



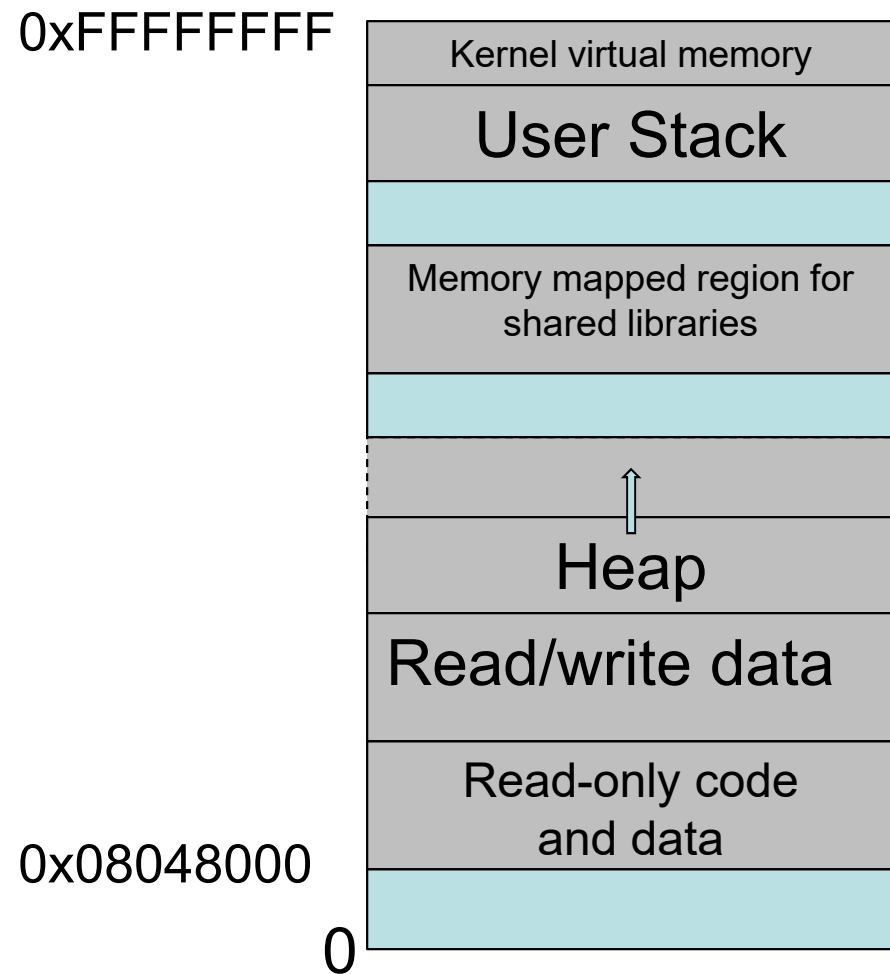
Introduction to Virtual Memory

- Learning Objectives
 - Explain how virtual memory forms the foundation of process isolation.
 - Explain what support hardware must provide to enable virtual memory.



What is Virtual Memory?

