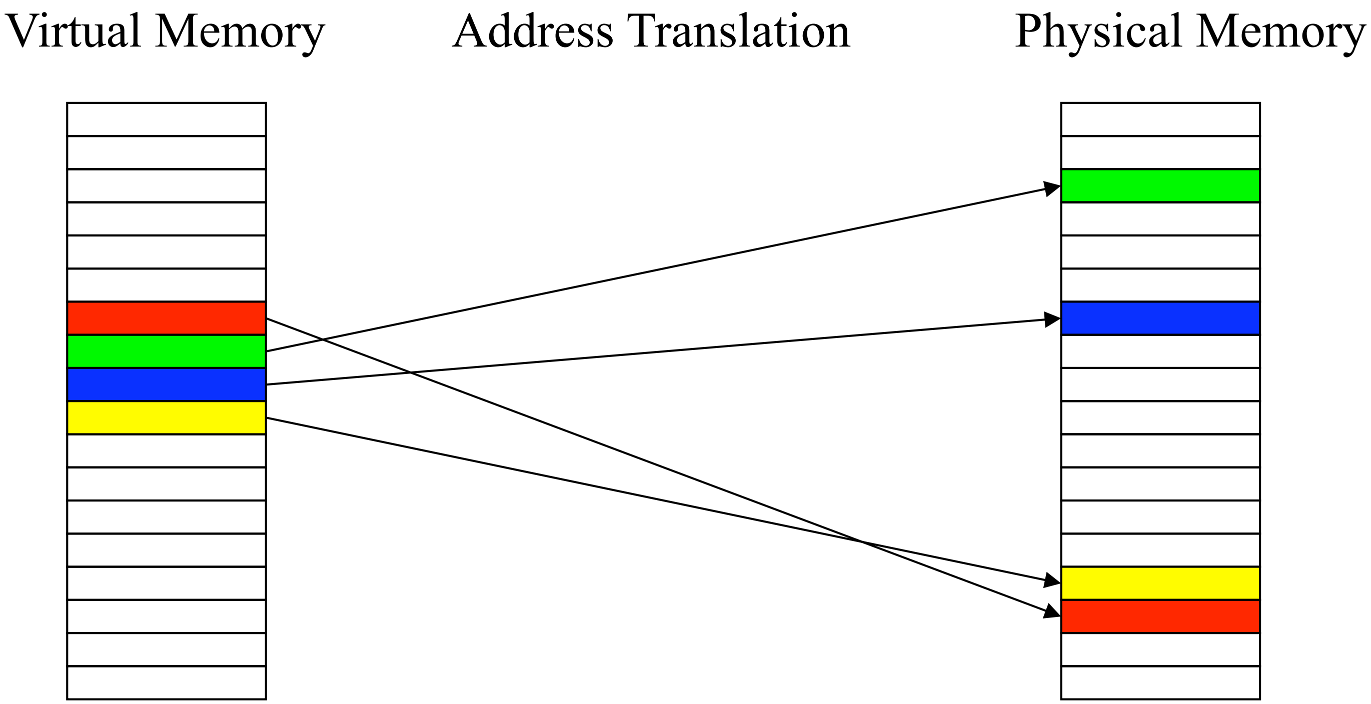
TLB Utilization

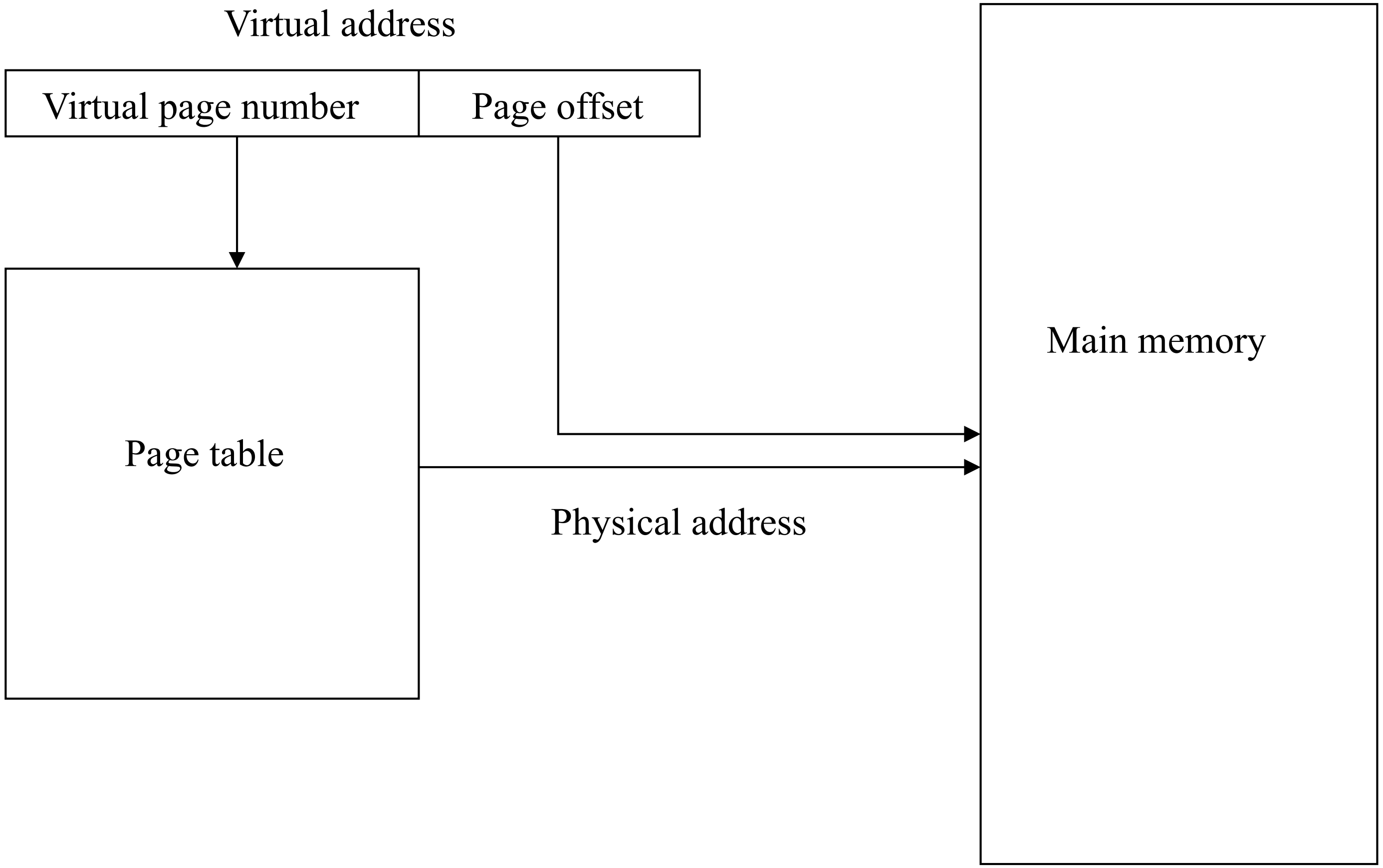
Background: Virtual Memory

- Modern programs operate in “virtual memory”
  - Each program thinks it has all of memory to itself
  - Fixed sized blocks (“pages”) vs variable sized blocks (“segments”)
- Virtual Memory benefits
  - Allow a program that is larger than physical memory to run
    - Programmer does not have to manually create overlays
  - Allow many programs to share limited physical memory
- Virtual Memory problems
  - Each virtual memory reference must be translated into a physical memory reference

Virtual Memory vs Physical Memory



Address Translation



Source: Computer Architecture A Quantitative Approach, by John L. Hennessy and David A. Patterson

Translation Speed

- Translation page table is stored in main memory
  - Each memory access logically takes twice as long – once to find the physical address, once to get the actual data
- Use a hardware cache of least recently used addresses
  - Called a Translation Lookaside Buffer or TLB

Quad-core improvements

- 2MB memory pages no longer limited to an 8 entry TLB
  - 2MB pages are not currently supported on XT, but support being tested and will be released soon
- 1GB memory pages
  - Cray currently has no plans to support these
- Improved Out-of-Order loads/stores
- Improved branch prediction