CS3281 / CS5281

Virtual Memory (contd.)

CS3281 / CS5281

Fall 2025

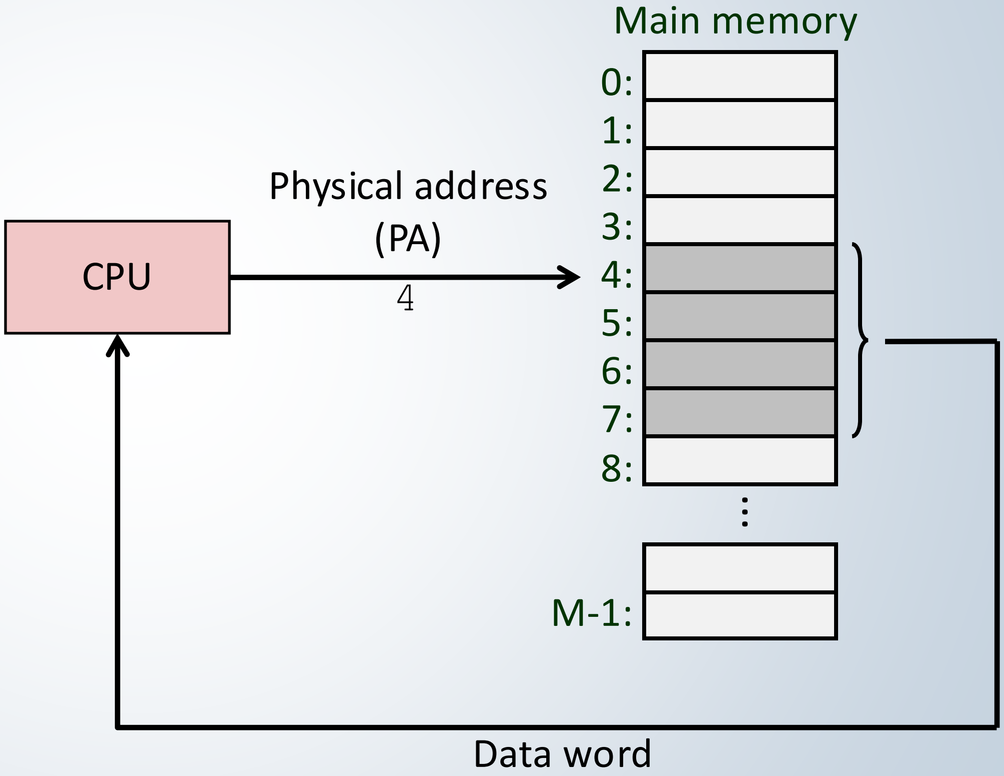
**Some lecture slides borrowed and adapted from CMU's "Computer Systems: A Programmer's Perspective", Ghena, St-Amour, Hardavellas, Bustamente (Northwestern), Bryant, O'Hallaron (CMU), Garcia, Weaver (UC Berkeley)*

Tel (615) 343-7472 | Fax (615) 343-7440
1025 16th Avenue South Nashville, TN 37212
www.isis.vanderbilt.edu

VANDERBILT UNIVERSITY

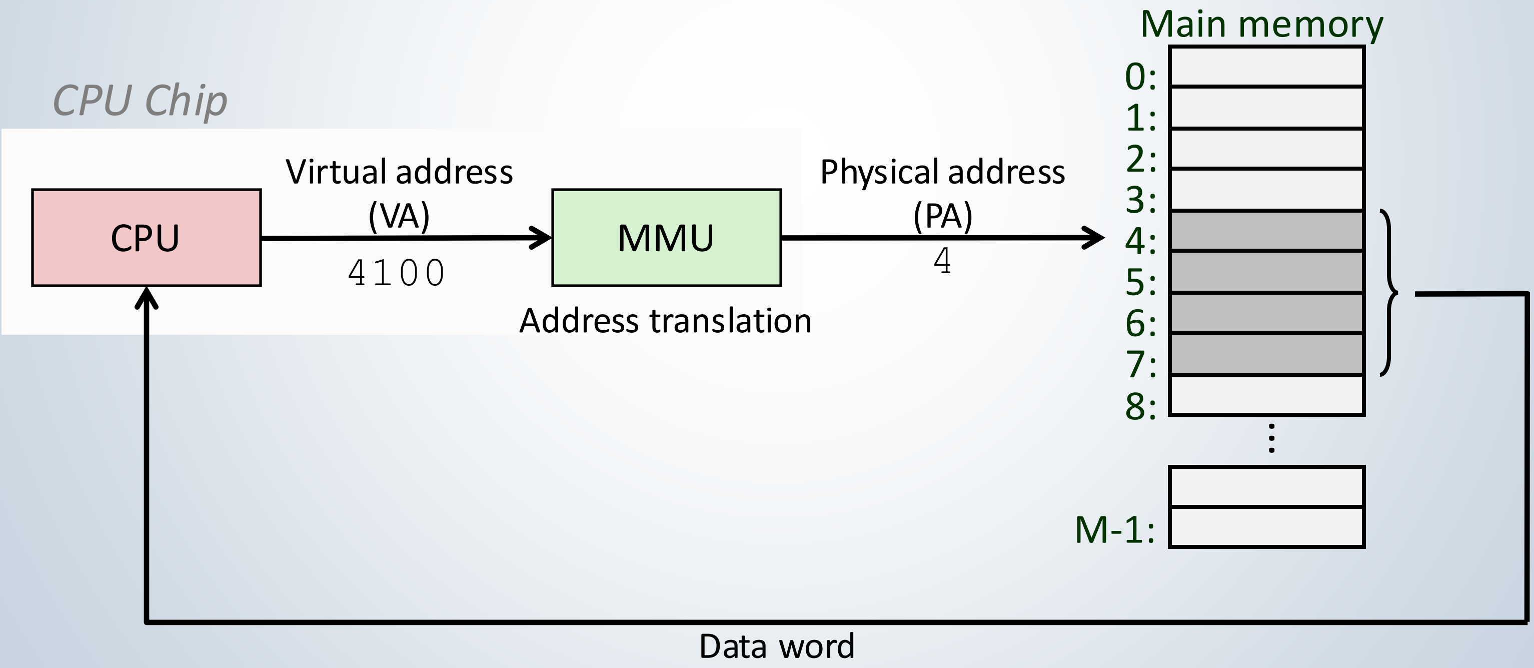
A System Using Physical Addressing

- Used in "simple" systems like embedded microcontrollers in devices like cars, elevators, and digital picture frames



A System Using Virtual Addressing

- Used in all modern servers, laptops, and smart phones
- One of the great ideas in computer science



Address Spaces

- Linear address space: Ordered set of contiguous non-negative integer addresses:

