Course: Virtualization

Unit -3 Memory Virtualization

Prof. Rahul Shrimali

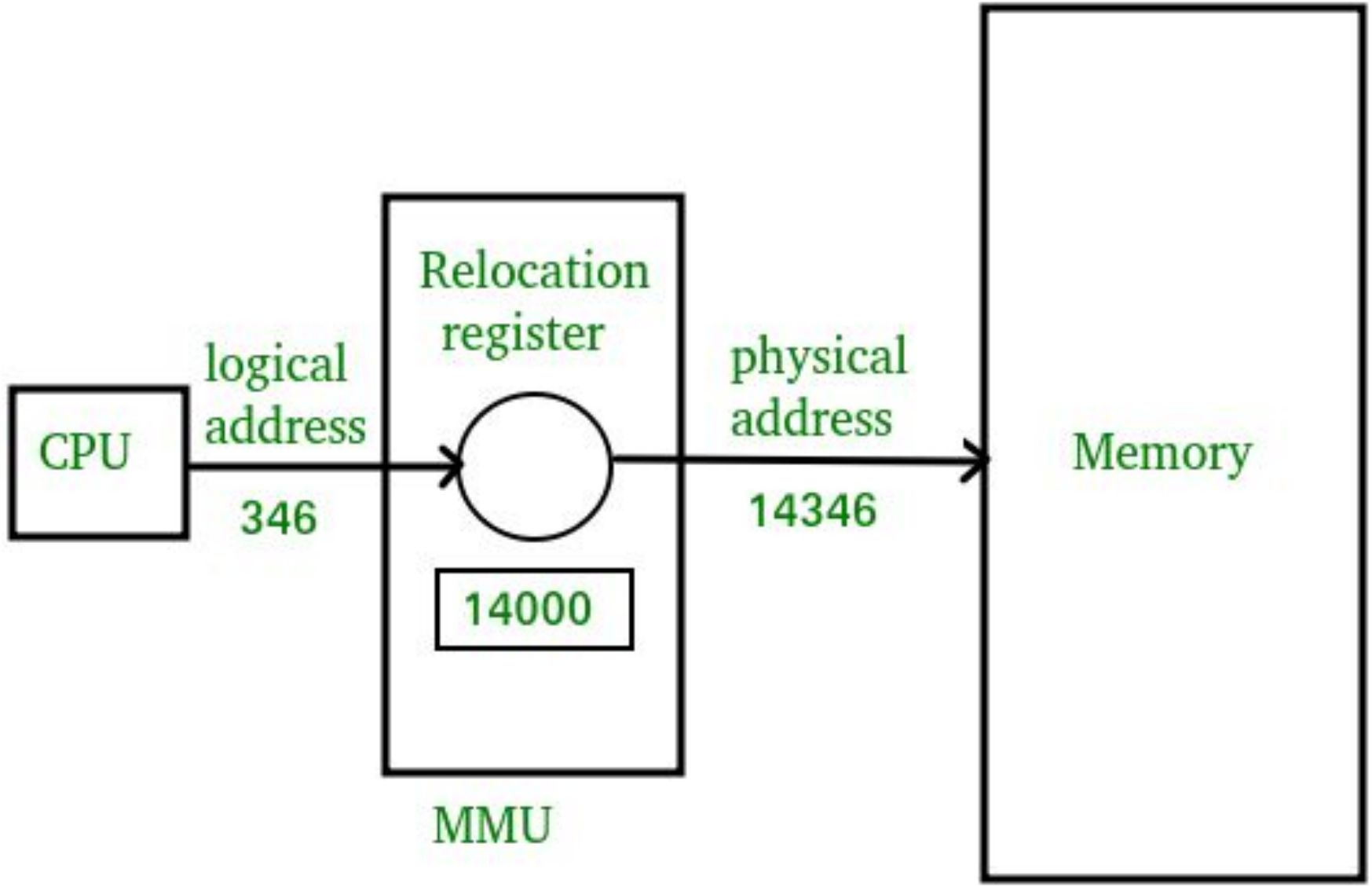
Institute Of Computer Technology

Ganpat University
॥ विद्यया समाजोत्कर्षः ॥

Outline

- Memory Management - Basics
- Role of MMU
- Page Tables & TLBs
- Memory Virtualization
  - What is to be Virtualized?
  - Why to Virtualize Main Memory?
  - How to Virtualize Main Memory?
- Comparisons

