

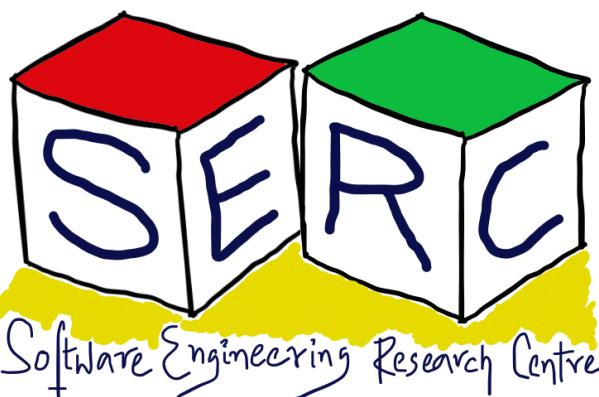
# CS3.301 Operating Systems and Networks

OS: Overall Run Down and Concluding Thoughts!

Karthik Vaidhyanathan

<https://karthikvaidhyanathan.com>

1



# Acknowledgement

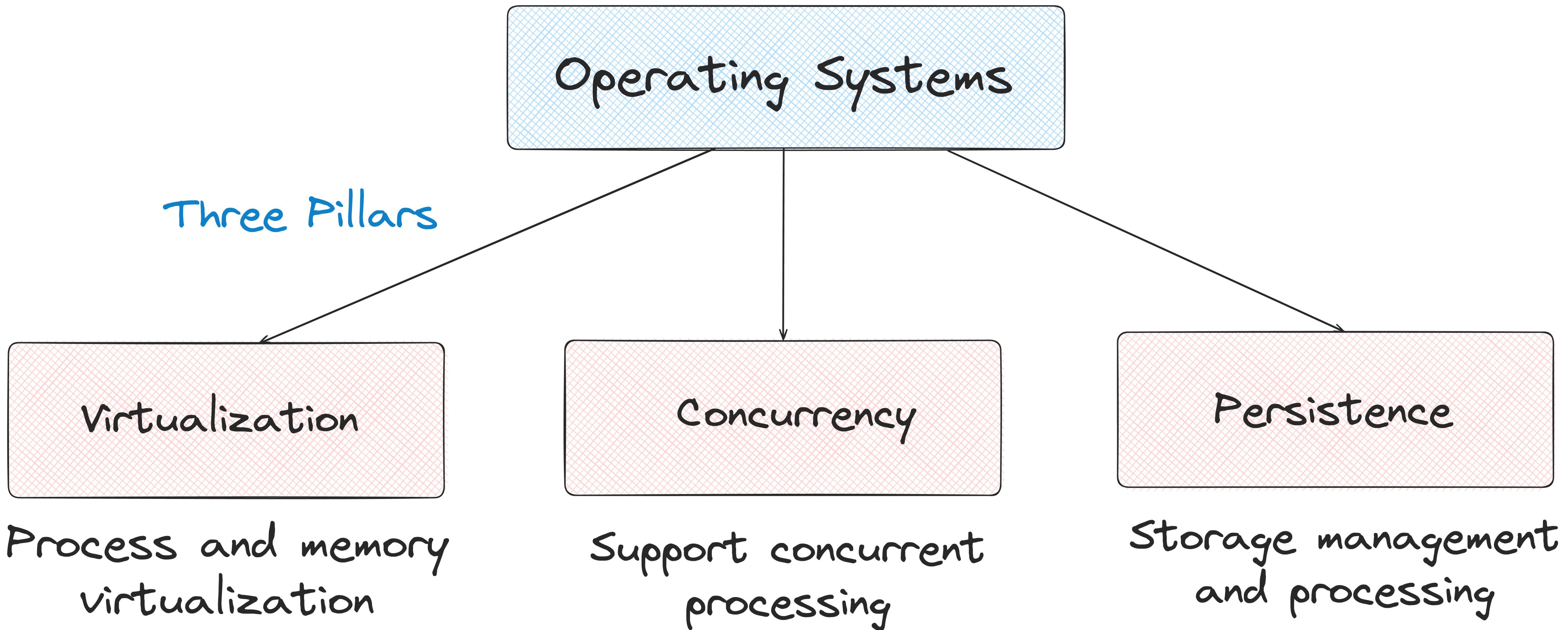
The materials used in this presentation have been gathered/adapted/generate from various sources as well as based on my own experiences and knowledge -- Karthik Vaidhyanathan

## Sources:

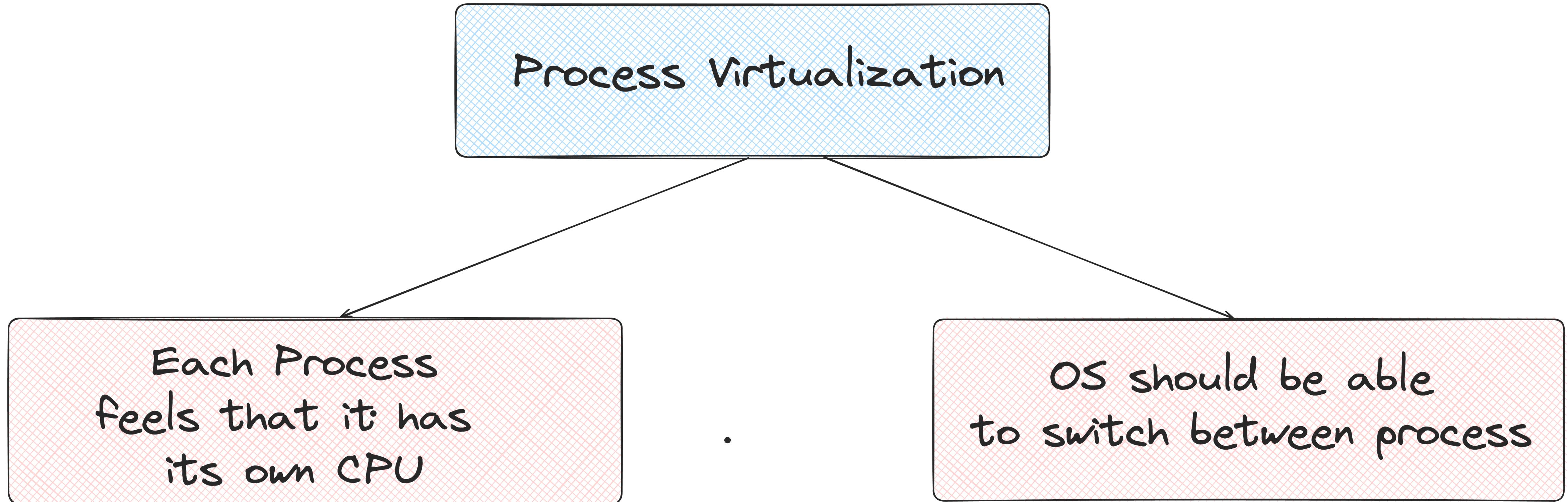
- Operating Systems in Three Easy Pieces by Remzi et al.
- Different materials used throughout the course



# OS: An Overview

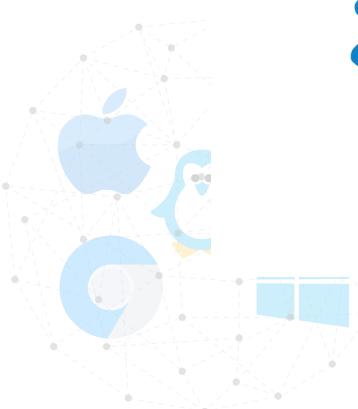


# Process Virtualization



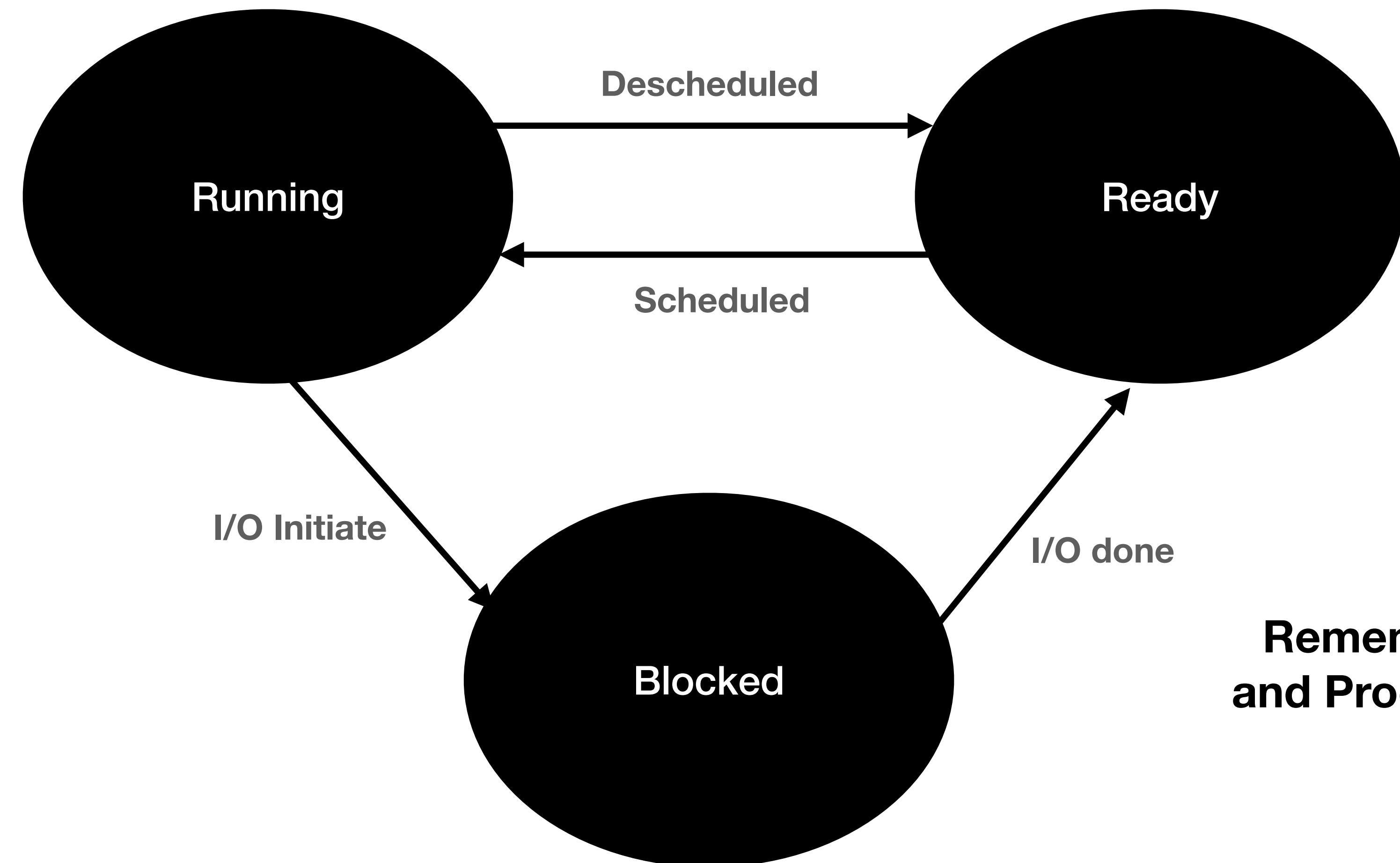
- 1. Process has different states
- 2. Process management API