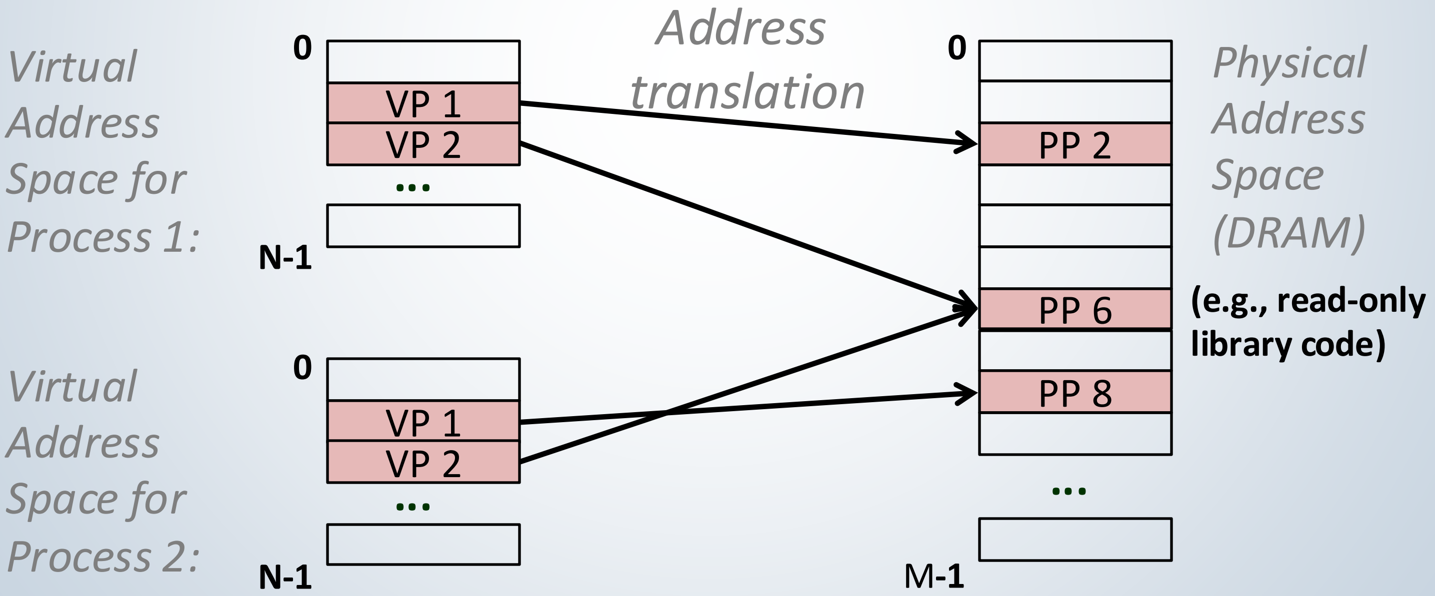
  {0, 1, 2, 3 … }

- Number of bits for virtual/physical addressing tell us how big they are

Tel (615) 343-7472 | Fax (615) 343-7440
1025 16th Avenue South Nashville, TN 37212
www.isis.vanderbilt.edu

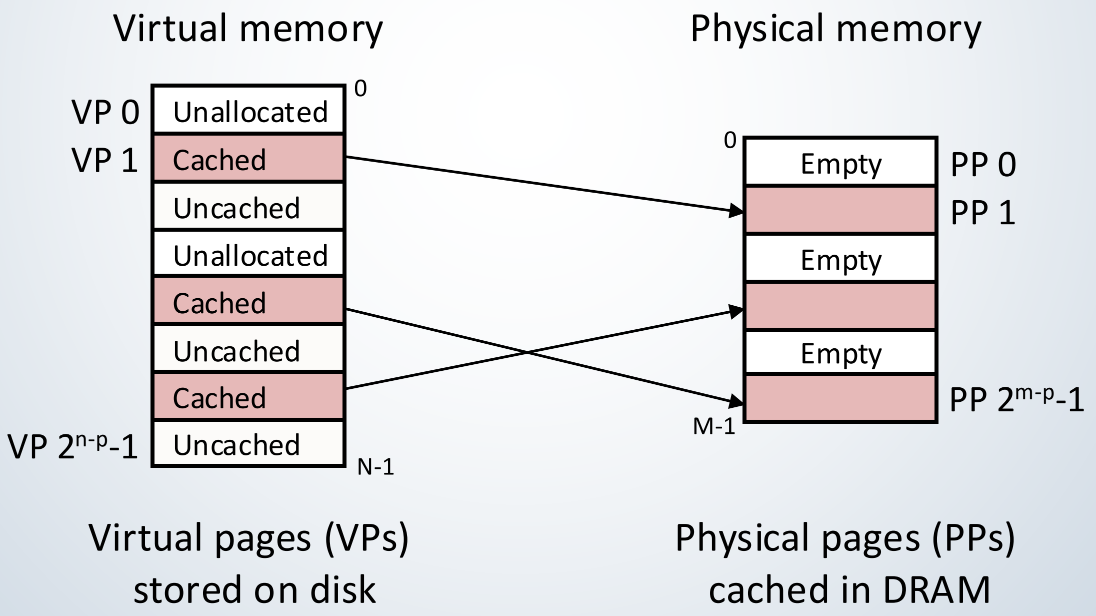
VM as a Tool for Memory Management

- **Key idea: each process has its own <u>virtual address space</u>**
  - It can view memory as a simple linear array
  - Mapping function scatters addresses through physical memory



VM as a Tool for Caching

- Conceptually, *virtual memory* is an array of N contiguous bytes stored on disk.
- The contents of the array on disk are cached in *physical memory* (*DRAM cache*)
  - These cache blocks are called *pages* (size is $P = 2^p$ bytes). Pages may (not) be contiguous in the physical memory



DRAM Cache Organization

- DRAM cache organization driven by the enormous miss penalty
  - DRAM is about ***10x*** slower than SRAM (cache)
  - Disk is about ***10,000x*** slower than DRAM

- Consequences
  - Only a subset of virtual pages are stored in the main memory
  - Highly sophisticated, expensive replacement algorithms
  - Large page (block) size: typically 4 KB
    - CPU caches (SRAM) move data at cache-line granularity (aka "cache blocks"), e.g., 64 B, between SRAM↔DRAM.
    - Virtual memory moves data at page granularity, e.g., 4 KB, between DRAM↔disk. Because the miss penalty to disk is enormous, the transfer unit is much larger than a cache line.

Why pages help: Spatial Locality

- Programs tend to touch nearby addresses soon after one another (arrays, stack frames, sequential code).
- A page fault brings in a whole page (≈4 KB), not just one word → the next few KB of nearby data/code are already in DRAM.
  - This amortizes the enormous disk miss penalty (disk ⋙ DRAM), turning one slow I/O into many fast DRAM hits.
- Typical wins:
  - Instruction streams (sequential fetch within functions)
  - Stacks (new frames/locals)
  - Arrays & buffers (for i = 0…N loops)
  - Contiguous allocations

*Works **only if** locality exists; random sparse access wastes I/O and can cause thrashing.*

Tel (615) 343-7472 | Fax (615) 343-7440
1025 16th Avenue South Nashville, TN 37212
www.isis.vanderbilt.edu

