

## Procesos

### Semana 2: Procesos

En esta semana conoceremos el concepto de procesos y entenderemos cómo hace el sistema operativo para implementar de forma segura y eficiente esta importante abstracción en los computadores modernos.

Te invitamos a revisar el material que se detalla en la tabla mostrada a continuación:

Material	Descripción
Texto guía	• Capítulo 3 - A Dialogue on Virtualization ( <a href="#">link</a> )
	• Capítulo 4 - The Abstraction: The Process ( <a href="#">link</a> )
	• Capítulo 6 - Mechanism: Limited Direct Execution ( <a href="#">link</a> )
Diapositivas de apoyo	• Clase 2 - Procesos ( <a href="#">link</a> )
	• Clase 3 - Ejecución Directa Limitada ( <a href="#">link</a> )
Capítulo complementario	• Capítulo 5 - Interlude: Process API ( <a href="#">link - código</a> )

Introducción

Capítulo del Libro

Resumen

Diapositivas de la clase de hoy (19/08/2025)

### 1. Avisos:

#### ☒ Academia Red Hat

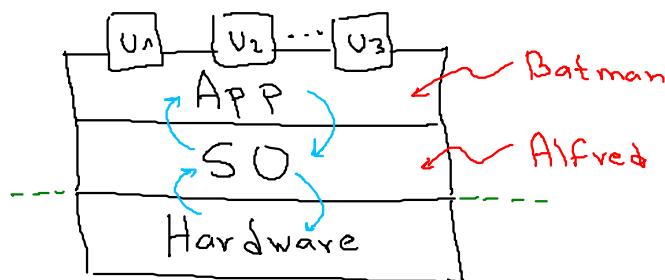
- Red Hat OpenShift Development I: Introduction to Containers with Podman ✓
- Red Hat Enterprise Linux Automation with Ansible ✓
- Red Hat OpenStack Administration I: Core Operations for Domain Operators ✓
- New Features in Red Hat Enterprise Linux
- Red Hat Application Development I: Programming in Java EE
- Administración de Sistemas Red Hat II
- Red Hat Training - Introduction to Python Programming
- Network Automation with Red Hat Ansible Automation Platform ✓

Opcional / Nota dentro del curso

#### ☒ Quiz semana 1 (Hasta hoy a las 23:59)

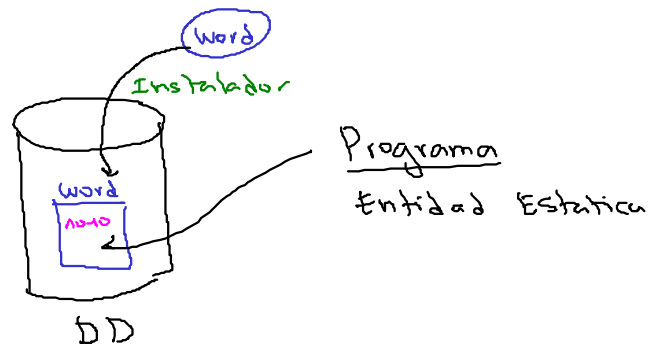
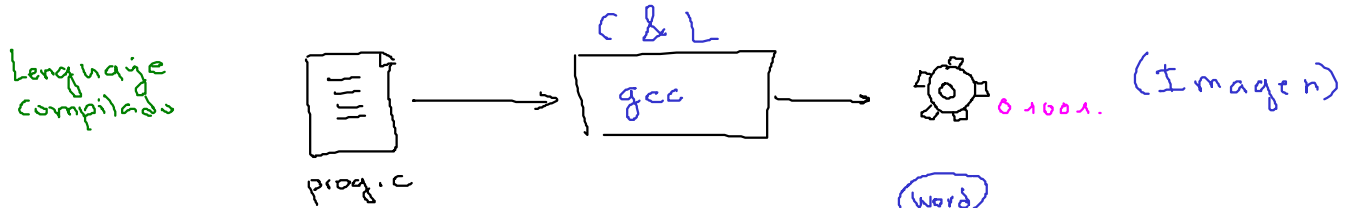
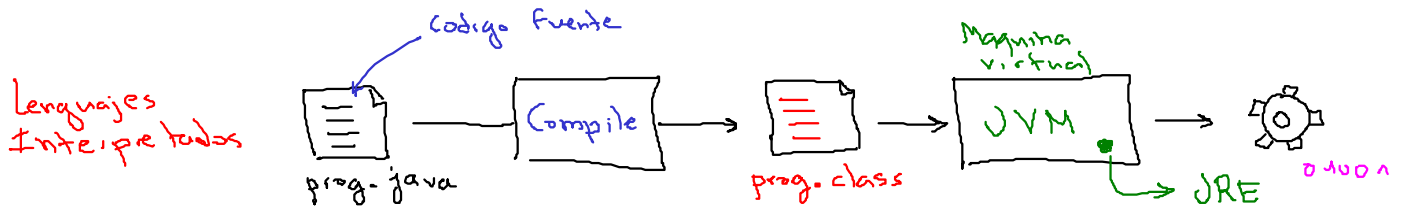
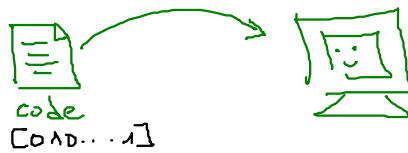
### 2. Repaso clase anterior

