- [x] 1. Last time
- [x] 2. Intro to virtual memory
- [ ] 3. Paging
  - [x] Intro
  - [ ] Key data structure: page table
  - [ ] Multilevel page table
  - [ ] Alternatives/tradeoffs

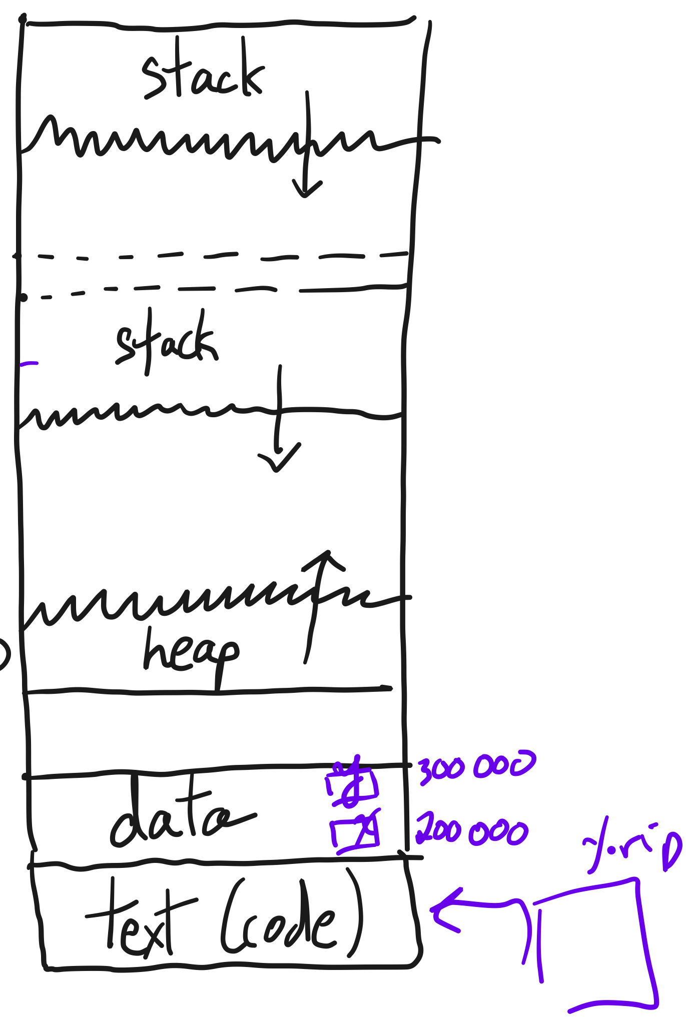
---

2. Intro to virtual memory

process "sees"

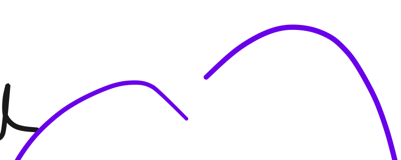
program excerpt:

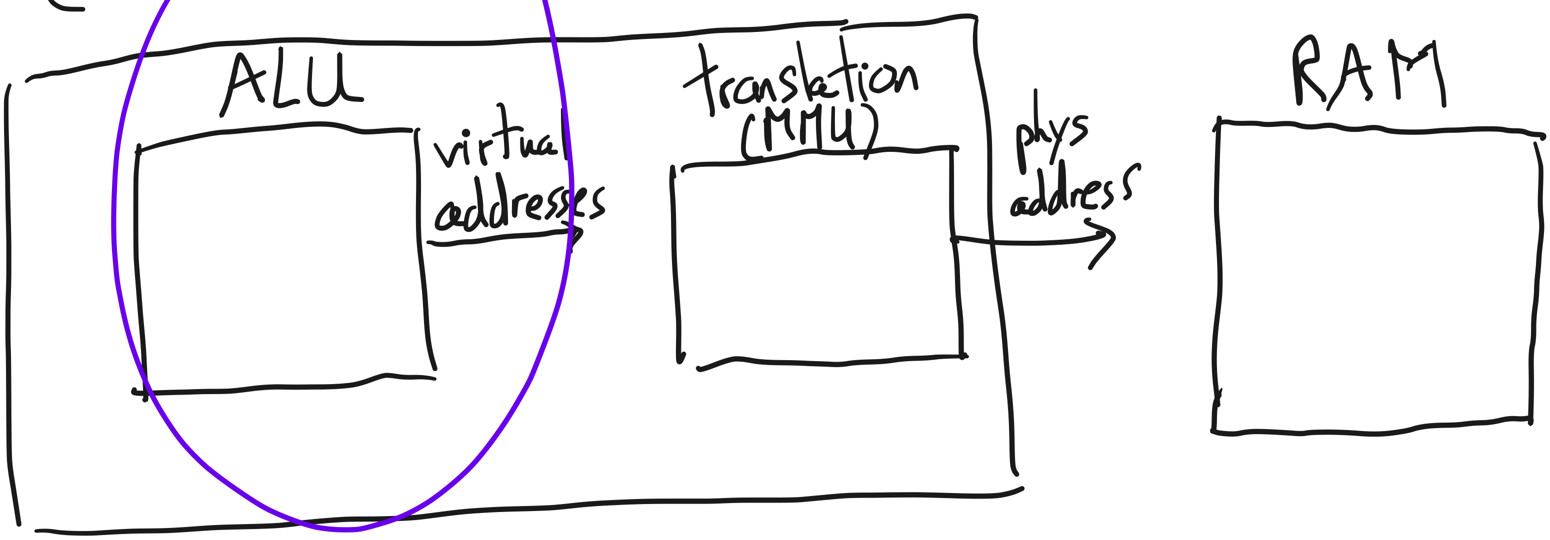
code address	instruction
0x500	movq 0x200000, %rax
0x508	incq 1, %rax
0x510	movq %rax, 0x300000



$y = x + 1$

CPU





Benefits of virtual memory:

- programmability
  - (a) program <u>thinks</u> it has lots of memory
  - (b) programs can use "easy" addresses: compiler and linker don't have to worry about where program lives in physical memory
  - (c) multiple instances of a program can be loaded and not collide

- protection
    - processes cannot read/write each other's memory
    - enables isolation (which is essential)

- effective use of resources

shmget()
mmap()

- sharing

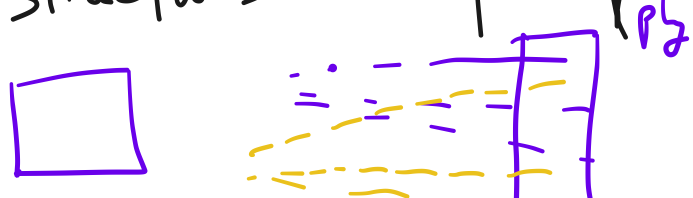
- - - - -

How is translation implemented?

- hardware does it, in MMU

OS sets up data structures that the hardware "sees".

These data structures are per-process.



3. Paging

A. Intro

- Divide memory (virtual + physical) into fixed-size chunks
- These chunks are called PAGES
- PAGE SIZE

- x86-64: 4096 B = 4 KB = 2^{12} bytes

8 bits = 1 byte

Aside:

2^{10}: kilo, ~1000 (1024)

2^{20}: mega, ~1 million (1024 × 1024)

2^{30}: giga, ~1 billion

2^{40}: tera, ~1 trillion

2^{50}: peta, ~1 quadrillion

2 : peta, quadrillion

How many pages are there on a 32-bit architecture?

addr

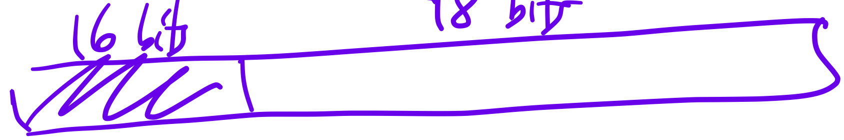
4 bytes

2^{32} bytes

2^{32} bytes / (2^{12} bytes / page)

$= 2^{20}$ pages

What if 48 bits are used to address memory?