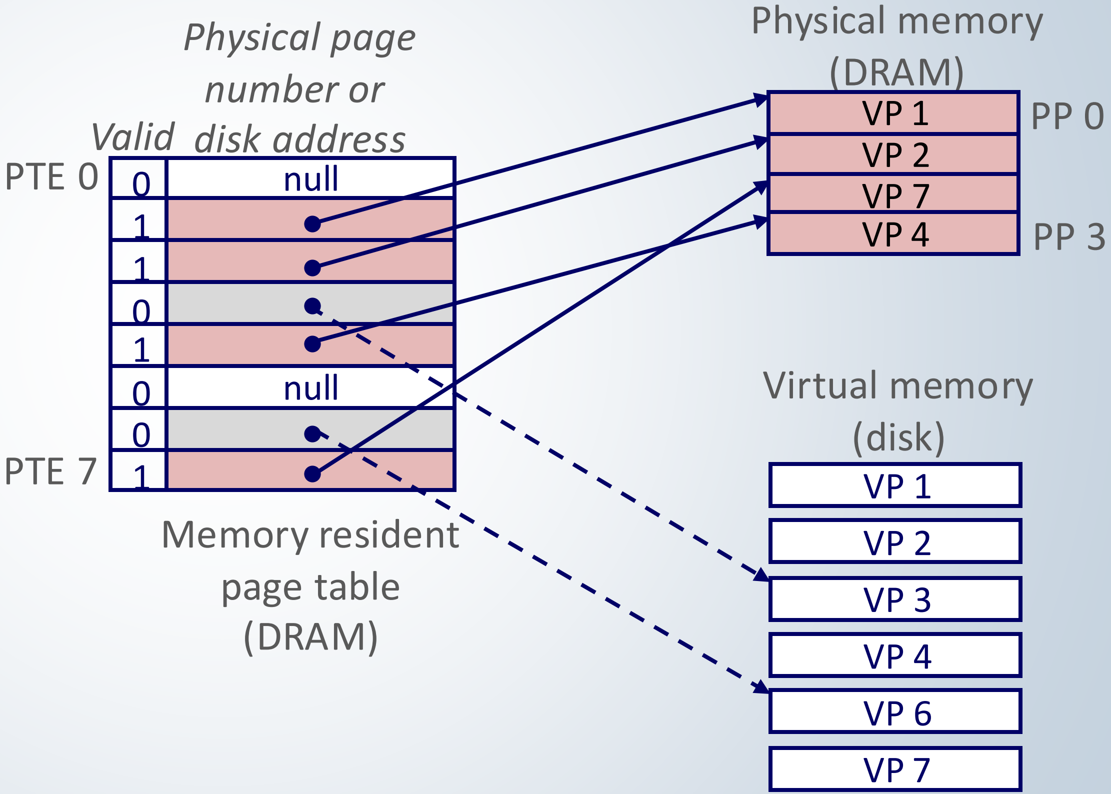
Internal Fragmentation

- Depending on a page size, a program size may not be a multiple of the number of pages. Thus, the last page is partially filled. This loss of usable memory is known as internal fragmentation
- malloc() usually tries to make unused spaces available

Tel (615) 343-7472 | Fax (615) 343-7440
1025 16th Avenue South Nashville, TN 37212
www.isis.vanderbilt.edu

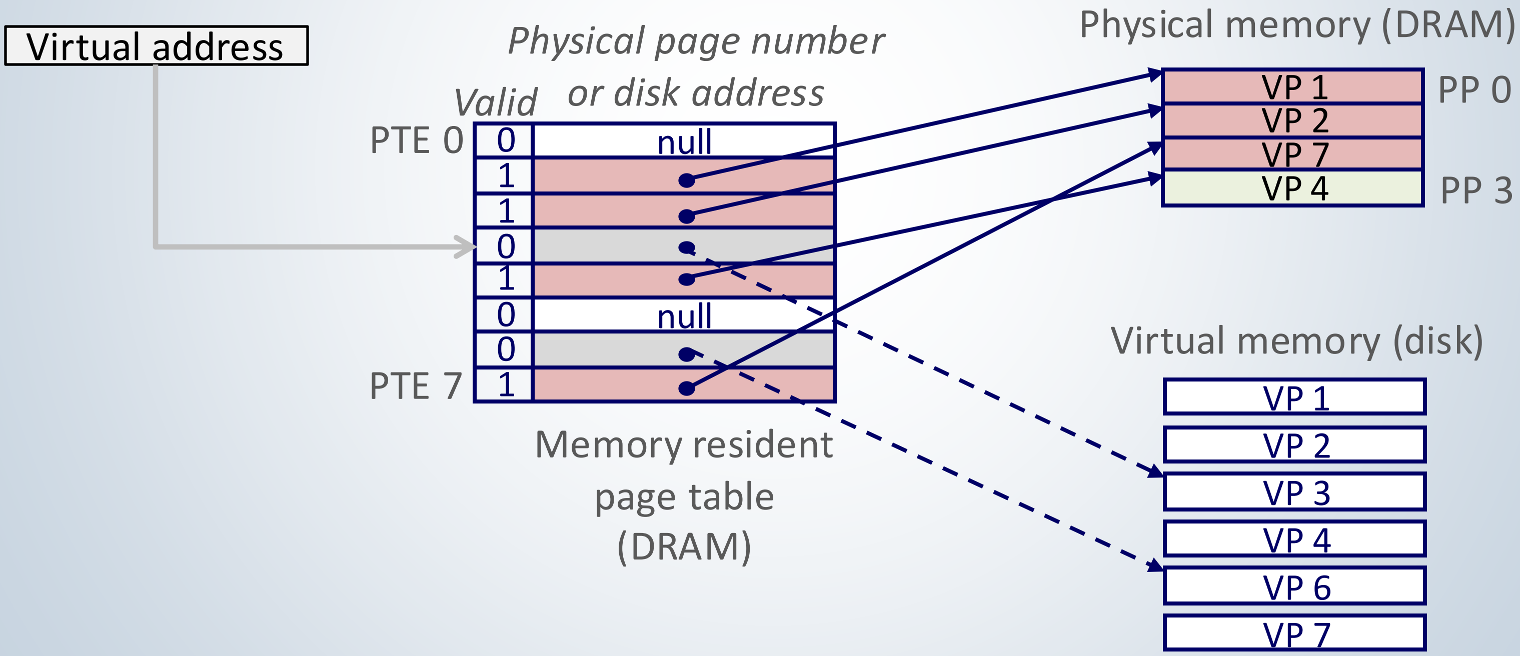
Enabling Data Structure: Page Table

- A *page table* is an array of page table entries (PTEs) that maps virtual pages to physical pages.
  - Per-process kernel data structure in DRAM