1. Que es un SO → Programa: Que hace?  
- Virtualizar (CPU, Memoria)



- Administración eficiente de recursos:
  1. CPU
  2. Mem
  3. I/O
  - ...
- Manejar fallos

### 3. Sobre los programas

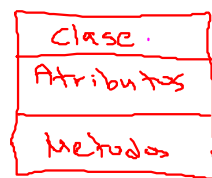


Abstracción

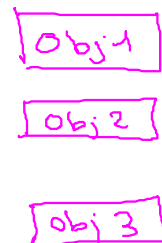


POO

CLASE  
(Plantilla)



OBJETOS

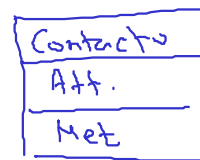


Aplicación: Agenda

Apps



Persona  
- Datos  
- ...



German  
Isabel  
Jonas  
...

Virtualizador (CPU)

SO



Manual de la CPU

Estructura de datos

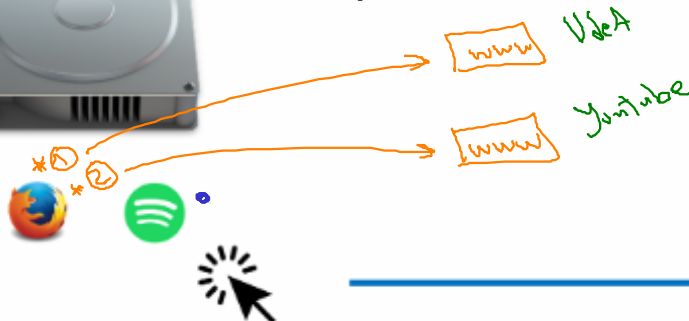
Como es?



