

- Multi-level caches
- sub-blocks
- Write buffers
- etc.

What else?



- Better programs / compilers?

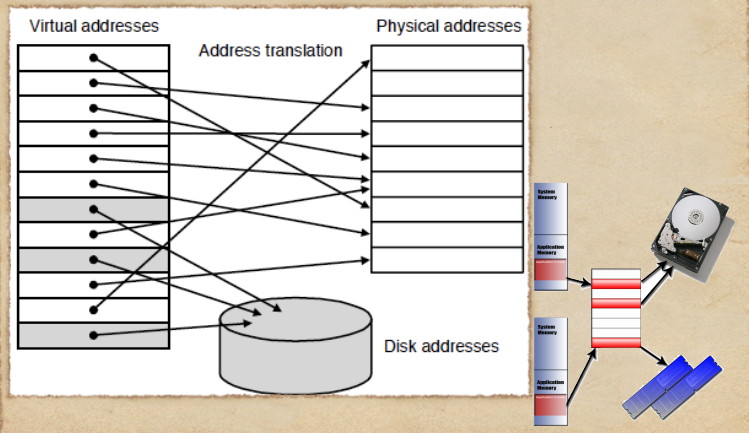
```
for (i=0;i<1000;i++)
  for (j=0;j<1000;j++)
    sum=b[i][j]+c[i][j];
```

```
for (j=0;j<1000;j++)
  for (i=0;i<1000;i++)
    sum=b[i][j]+c[i][j];
```

What else?

- Prefetch?
- More bit-level parallelism?

Virtual Memory



Parameters

- Page size
  - TLB
  - Fragmentation
  - Disk Access
- Replacement algorithm

Put it all together

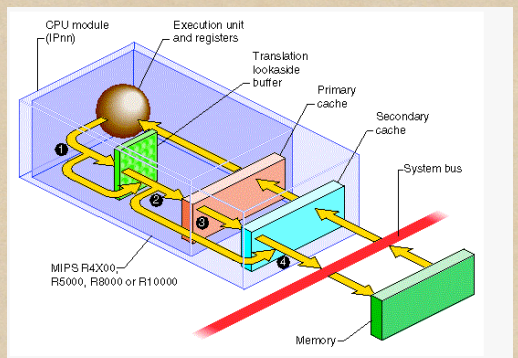


Figure 1-1 CPU Access to Memory