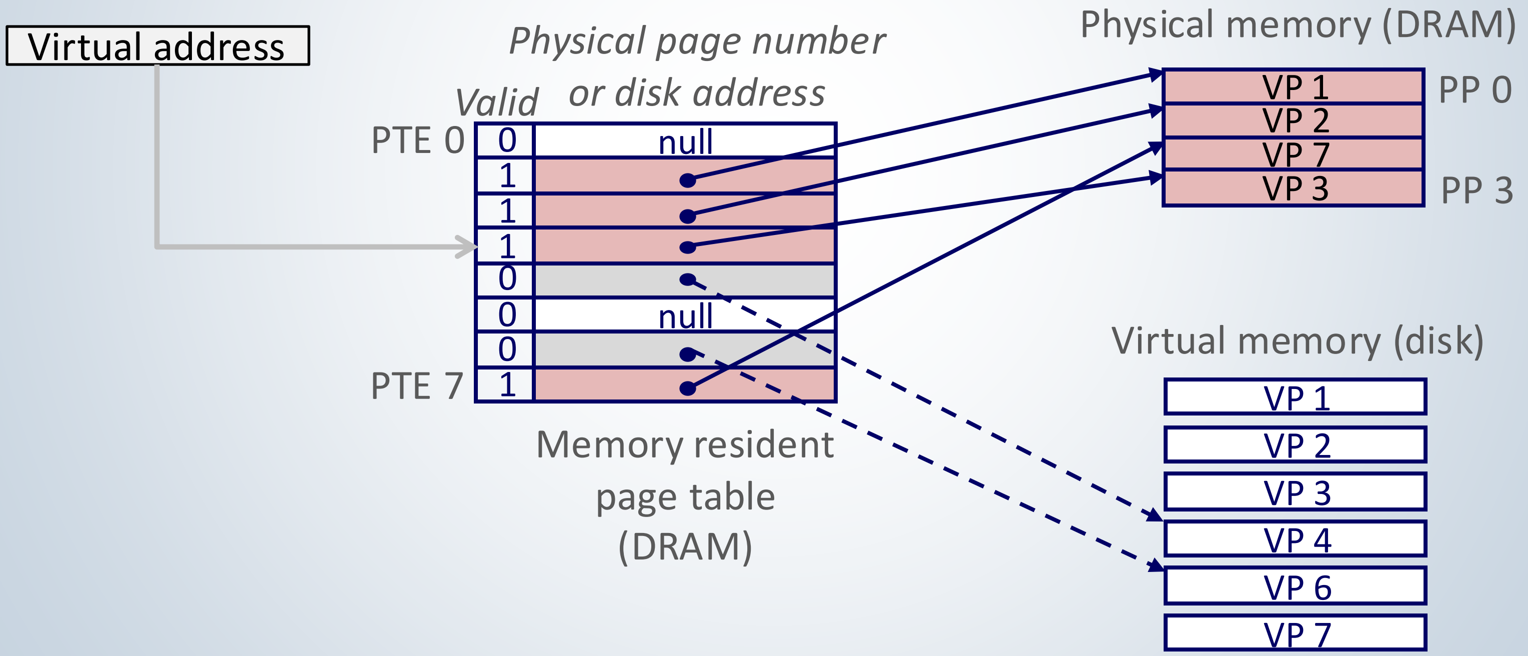
Handling Page Fault

- Page miss causes page fault (an exception)
- Page fault handler selects a victim to be evicted (here VP 4)



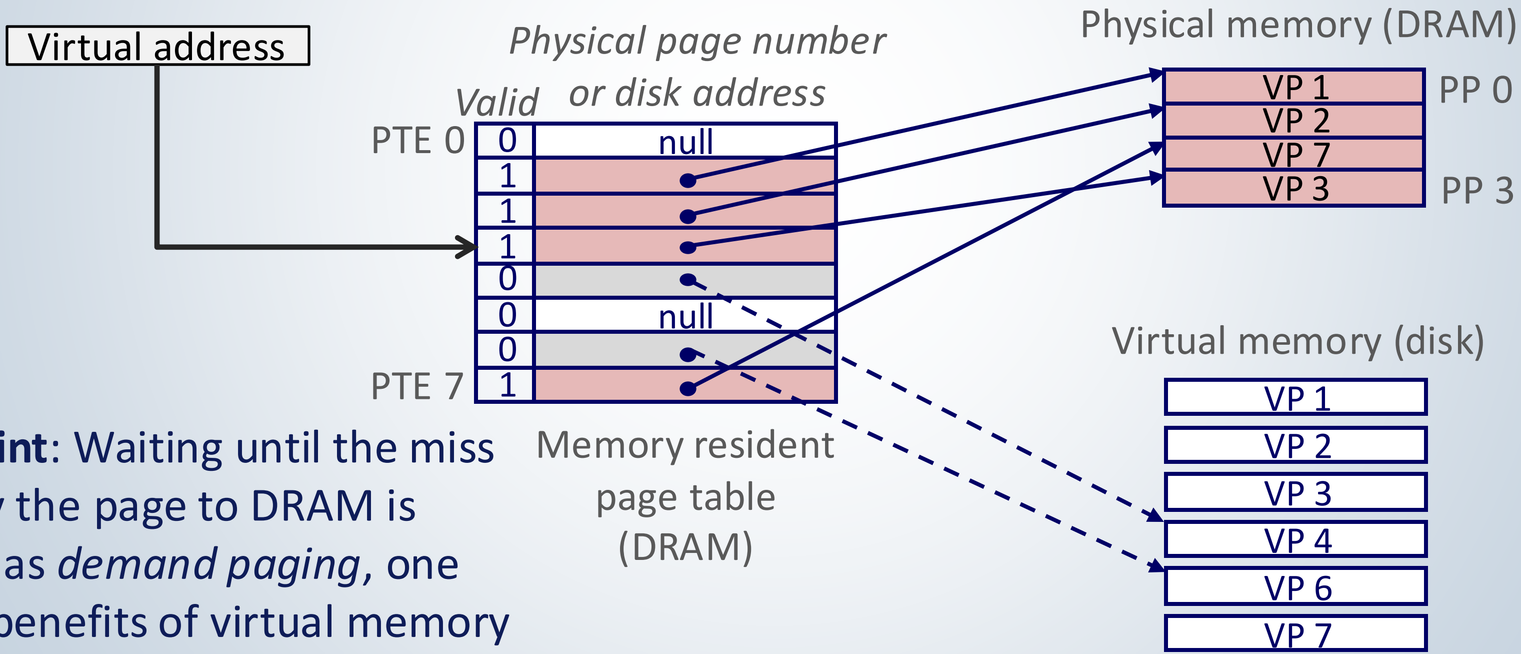
Handling Page Fault

- Page miss causes page fault (an exception)
- Page fault handler selects a victim to be evicted (here VP 4)