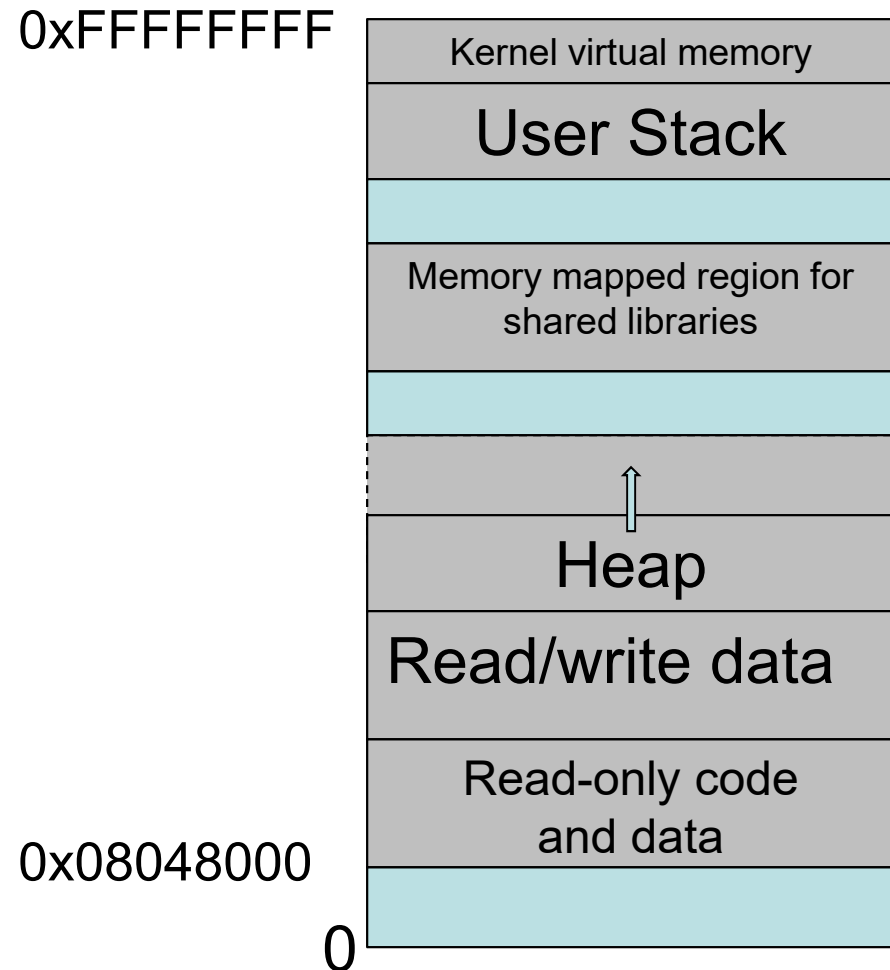
- Recall





What is Virtual Memory?

- Recall
- But: fork creates a second address space.
- How can two 32-bit address spaces both reside on a single 32-bit machine?





It's all a lie!

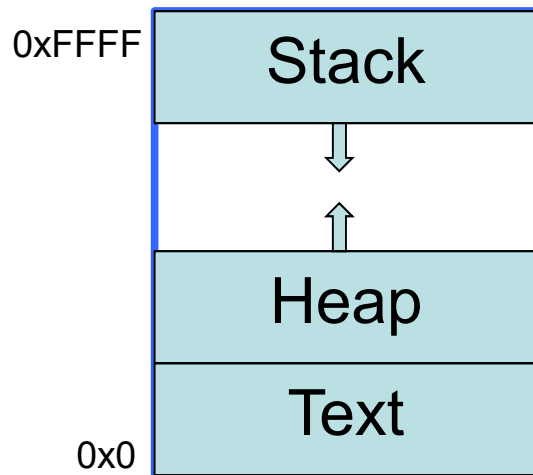
- There are two different kinds of addresses:
 - **Physical addresses:**
 - Refer to specific locations in the actual memory (DRAM).
 - You may very well have less physical memory than your virtual address space is capable of addressing.
 - E.g., you can have a 32-bit machine with less than 4 GB of memory.
 - E.g., you can certainly have a 64-bit machine (with a 64-bit address space) that has less than 64 EB (Exabytes).
 - **Virtual addresses:**
 - These are the addresses that programs use.
 - Require that there is **a mapping from virtual addresses to physical addresses**.



Why VM?

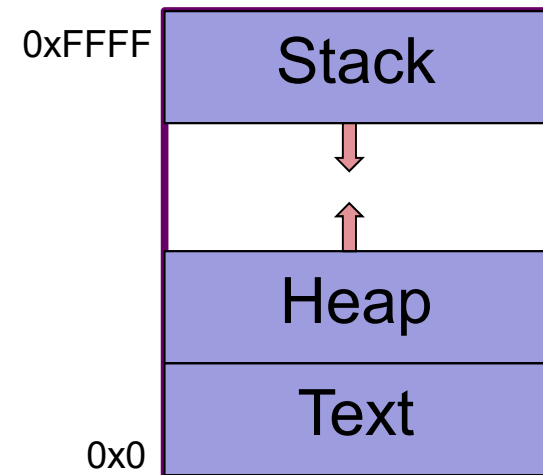
I have the entire address space to myself!