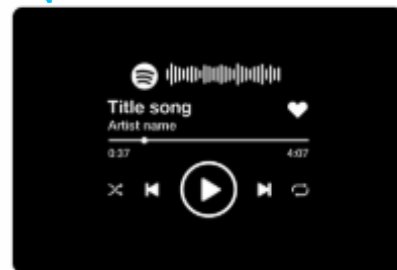
## Programa



- Estático
- Almacenado en DD



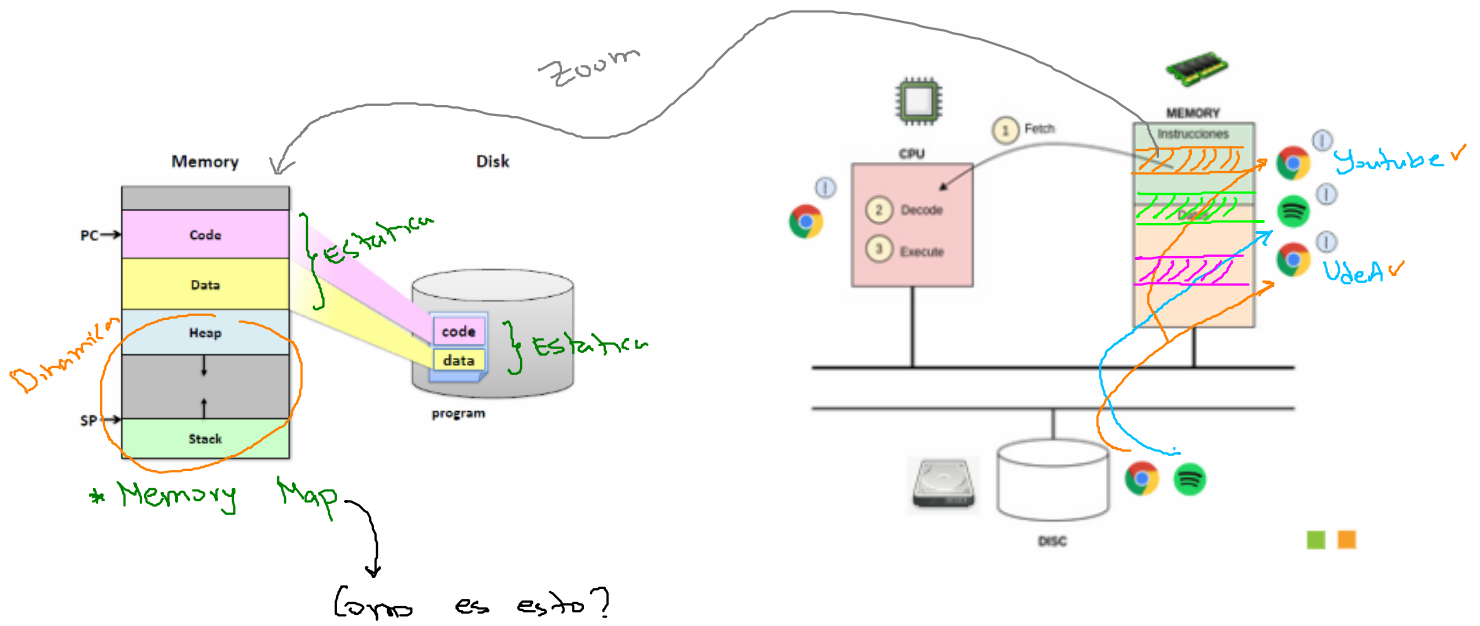
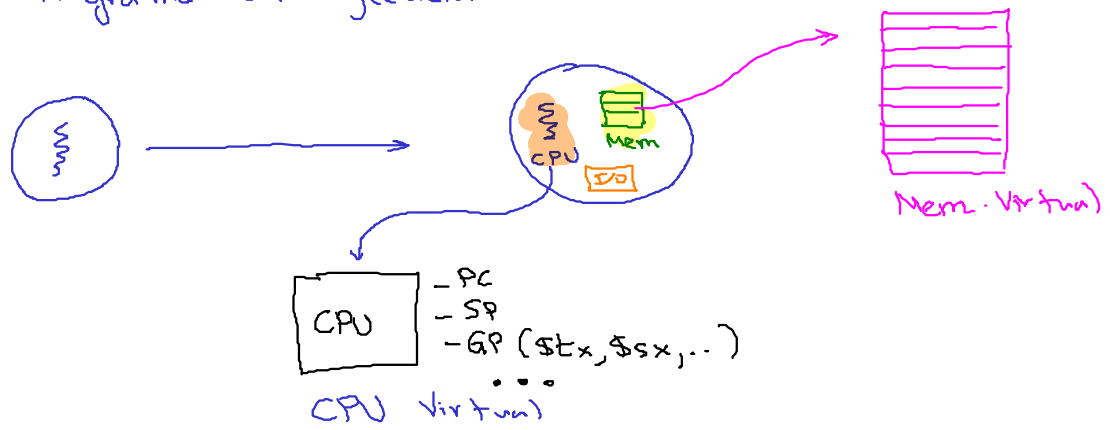
Programa en ejecución Vida



- Estados

The diagram illustrates the abstraction of a process into its core components. At the top left, a mouse cursor icon points to a blue circle containing a wavy line, representing a process. A blue arrow points from this process to a larger blue circle at the top right, also containing a wavy line. A green arrow labeled "Spotify" points from the "Spotify" text to the larger blue circle. Below the larger blue circle, a list of components is shown: CPU ① (red), Memoria ② (green), and I/O ③ (blue). A blue arrow points from the "Como abstraer la CPU?" text to the CPU component. The CPU component is a yellow box labeled "CPU ①" containing "PC", "SP", and "...". Below the CPU box is the label "Registers". To the right of the CPU box is the "MM ②" component, which is a stack of boxes labeled "Code", "Data", "heap", and "stack". Below the MM box is the label "Address Space". At the bottom of the diagram is a green box labeled "I/O ③". A magnifying glass icon is positioned over the CPU component.