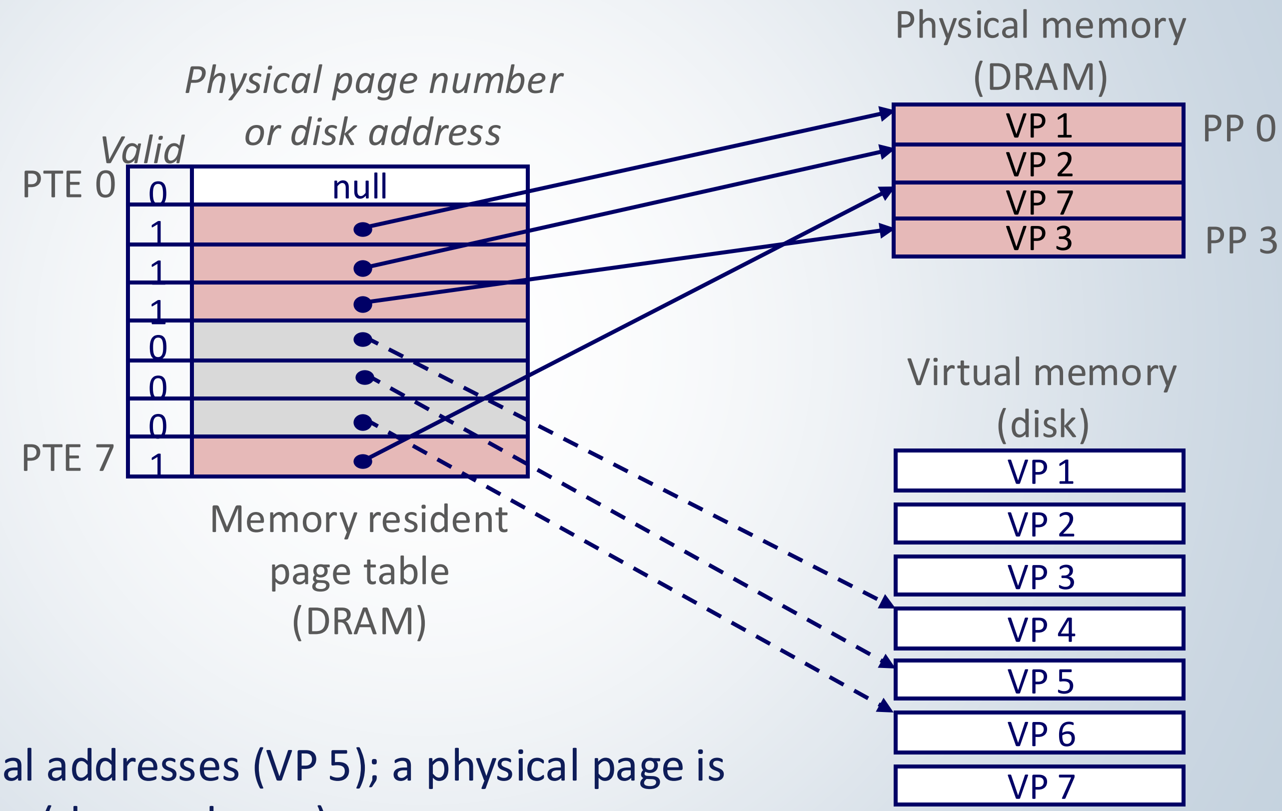
Handling Page Fault

- Page miss causes page fault (an exception)
- Page fault handler selects a victim to be evicted (here VP 4)
- Offending (faulting) instruction is restarted: page hit! On a **page fault**, the OS brings the page **from disk into DRAM** (or allocates a zero-filled page for anonymous memory), updates the PTE, and **restarts the faulting instruction**.



Key point: Waiting until the miss to copy the page to DRAM is known as *demand paging*, one of the benefits of virtual memory

Allocating Pages



malloc() reserves virtual addresses (VP 5); a physical page is allocated on first access (demand-zero)

VM Address Translation

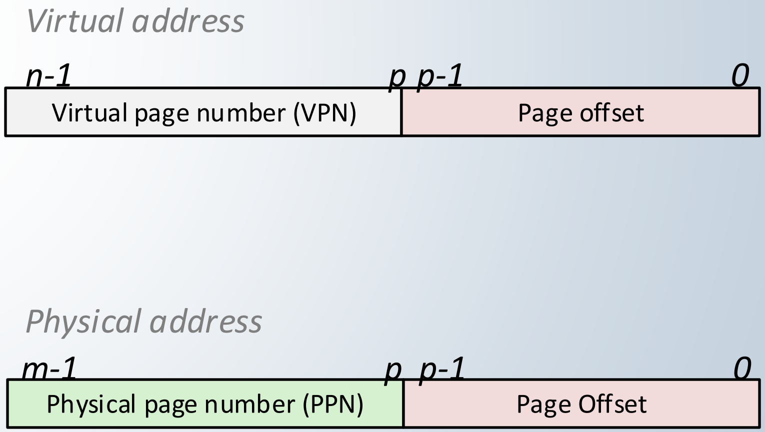
- Virtual Address Space
  - $V = \{0, 1, \ldots, N-1\}$
- Physical Address Space
  - $P = \{0, 1, \ldots, M-1\}$
- Address Translation
  - $\boldsymbol{MAP: \; V \rightarrow P \; U \; \{\varnothing\}}$
  - For virtual address ***a***:
    - $\boldsymbol{MAP(a) = a'}$ if data at virtual address ***a*** is at physical address ***a'*** in ***P***
    - $\boldsymbol{MAP(a) = \varnothing}$ if data at virtual address ***a*** is not in physical memory (page fault)

Breaking down virtual addresses

- Basic Parameters
  - **$N = 2^n$** : Number of addresses in virtual address space
  - **$M = 2^m$** : Number of addresses in physical address space. $m \leq n$ (usually much less)
  - **$P = 2^p$** : Page size (bytes)

- Components of the virtual address (VA)
  - Virtual page number (VPN): **n-p** bits
  - Page Offset: **p** bits

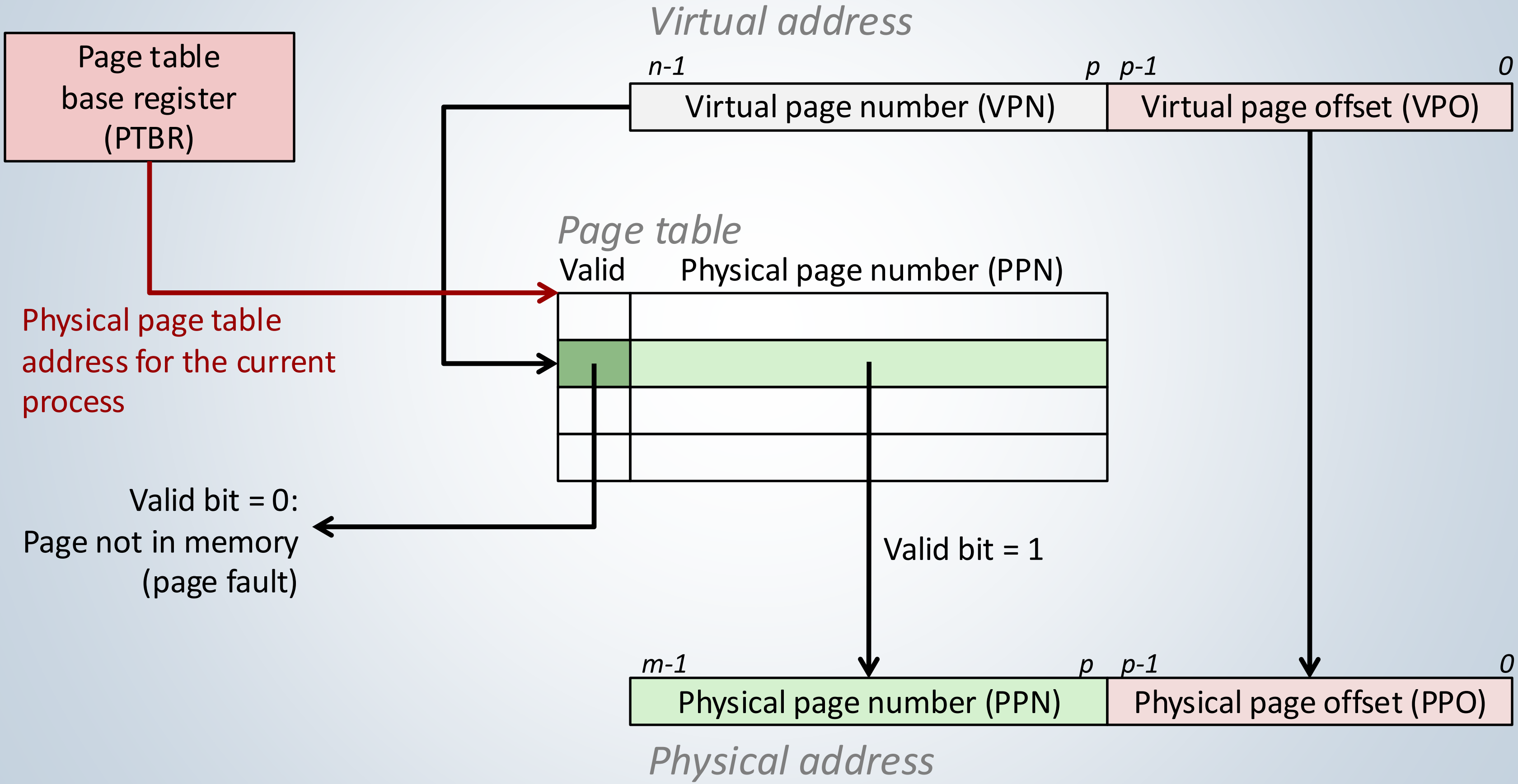
- Components of the physical address (PA)
  - Physical page number (PPN): **m-p** bits
  - Page Offset (same offset as VA): **p** bits



Tel (615) 343-7472 | Fax (615) 343-7440
1025 16th Avenue South Nashville, TN 37212
www.isis.vanderbilt.edu

VANDERBILT UNIVERSITY

Address Translation with a Page Table



Virtual memory example

- Parameters
  - Virtual addresses are 12-bits
  - Physical addresses are 16-bits
  - Page size is 64 bytes

For this toy example, which space is larger isn't important. **In real systems, virtual space is usually much larger than physical**.