2^{48} bytes / (2^{12} bytes / pages) $= 2^{36}$ pages

$36 = 30 + 6$

Page 0: [0, 4095]

Page 1: [4096, 8191]

VPN PPN

Page $2^{20}-1$:

$[\ \ldots\ \ldots,\ 2^{32}-1\ ]$

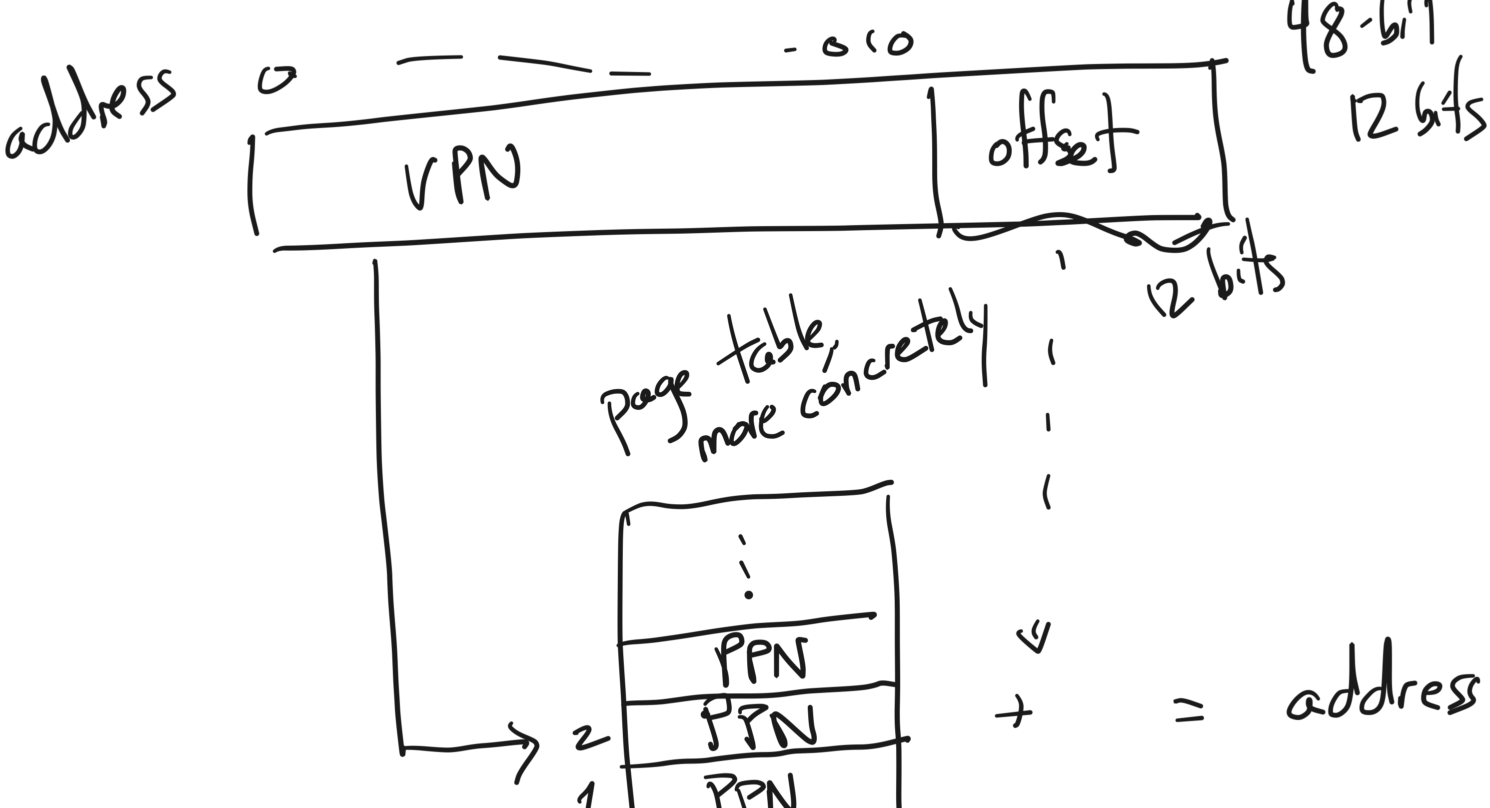
B. Key data structure: page table (per-process)

