```
Week 11.a
CS6640
11/14 2023
https://naizhengtan.github.io/23fall/
```
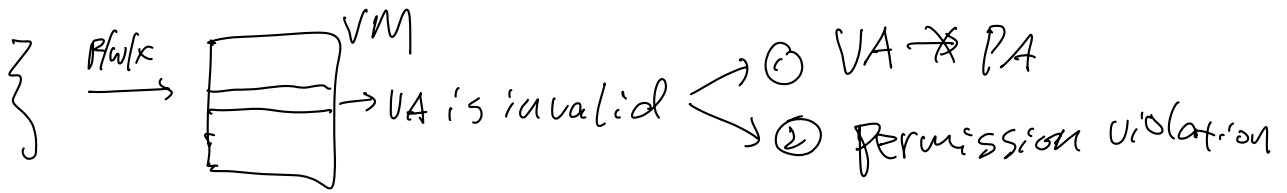
```
1. page fault intro
2. on-demand allocation
3. classic use cases
---
```

1. PF intro

- on-demand allocation
- COW   fork()
- memory mapped file

- PF is a type of exception.

$$\Big\{ \xrightarrow{R/W/x} \boxed{}^{VM} \rightarrow UA \text{ is invalid} \Big\langle \begin{array}{l} ① \; VA \not\rightarrow PA \\ ② \; \text{Permission violation} \end{array}$$

- VM:
  - ① translation: $VA \rightarrow PA$ $\nearrow PTE$
  - ② protection: isolation + AC
  - ③ a level-of-indirection : $\Rightarrow$ PF

> "Any problem in computer science can be solved with
> another level of indirection."
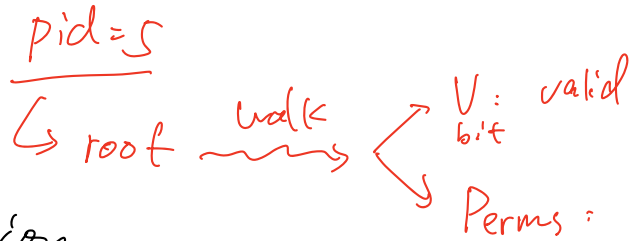>   -- Butler Lampson and David J. Wheeler

- RISC-V:

  Q: User or Kernel?

  $\Big\{ \begin{array}{l} ① \; 3 \text{ of } cb \rightarrow PF \; (mcause) \\ ② \; mtval \rightarrow faulted \; UA. \\ ③ \; mepc \rightarrow instr \; triggered \; the \; page \; fault \end{array}$

  - user-level app is running
  - MPP → $\{ \begin{array}{l} 1 \rightarrow kernel \\ 0 \rightarrow user \end{array}$
    - SP
    - mepc

Q: VA $\neq$ PA or Perm Violation

• Walk PT

pid=5
$\hookrightarrow$ root $\xrightarrow{walk}$ $\begin{cases} V: \text{valid} \\ \text{bit} \\ \text{Perms}: \end{cases}$

2. on-demand allocation

$0x82657C4$


```
      } size
malloc } heep
fork
```
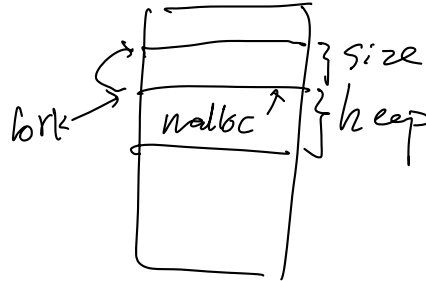
3. classic use cases

   * copy-on-write fork

   * overcommitting memory

* memory-mapped files

* pgfaults for user applications

RDMA

* DSM: distributed shared memory

DSM

M1   M2   M$$d

evict

1TB   1TB   1TB

100TB