BASIC CONCEPT OF MEMORY MANAGEMENT

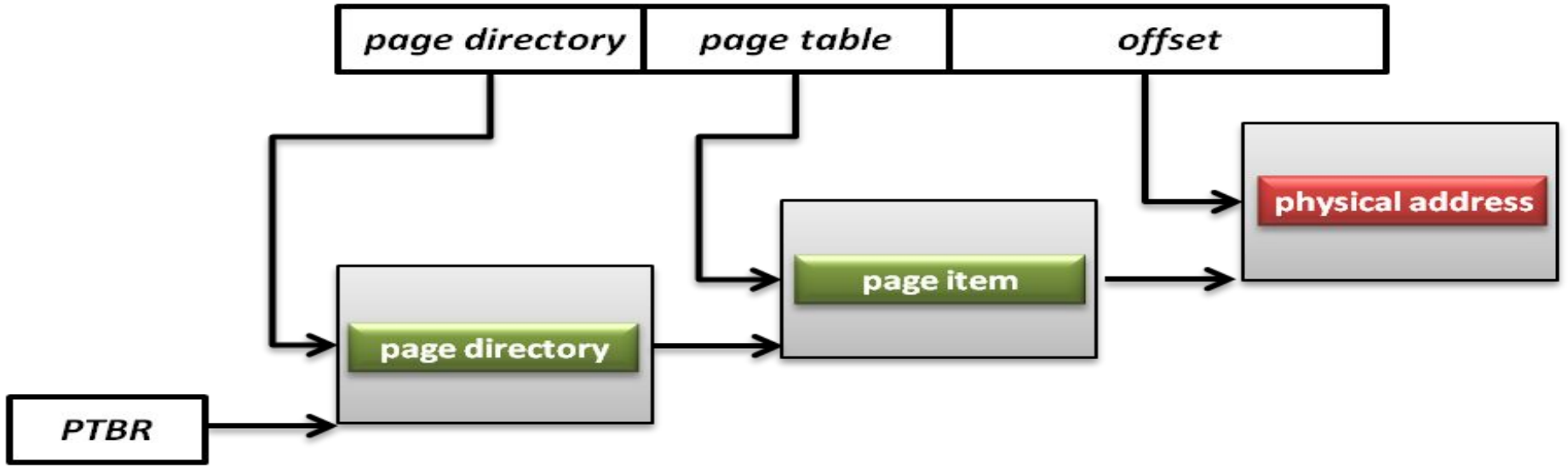

Relocation
register
logical
address
CPU
346
14000
physical
address
14346
Memory
MMU

Memory Management Unit

- A hardware component responsible for handling accesses to memory requested by the CPU
  - Address translation: virtual address to physical address (VA to PA)
  - Memory protection
  - Cache control
  - Bus arbitration (The **Bus Arbiter** decides who would become the current bus master.)
- Page tables are maintained by operating system, and MMU only references them.
- TLB updates are performed automatically by page-table walking hardware

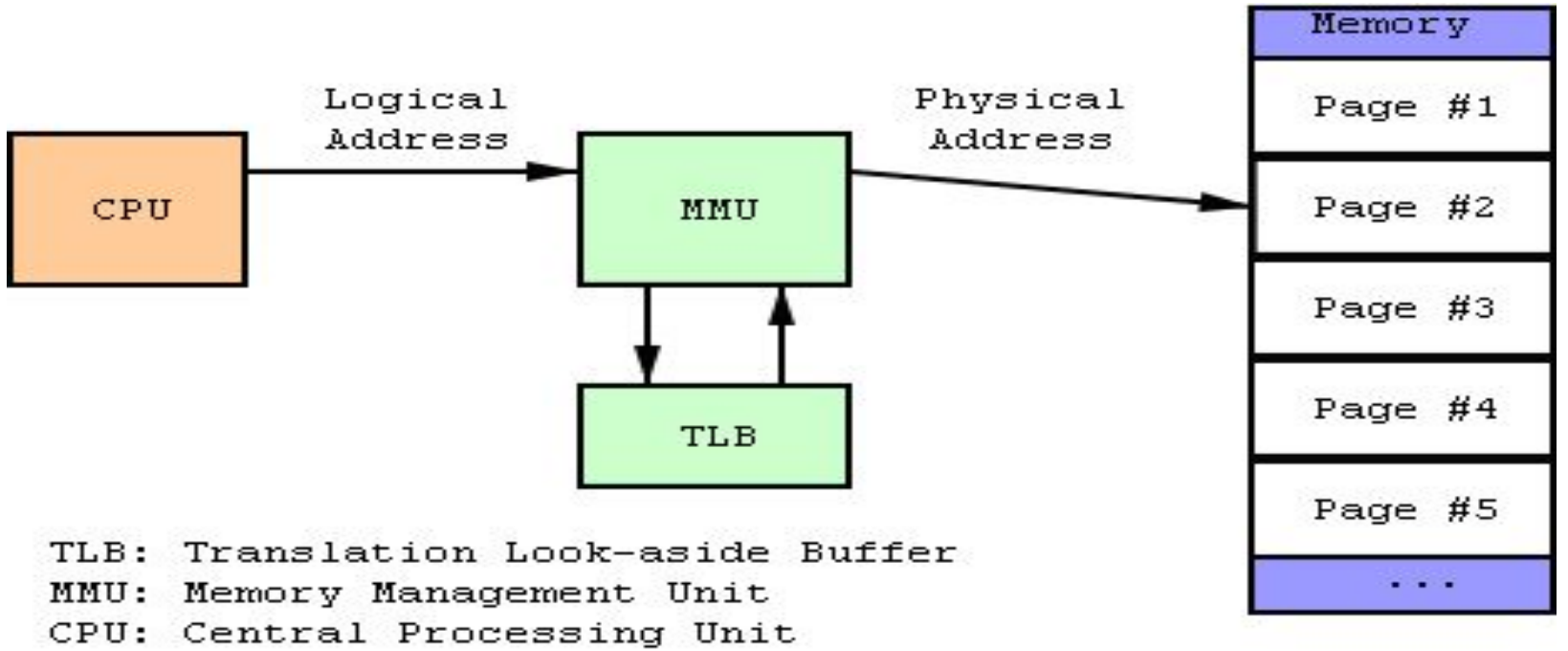
Page Tables

- A page table is the data structure used by a virtual memory system to store the mapping between virtual addresses and physical addresses
- Translation table base register(TTBR)
  - Also called page table base register
  - A register that stores the address of the base page table for MMU



TLB

- Translation look-aside buffer
  - A CPU cache that memory management hardware uses to improve virtual address translation speed
  - The TLB is typically implemented as content-addressable memory (CAM)
  - The CAM search key is the virtual address and the search result is a physical address