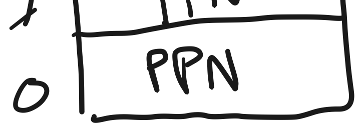
conceptually: a map from

VPN → PPN (PFN)

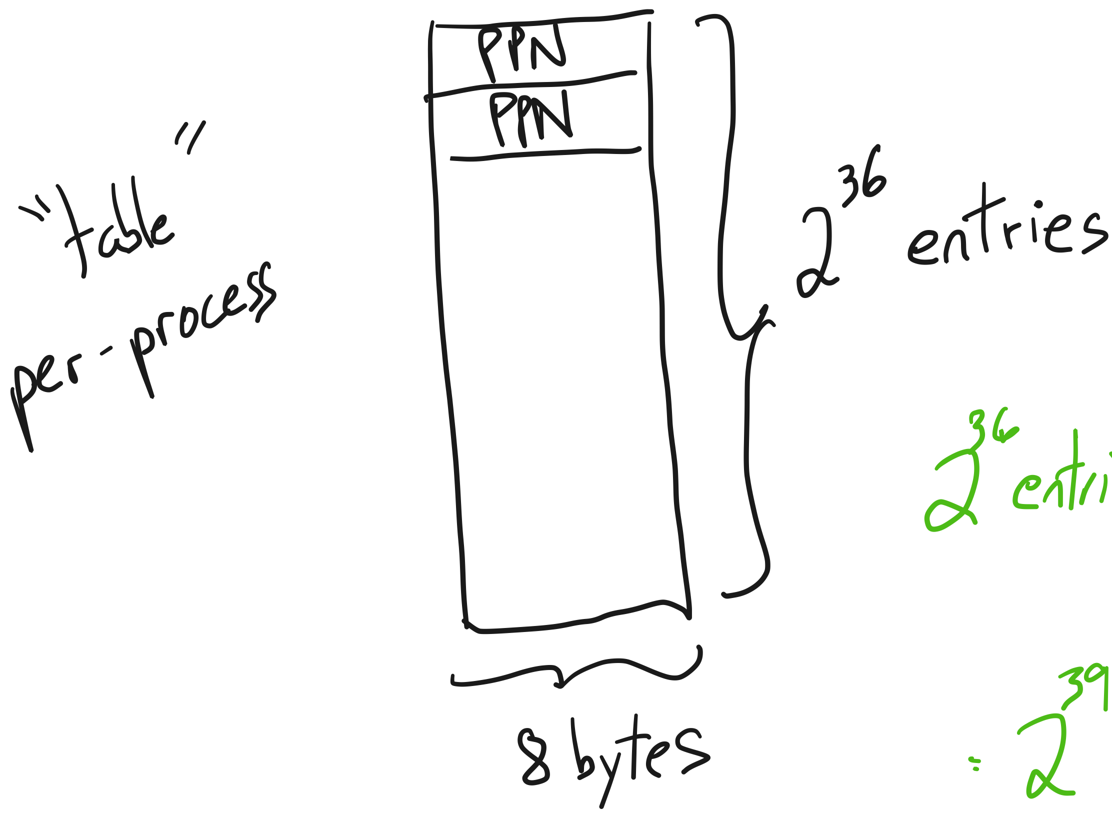
32-bit: 2^{20}, 12 bits

48-bit: 12 bits





assume: 48-bit addresses, and 4KB pages (2^{12} bytes)