Proceso = Programa en ejecución



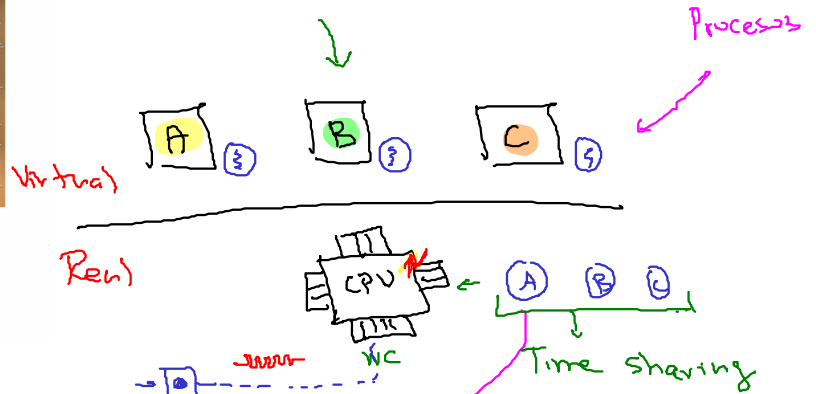
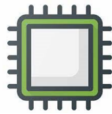
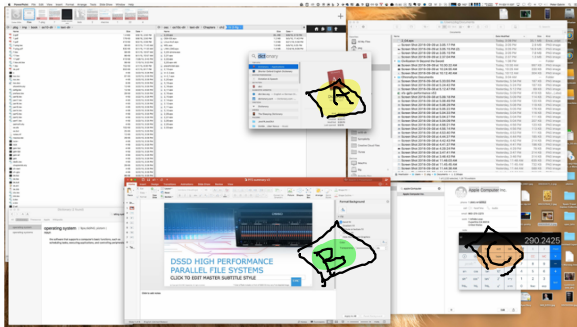


21/08/2025 - Sistemas Operativos (Ude@)

Repaso clase anterior

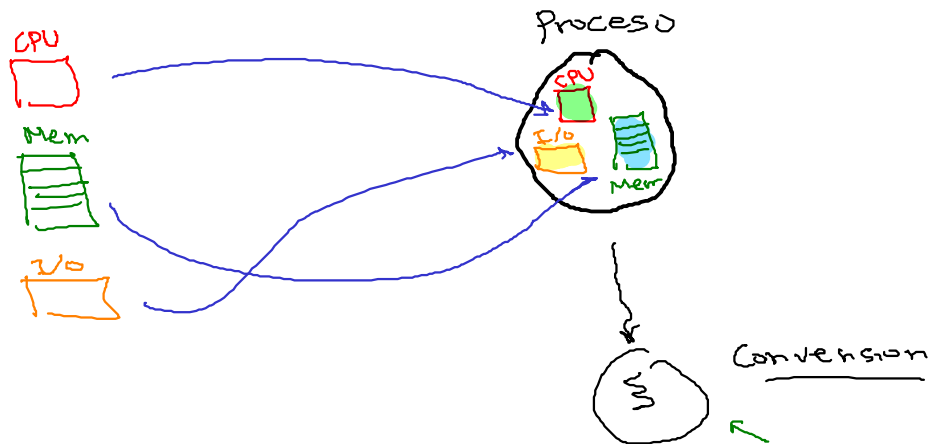
? Como es posible la **conurrencia**?

Como es posible que  
varios programas corran  
a la vez si solo tengo  
una CPU.

