Memory Management: Address Translation

Daniel R. Cender

Grand Canyon University: CST-221-O500

January 31, 2020

**Translate a Virtual Address to a Physical Address**

**Explanation**

The MMU (Memory Management Unit) is a piece of hardware that sits between the CPU and the memory buses in a computer system. When an operating system allows a process to access a memory address, the MMU takes the translates the virtual address to a real slot in the physical memory store. When the MMU receives an address, it scans in parallel for it in the TLB (Translation Lookaside Buffer). If it finds the desired page frame and the request doesn’t violate protection bits, it accesses the physical address in the page table noted by the TLB entry (Tanenbaum and Bos, 2015). If not, the MMU goes around the TLB (a “miss”) and searches in the page table for the page frame containing the physical memory address. This “miss” might be *hard* or *soft*, depending on whether the frame is already in the page table or not (requiring physical memory swapping in a *page fault*), a speed difference of many clock cycles. If the TLB is full, then the MMU must manage a TLB entry swap, otherwise it just adds the page frame to the TLB.

When a TLB miss occurs, there is oftentimes a trap to the operating system to handle the page frame lookup is performed, so more space and power can be reserved in the CPU package for caches and processing speed improvements (Tanenbaum and Bos, 2015). The basic process described above is shown via a simplified pseudocode implementation below and shown more properly formatted in Figure 1.

**Pseudocode Process**

function translateVirtualToPhysicalAddress(addr) {

var physicalAddr

var pageOffset = parseOffset(addr)

var tlbRecord = findMatchInTLB(addr)

if(tlbRecord is valid/present) {

physicalAddr = getPhysicalAddrFromFrame(tlbRecord, pageOffset)

} else {

var pageTableRecord = getFrameFromPageTable(addr)

if (pageTableRecord not valid/present) {

pageTableRecord = swapPageTableRecords(addr)

}

var newTLBEntry = swapOrAddTLBEntry(addr, pageTableRecord)

physicalAddr = getPhysicalAddrFromFrame(newTLBEntry, pageOffset)

}

return physicalAddr

}

**Formatted Screenshot**

*Figure 1.* Formatted pseudocode for basic algorithm of translating a virtual memory address into a physical memory address.

**A black and silver text

Description automatically generated**

References

Tanenbaum, A.S. & Bos, H. (2015).*Modern Operating Systems.*Chapter 3.