Program 1

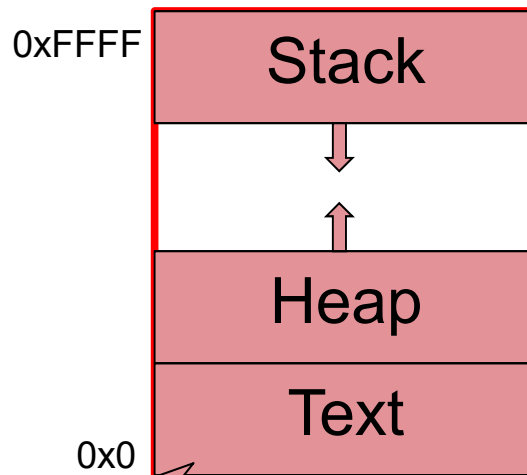


You're both wrong – I have the whole address space

Program 3



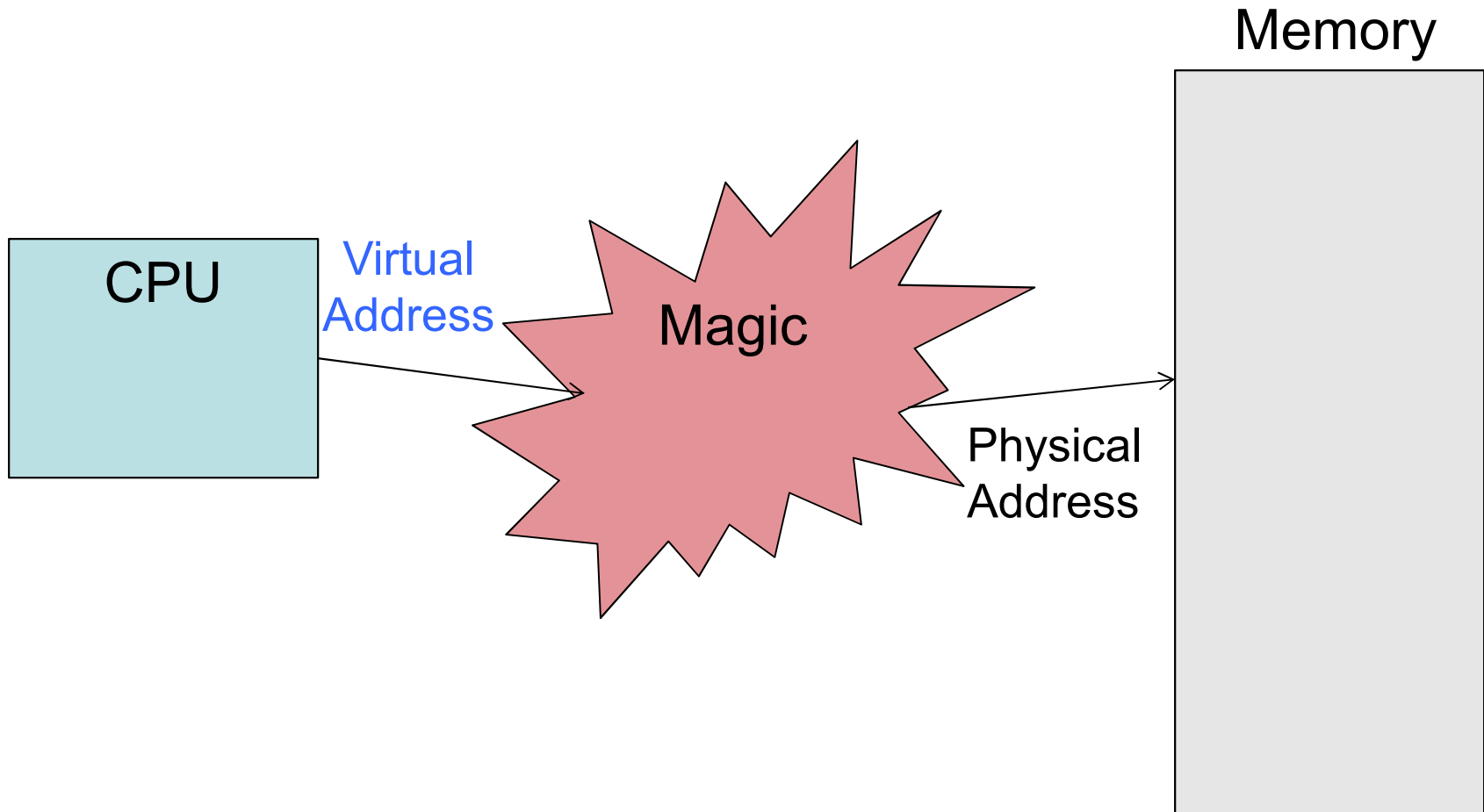
Program 2



Hey – I have the entire address space!