Concepts

Shadow page table

Hardware assistance

Comparison

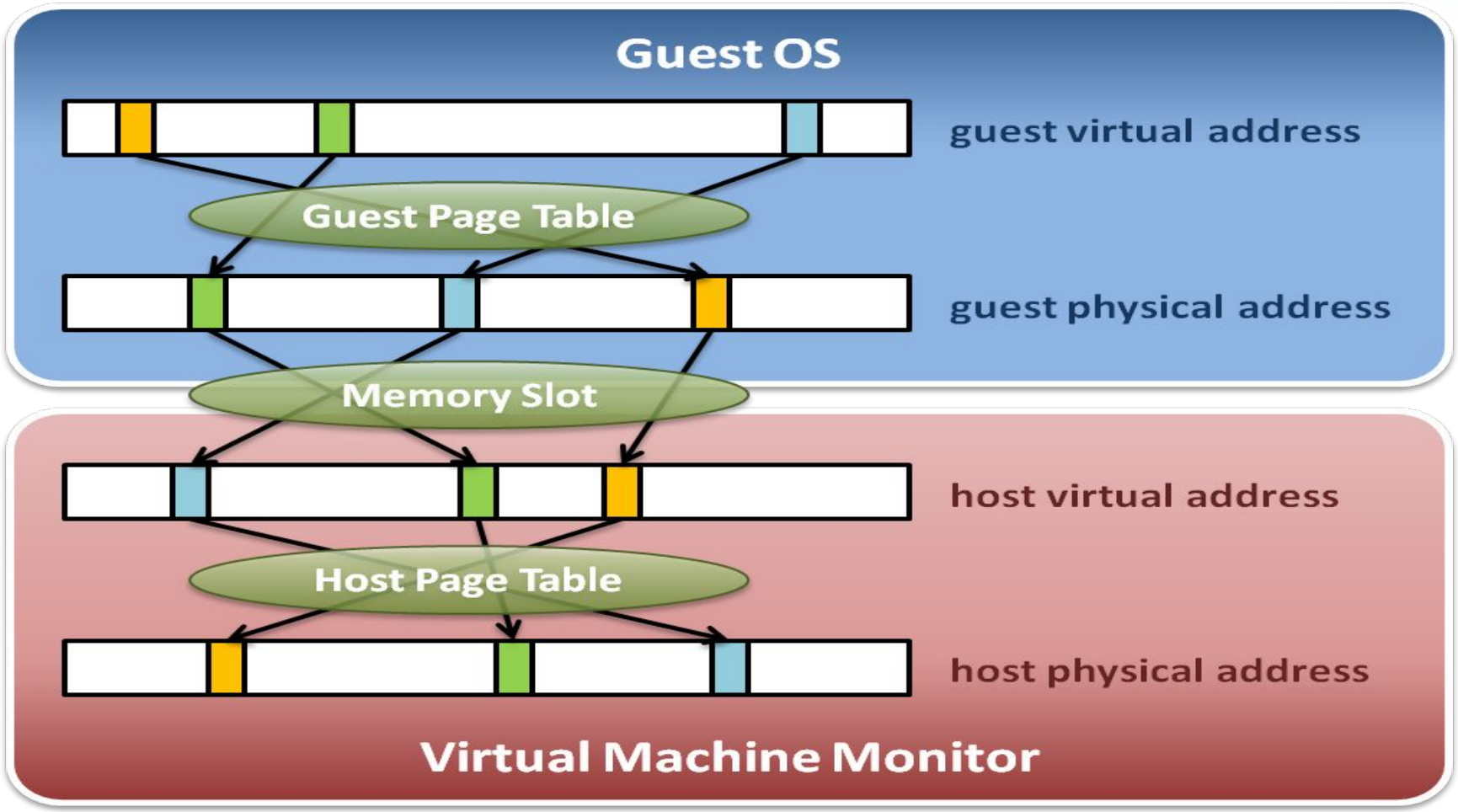
MEMORY VIRTUALIZATION

Memory Management on a VM

- Traditionally, OS fully controls all physical memory space and provides a continuous addressing space to each process
- Guest OS is just one of user space processes of host OS
- If guest OS is allowed to access the physical memory arbitrarily, then what happens?
- In system virtualization, VMM should make all virtual machines share the physical memory space