2^{36} entries $\times$ 2^{3} $\frac{\text{bytes}}{\text{entries}}$

$= 2^{39}$ bytes $=$ 512 GB

Ex: OS wants a process to use address

VA: 0x00402000 to refer to

PA: 0x00003000

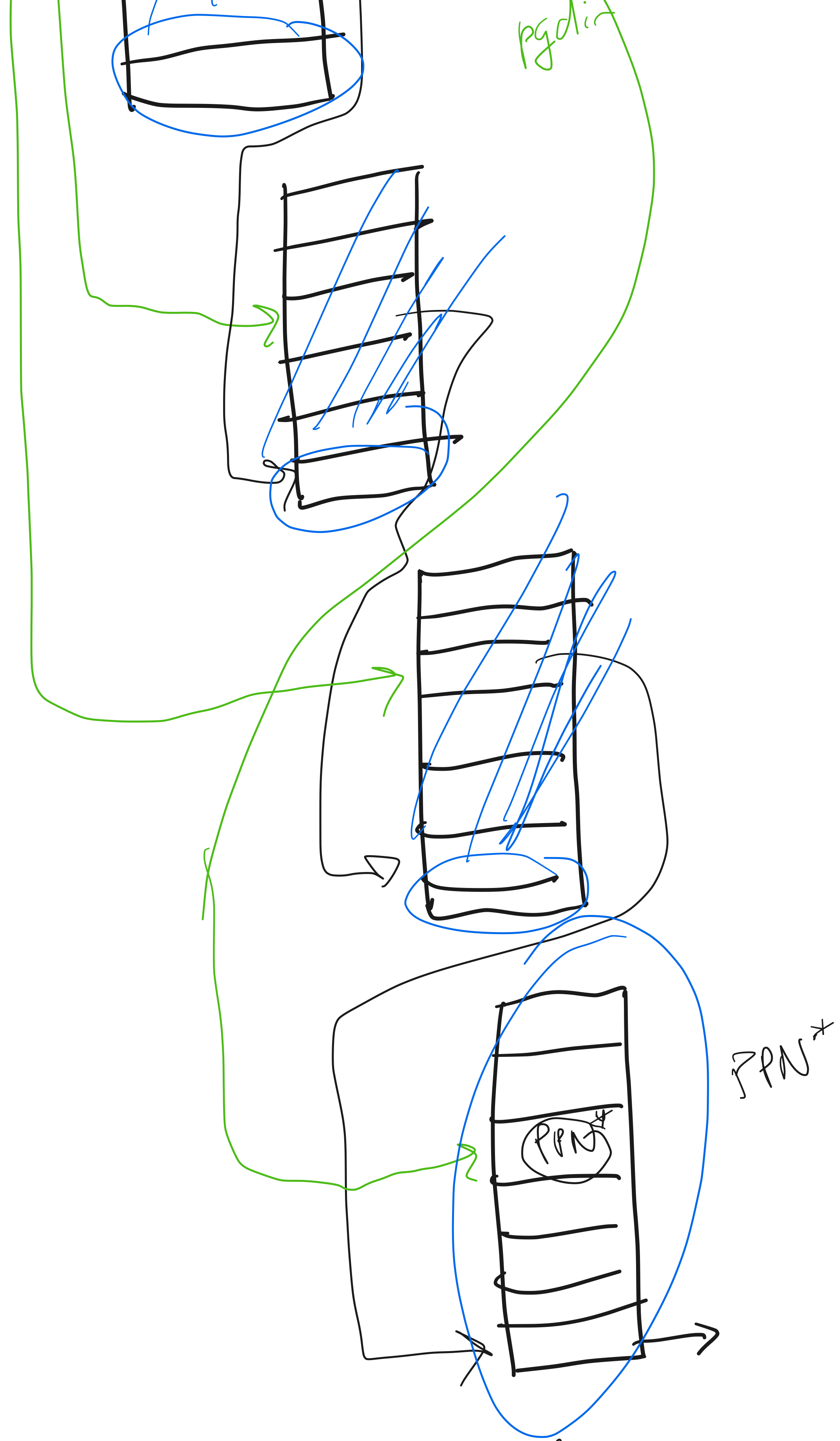
4 bits 4 bits 4 bits

table[0x00402] = 0x00003

1026

What's the issue?

C. Multilevel page tables



Ex: we want to map 2MB of physical memory

$[0, \ldots, 2^{21}-1]$ e $[0, 2^{21}]$ *

$$\left[100 * 2MB, \ldots, 101 * 2MB - 1\right]$$

in virtual space