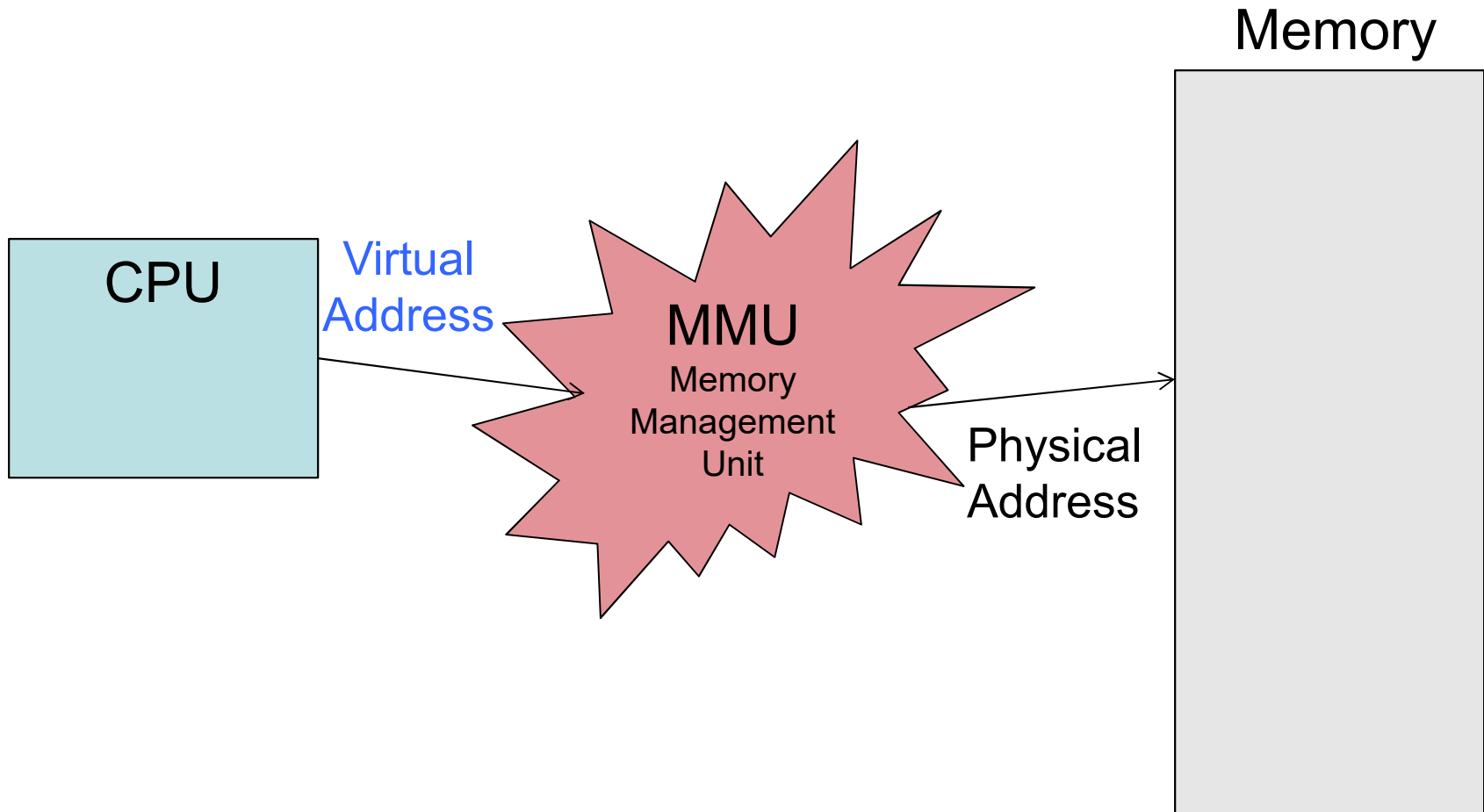


How?