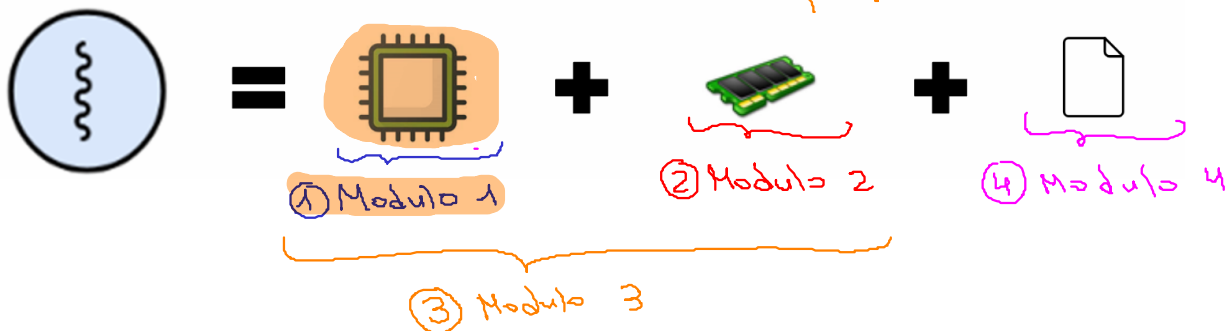


Como es un proceso en código.

Abstracción → **Proceso**



Proceso = CPU + Memory + I/O Info



# Programa .vs. Proceso

## Proceso

- Memoria Física (RAM) ✓
- Dinámico → Cambia: ☺ Live

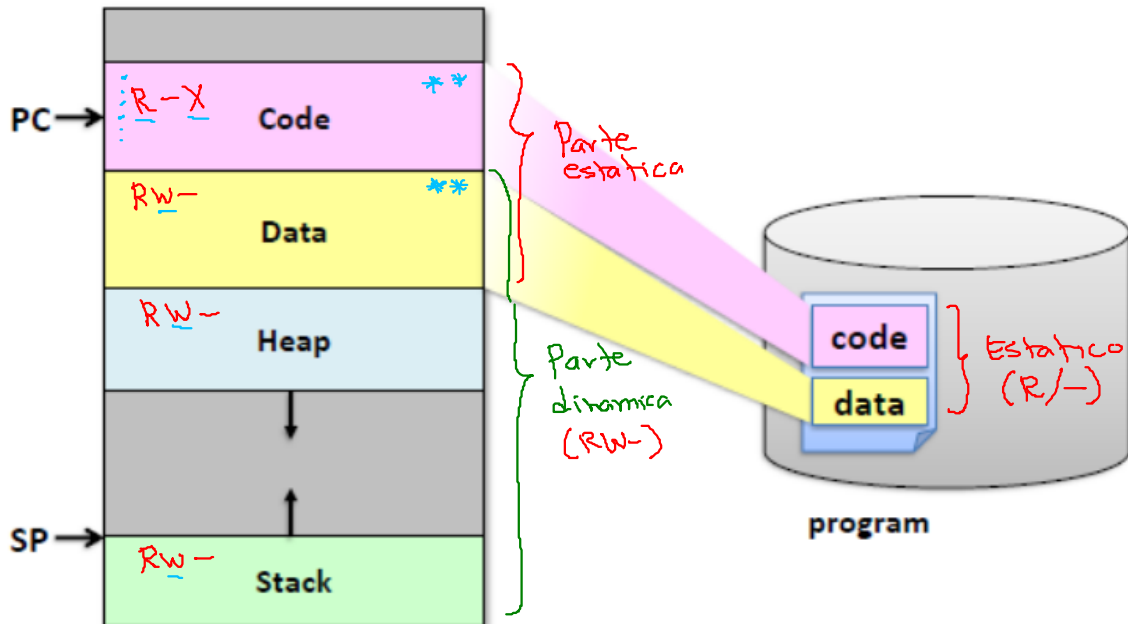
Memory

.vs.