- 1. Support for context switch
- 2. Process Scheduling



# States of the Process

## Process State Transitions

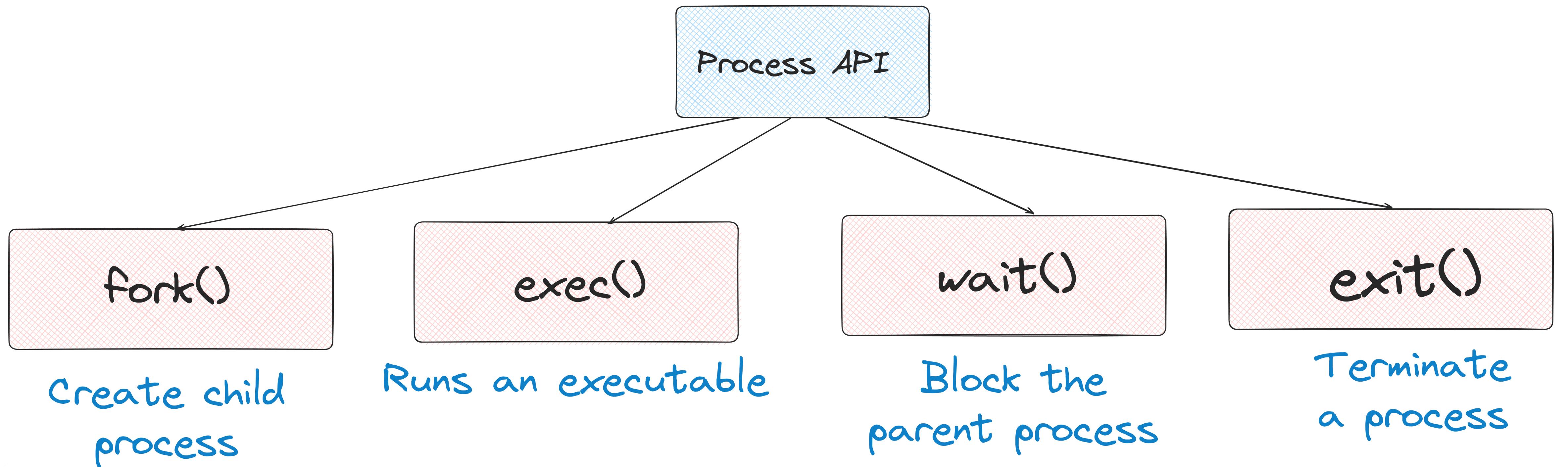


**Remember: Process lists  
and Process Control Blocks**

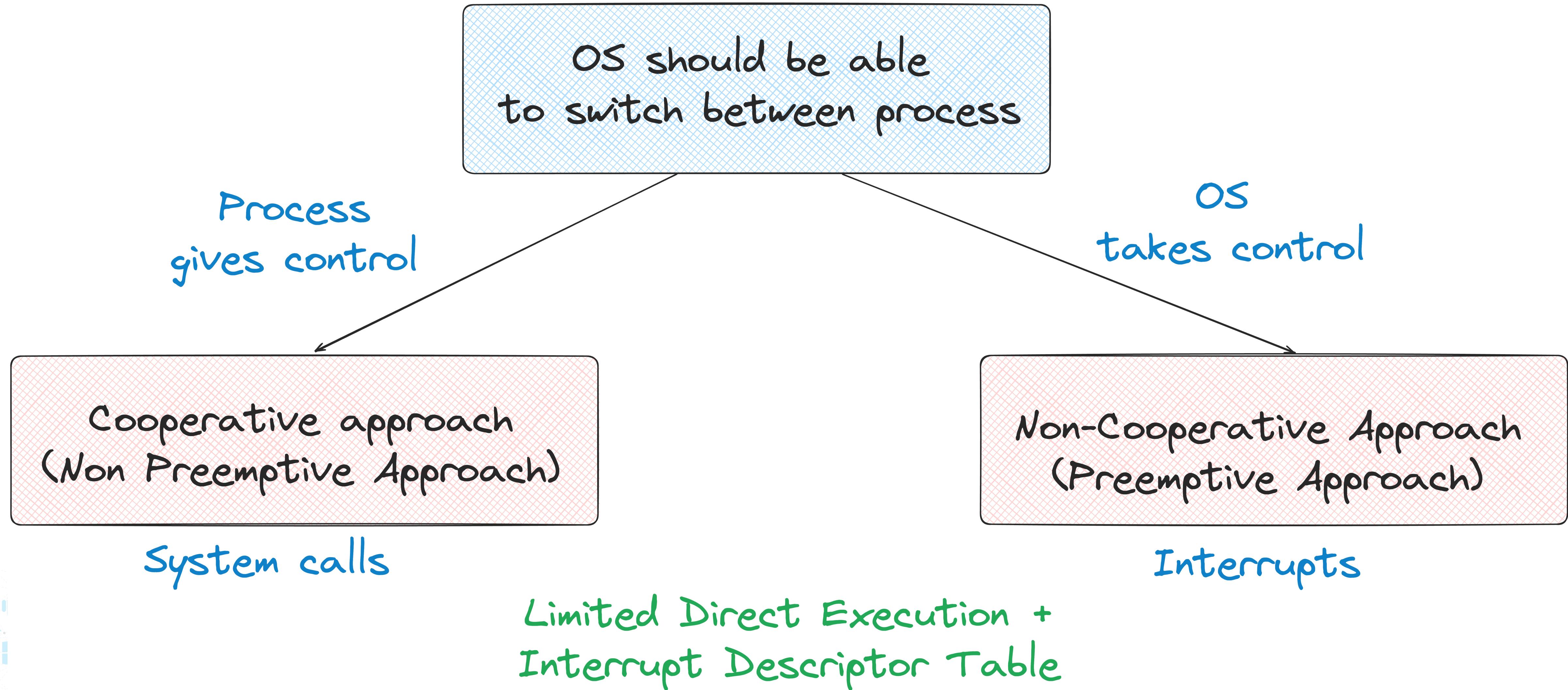


# Process Management API

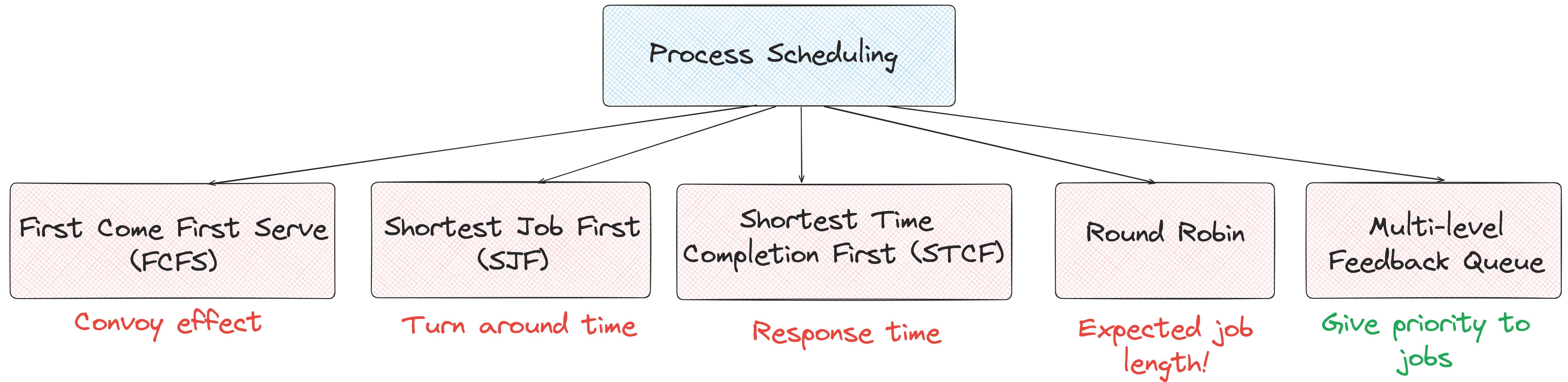
Memory image of a process - **Code, data, stack, and heap**