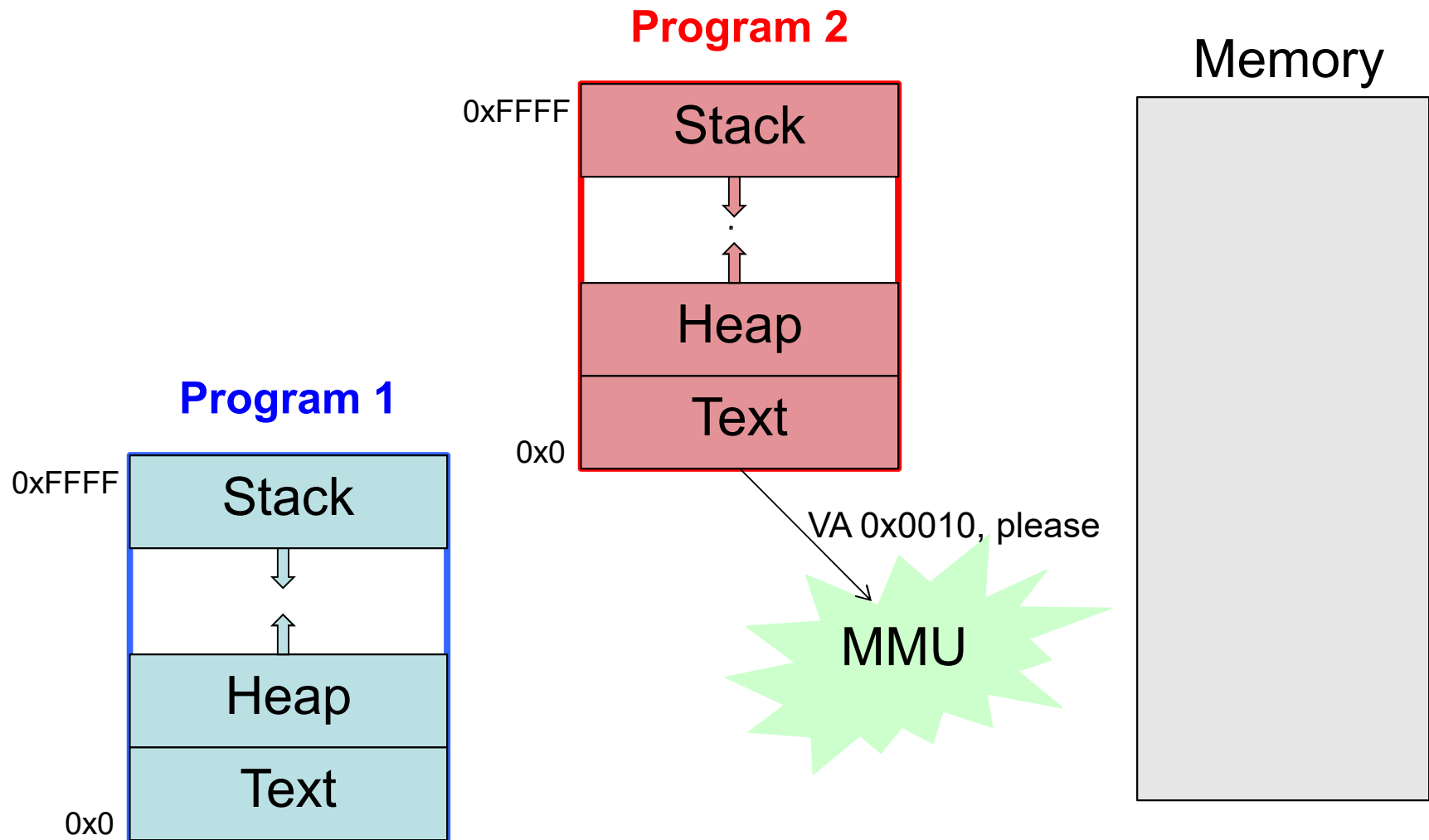


How?



