

# Sistemas Operativos

## Estructura de los sistemas operativos

---

Curso 2025

Facultad de Ingeniería, UDELAR

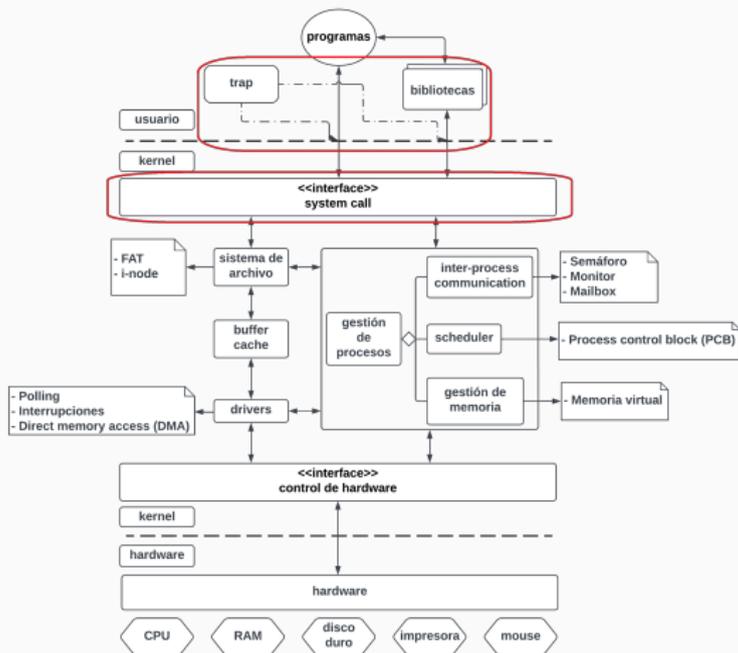
# Agenda

1. Llamados al sistema (**system calls**)
2. Estructura del sistema
3. Servicios del sistema operativo (**system services**)

# Llamados al sistema (**system calls**)

---

# Componentes de un sistema



## Llamados al sistema

- Los **llamados al sistema** (system calls) son una interfaz, provista por el núcleo, para que los procesos de usuarios accedan a los diferentes **servicios** que brinda el sistema operativo.
- Al principio los system calls estaban desarrollados en lenguaje de la arquitectura de la máquina.
- En los sistemas modernos están programados en lenguajes de programación de alto nivel como C o C++.
- Los servicios son invocados por los procesos en modo usuario, cuando ejecutan lo hacen en modo monitor, y al retornar vuelven al modo usuario.
- Típicamente a los system calls se les asocia un número que los identifica (en Linux son aproximadamente 350).