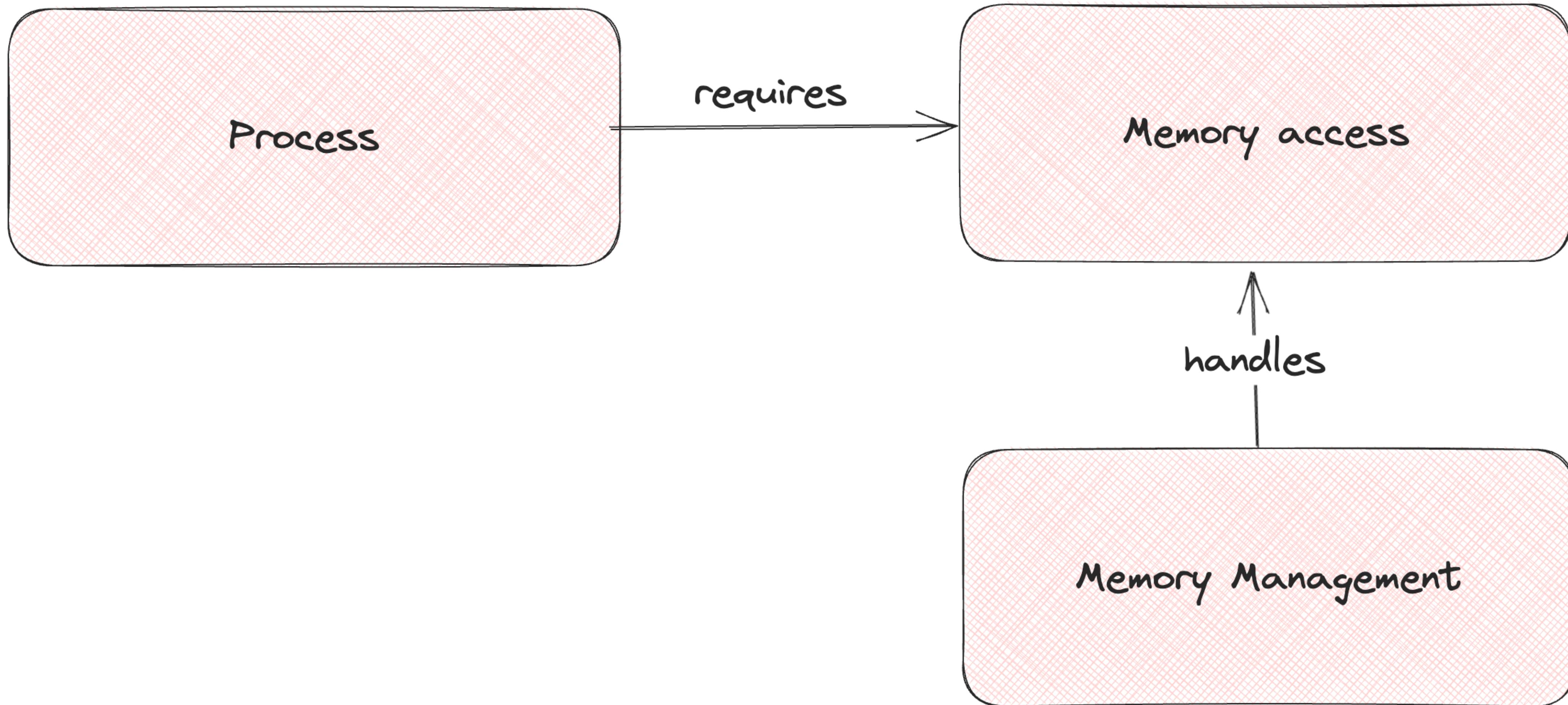
# Switching Between Process



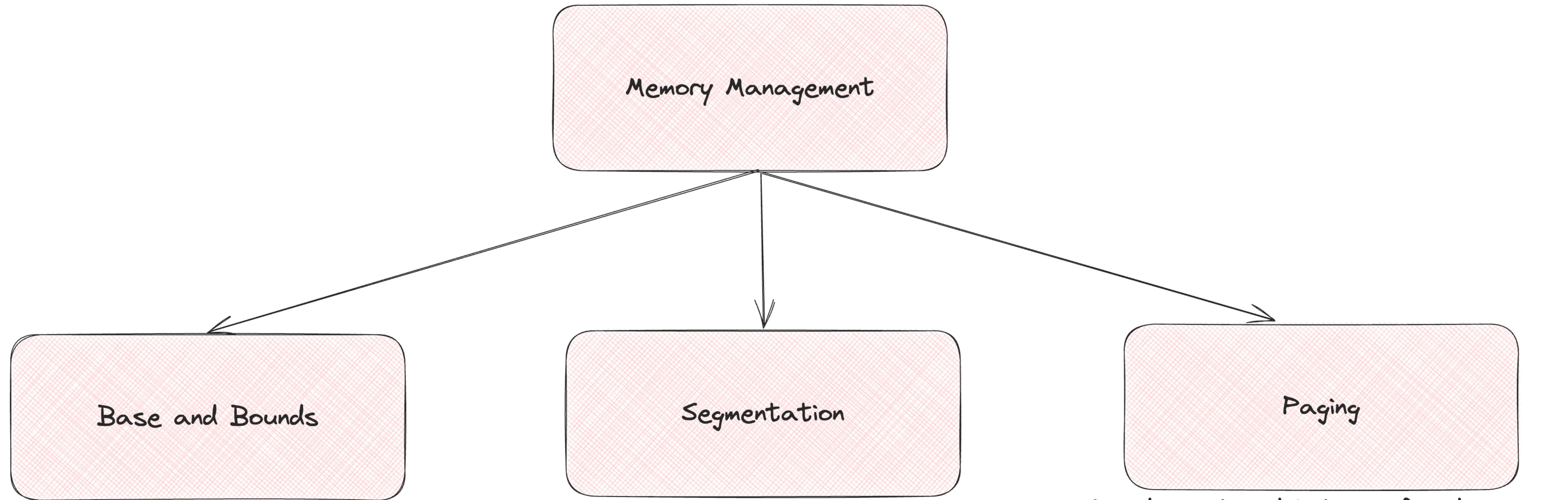
# Process Scheduling



# Memory Virtualization



# Memory Management



Base register  
and bounds register

(just add and check  
if less than bounds for  
translation)

Generalized Base and bounds

(for each segment there is a  
base and bounds that needs to  
be assigned)

Paging

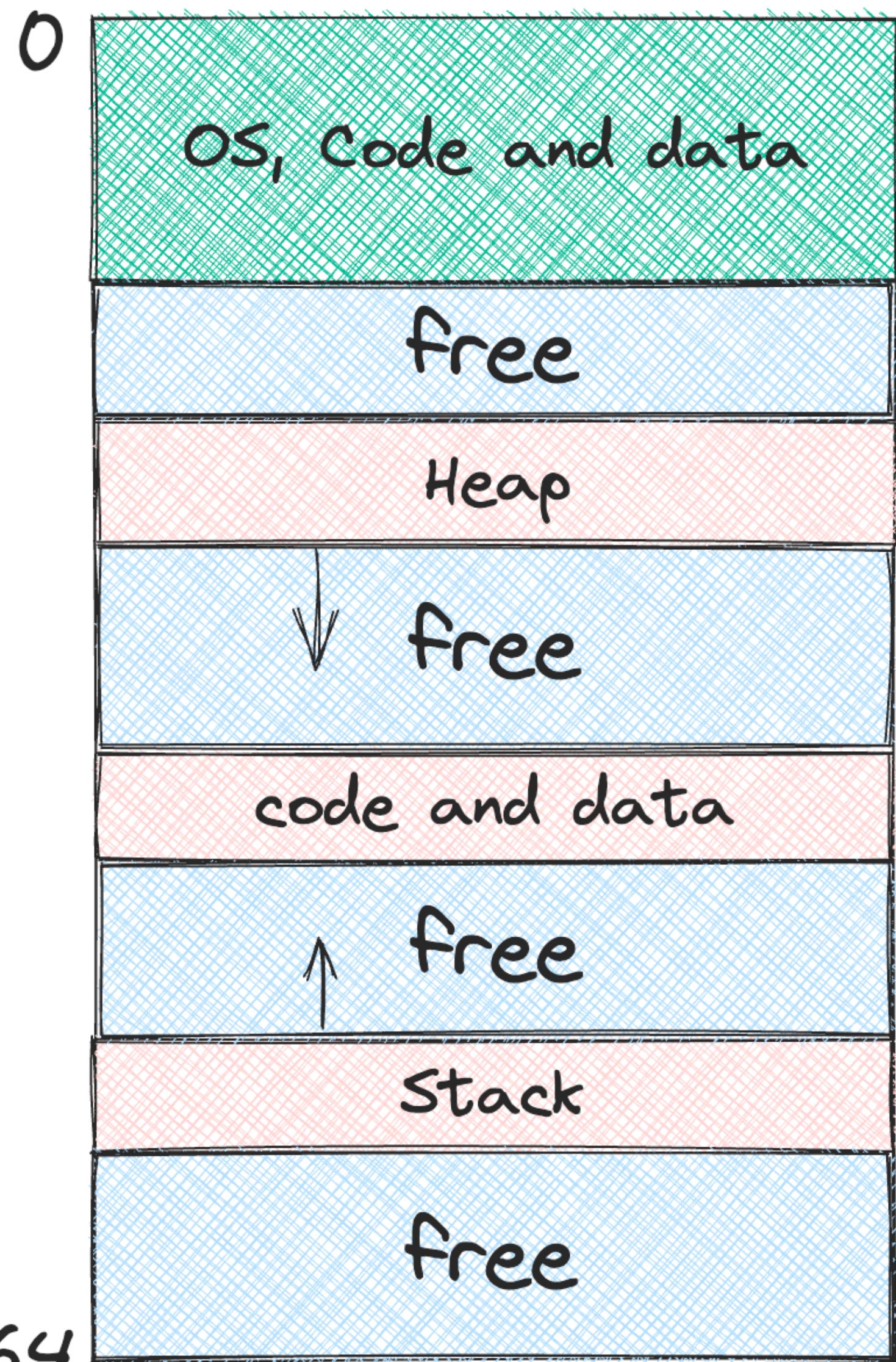
Dividing VA and PA into fixed chunks  
(Translation with the help of a page  
table)