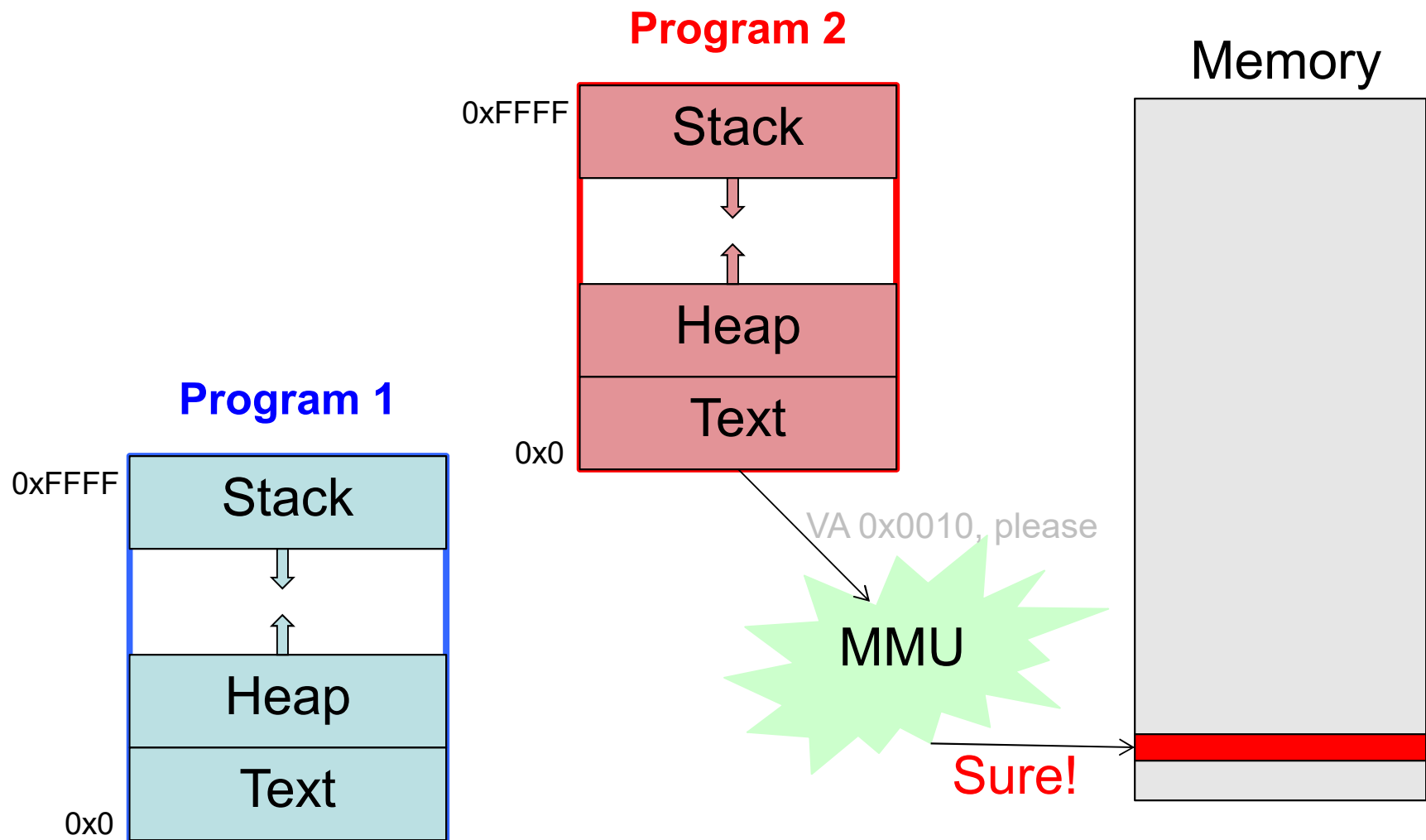


How VM Provides Process Isolation





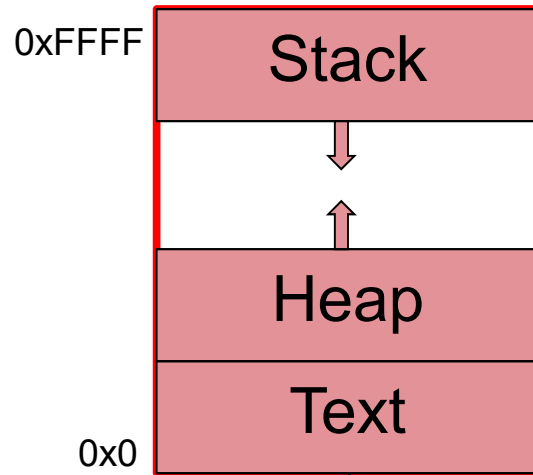
Protection



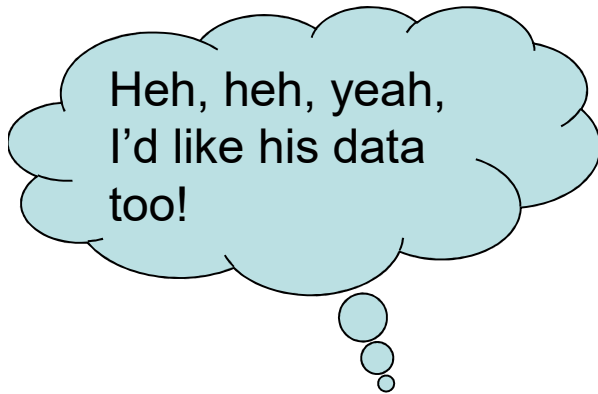