1. How do we split Virtual Addresses into VPN and Offset?

11	10	9	8	7	6	5	4	3	2	1	0

Tel (615) 343-7472 | Fax (615) 343-7440
1025 16th Avenue South Nashville, TN 37212
www.isis.vanderbilt.edu

VANDERBILT UNIVERSITY

Virtual memory example

- Parameters
  - Virtual addresses are 12-bits
  - Physical addresses are 16-bits
  - Page size is 64 bytes

Mapping can be anything, which is bigger doesn't really matter!

1. How do we split Virtual Addresses into VPN and Offset?
   - Offset is based on page size: 64-bytes ⇨ 6 bits. All the rest are VPN

11	10	9	8	7	6	5	4	3	2	1	0
Virtual Page Number						Page Offset					

2. How big are Physical Page Numbers?

Tel (615) 343-7472 | Fax (615) 343-7440
1025 16th Avenue South Nashville, TN 37212
www.isis.vanderbilt.edu

Virtual memory example

- Parameters
  - Virtual addresses are 12-bits
  - Physical addresses are 16-bits
  - Page size is 64 bytes

Mapping can be anything, which is bigger doesn't really matter!

1. How do we split Virtual Addresses into VPN and Offset?
   - Offset is based on page size: 64-bytes ⇨ 6 bits. All the rest are VPN

11	10	9	8	7	6	5	4	3	2	1	0
Virtual Page Number						Page Offset					

2. How big are Physical Page Numbers? 16-6 = 10 bits

Tel (615) 343-7472 | Fax (615) 343-7440
1025 16th Avenue South Nashville, TN 37212
www.isis.vanderbilt.edu

Virtual memory example

- Parameters
  - Virtual addresses are 12-bits
  - Physical addresses are 16-bits
  - Page size is 64 bytes

11	10	9	8	7	6	5	4	3	2	1	0
Virtual Page Number						Page Offset					

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Physical Page Number										Page Offset					

- Translate:
- Virtual address: 0x3F0
  - Binary:
  - VPN:
  - Offset:

Tel (615) 343-7472 | Fax (615) 343-7440
1025 16th Avenue South Nashville, TN 37212
www.isis.vanderbilt.edu

Virtual memory example

- Parameters
  - Virtual addresses are 12-bits
  - Physical addresses are 16-bits
  - Page size is 64 bytes

11	10	9	8	7	6	5	4	3	2	1	0
Virtual Page Number						Page Offset					

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Physical Page Number										Page Offset					

- Translate:
- Virtual address: 0x3F0
  - Binary: 0b001111110000
  - VPN: 0b001111
  - Offset: 0b110000

Tel (615) 343-7472 | Fax (615) 343-7440
1025 16th Avenue South Nashville, TN 37212
www.isis.vanderbilt.edu

Virtual memory example

- Parameters
  - Virtual addresses are 12-bits
  - Physical addresses are 16-bits
  - Page size is 64 bytes

11	10	9	8	7	6	5	4	3	2	1	0
Virtual Page Number						Page Offset					

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Physical Page Number										Page Offset					

- Translate:
- Virtual address: 0x3F0
  - Binary: 0b001111110000
  - VPN: 0b001111
  - Offset: 0b110000

VPN	PPN	Valid
0x00	0x123	1
0x01	0x156	1
0x02	0x143	1
0x03	0x16F	1
0x04	0x1FF	0
0x05	0x107	0
0x06	0x100	0
0x07	0x1C0	0
0x08	0x1D8	0
0x09	0x1BF	0
0x0A	0x000	1
0x0B	0x3FF	1
0x0C	0x308	0
0x0D	0x3FD	0
0x0E	0x111	1
0x0F	0x1F0	1

VPN	PPN	Valid
0x10	0x237	1
0x11	0x236	1
0x12	0x2B0	1
0x13	0x280	0
0x14	0x120	0
Continues on…		

- PPN:
- Offset:

Tel (615) 343-7472 | Fax (615) 343-7440
1025 16th Avenue South Nashville, TN 37212
www.isis.vanderbilt.edu

Virtual memory example

- Parameters
  - Virtual addresses are 12-bits
  - Physical addresses are 16-bits
  - Page size is 64 bytes

11	10	9	8	7	6	5	4	3	2	1	0
Virtual Page Number						Page Offset					

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Physical Page Number										Page Offset					

- Translate:
- Virtual address: 0x3F0
  - Binary: 0b001111110000
  - VPN: 0b001111
  - Offset: 0b110000

VPN	PPN	Valid
0x00	0x123	1
0x01	0x156	1
0x02	0x143	1
0x03	0x16F	1
0x04	0x1FF	0
0x05	0x107	0
0x06	0x100	0
0x07	0x1C0	0
0x08	0x1D8	0
0x09	0x1BF	0
0x0A	0x000	1
0x0B	0x3FF	1
0x0C	0x308	0
0x0D	0x3FD	0
0x0E	0x111	1
0x0F	0x1F0	1

VPN	PPN	Valid
0x10	0x237	1
0x11	0x236	1
0x12	0x2B0	1
0x13	0x280	0
0x14	0x120	0
Continues on…		

- PPN: 0b01 1111 0000
- Offset: 0b110000
- Physical address:

Tel (615) 343-7472 | Fax (615) 343-7440
1025 16th Avenue South Nashville, TN 37212
www.isis.vanderbilt.edu

Virtual memory example

- Parameters
  - Virtual addresses are 12-bits
  - Physical addresses are 16-bits
  - Page size is 64 bytes

11	10	9	8	7	6	5	4	3	2	1	0
Virtual Page Number						Page Offset					

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Physical Page Number										Page Offset					

- Translate:
- Virtual address: 0x3F0
  - Binary: 0b001111110000
  - VPN: 0b001111
  - Offset: 0b110000

VPN	PPN	Valid
0x00	0x123	1
0x01	0x156	1
0x02	0x143	1
0x03	0x16F	1
0x04	0x1FF	0
0x05	0x107	0
0x06	0x100	0
0x07	0x1C0	0
0x08	0x1D8	0
0x09	0x1BF	0
0x0A	0x000	1
0x0B	0x3FF	1
0x0C	0x308	0
0x0D	0x3FD	0
0x0E	0x111	1
0x0F	0x1F0	1

VPN	PPN	Valid
0x10	0x237	1
0x11	0x236	1
0x12	0x2B0	1
0x13	0x280	0
0x14	0x120	0
Continues on…		

- PPN: 0b01 1111 0000
- Offset: 0b110000
- Physical address:
  - 0b0111110000110000
  - 0x7C30

Tel (615) 343-7472 | Fax (615) 343-7440
1025 16th Avenue South Nashville, TN 37212
www.isis.vanderbilt.edu

- Parameters
  - Virtual addresses are 12-bits
  - Physical addresses are 16-bits
  - Page size is 64 bytes