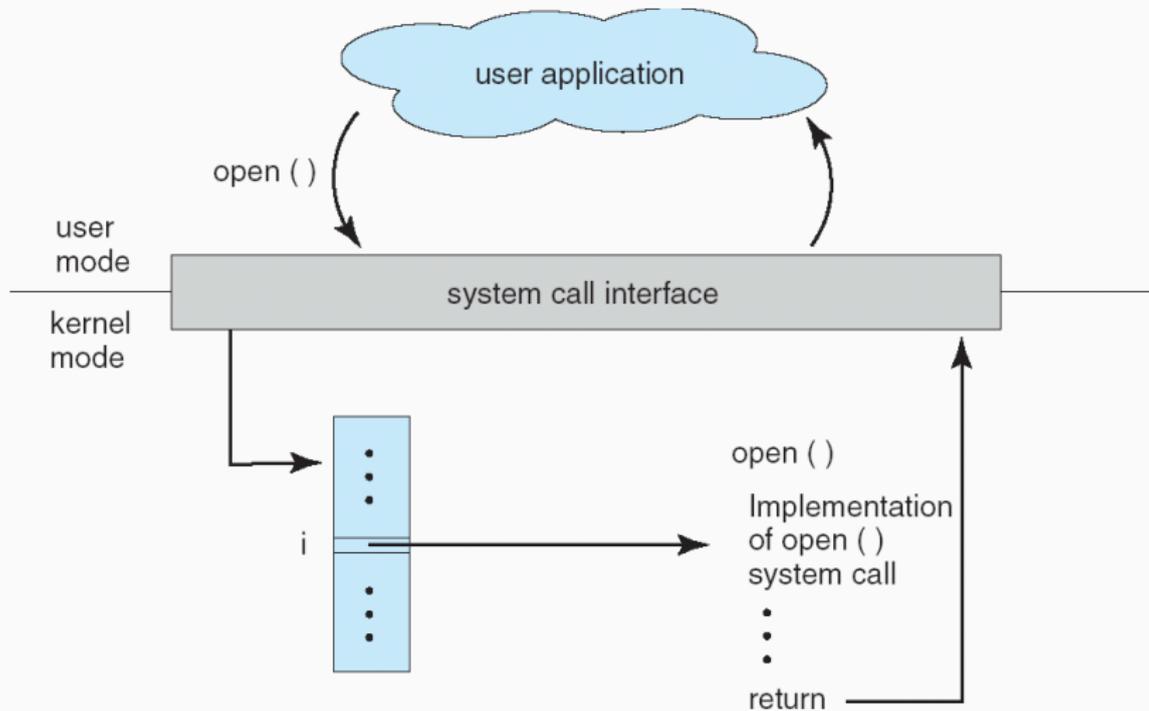
## Llamados al sistema

- La llamada a un **system call** incluye las siguientes tareas:
  - Cargar los parámetros en el lugar adecuado (stack o registros).
  - Cargar el número de system call en algún registro específico (Ej: eax en Intel).
  - Invocar a la interrupción por software (trap) adecuada (**system call handler**).
  - El hardware cambia el bit de modo a monitor e invoca al manejador de la interrupción que controla que el número de system call pasado en el registro sea menor que el mayor del sistema y, finalmente, invoca al system call correspondiente.
  - El valor retornado por el system call es puesto en un registro específico (Ej.: eax en Intel).
  - Se vuelve el sistema a modo usuario y se retorna el control al proceso que invocó la system call (o eventualmente a otro)

## Llamados al sistema

- Existen 3 formas de pasar los parámetros al sistema operativo:
  - A través de los **registros**: Se utilizan un conjunto de registros para pasar los parámetros. Tiene el problema de la cantidad de parámetros es fija y que restringe el tamaño del valor.
    - En Intel se utilizan 5 registros: ebx, ecx, edx, esi, y edi.
  - Un **bloque de memoria** apuntado a través de un registro.
  - En el **stack** del proceso que realiza el llamado. El proceso guarda los parámetros con operaciones push sobre el stack y el sistema operativo los saca con la operación pop.
- De la misma forma se pueden recibir los datos de respuesta

# Llamados al sistema



# Llamados al sistema

- Los system calls se clasifican en distintos tipos:
  - **Control de procesos**
    - Cargar, ejecutar, finalizar, abortar, obtener atributos, cargar atributos, esperar por tiempo, esperar por un evento o señal, obtener o liberar memoria, etc.
  - **Gestión de archivos**
    - Crear, borrar, abrir, cerrar, leer, escribir, obtener o cargar atributos, etc.
  - **Gestión de dispositivos**
    - Requerir o liberar un dispositivo, leer o escribir, buscar o cargar atributos de un dispositivo, etc.
  - **Gestión de información del sistema**
    - Obtener o cargar la hora del sistema, datos del sistema, de procesos, etc.
  - **Comunicaciones**
    - Crear o destruir conexiones, enviar o recibir mensajes, etc.

