

CSCE 410

MP4

Kranthi Kumar Katam
225009204

Page Table:

As we take our Page table from the previous MP. But, at start CPU will issue only logical addresses. Get the page directory and page table entry from process memory pool. And make the last entry of the page directory point to itself by making the 1023 address to valid address. The first 10 bits point to page directory, next 10 bits point to page table entry, and next 12 bits are offset. The MMU thinks this as a reference to page table and it traverses through the same table again. Then it goes to the last 1023 frame and that will again reference to the page directory. Then the actual offset value is available which points to the memory address.

Create an object in VMpool i.e., `no_regions` which are number of regions in the memory and number of pools in that region is given by `no_pools`. In page fault handler check whether the fault address is legitimate or not. If legitimate then proceed to page fault handling.

Register pool maintains a register of all the used pools in the memory. These pools are stored in `no_regions` and `no_pools`.

Free page will call the release frame function in `contframepool` and release the frame with the page number i.e., the offset. And write the page directory address to CR3 register to flush the TLB.

Virtual Memory Pool:

In VMpool class base address is the starting address of the memory. Size defines the total size of the memory. First register current pool to register pool which will store it as the first region.

Allocator is store the start address and size of a region in virtual memory table. Restricted the number of regions to 512. The first page of table will an array of region descriptors. Initialized to -1 and 1 for start address and size of the region. Allocator requests for the size of the address space, then the allocator function will iterate through 512 memory regions and if the size of region > size then it will return the start address the region which is free.

Release function releases the previously allocated memory. Initialize page address with start address from release function. Then iterate through virtual memory

tables and check for this page address, if present break and again iterate through the region with the start address and release the frames in the region.

Is_legitimate function checks whether address is part of the allocated memory or not. It takes the address and appends that to page address. Then iterate through 512 regions if it is present in the any of the region if present then return true else return false.