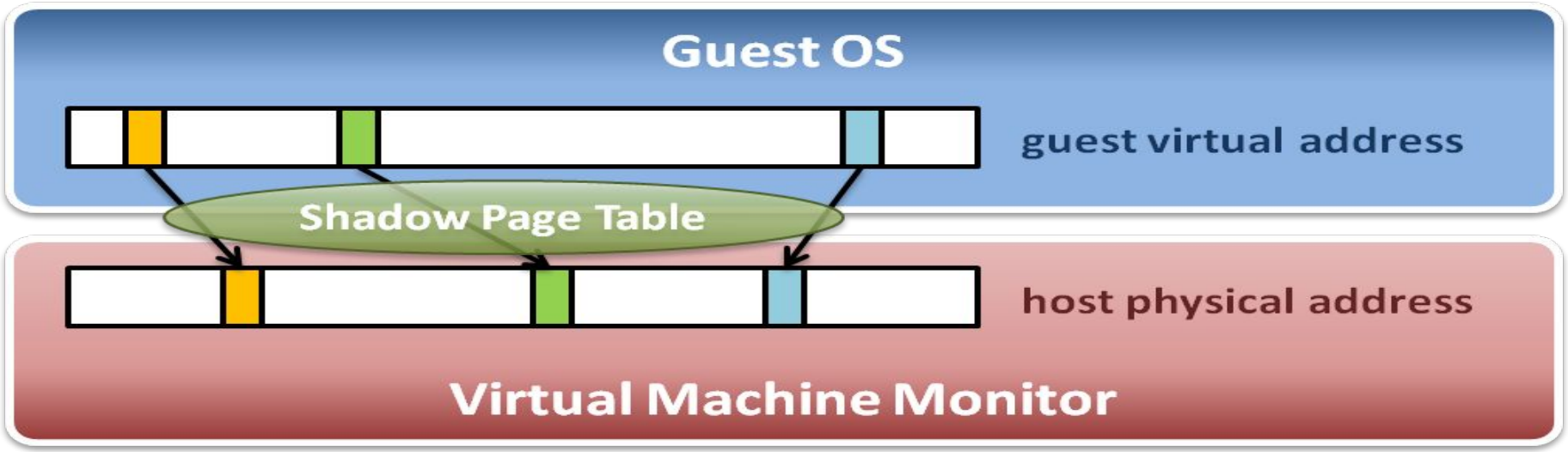
Memory Virtualization

- Memory virtualization architecture – Basic Design



Memory Virtualization

- The performance drop of memory access is usually unbearable. VMM needs further optimization.
- VMM maintains shadow page tables :
  - Direct virtual-to-physical address mapping
  - Use hardware TLB for address translation



Goals of Memory Virtualization

- Address Translation
  - Control table-walking hardware that accesses translation tables in main memory.
- Memory Protection
  - Define access permission which uses the Access Control Hardware.
- Access Attribute
  - Define attribute and type of memory region to direct how memory operation to be handled.

- How to implement?
  - Software solution: shadow page table
  - Hardware solution
    - NPT on SVM from AMD
    - EPT on VMX from Intel
    - ARM v7 VMSA (Virtual Memory System Architecture) with virtualization extension