## Programa

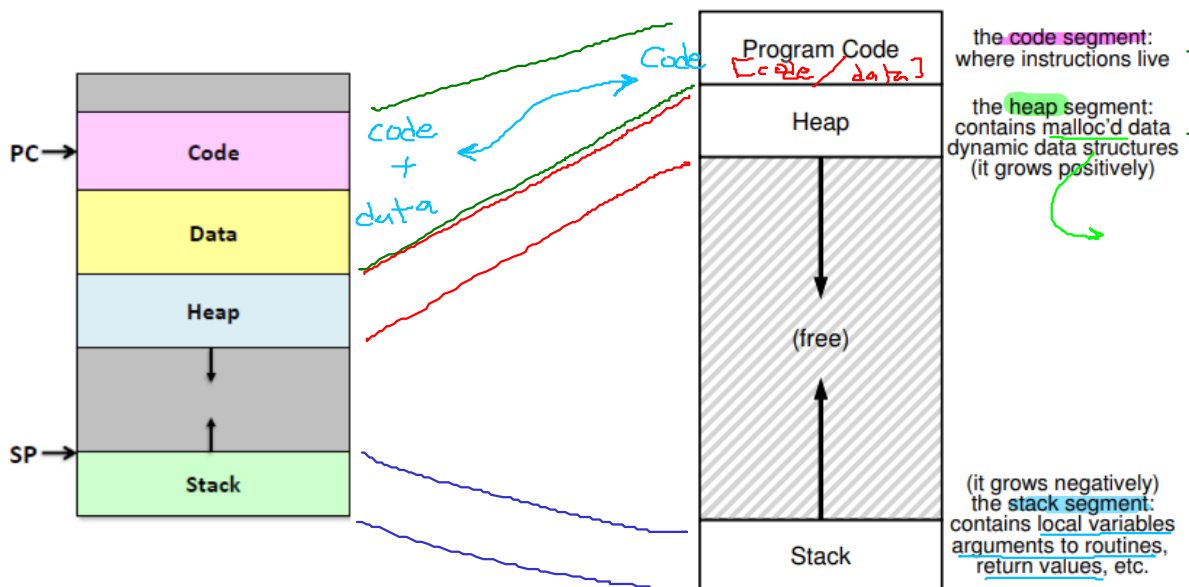
- Disco duro ✓
- Estático: ☹ Dead

Disk



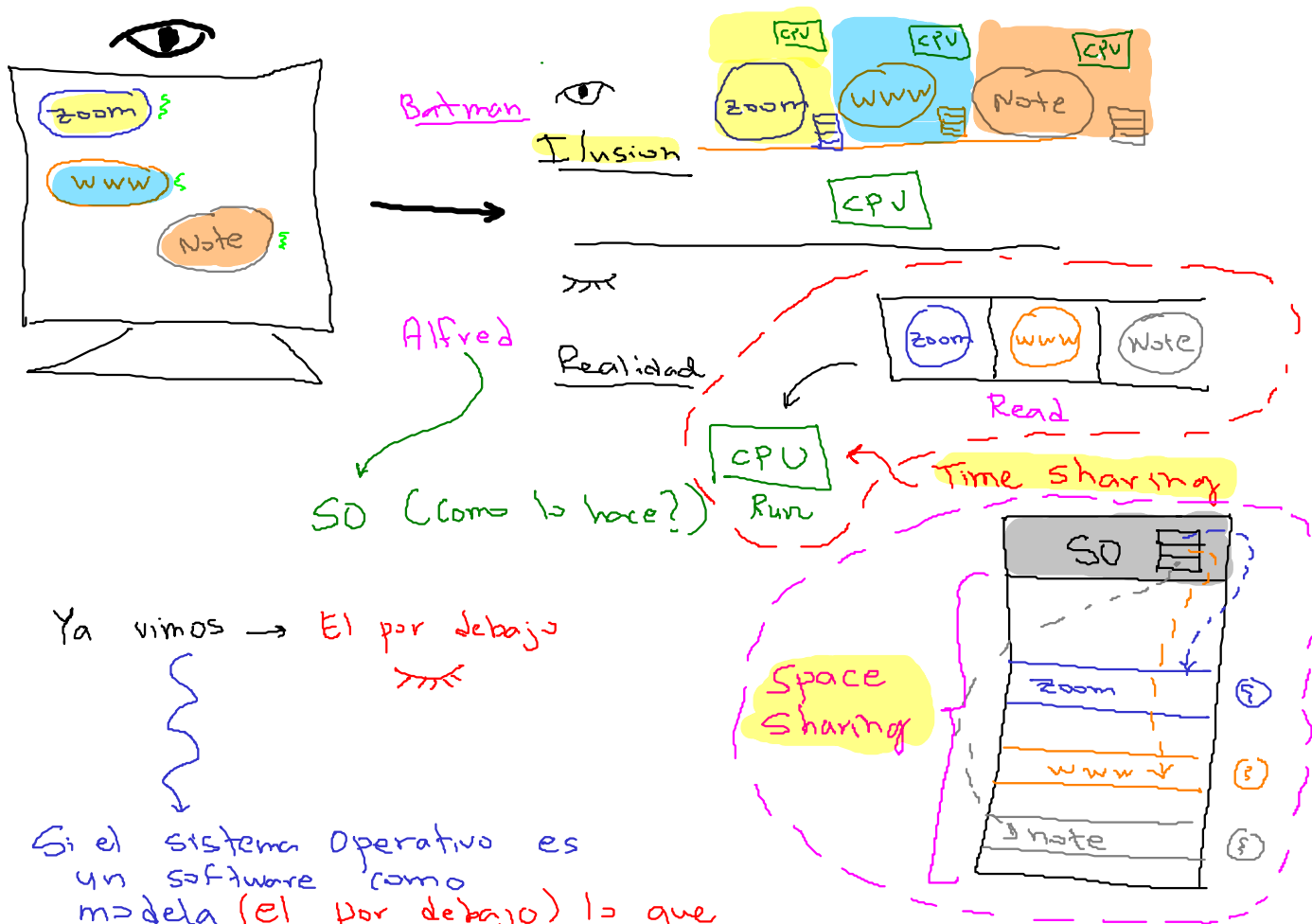
¿Cómo se organizan los datos? → Mapa de Memoria

