

# CS 537

## Project 4

### Midterm 2

A-K : Von Vleck B102

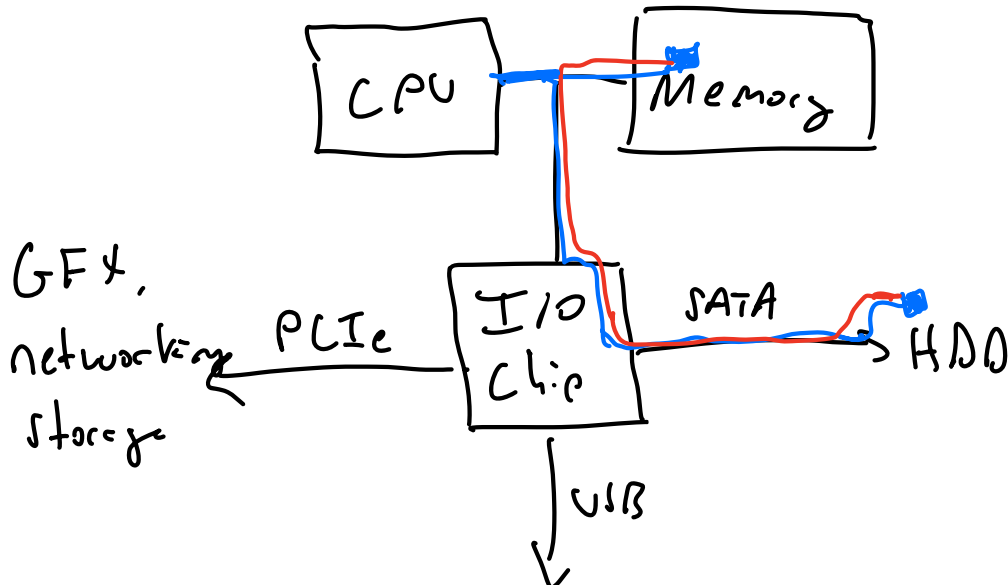
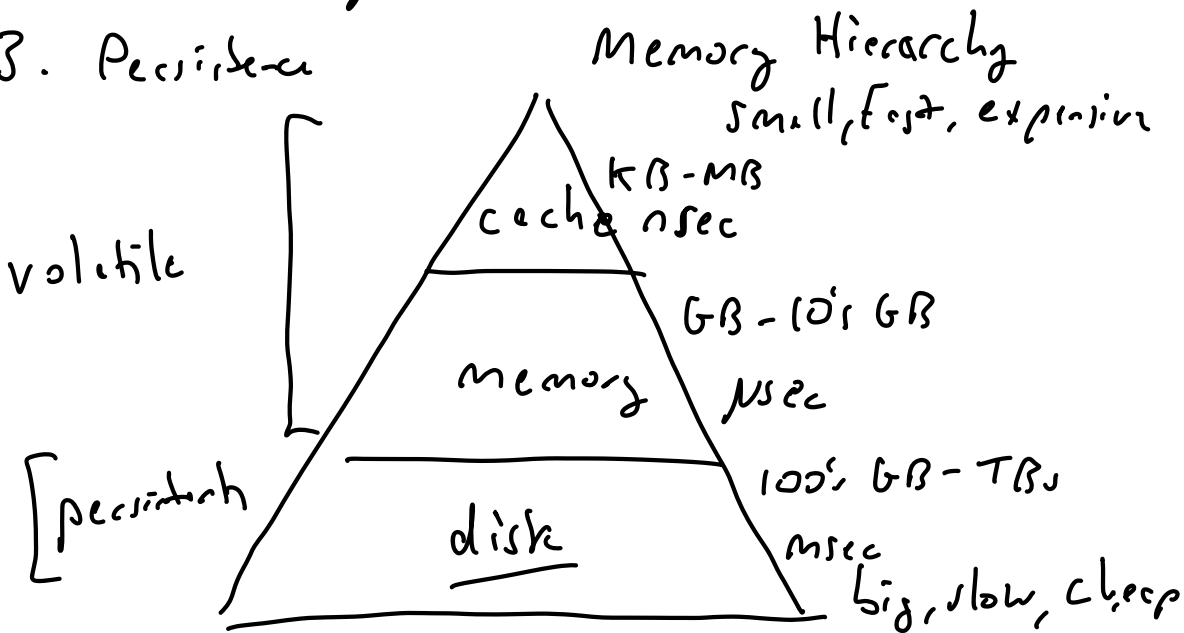
L-Z : Ingraham B10