Concepts

Shadow page table

Hardware assistance

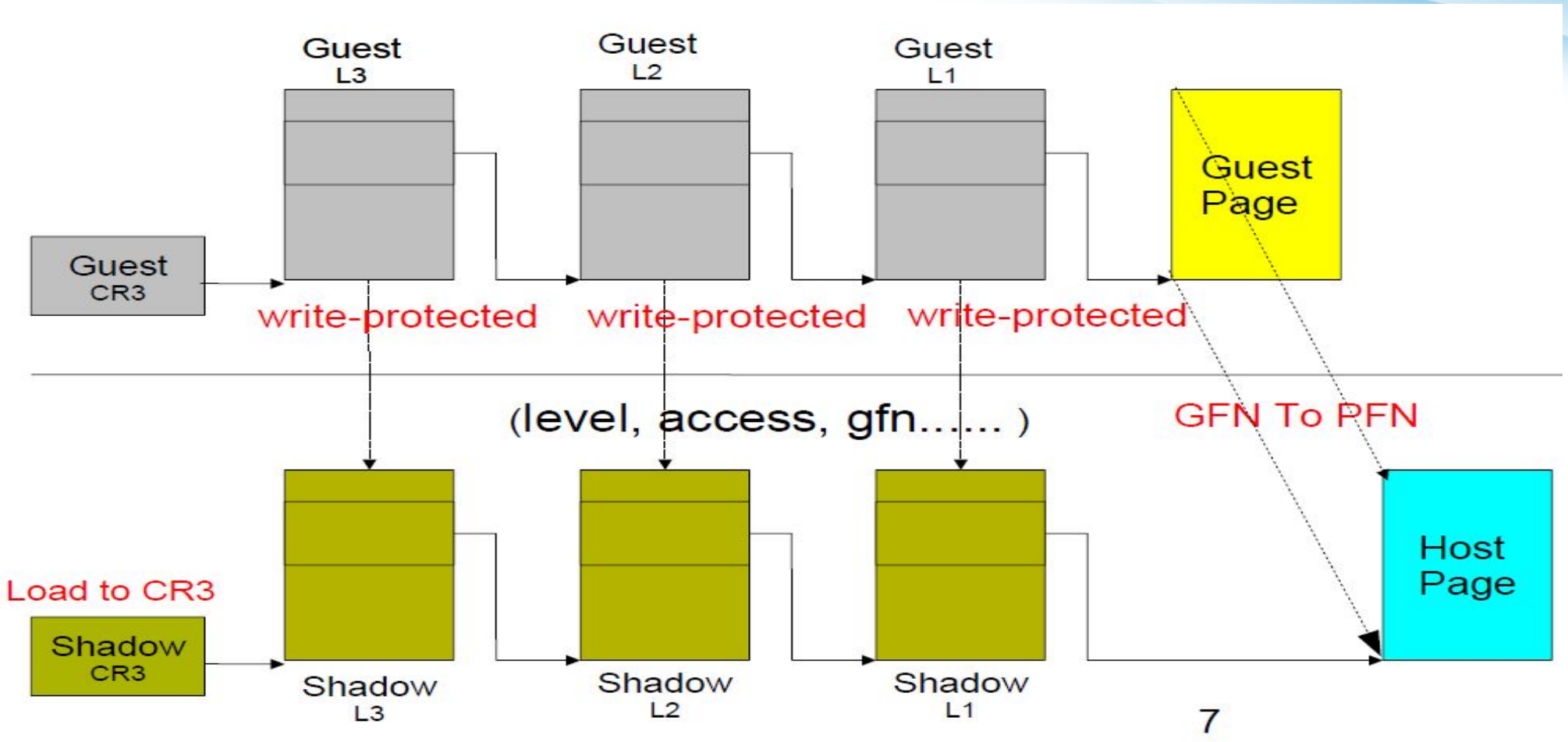
Comparison

MEMORY VIRTUALIZATION

Shadow Page Table

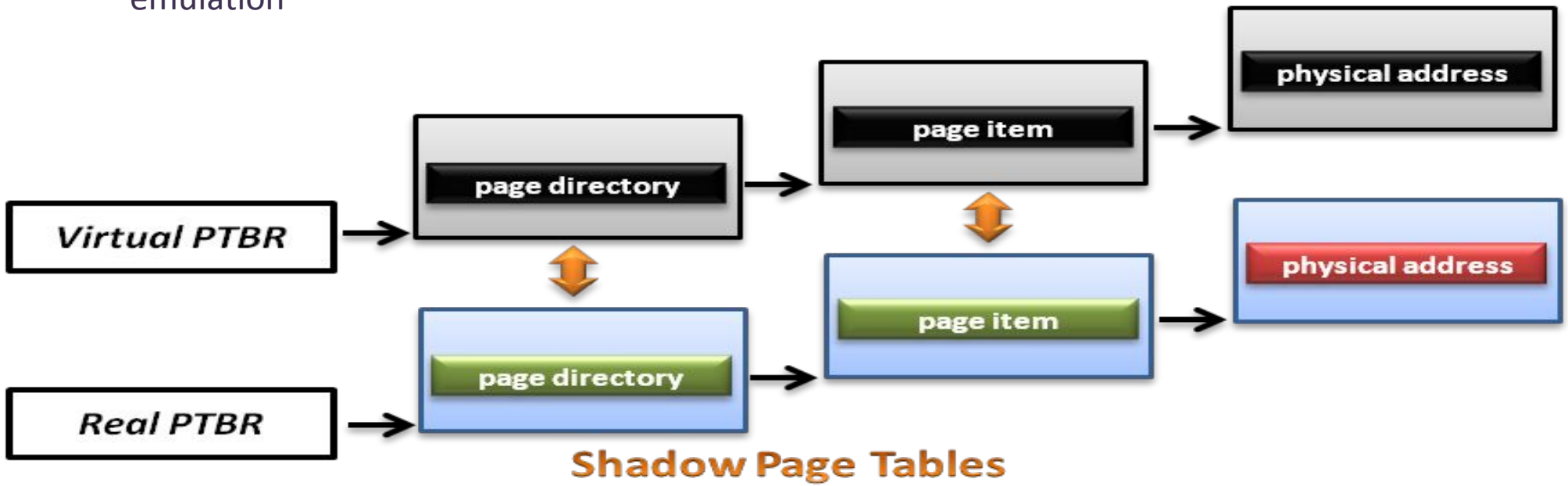
- Map guest virtual address to host physical address
  - Shadow page table
    - Guest OS will maintain its own virtual memory page table in the guest physical memory frames.
    - For each guest physical memory frame, VMM should map it to host physical memory frame.
    - Shadow page table maintains the mapping from guest virtual address to host physical address.
  - Page table protection
    - VMM will apply write protection to all the physical frames of guest page tables, which lead the guest page table write exception and trap to VMM.

Shadow Page Table: Overview



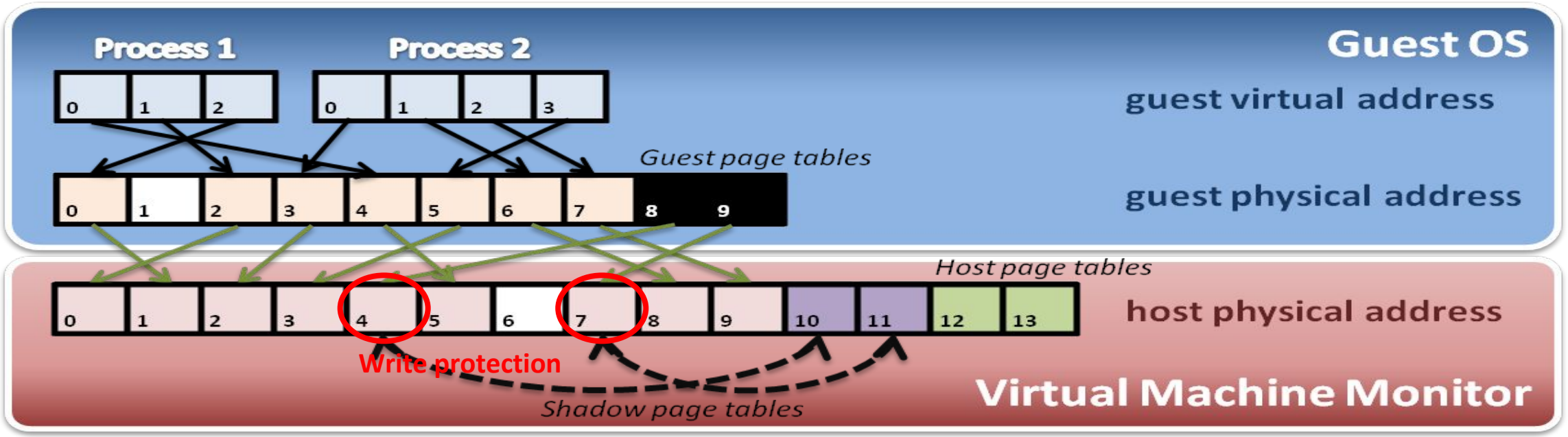
Shadow Page Table

- How does this technique work ?
  - VMM should make MMU virtualized
    - VMM manages the real PTBR and a virtual PTBR for each VM
    - When a guest OS is activated, the real PTBR points to the corresponding shadow page table of the guest OS
    - When the guest OS attempts to modify the PTBR, it will be intercepted by VMM for further emulation