Convencion → Convencion de Remzi



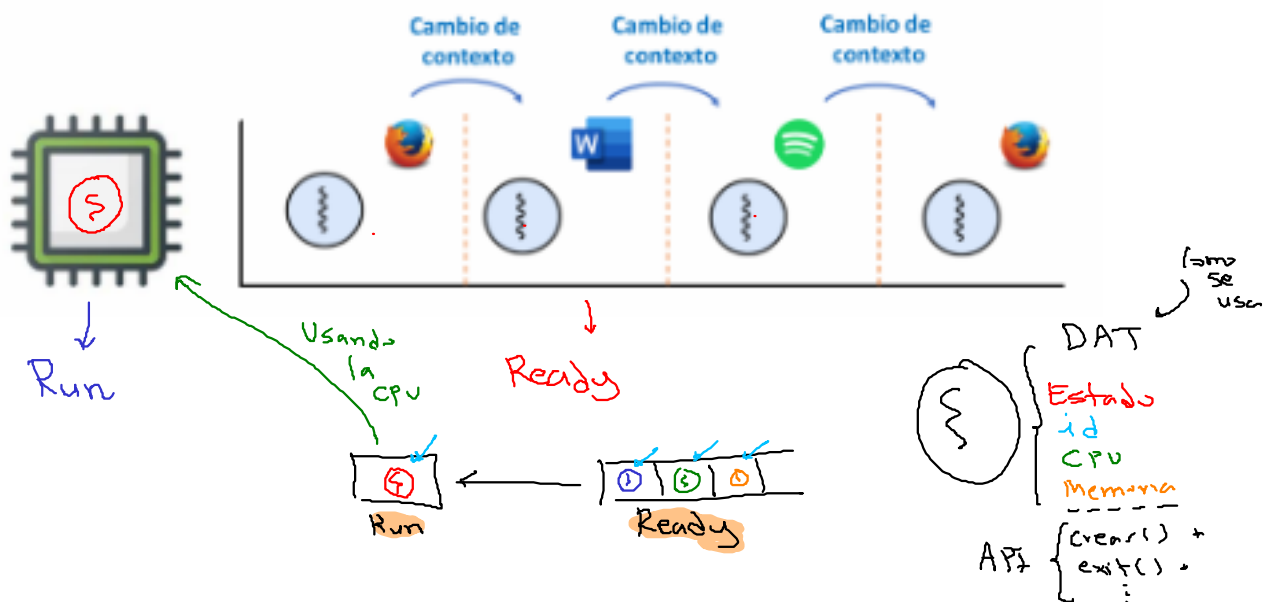
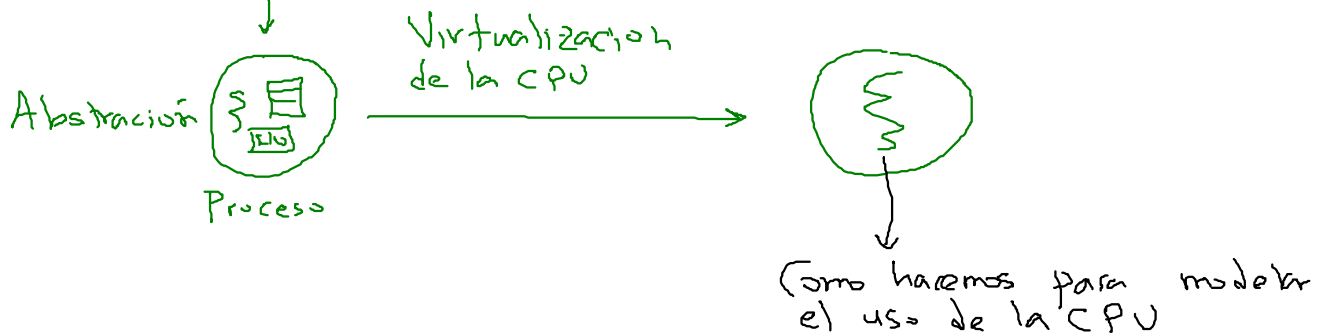