McBurney 8:45-8:00pm CS 1257

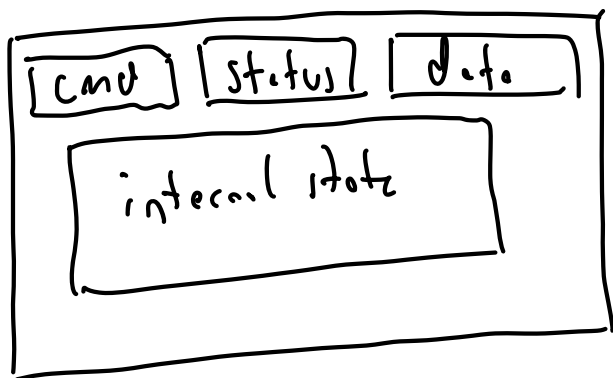
1. Virtualization  $\left\{ \begin{array}{l} \text{memory} \\ \text{CPU} \end{array} \right.$

2. Concurrency

3. Persistence



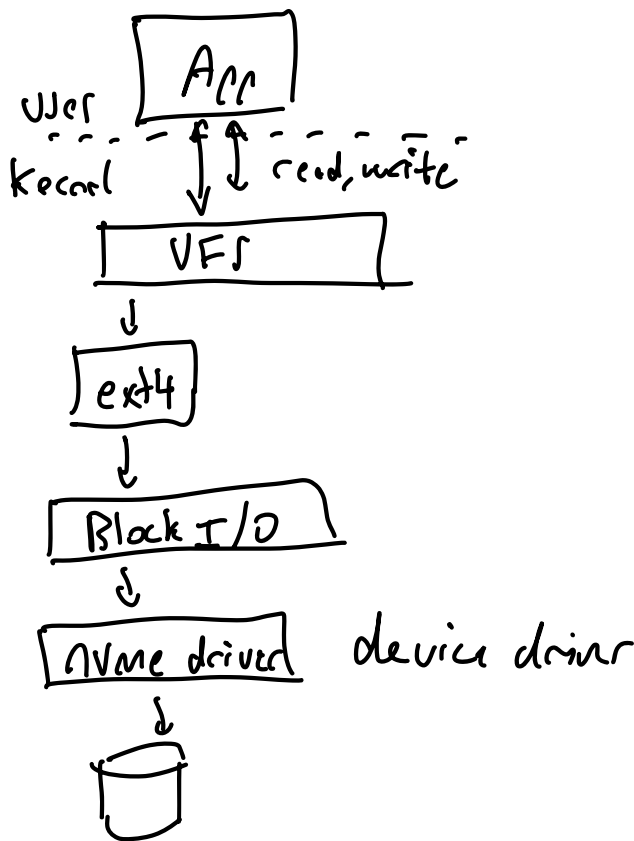
keyboard,  
mouse



Device interface  $\Leftrightarrow$  API

OS code to work with devices

device driver



1) Direct I/O commands  
(in, out)

2) Memory-mapped I/O

status = BUSY | READY

```
while (status == BUSY)
    ; // spin
```

```
for (ptr : data)
```

```
    memcpy (data, ptr)
```

```
    write (cmd)
```

```
    while (status == BUSY)
        ; // spin
```

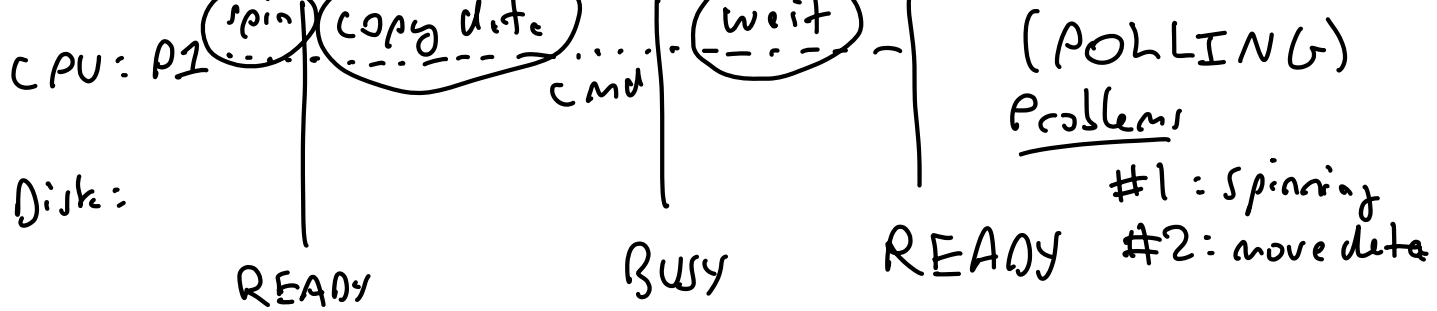
1) check if available

2) move data



3) issue command

4) wait for completion



#1 instead of spin → sleep

How? Hardware interrupt

while (busy)

```
sem_wait(&io) | cond_wait(&io, &mutex)
```

```
OS on interrupt: sem_post(&io) | cond_signal(&io, &mutex)
```

#2 move data. Direct Memory access (DMA)