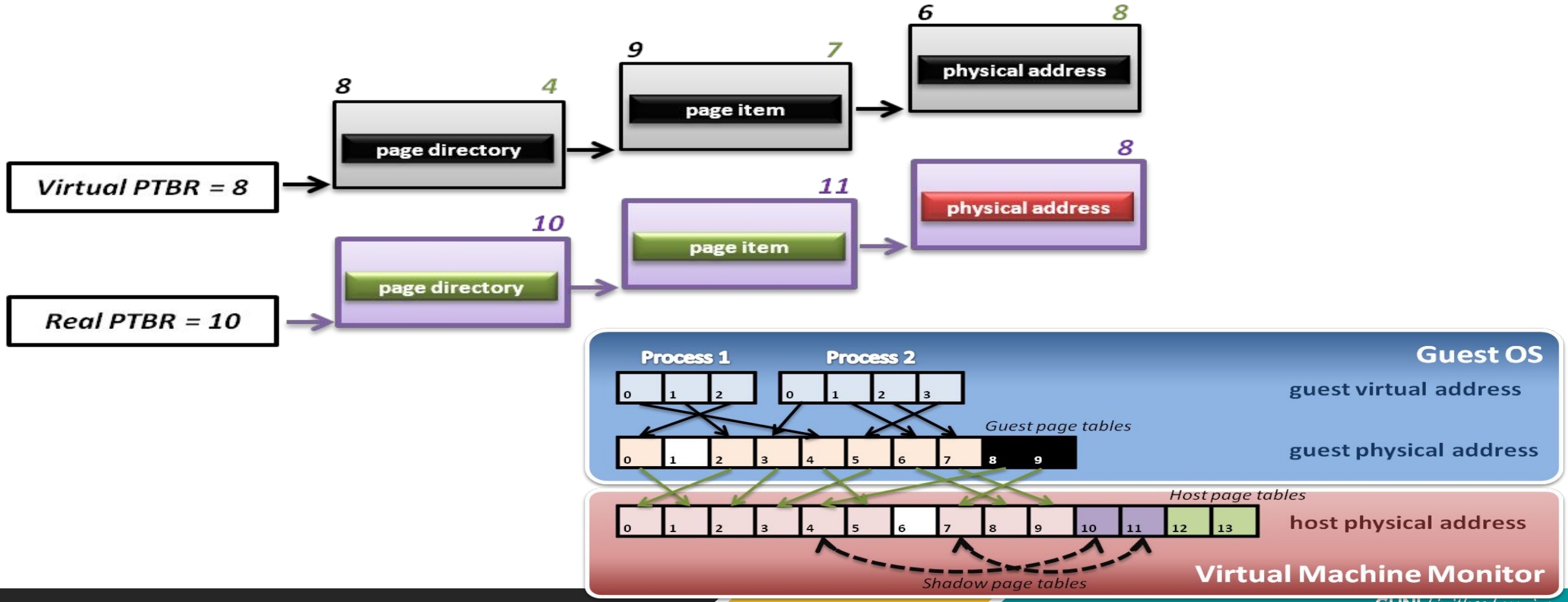
Shadow Page Table

- Construct shadow page table
  - Guest OS will maintain its own page table for each process.
  - VMM maps each guest physical page to host physical page.
  - Create shadow page tables for each guest page table.
  - VMM should protect host frame which contains guest page table. (that means to write protect the guest page tables in host memory)



Shadow Page Table

- Implement with PTBR :
  - For example, process 2 in guest OS wants to access its memory whose page number is 1.