11	10	9	8	7	6	5	4	3	2	1	0
Virtual Page Number						Page Offset					

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Physical Page Number										Page Offset					

- Translate:
- Virtual address: 0x500
  - Binary:
  - VPN:
  - Offset:

VPN	PPN	Valid
0x00	0x123	1
0x01	0x156	1
0x02	0x143	1
0x03	0x16F	1
0x04	0x1FF	0
0x05	0x107	0
0x06	0x100	0
0x07	0x1C0	0
0x08	0x1D8	0
0x09	0x1BF	0
0x0A	0x000	1
0x0B	0x3FF	1
0x0C	0x308	0
0x0D	0x3FD	0
0x0E	0x111	1
0x0F	0x1F0	1

VPN	PPN	Valid
0x10	0x237	1
0x11	0x236	1
0x12	0x2B0	1
0x13	0x280	0
0x14	0x120	0
Continues on…		

- PPN:
- Offset:
- Physical address:

Tel (615) 343-7472 | Fax (615) 343-7440
1025 16th Avenue South Nashville, TN 37212
www.isis.vanderbilt.edu

VANDERBILT UNIVERSITY

- Parameters
  - Virtual addresses are 12-bits
  - Physical addresses are 16-bits
  - Page size is 64 bytes

11	10	9	8	7	6	5	4	3	2	1	0
Virtual Page Number						Page Offset					

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Physical Page Number										Page Offset					

- Translate:
- Virtual address: 0x500
  - Binary: 0b010100000000
  - VPN: 0b010100
  - Offset: 0b000000

VPN	PPN	Valid
0x00	0x123	1
0x01	0x156	1
0x02	0x143	1
0x03	0x16F	1
0x04	0x1FF	0
0x05	0x107	0
0x06	0x100	0
0x07	0x1C0	0
0x08	0x1D8	0
0x09	0x1BF	0
0x0A	0x000	1
0x0B	0x3FF	1
0x0C	0x308	0
0x0D	0x3FD	0
0x0E	0x111	1
0x0F	0x1F0	1

VPN	PPN	Valid
0x10	0x237	1
0x11	0x236	1
0x12	0x2B0	1
0x13	0x280	0
0x14	0x120	0
Continues on…		

- PPN: INVALID
- Offset:
- Physical address:
  - **Page Fault**

Tel (615) 343-7472 | Fax (615) 343-7440
1025 16th Avenue South Nashville, TN 37212
www.isis.vanderbilt.edu

- Parameters
  - Virtual addresses are 12-bits
  - Physical addresses are 16-bits
  - Page size is 64 bytes

11	10	9	8	7	6	5	4	3	2	1	0
Virtual Page Number						Page Offset					

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Physical Page Number										Page Offset					

- Translate:
- Virtual address: 0x0D6
  - Binary:
  - VPN:
  - Offset:

VPN	PPN	Valid
0x00	0x123	1
0x01	0x156	1
0x02	0x143	1
0x03	0x16F	1
0x04	0x1FF	0
0x05	0x107	0
0x06	0x100	0
0x07	0x1C0	0
0x08	0x1D8	0
0x09	0x1BF	0
0x0A	0x000	1
0x0B	0x3FF	1
0x0C	0x308	0
0x0D	0x3FD	0
0x0E	0x111	1
0x0F	0x1F0	1

VPN	PPN	Valid
0x10	0x237	1
0x11	0x236	1
0x12	0x2B0	1
0x13	0x280	0
0x14	0x120	0
Continues on...		

- PPN:
- Offset:
- Physical address:

- Parameters
  - Virtual addresses are 12-bits
  - Physical addresses are 16-bits
  - Page size is 64 bytes

11	10	9	8	7	6	5	4	3	2	1	0
Virtual Page Number						Page Offset					

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Physical Page Number										Page Offset					

- Translate:
- Virtual address: 0x0D6
  - Binary: 0b000011010110
  - VPN: 0b000011
  - Offset: 0b010110

VPN	PPN	Valid
0x00	0x123	1
0x01	0x156	1
0x02	0x143	1
0x03	0x16F	1
0x04	0x1FF	0
0x05	0x107	0
0x06	0x100	0
0x07	0x1C0	0
0x08	0x1D8	0
0x09	0x1BF	0
0x0A	0x000	1
0x0B	0x3FF	1
0x0C	0x308	0
0x0D	0x3FD	0
0x0E	0x111	1
0x0F	0x1F0	1

VPN	PPN	Valid
0x10	0x237	1
0x11	0x236	1
0x12	0x2B0	1
0x13	0x280	0
0x14	0x120	0
Continues on…		

- PPN: 0b010 110 1111
- Offset: 0b010110

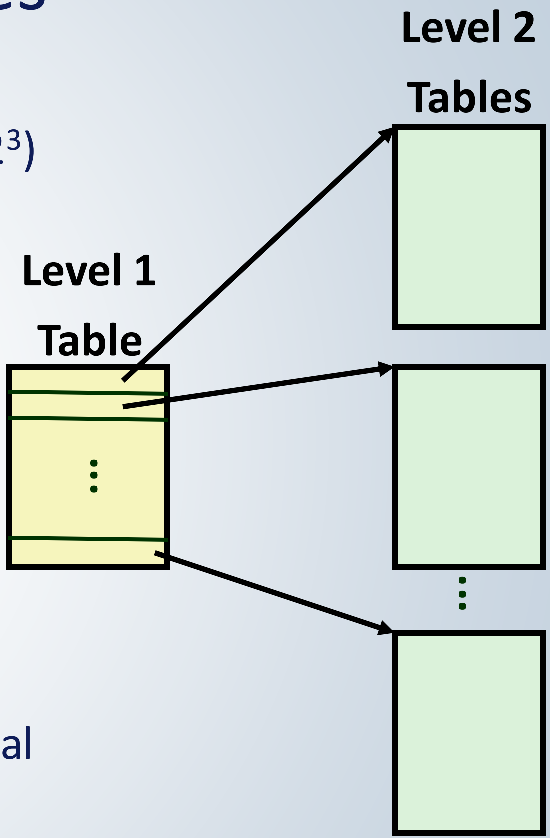
- Physical address:
  - 0b0101101111010110
  - 0x5BD6

Tel (615) 343-7472 | Fax (615) 343-7440
1025 16th Avenue South Nashville, TN 37212
www.isis.vanderbilt.edu