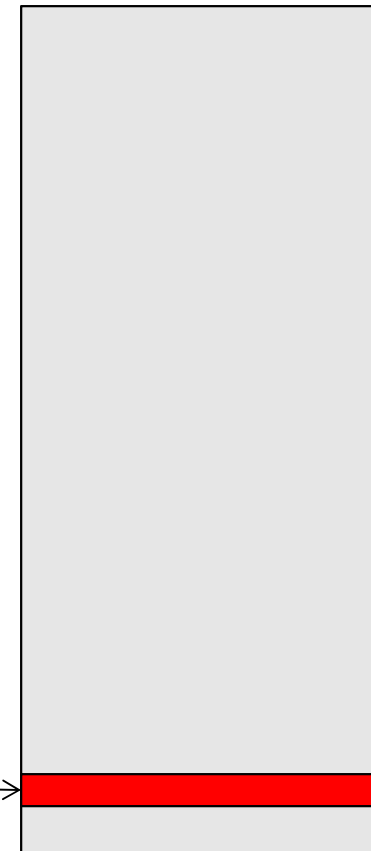


Protection

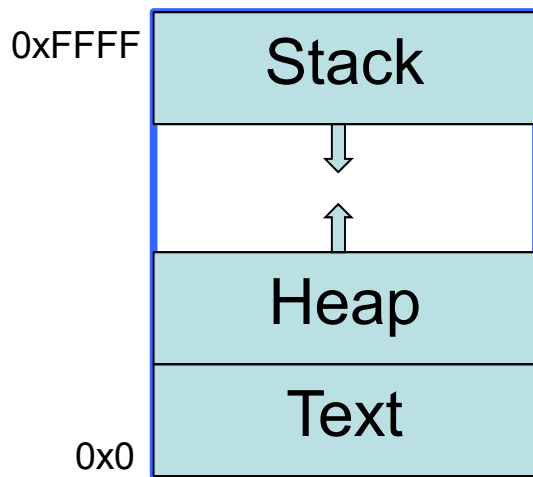
Program 2



Memory



Program 1

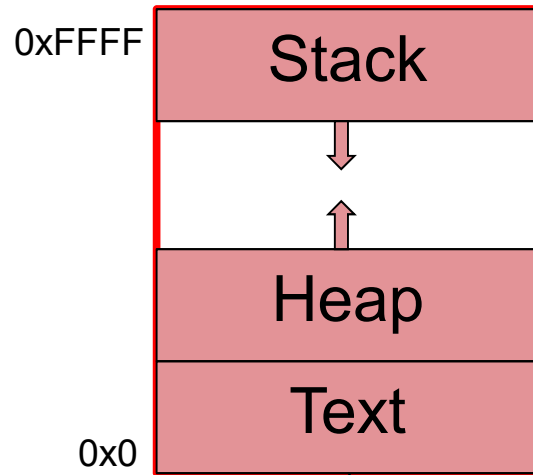


Sure!