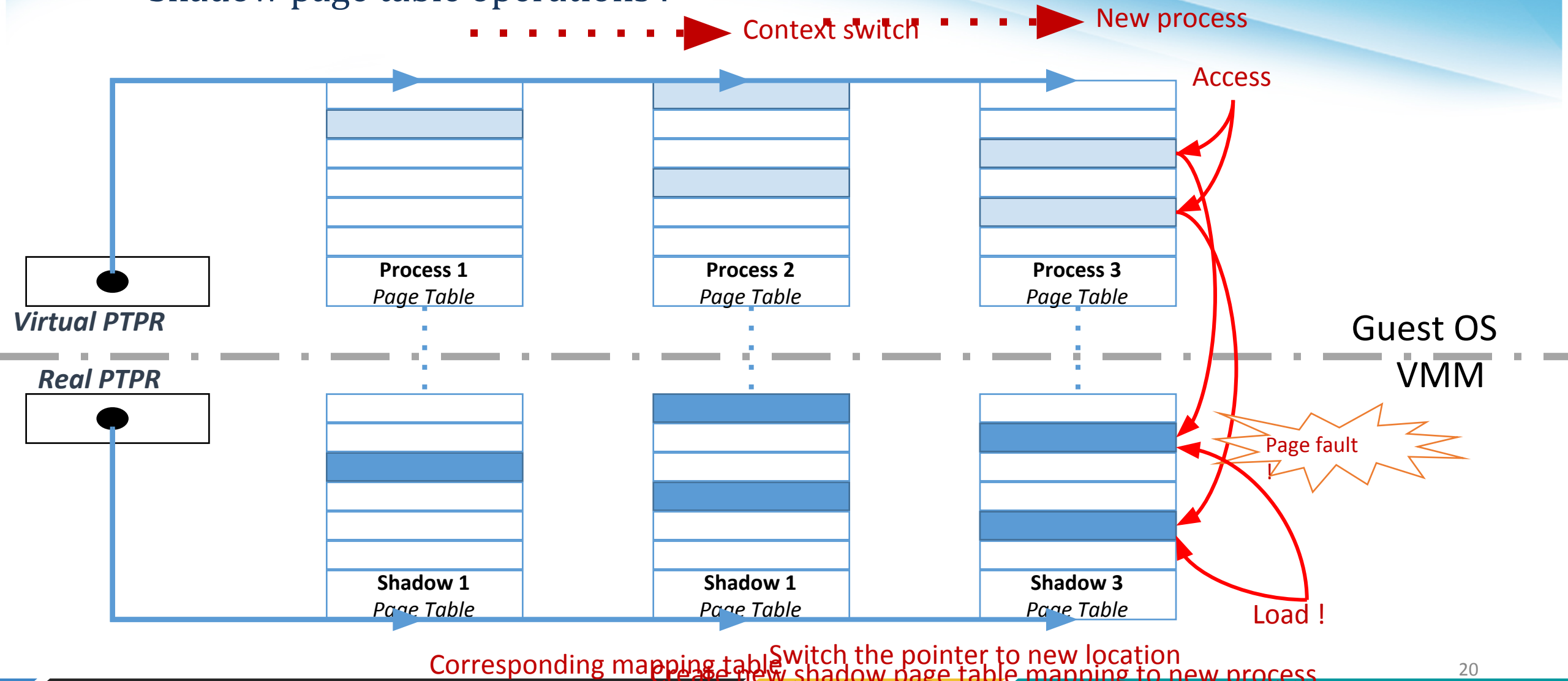


SPT Maintenance

- If guest OS modify one of its page tables, then the corresponding entry of SPT must be updated.
  - We call it “shadow” because SPT is just like the shadow of page tables of guest OS.
- How to identify this kind of modification?
  - Guest OS could read/write a physical frame with the help of SPT.
  - Mark those physical frames used as guest page tables read-only, so that when a guest OS tries to modify its guest page table, an exception would be triggered.
  - Then VMM checks the modification and updates the corresponding entry on SPT

Shadow Page Table

- Shadow page table operations :



Concepts

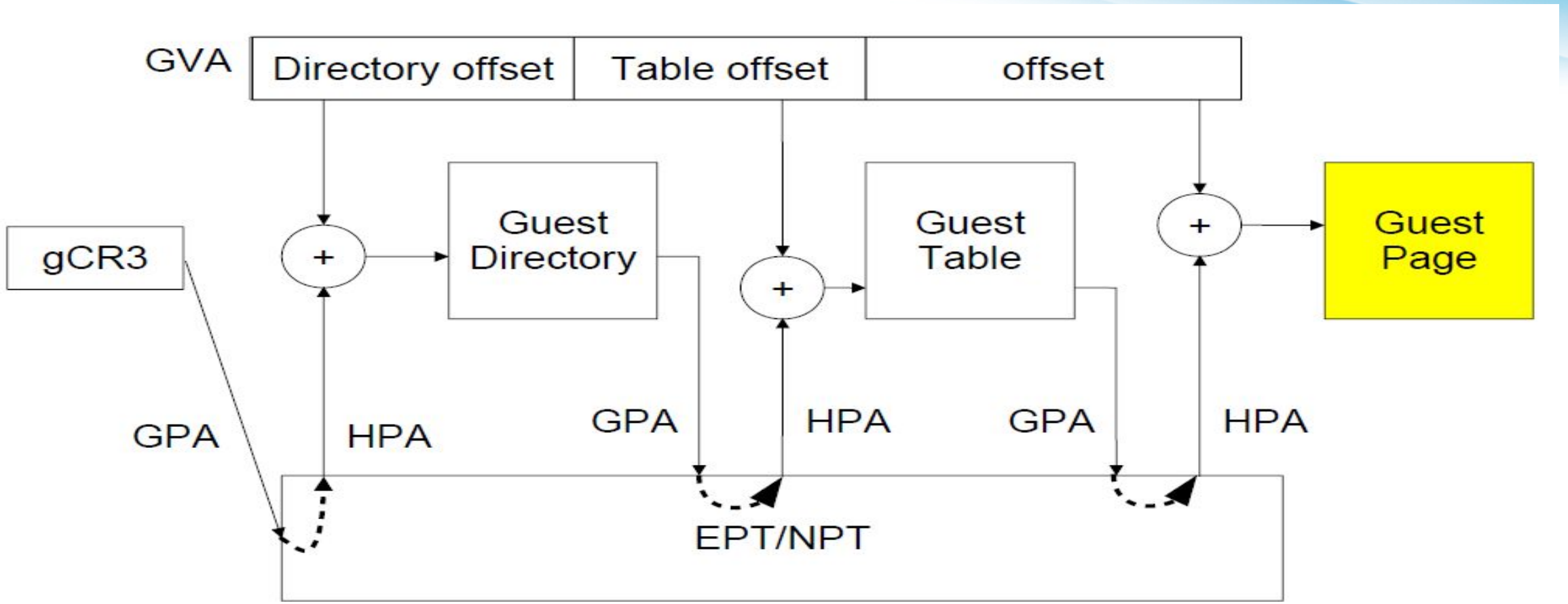
Shadow page table

Hardware assistance

Comparison

MEMORY VIRTUALIZATION

Hardware Assistance: Overview



Hardware Solution

- Difficulties of shadow page table technique :
  - Shadow page table implementation is extremely complex.
  - Page fault mechanism and synchronization issues are critical.
  - Host memory space overhead is considerable.
- But why we need this technique to virtualize MMU ?
  - MMU do not first implemented for virtualization.
  - MMU is knowing nothing about two level page address translation.
- Now, let us consider hardware solution.