# **Estructura del sistema**

---

# Estructura del sistema

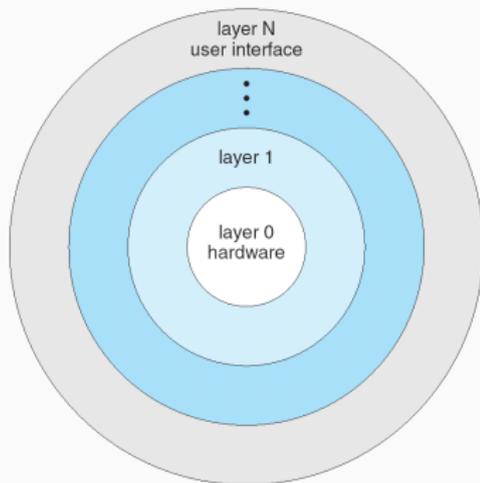
- Las estructuras interna de los sistemas operativos pueden ser muy diferentes.
- Se deben tener en cuenta:
  - Metas de los usuarios: ser amigable, intuitivo, confiable, seguro, rápido, etc.
  - Metas del sistema: fácil de diseñar, implementar y mantener, también flexible, confiable y eficiente.
- Diseño del sistema:
  - Sistema **monolítico**.
  - Sistema **en capas**.
  - Sistema con micronúcleo (**microkernel**).

# Sistema monolítico

- No se tiene una estructura definida.
- El sistema es escrito como una colección de procedimientos, que pueden ser invocados por cualquier otro.
- No existe “ocultación de información”, ya que cualquier procedimiento puede invocar a otro.
- Si bien todo procedimiento es público y accesible a cualquiera, es posible tener buenos diseños y lograr, de esa forma, buena eficiencia en el sistema.
- Ej.: **MS-DOS**.
  - Los componentes pueden invocar procedimientos de cualquiera.
- Ej.: **Linux**
  - Linux es un núcleo monolítico que ha logrado un buen diseño orientado a objetos (sistema modular).

# Sistema en capas

- Se organiza el diseño en una jerarquía de capas construidas una encima de la otra.
- Los servicios que brinda cada capa son expuestos en una interfase pública y son consumidos solamente por los de la capa de arriba.
- La capa 0 es el hardware y la N es la de procesos de usuario.

