Putting It All Together

We've covered a lot of pretty dense concepts in this chapter, so let's just recap on what happens when a memory address is accessed by a process.

Figure 7.4 illustrates what happens when a process requests a virtual memory address.

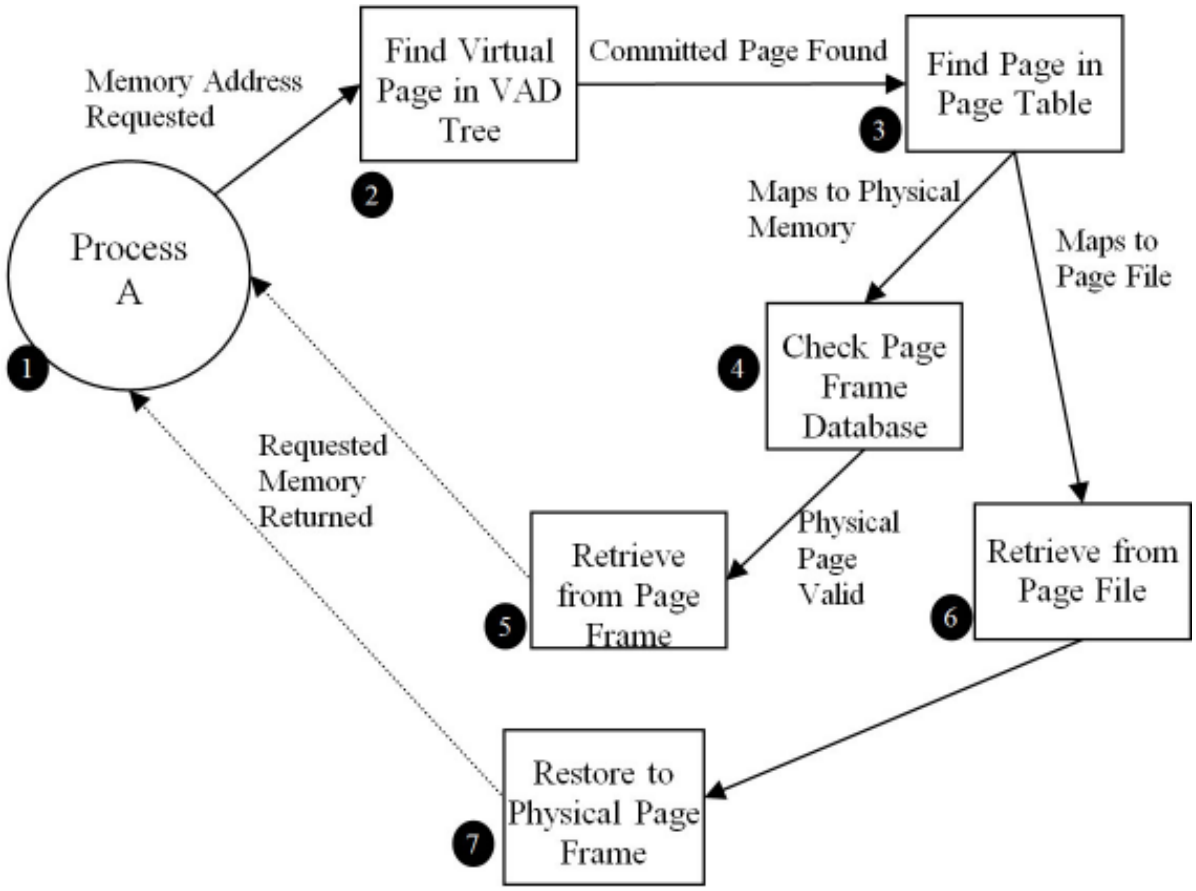

Figure 7.4: Mapping a virtual address.

When a process requests to read a virtual memory address the steps below occur.

1. Process A requests a virtual memory address.
2. The VAD tree is checked to ensure the address falls within a valid committed virtual page. If not, an exception is thrown.
3. The virtual address is broken down into its constituent parts and used to find the PTE in the page table.
4. If the PTE thinks the page is in physical memory, then the page frame entry is retrieved from the PFD.
5. It is checked for validity, and used to retrieve the actual data from the direct location in physical memory, which is then returned to process A.
6. If the page isn't in physical memory, then a page fault occurs and it has to be loaded in from the page file. An existing physical page may have to be replaced to achieve this.
7. Once restored, the requested memory can be returned from physical memory.

It really is that simple!

OK, that was a joke. The truth is that it *is* a bit convoluted and complicated, but if you had designed a system to achieve something similar, you would probably have used many of the same structures and processes.