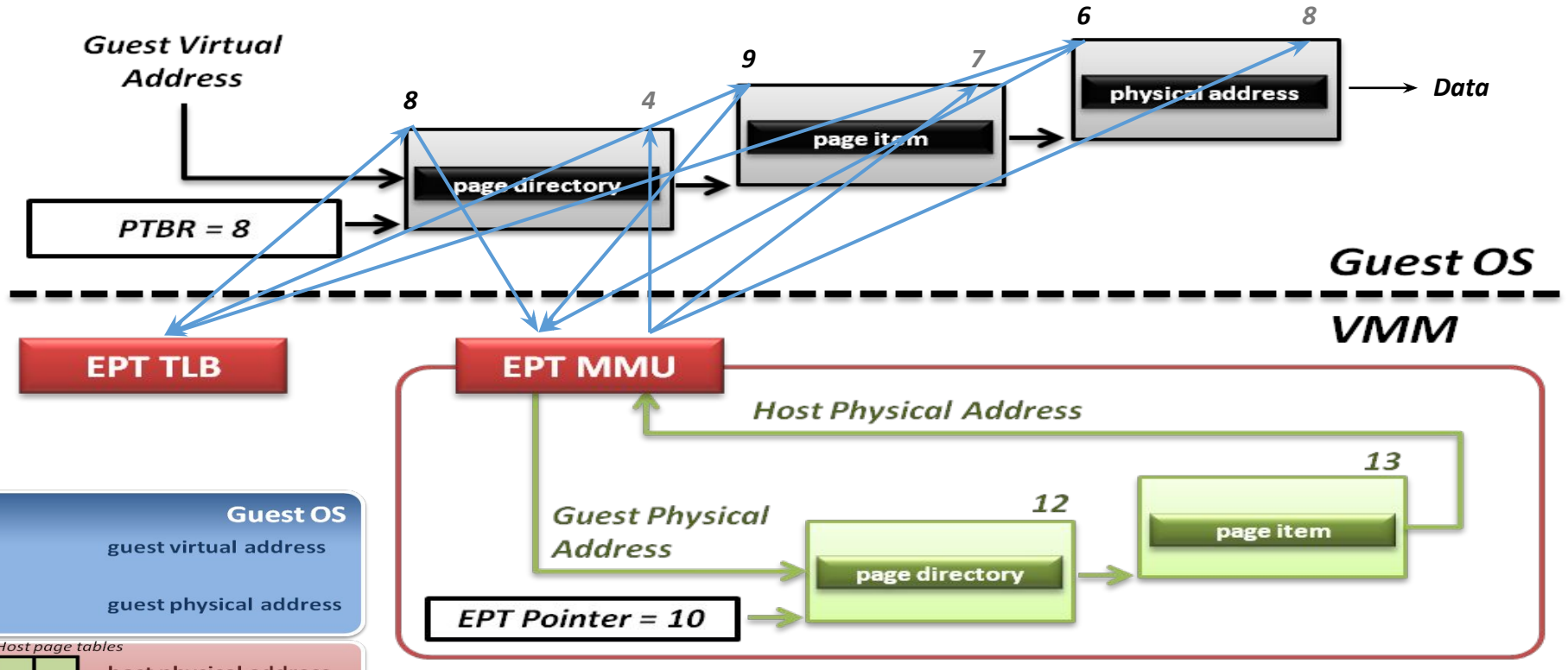
- Extended page tables are Intel's implementation of Second Level Address Translation (SLAT), also known as nested paging, which is used to more efficiently virtualize the memory of guest VMs.
- Basically, guest virtual addresses are first translated to guest physical addresses, which are then translated to host physical addresses. This is all done in hardware (by the MMU) to avoid extra work needing to be done in software by the VMM.

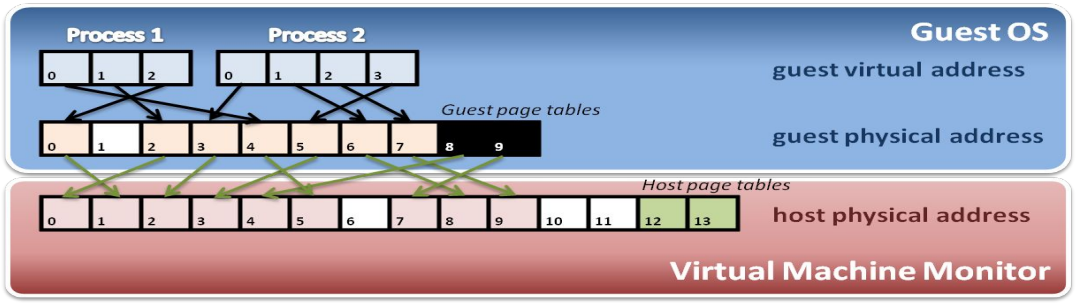
Extended Page Table

- Concept of Extended Page Table (EPT) :
  - Instead of walking along with only one page table hierarchy, EPT technique implement one more page table hierarchy.
    - One page table is maintained by guest OS, which is used to generate guest physical address.
    - The other page table is maintained by VMM, which is used to map guest physical address to host physical address.
  - For each memory access operation, EPT MMU directly gets the guest physical address from guest page table, and then gets the host physical address by the VMM mapping table automatically.

Extended Page Table

- Memory operation :





Concepts

Shadow page table

Hardware assistance

Comparison

MEMORY VIRTUALIZATION

Question

- Computer architecture with virtualization extension is a trend.
- Hardware-assisted techniques replace many software methods of virtualization.
- However, is hardware-assisted implementation a definite winner?

Memory Virtualization Summary

- Software implementation
  - Memory architecture
    - MMU (memory management unit)
    - TLB (translation look-aside buffer)
  - Shadow page table
    - MMU virtualization by virtual PTBR
    - Shadow page table construction
    - Page fault and page table protection
- Hardware assistance
  - Extended page table
    - Hardware walk guest and host page table simultaneously

Reference

- Selective hardware/software memory virtualization
  - http://www.cs.mtu.edu/~zlwang/papers/vee11.pdf
- ARM® Architecture Reference Manual: ARMv7-A and ARMv7-R edition