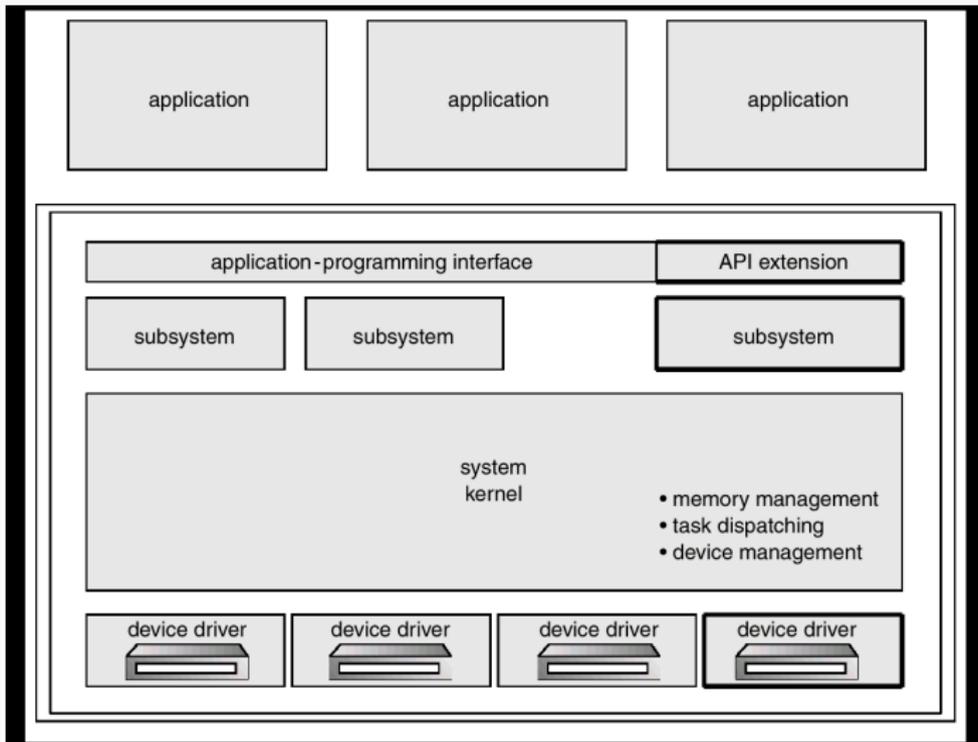


# Sistema en capas

- Ventajas:
  - Modularidad.
  - Depuración y verificación de cada capa por separado.
- Desventajas
  - Alto costo de definición de cada capa en la etapa de diseño.
  - Menos eficiente frente al sistema monolítico ya que sufre de **overhead** al pasar por cada capa.

# Sistema en capas

- Ej.: en capas – OS/2.



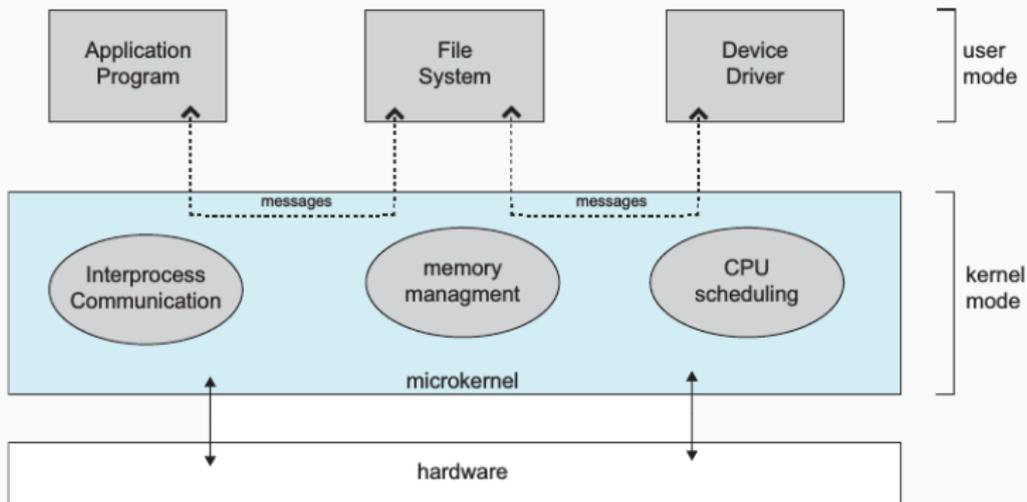
## Sistema con micronúcleo (microkernel)

- Se constituye de un núcleo que brinde un manejo mínimo de procesos, memoria y, además, provea de una capa de comunicación entre procesos.
- La capa de comunicación es la funcionalidad principal del sistema.
- Los restantes servicios del sistema son construidos como procesos separados al micronúcleo que ejecutan en modo usuario.
- El acceso los servicios del sistema se realiza a través de pasaje de mensajes.

- Ventajas:
  - Aumenta la portabilidad y escalabilidad ya que encapsula las características físicas del sistema
  - Para incorporar un nuevo servicio no es necesario modificar el núcleo.
  - Es más seguro ya que los servicios corren en modo usuario.
  - El diseño simple y funcional típicamente resulta en un sistema más confiable.

# Sistema con micronúcleo

- Ej:



# Servicios del sistema operativo (**system services**)

---

- El sistema brindará un entorno de ejecución de programas dónde se dispondrá de un conjunto de servicios que serán accesible mediante una interfase bien definida.
- Servicios básicos que debe brindar un sistema operativo:
  - Ejecución de programas y administración de procesos.
  - Operaciones de Entrada/Salida.
  - Manipulación de sistemas de archivos.
  - Comunicación entre procesos.
  - Detección y manipulación de errores (excepciones).

- Otros servicios de propósito general que deberá brindar el sistema operativo son:
  - Asignación de recursos.
  - Contabilización.
  - Protección, manejo de usuarios y permisos.
- Una vez que están definidos los servicios que brindará el sistema operativo, se puede empezar a desarrollar la estructura del sistema.