

## # Introducción

### ## Introducción 1

#### ### Sistema Operativo

> Qué es?

Es el software principal o conjunto de programas de un sistema informático que gestiona los recursos de hardware y provee servicios a los programas de aplicación de software, ejecutándose en **\*\*modo \*\*privilegiado\*\*\*\*** respecto de los restantes (aunque puede que parte de él se ejecute en espacio de usuario).

En ciertos textos, el sistema operativo es llamado indistintamente como núcleo o kernel, pero debe tenerse en cuenta que la diferencia entre kernel y sistema operativo solo es aplicable si el núcleo es monolítico, lo cual fue muy común entre los primeros sistemas. En caso contrario, es incorrecto llamar al sistema operativo núcleo.

> Unica definición?

Existen dos perspectivas al momento de definir lo que es un sistema operativo.

#### #### Perspectiva de arriba hacia abajo

- Abstracción con respecto a la arquitectura
- **\*\*Arquitectura\*\***, conjunto de instrucciones, organización de memoria, E/S, estructura de bus)
- El SO “oculta” el HW y presenta a los programas abstracciones más simples de manejar.
- Los programas de aplicación son los *\_clientes\_* del SO.
- **\*\*Comparación\*\***, uso de escritorio y uso de comandos de texto
- Comodidad, “amigabilidad” (friendliness)

#### #### Perspectiva de abajo hacia arriba

- Visión del SO como un administrador de recursos
- Administra los recursos de HW de uno o más procesos

- Provee un conjunto de servicios a los usuarios del sistema
- Maneja la memoria secundaria y dispositivos de I/O.
- Ejecución simultánea de procesos
- Multiplexación en tiempo (CPU) y en espacio (memoria)

### ### Objetivos del Sistema Operativo

- Comodidad
- Hacer mas fácil el uso de la PC
- Eficiencia
- Hacer un uso más eficiente de los recursos del sistema
- Evolución
- Permitir la introducción de nuevas funciones al sistema sin interferir con funciones anteriores

### ### Componentes de un Sistema Operativo

#### #### Kernel

*\_Porción de código\_* que se encuentra en memoria principal y se encarga de la administración de los recursos. Implementa servicios esenciales tales como Manejo de Memoria, Administración de procesos, Comunicación y Concurrencia, y Gestión de la E/S.

*\_Ver en diapositivas Servicios de un SO\_*

### ### Problemas que un SO debe evitar

Un sistema operativo debe resolver el acceso a los recursos compartidos. Entre ellos, gestionar el uso de la CPU, detectar accesos ilegales a la memoria, detectar intentos de ejecución de instrucciones de E/S ilegales, proteger el vector de interrupciones, etc.

#### #### Apoyo del HW

- Modos de Ejecución
- Interrupción de Clock
- Protección de la memoria
  - Memoria Lógica
  - Memoria Física

#### #### Modos de Ejecución

- El bit en la CPU indica el modo actual
- Las instrucciones privilegiadas deben ejecutarse en modo **\*\*Supervisor\*\*** o **\*\*Kernel\*\***
  - necesitan acceder a estructuras del kernel, o ejecutar código que no es del proceso
- En modo Usuario, el proceso puede acceder sólo a su espacio de direcciones, es decir a las direcciones *\_propias\_*.
- El kernel del SO se ejecuta en modo supervisor
- El resto del SO y los programas de usuario se ejecutan en modo usuario (subconjunto de instrucciones permitidas)

#### ##### Como lo distinguimos

- Cuando se arranque el sistema, arranca con el bit en modo supervisor.
- Cada vez que comienza a ejecutarse un proceso de suario, este bit se DEBE PONER en modo usuario.
  - Mediante una Instrucción especial.
- Cuando hay un trap o una interrupción, el bit de modo se pone en modo Kernel.
  - Única forma de pasar a Modo Kernel
  - No es el proceso de usuario quien hace el cambio explícitamente.

#### ##### Como actua

Cuando el proceso de usuario intenta por sí mismo ejecutar instrucciones que pueden causar problemas (las llamadas instrucciones privilegiadas) esto el HW lo detecta como una operación ilegal y produce un trap al SO.

#### ##### Resumen

- **\*\*Modo kernel\*\***
  - Modo privilegiado
  - Manejo estricto de pautas de confabilidad/seguridad
  - Manejo de:
    - CPU, memoria, operaciones de entrada/salida
    - Administración multiprocesador, diagnosticos, testing
    - Estructura del Filesystem
    - Comunicaciones: interfaz de red
- **\*\*Modo user\*\***
  - Más flexible
  - Funciones de Mantenimiento más simples, debugging
    - Compiler, assembler, interpreter, linker/loader
    - File system management, telecommunication, network management
    - Editors, spreadsheets, user applications

#### #### Protección de la Memoria

- Delimitar el espacio de direcciones del proceso
- Poner límites a las direcciones que puede utilizar un proceso
  - Por ejemplo: Uso de un registro base y un registro límite
  - El kernel carga estos registros por medio de instrucciones privilegiadas. Esta acción sólo puede realizarse en modo Kernel

\* \* \*

La memoria principal aloja al SO y a los procesos de usuario

- El kernel debe proteger para que los procesos de usuario no puedan acceder a donde no les corresponde
- 
- El kernel debe proteger el espacio de direcciones de un proceso del acceso de otros procesos.

*\_Ver imagen diapositiva 14 - Introducción 2\_*

#### #### Protección de la E/S

Las instrucciones de E/S se definen como privilegiadas por lo que deben ser ejecutadas en **\*\*Modo Kernel\*\*** para ser gestionadas por el `_kernel_` del sistema operativo.

#### #### Protección de la CPU

Para la protección de la CPU se utiliza interrupción por clock para evitar que un proceso se apropie de la CPU.

Se implementa normalmente a través de un clock y un contador.

Para esto, el kernel le da un valor al contador que se decrementa con cada tick del reloj y al llegar a 0 puede expulsar este proceso para gestionar otro.

Las instrucciones que modifican el funcionamiento del reloj son privilegiadas.

Se le asigna al contador el valor que se quiere que se ejecute un proceso.

Se la usa también para el cálculo de la hora actual, basándose en cantidad de interrupciones ocurridas cada tanto tiempo y desde una fecha y hora determinada.

#### ### System Calls

Los programas acceden a los servicios del kernel del Sistema Operativo. Consecuentemente, las System Calls se ejecutan en modo Kernel o Supervisor.

Los parámetros asociados a las llamadas pueden pasarse de varias maneras: por registros, bloques o tablas en memoria ó la pila.

```
```c
count = read(file, bufer, nbytes);
```
```

*\_Ver esquema de memoria en diapositiva 19\_*

### **\*\*Categorías de system calls\*\***

- Control de Procesos
- Manejo de archivos
- Manejo de dispositivos
- Mantenimiento de información del sistema
- Comunicaciones

*\_Ver ejemplos y categorias en diapositiva 21\_*

\* \* \*

*\_Ver Anexos\_*