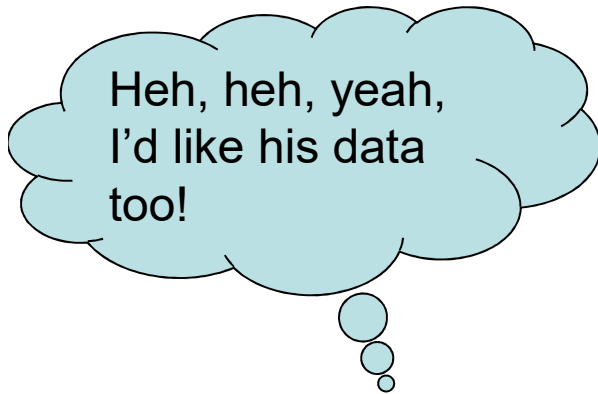
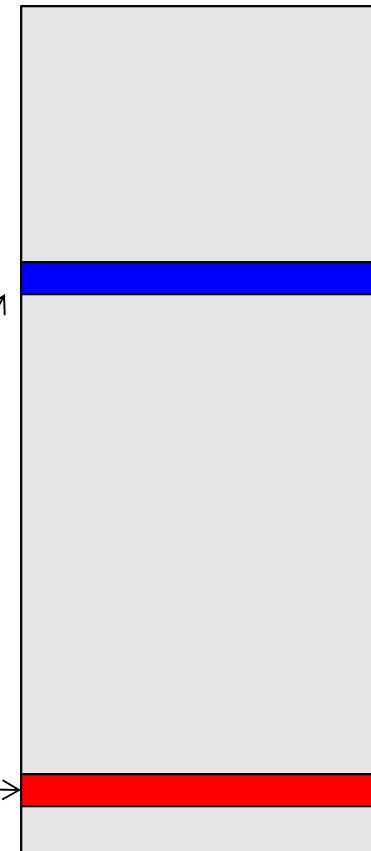


Protection

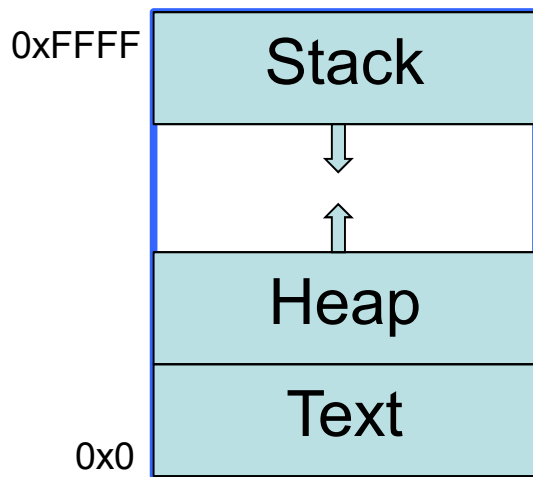
Program 2



Memory



Program 1



VA 0x0010, please

VA 0x0010, please

MMU

Sure!

Sure!



VM: A Hardware/Software Partnership

- We need hardware support to provide virtual memory.
 - Why?
 - Invoking the operating system on every address access would be too slow.
- The Hardware
 - Provides a (fast) mechanism to map from a virtual address to a physical address
- The Software
 - Sets up the mappings that the hardware will execute
 - Manages the allocation of physical memory
 - Implements policies about how memory is shared.



Constraining the Mapping

- Recall that different parts of an address space support different operations:
 - Read-only text/data: cannot be modified
 - Implication: We must not be able to write certain parts of memory
 - Data should not be executed:
 - Implication: Not all parts of memory should be enabled for execution.
- The operating system is **special** software
 - We want the operating system to do many things that we don't want normal processes to do: interact with devices, touch any process's memory, etc.
 - Implication: The OS should have access to some memory locations that are inaccessible to regular processes.



Putting it all Together

- We ask the hardware to map a triple of:
 - Virtual address
 - Type of access (read, write, execute)
 - **Privilege** level
- The hardware will either:
 - Produce a physical address
 - Fault, due to:
 - The type of access is not allowed to the memory requested
 - The process requesting access does not have the appropriate privilege level to access the memory.