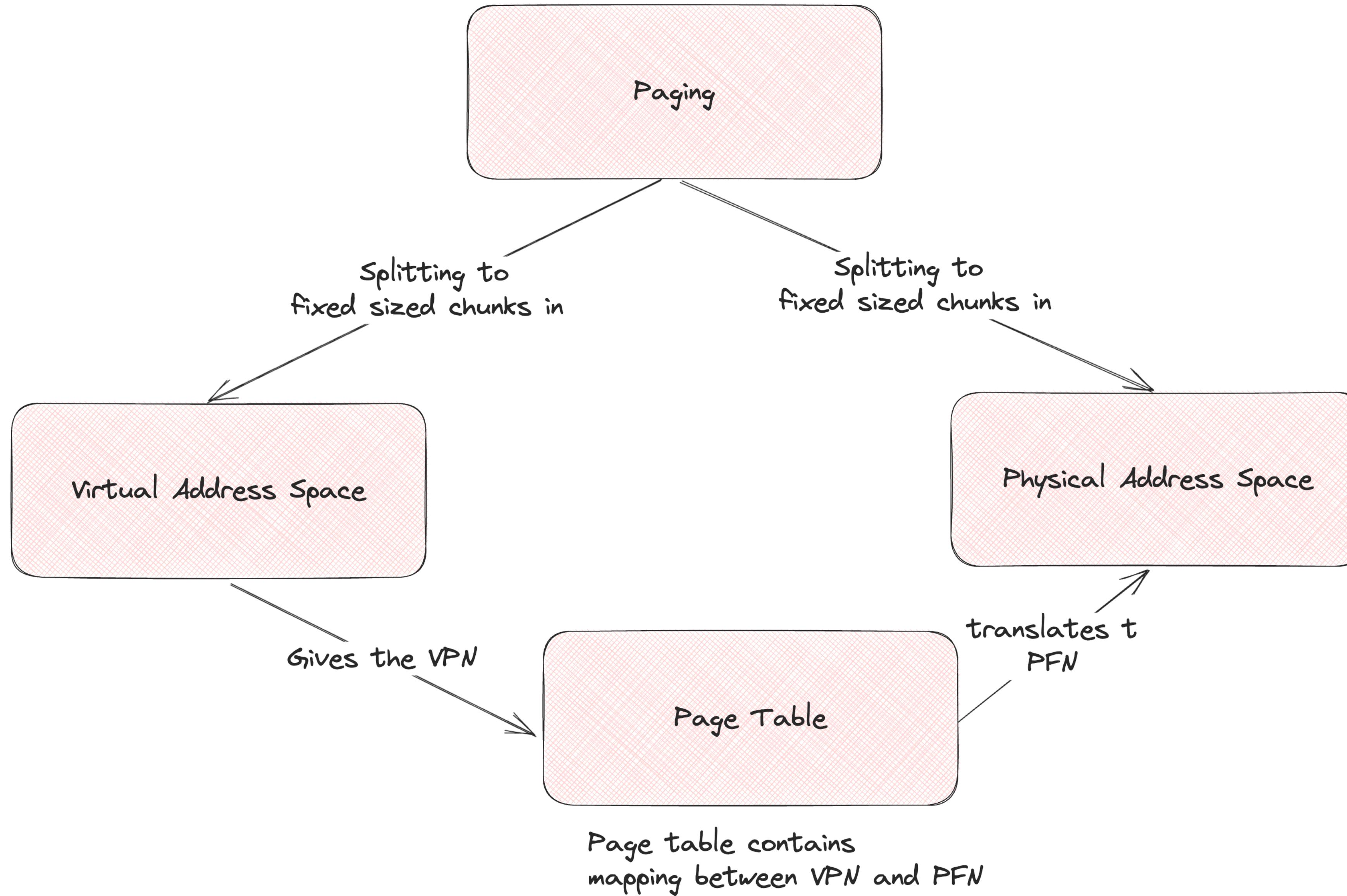
# Segmentation

## Generalized Base and Bounds

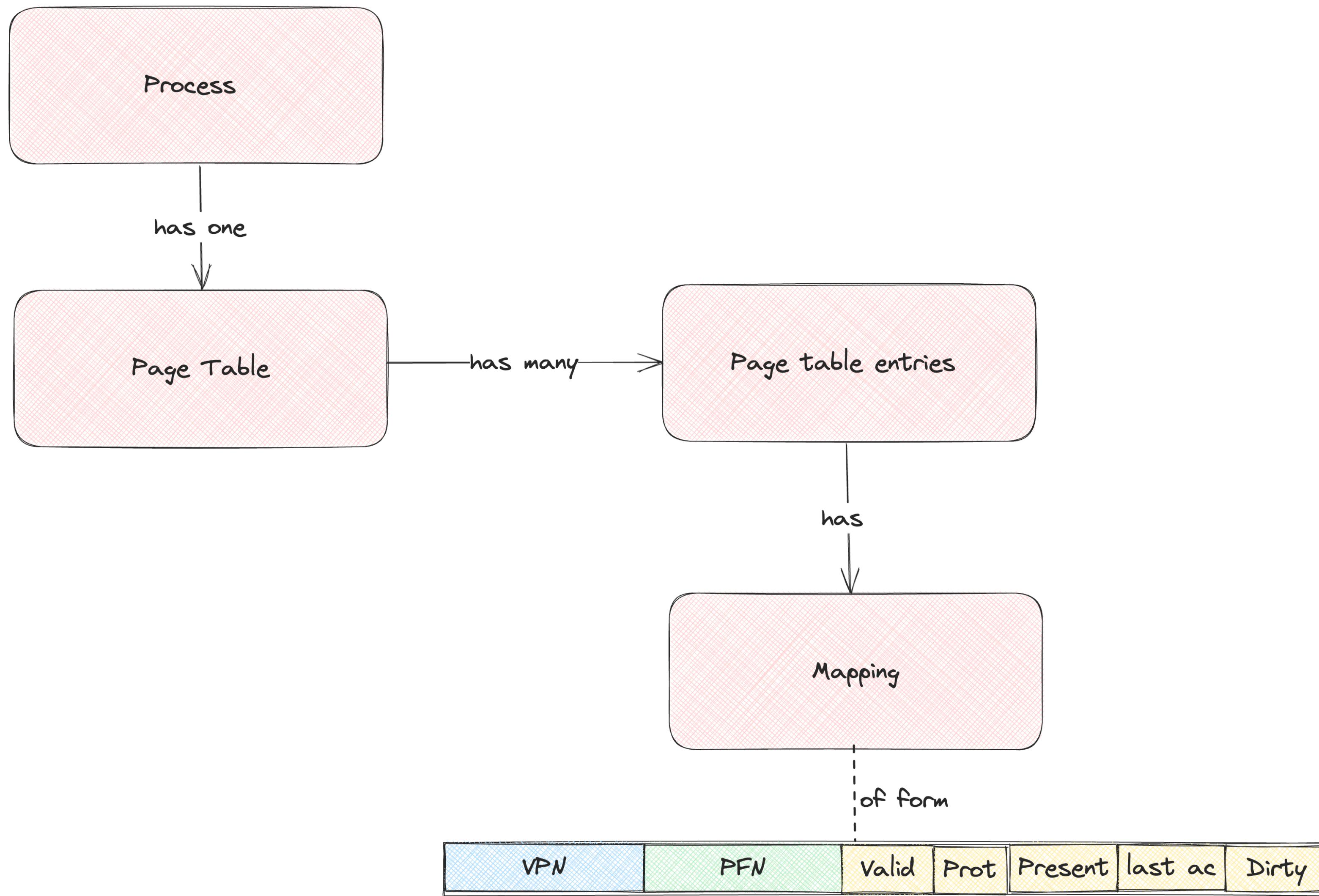
- Only used memory is allocated in physical memory
  - Allows allocating large address space
  - Sparse address space
- Different segments per process - code, stack, heap
- **For translation:** use first bits to identify segments and perform translation
- Results in **External fragmentation**



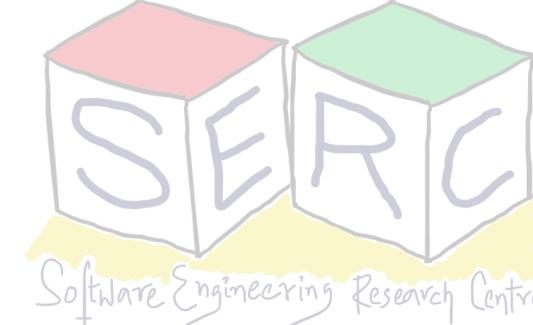
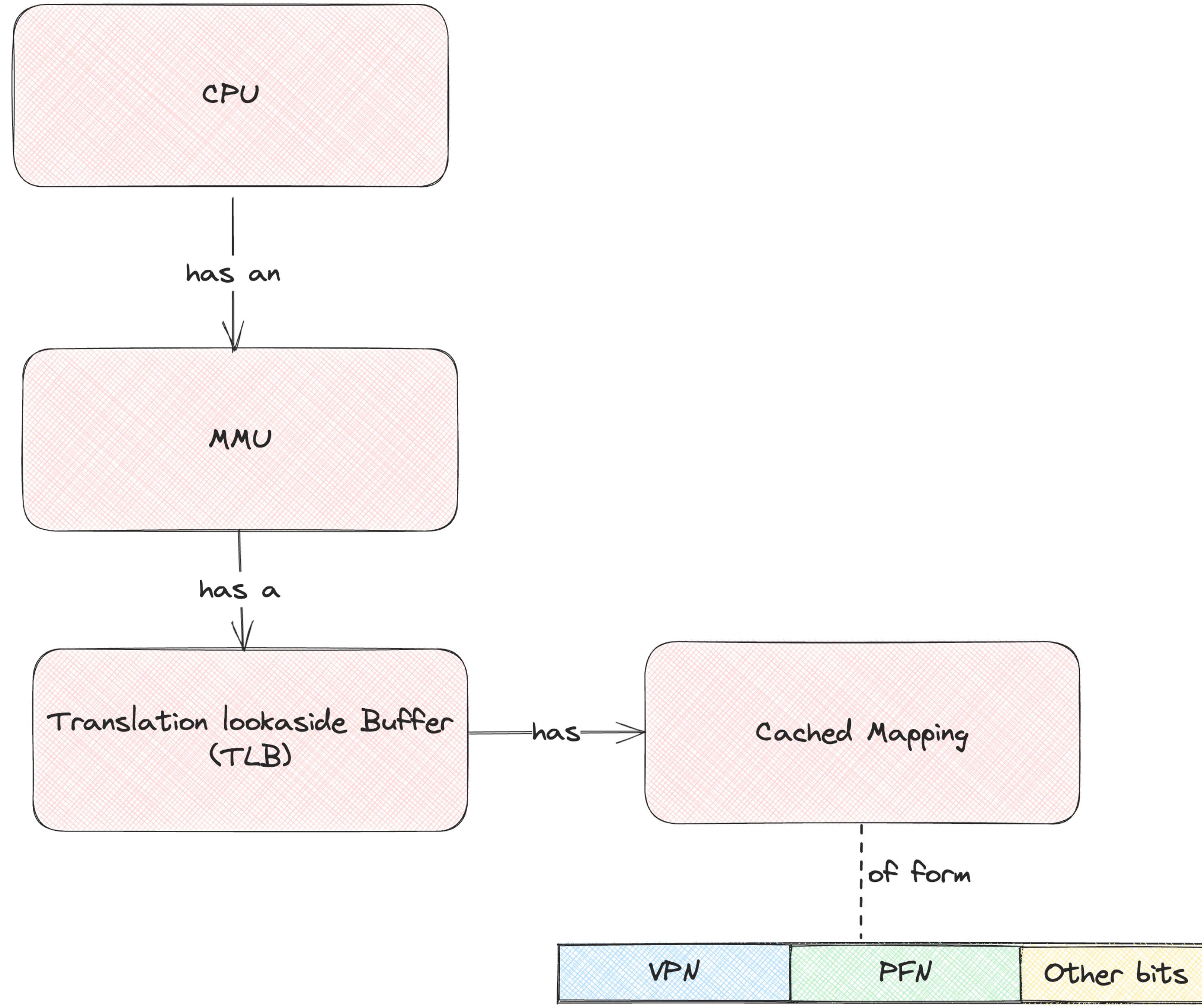
# Paging