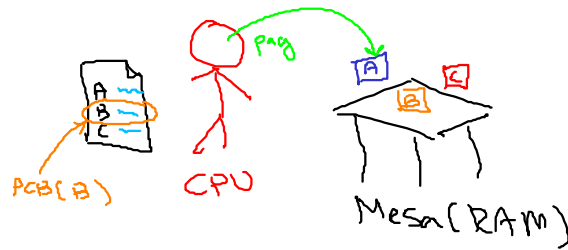


Ya vimos → El por debajo

Si el sistema operativo es un software como modela (el por debajo) lo que no vemos

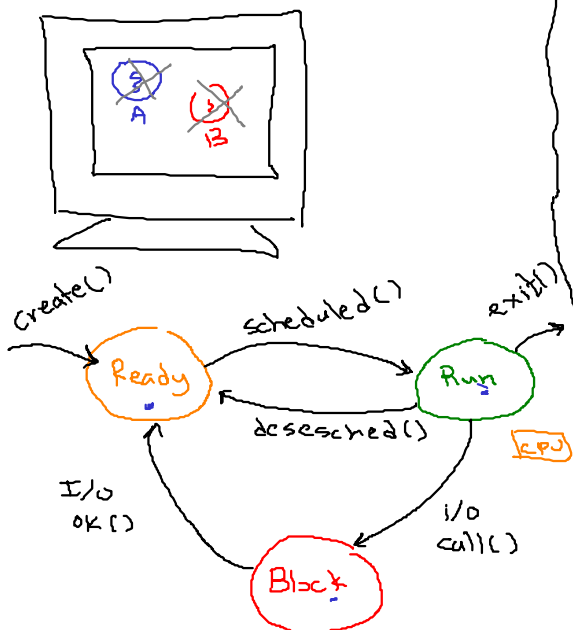


Ir a la Biblioteca



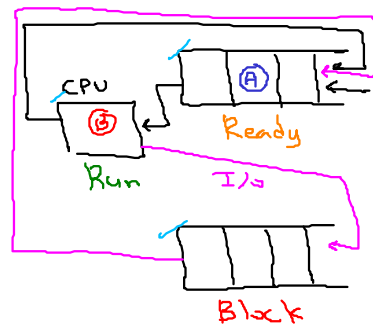
API → El Sistema Operativo controla los procesos usando el API de procesos

Abstracción  
↓  
Proceso

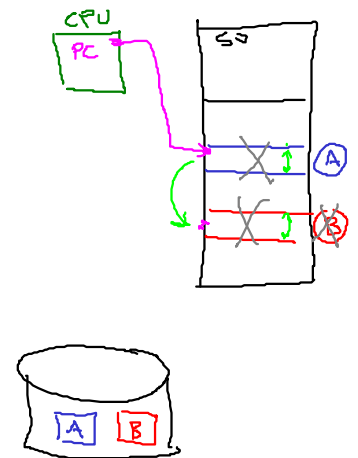


Modelo de 3 estados

Estructuras de Datos



HW



Que estructuras de datos necesito?



Proceso = CPU + Mem + I/O

Funciones

Función	Descripción
Crear (create)	Crea un nuevo proceso para ejecutar un programa.
Eliminar (destroy)	Forzar la detención de un programa en ejecución.
Esperar (wait)	Espera que un proceso termine su ejecución.
Operaciones de control	Ej. Suspender un proceso.
Estado (status)	Información de estado del proceso.

Información

PCB (Process control block)

(A) → PCB(A)

(B) → PCB(B)