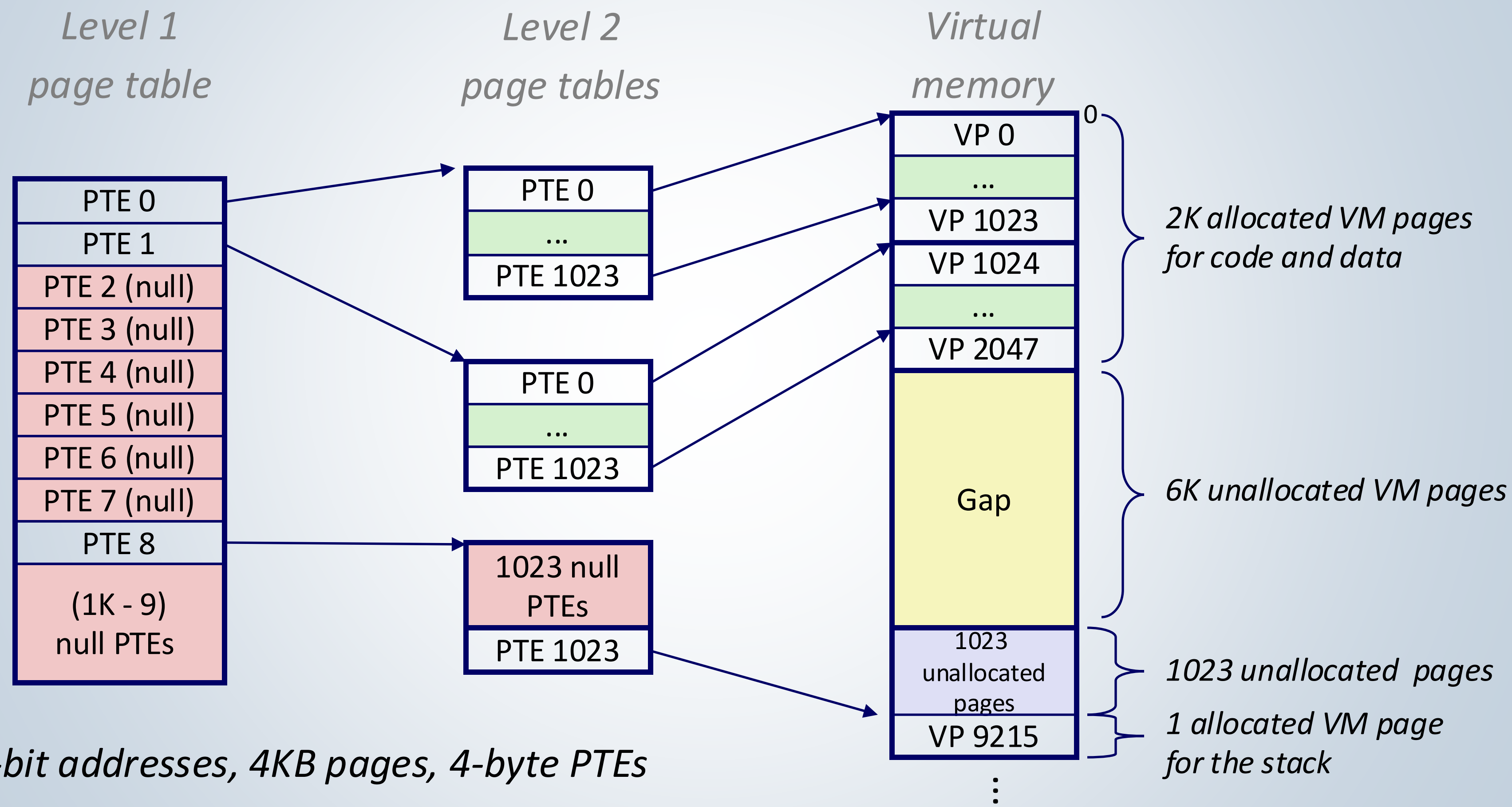
VANDERBILT UNIVERSITY

Multi-Level Page Tables

- Suppose:
  - 4KB (2^{12}) page size, 48-bit address space, 8-byte PTE (2^3)
- Problem:
  - Would need a 512 GB page table!
    - $2^{48} * 2^{-12} * 2^3 = 2^{39}$ bytes
- Common solution: Multi-level page table
- Example: 2-level page table
  - Level 1 table: each PTE points to a page table (always memory resident)
  - Level 2 table: each PTE points to a data page (paged in and out like any other data)
  - Huge advantage: Allocate an L2 table when some virtual pages in that region are actually used

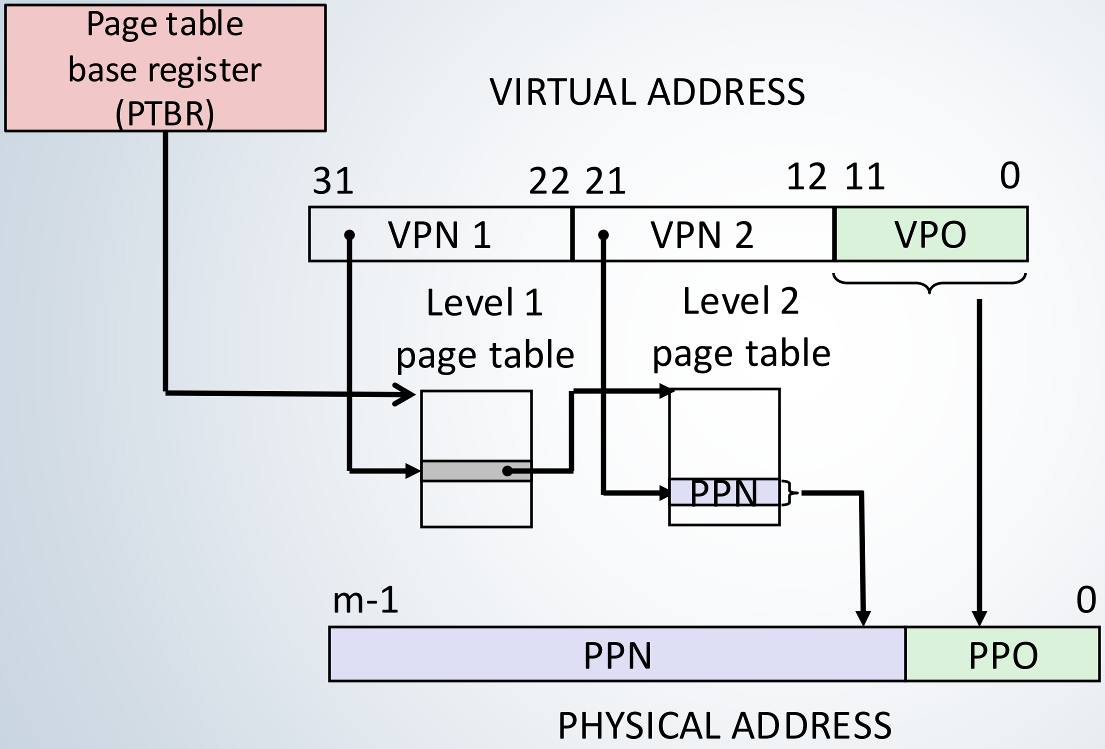


Two-Level Page-Table Hierarchy



32-bit addresses, 4KB pages, 4-byte PTEs

Translating with a 2-Level Page Table



Tel (615) 343-7472 | Fax (615) 343-7440
1025 16th Avenue South Nashville, TN 37212
www.isis.vanderbilt.edu

Multi-level page table (example)

- Consider a 3-level page table. A page table has 64 entries. If the page size is 4K bytes, how much is the size of the virtual address space?

Tel (615) 343-7472 | Fax (615) 343-7440
1025 16th Avenue South Nashville, TN 37212
www.isis.vanderbilt.edu

Multi-level page table (example)

- Consider a 3-level page table. A page table has 64 entries. If the page size is 4K bytes, how much is the size of the virtual address space?
- 64 entries per table: 6 bits of index per level
- Page size = 4KB, 2^12 -> 12 offset bits
- Bits view
  – 3 levels * 6 bits = 18 + offset = 30 bits
  – The virtual address space size is 2^{30}bytes = **1 GiB**

Tel (615) 343-7472 | Fax (615) 343-7440
1025 16th Avenue South Nashville, TN 37212
www.isis.vanderbilt.edu

Multi-level page table (example)

- Another view:
- L3 table maps 64 pages: 64 x 4KB = 256KB
- L2 table has 64 entries, each mapping to an L3: 64 x 256KB = 16MB
- L1 has 64 entries, each mapping to an L2: 64 x 16MB: 1GB