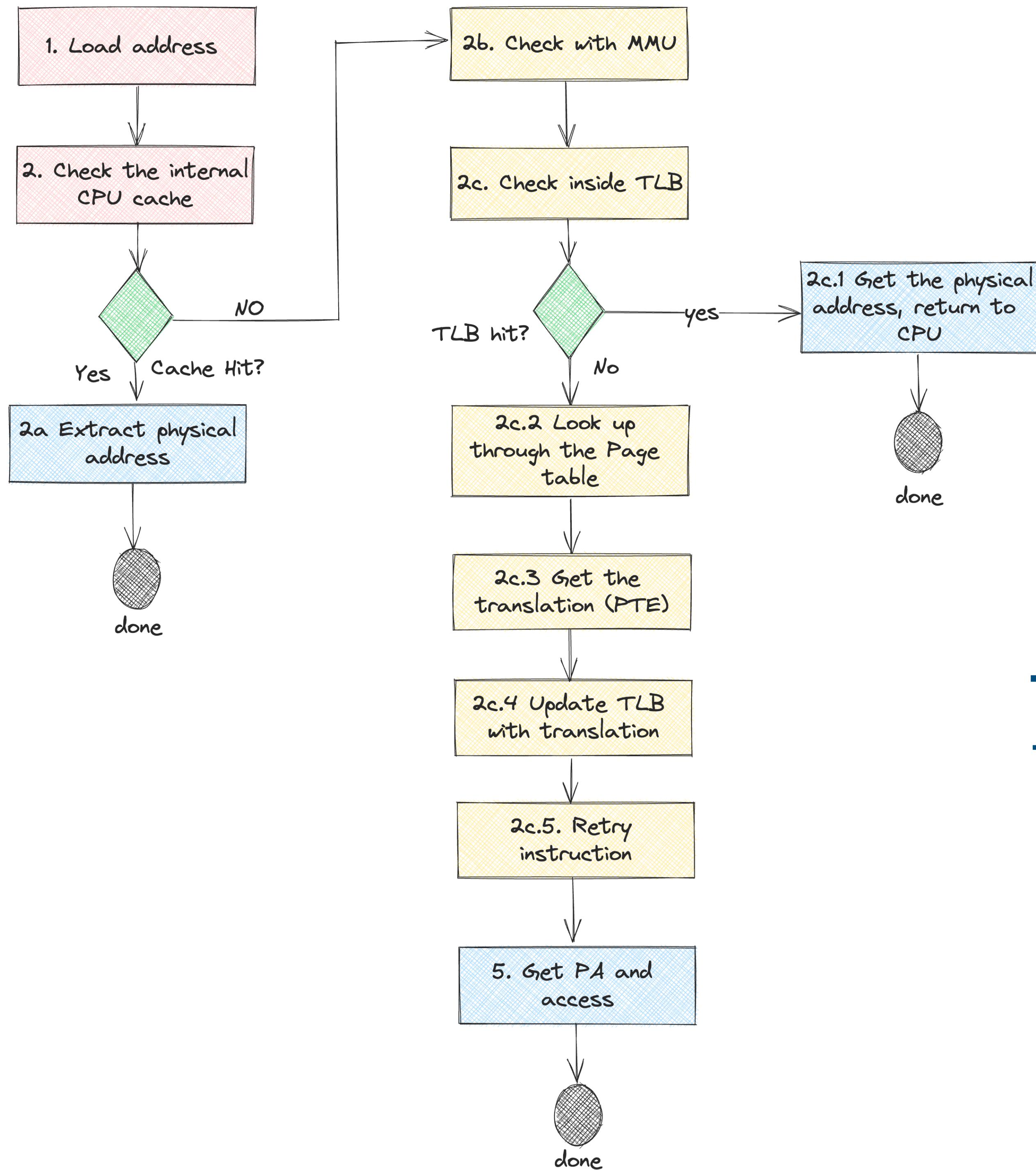


# Page Tables

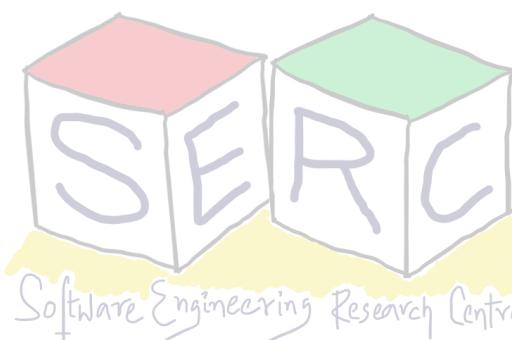


# Paging - TLB



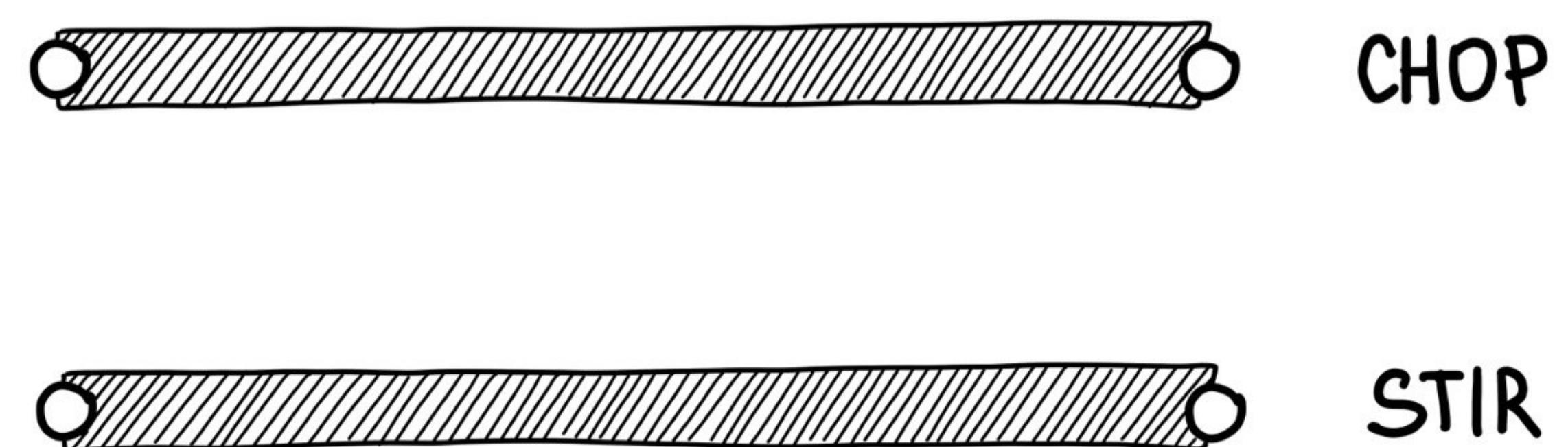
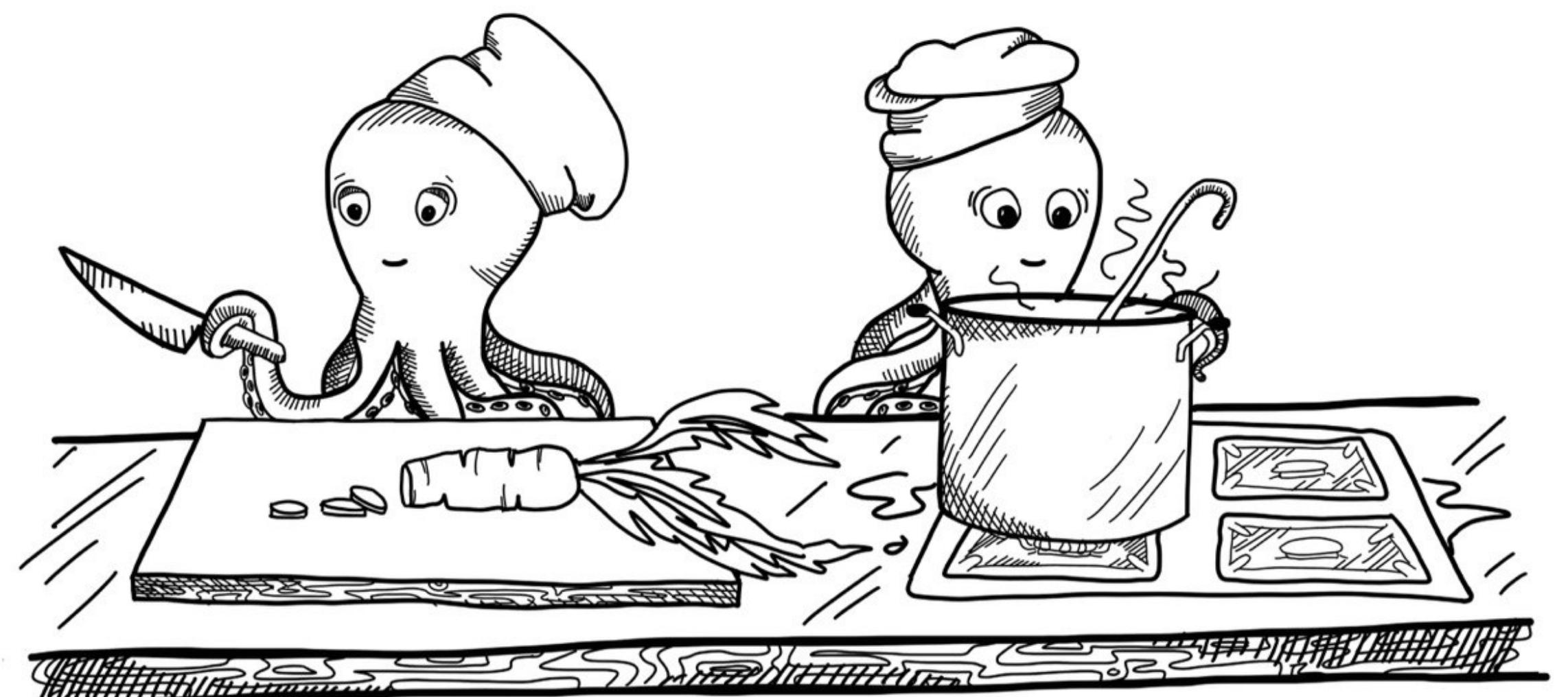
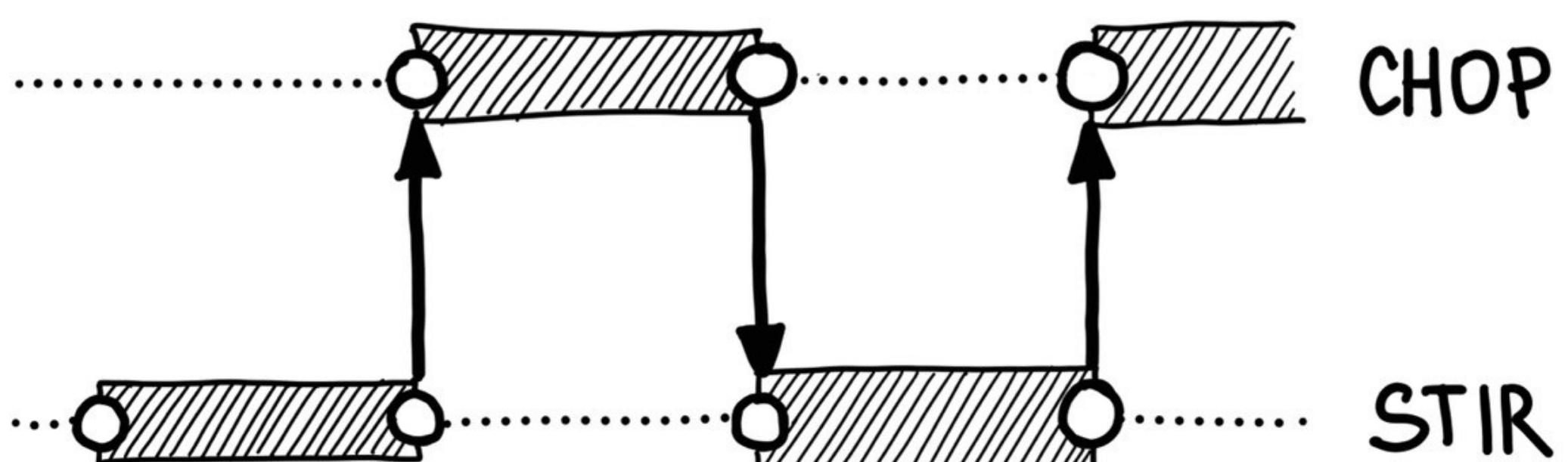
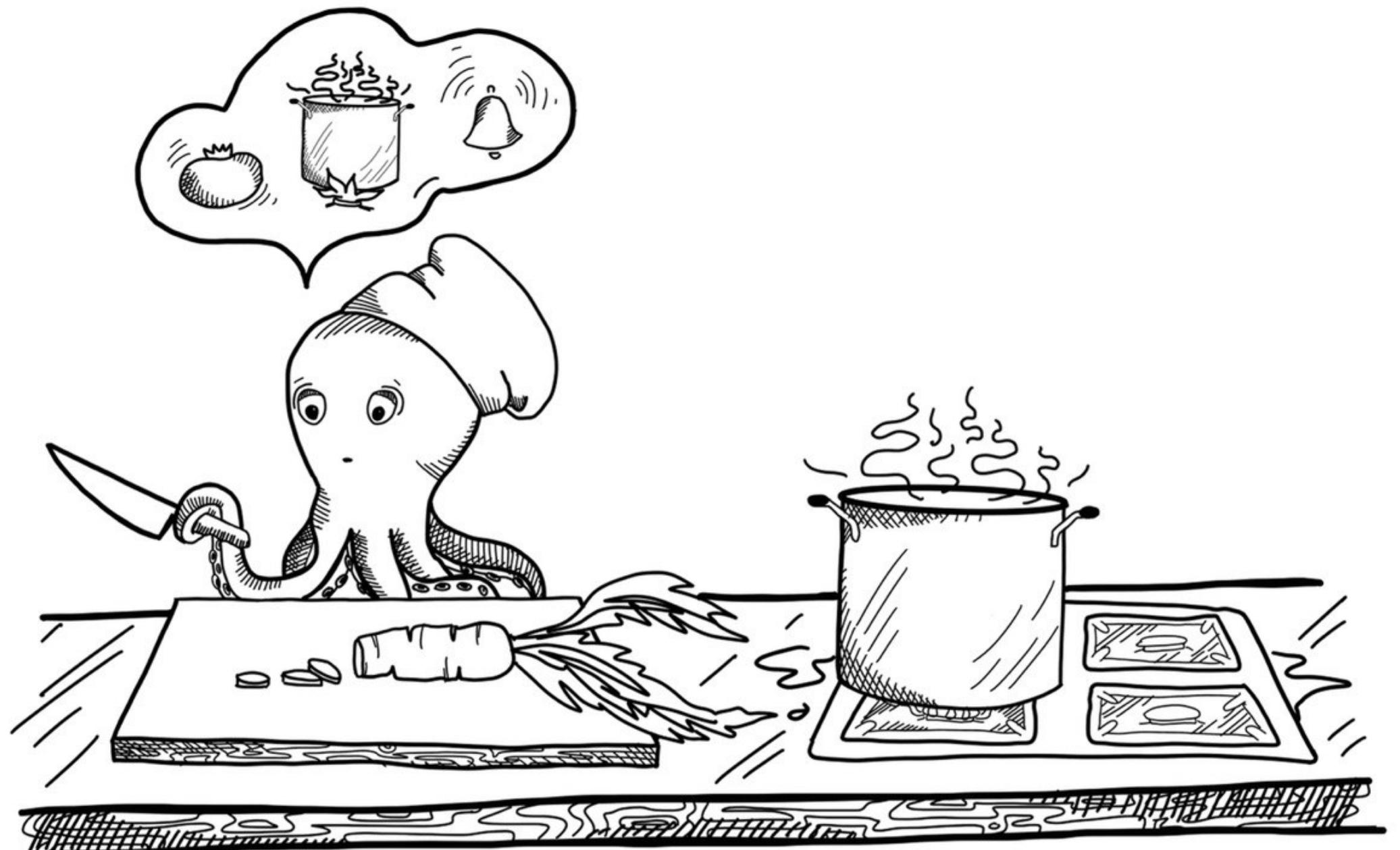


## The overall address translation Process

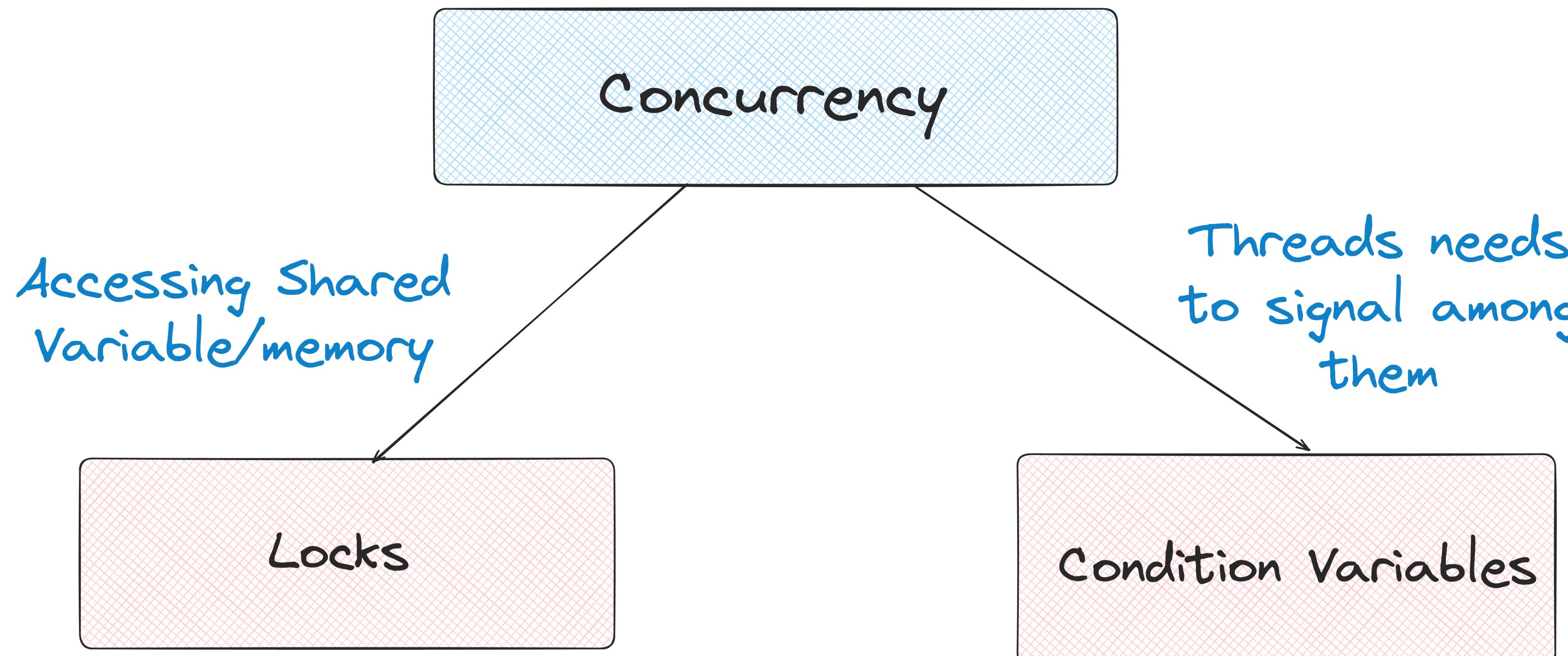


# Concurrency and Parallelism

What is what?



# Locks and Condition Variables



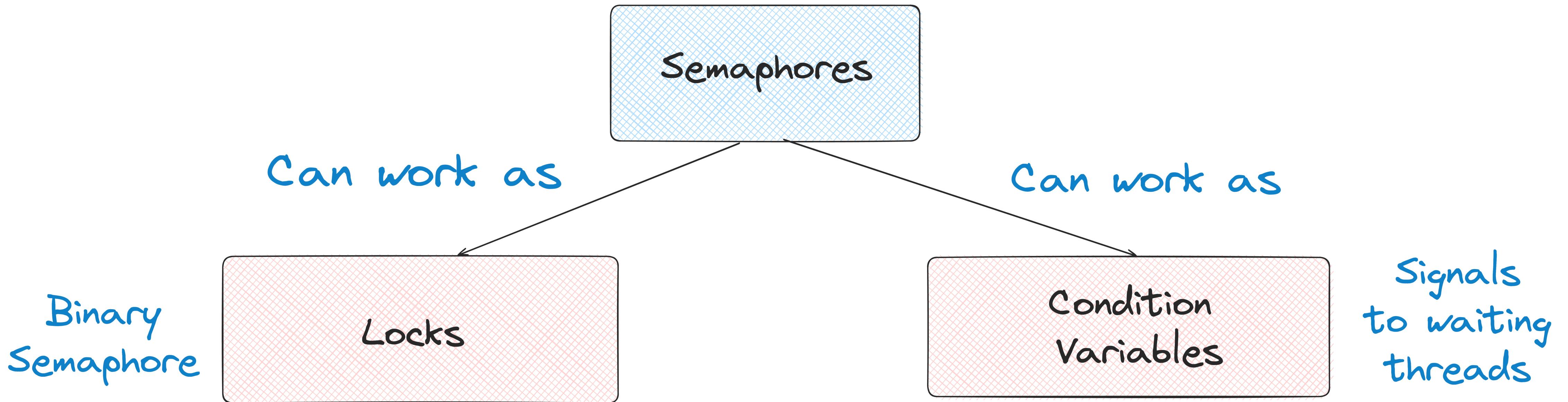
Hardware Primitives  
Software locks

Use Signal and Wait  
Adds threads to queues  
when waiting

Combination of Locks and Condition variables may be required  
to accomplish different tasks



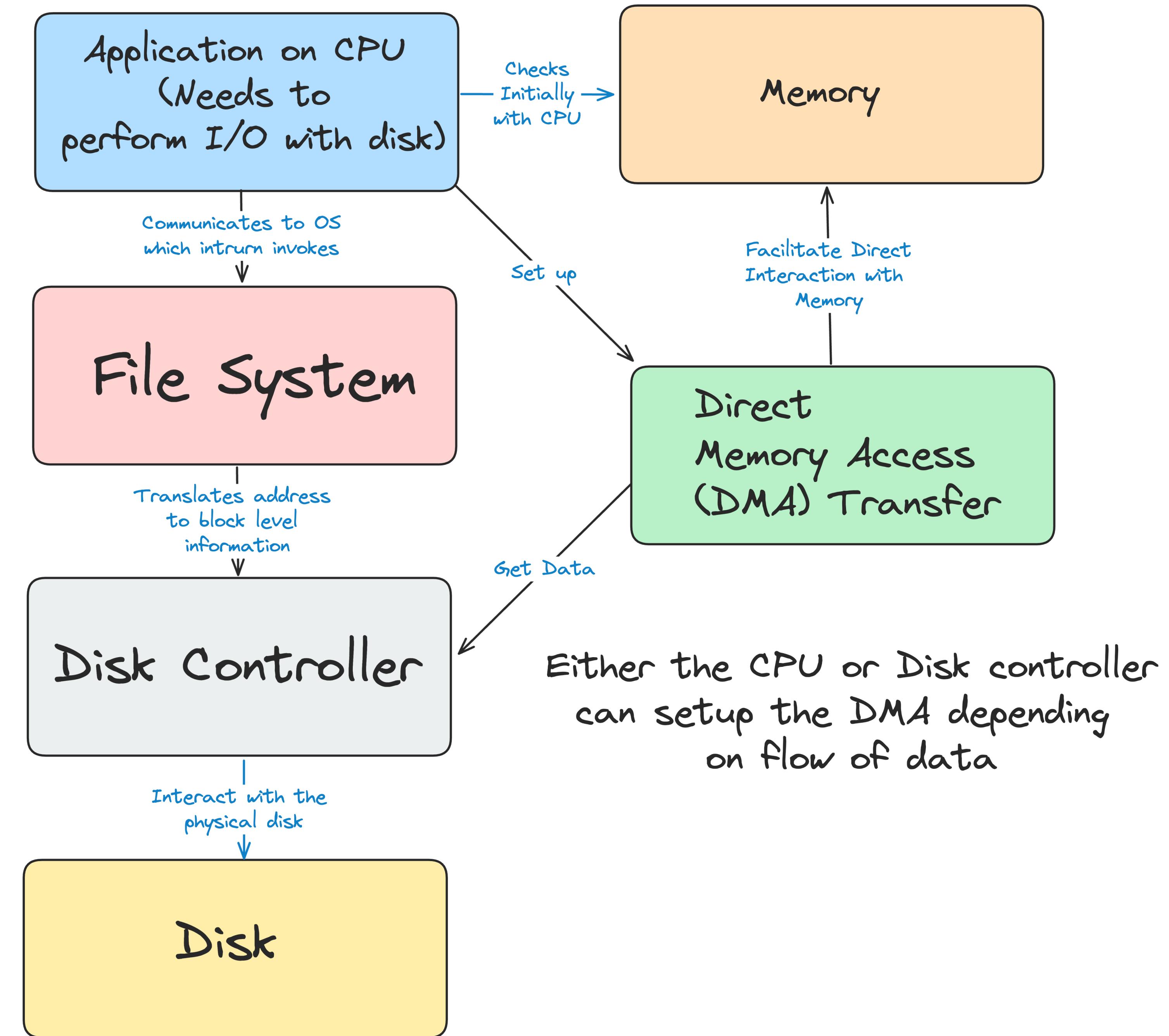
# Semaphores



Two key operations: wait () and post()  
Initialization of semaphore holds the key

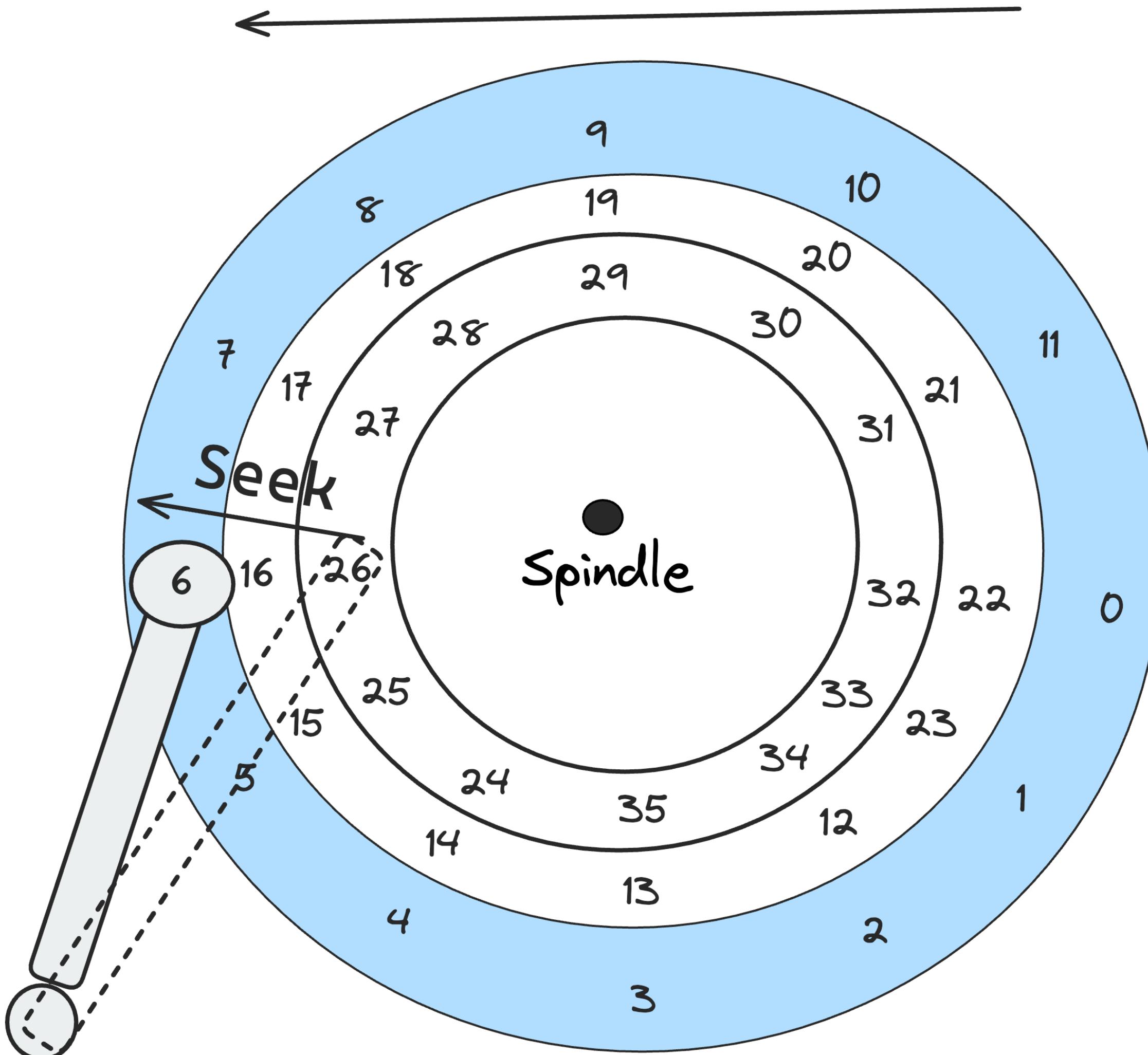


# Persistence



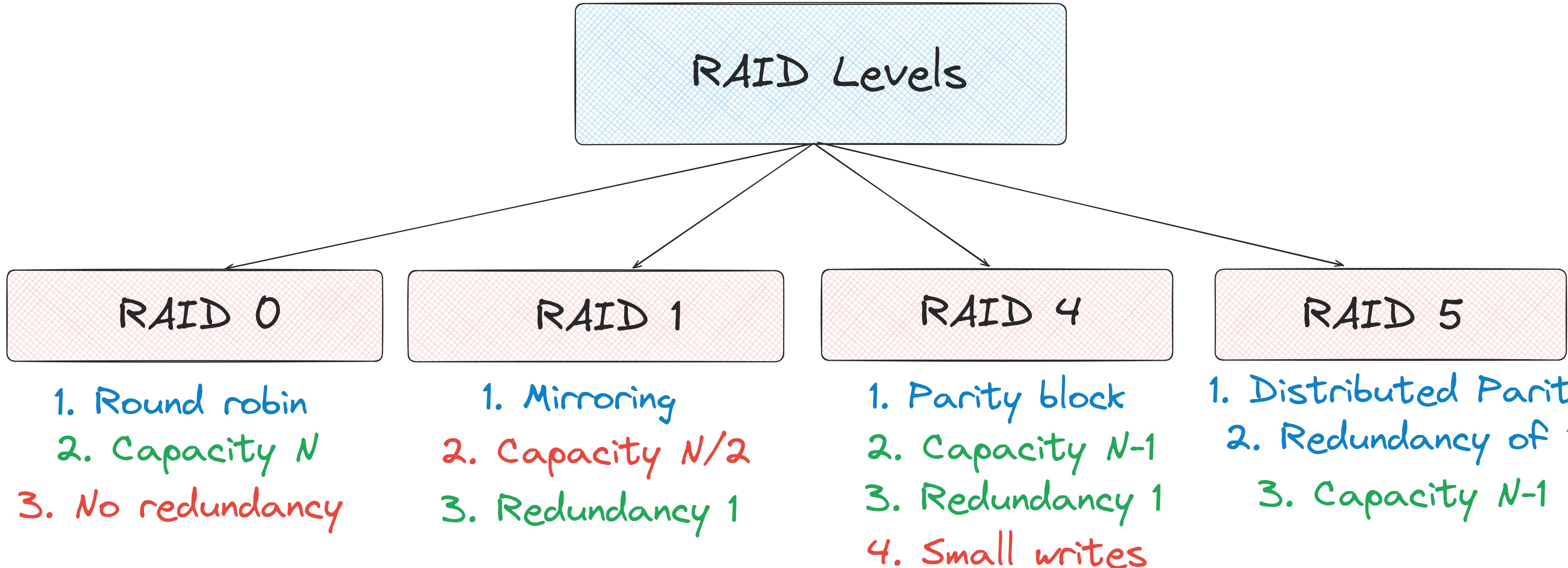
# Disks: An Overview

Rotates this way



- Disk rotates on a spindle
  - The arm can move across (seek) or stay as the disk rotates
  - The head is used to read/write
- Data is arranged in tracks as blocks/sectors
- There are 100s of tracks on a single disk
- **Seek, rotate and transfer** - three key phases

# RAIDs



# Breaking down into two main aspects

- We worked on building a **Very Simple File System (VSFS)**
- In any FS, two key things make the difference

## Data Structures

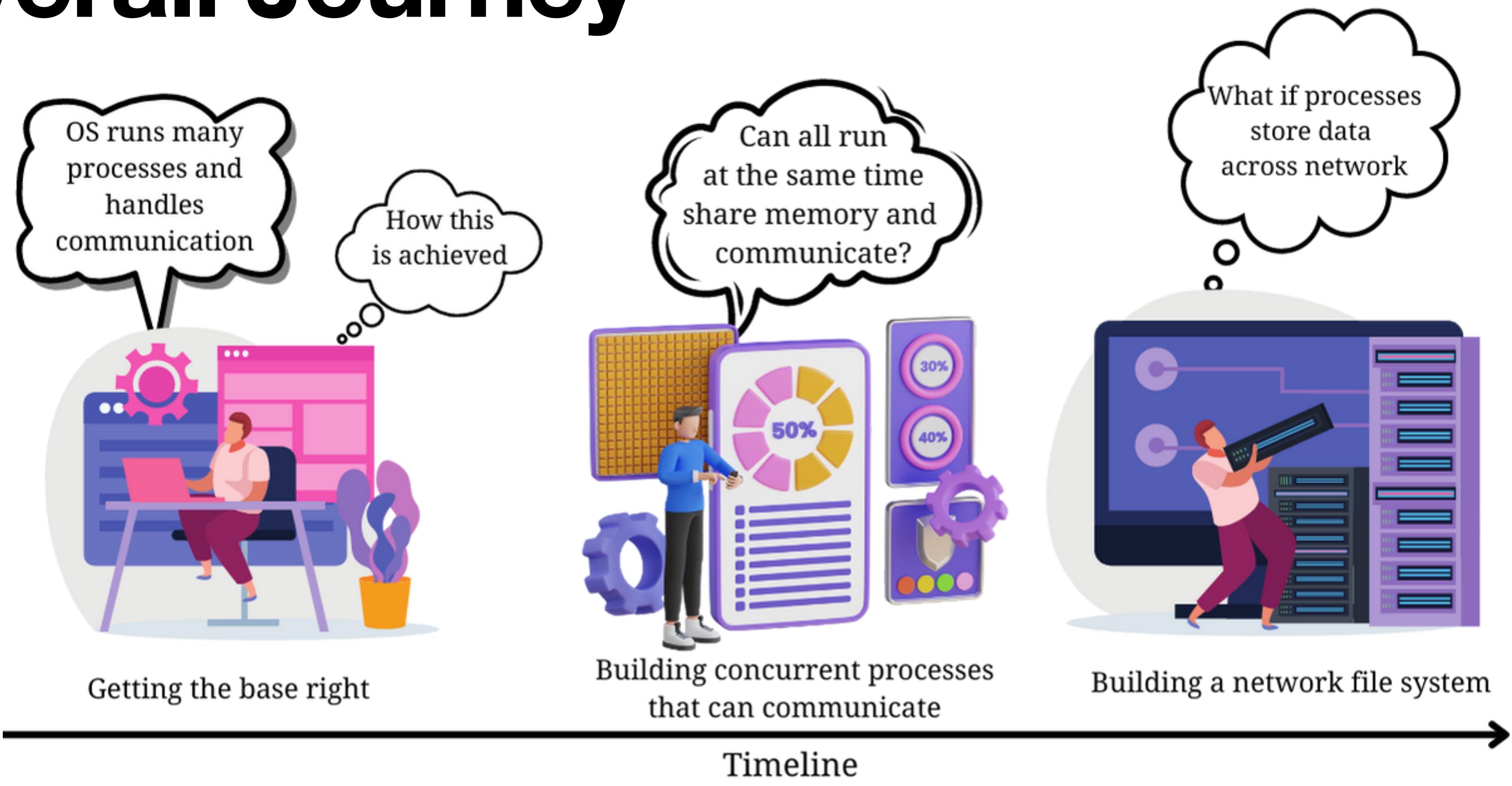
- Inode - Data structure for each file
- Store inodes, data, mapping to inodes, etc in a large array

## Access Methods

- Start with the root
  - Traverse through the path using inode mapping
- Caching can be used to improve efficiency



# The overall Journey



## This Course

Process and Memory Virtualization

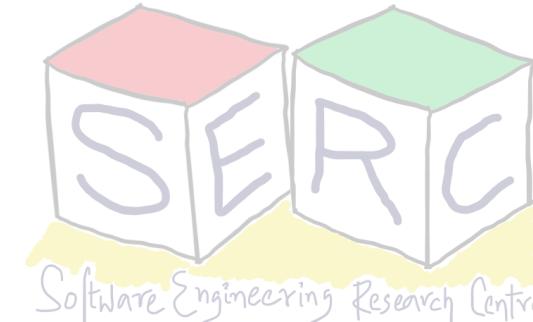
Networking intro

Concurrency

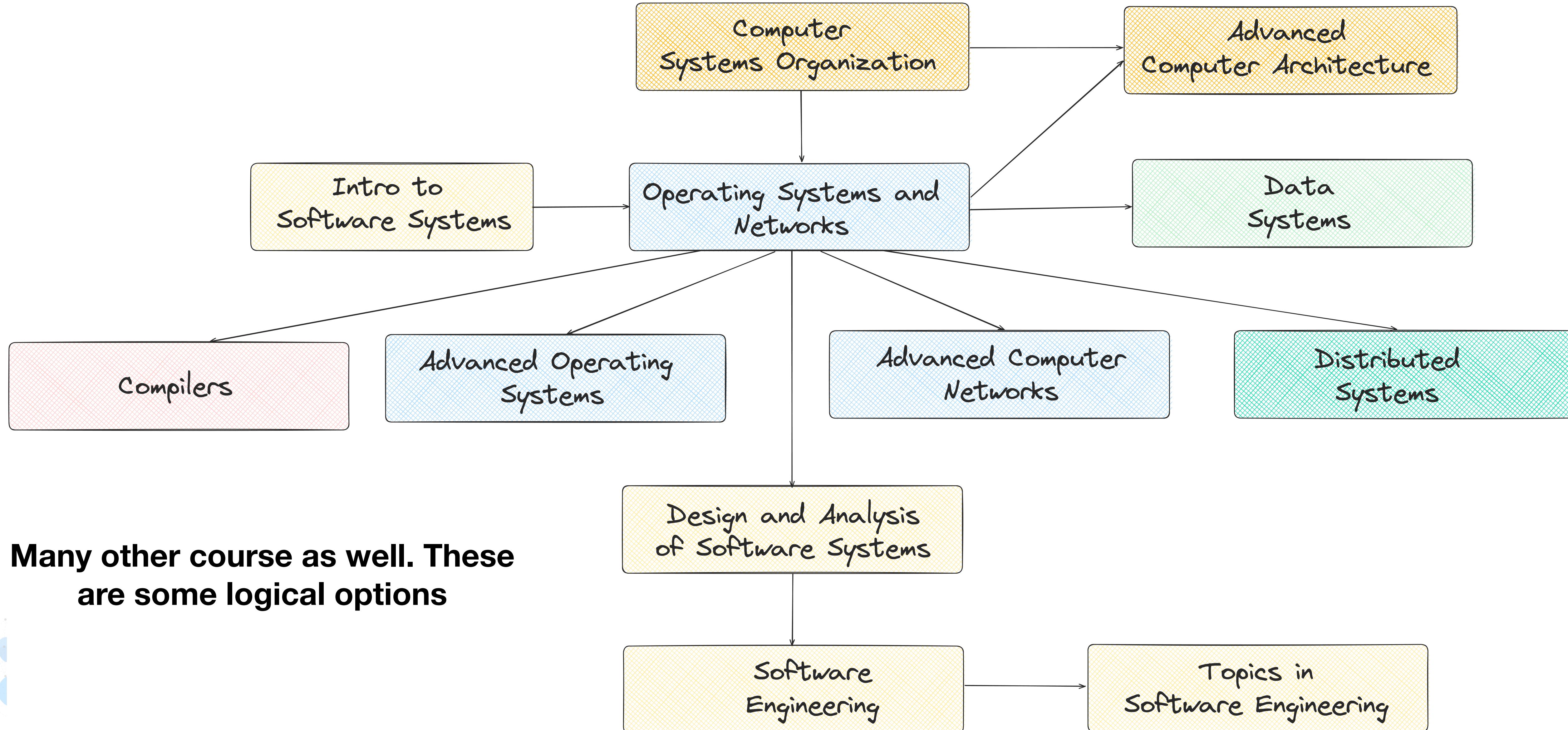
Addressing and Routing

Persistence

Network file Systems



# What next?



# Course Restructuring

- Fresh perspective to the OSN course this year
- Adopted different set of books:
  - Modified the grading scheme - More weightage to projects
  - Introduced **course project (group)**
  - 3 Mini projects
  - Every project had a network component
  - OS + Networks were kept **more intertwined**
  - Countless hours of brainstorming with former students + TAs
  - Feedbacks are always welcome!!



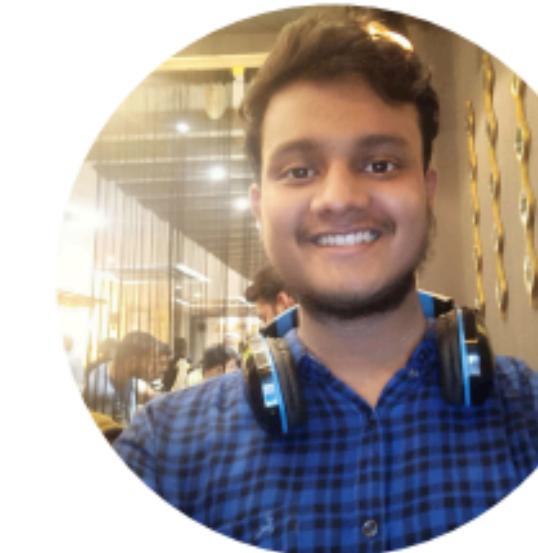
# The Team! Many Thanks to all TAs



VJS Pranavasri



Ashna Dua



Divij D



Hitesh Goel



Karthik Vaidhyanathan



Jhalak Akhilesh Banzal



Prince Varshney



Roja Lakshmi Sahoo



Sarthak Bansal



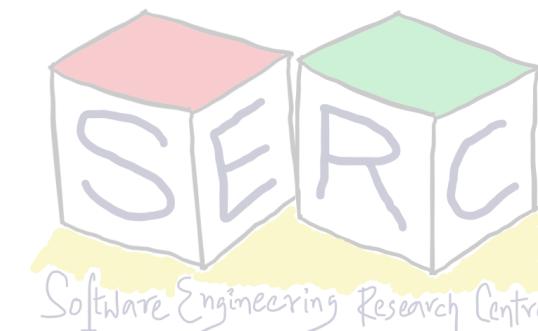
Swayam Agrawal



Vineeth Bhat



Vyom Goyal





**Thank you**

**Course site:** [karthikv1392.github.io/cs3301\\_osn](https://karthikv1392.github.io/cs3301_osn)

**Email:** [karthik.vaidhyanathan@iiit.ac.in](mailto:karthik.vaidhyanathan@iiit.ac.in)

**Twitter:** [@karthyishere](https://twitter.com/karthyishere)

