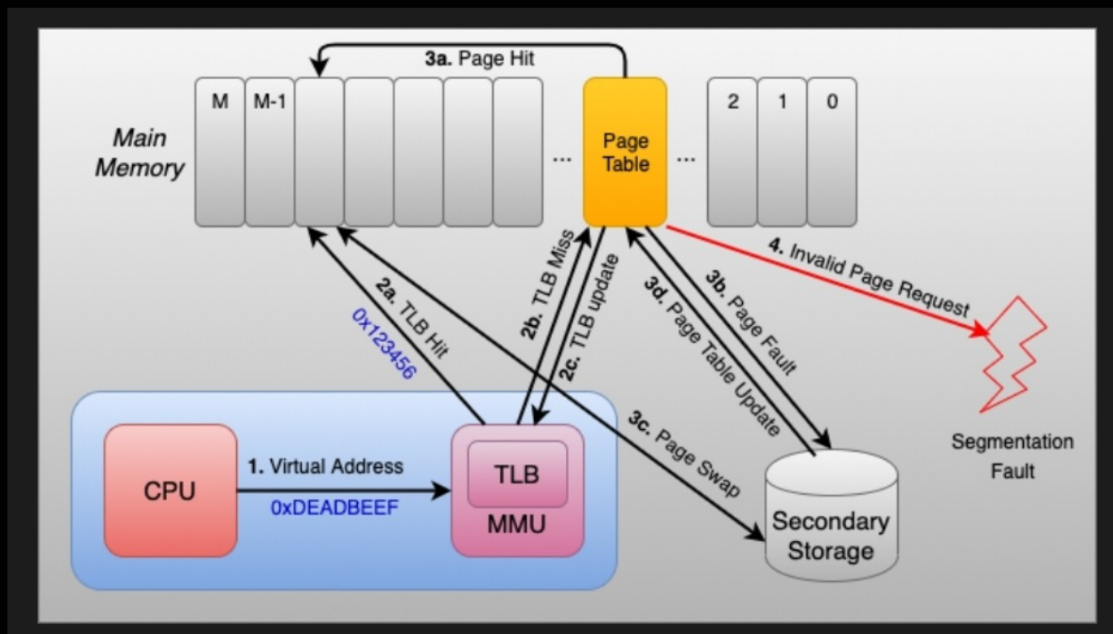


Deep Dive: MMU Virtualization with Xen on ARM — Star Lab Sof...

This article will explore how the Xen hypervisor uses the Memory Management Unit (...)

www.starlab.io

- When one does not have enough memory they (used) use "overlays". It's basically swapping.
 - Multiple Async rotating drums & auto high speed mem ops (pages) was **JTE paper** for page concepts → Must read (if you can find)
- x — x — x —
- MMU are one that help convert b/w virt & physical. MMU does this by splitting virt addr space to pages. Any page table in memory has mapping from virt to physical. TLB is another structure that caches frequently used.
 - XEN uses a similar concept for MMU. a workflow as such:



- every block has 1 entry in the table. implies that the address of virt are split into 2 parts:

Upper bits | low bits
(block addr) (offset in block)

often OS create a tree like page table entry in each level point to the entry in next level.

— x — x — x —

Up finally, let's look at how Xen does MMU.

- every memory call by guest OS undergo 2 layers of translations.
- Guest OS itself does the first, Vmm does the second.
- The guest OS MMU are now converting "virt" address to "intermediate" address.
- when the OS (conceive that it's not physical) send the request to Vmm.
- here the Vmm does its own translation when also checking policies.

- As for TLB lookups, it is also done in 2 stages where each VM has a VMID that acts as a address for the VM. finally looks some thing like this:

