

## TEMA 1 INTRODUCCIÓN

### 1.1 QUE ES UN SISTEMA OPERATIVO

Es un grupo de programas de proceso con las rutinas de control necesarias para mantener continuamente operativos dichos programas.

**Objetivo:** Optimizar todos los recursos del sistema. El sistema operativo es el programa fundamental.

**Programas de sistema:** Controlan la operación de la computadora.

**Programas de aplicación:** Resuelven problema usuarios.

El s. O. Protege y libera a los programadores de la complejidad del hardware, controlar todas las partes del sistema, presentar maquina virtual.

**Microprograma:** Es un software que se localiza en la memoria de solo lectura. Busca las instrucciones de lenguaje de maquina para ejecutarlas como una serie de pequeños pasos, define lo que es el lenguaje de maquina.

**Lenguaje de maquina:** Posee entre 50 y 300 instrucciones, (desplazar datos, hacer operaciones aritméticas y comparar valores.) El S.O. Se ejecuta en modo central o modo de supervisión, Los compiladores, editores y demás programas se ejecutan en modo usuario.

**Principales características de los s. O.:**

- |  |                                |
|--|--------------------------------|
| 1. Definir la “interfaz del usuario”.                | 4. Facilitar la e/s.           |
| 2. Compartir el hardware y los datos entre usuarios. | 5. Recuperarse de los errores. |
| 3. Planificar recursos entre usuarios                |                                |

**Principales recursos administrados por los s. O.:**

\*Procesadores. \*Almacenamiento. \*Dispositivos de e/s. \*Datos.

**Los s. O. Son una interfaz con:**

\*Operadores. \*Programadores \*Hardware. \*Usuarios.

### 1.2 HISTORIA DE LOS S.O – GENERACIONES

**Primera generación cero (década del 1940):**

\*Carencia de S.O. \*Completo acceso al lenguaje de maquina.

**Primera generación (1945-1955): bulbos y conexiones:**

\*Carencia de s. O. \*Transición entre trabajos, haciendo la misma mas simple.

**Segunda generación (1955-1965): transistores y sistemas de procesamiento por lotes (batch):**

*Multiprogramación:* varios programas de usuarios en el almacenamiento principal, cambiando el procesador

*Multiprocesamiento:* varios procesadores. Independencia de dispositivo

**Tercera generación (1965-1980): circuitos integrados y multiprogramación:**

- |  |   |
|--|---|
| • Difusión de la multiprogramación:  | • Son sistemas de modos múltiples y de propósitos generales |
| • Partición de la memoria.   | • Aparecen los lenguajes de control de trabajos             |
| • Aprovechamiento del tiempo de espera.  | • Soportan timesharing (tiempo compartido)                  |
| • Aparición de técnicas de spooling (operación simultanea y en línea de periféricos) | • Aparecen los sistemas de tiempo real                      |
| • Almacenamiento de trabajos de e/s en discos  |   |

**Cuarta generación (1980-1990): computadoras personales:**

Software amigable con el usuario y desarrollo de sistemas operativos de red y sistemas operativos distribuidos.

**S.O de red:** usuarios conscientes de varias computadoras conectadas donde C/maquina ejecuta su propio S.O. Local. Son similares a los s. O. De un solo procesador

**S.O distribuidos:** Aparece como un s. O. De un solo procesador. Donde los usuarios no son conscientes del lugar donde se ejecutan sus programas o donde se encuentran sus archivos. Un programa se puede ejecutar en varios procesadores a la vez, maximizando el paralelismo. Además aparecen los emuladores de terminal para el acceso a equipos remotos desde computadoras personales.

### 1.3 CONCEPTOS DE LOS SISTEMAS OPERATIVOS

**Las llamadas al sistema:** es la interfaz entre el S.O. y los programas (**instrucciones ampliadas**)

**Procesos:** es un programa en ejecución que consta del programa ejecutable, sus datos, pila y contador y otros registros. La información de control se almacena en la **tabla de procesos**. Un proceso (suspendido) consta de un espacio de dirección y los datos de la tabla.

**Archivos:** los S.O. deben brindar independencia de dispositivos. La forma de agrupar archivos es mediante

“directorios jerárquicos”, donde cada archivo tiene una ruta de acceso.

**Llamada al sistema:** Permiten a los programas comunicarse con el s. O. Y solicitarle servicios. A cada llamada le corresponde un procedimiento:

- instrucción tipo “trap”.
- s. O. Recupera el control y ejecuta el trabajo solicitado.
- el s. O. Coloca un código de estado y ejecuta “return from trap”
- El procedimiento regresa

#### 1.4 ESTRUCTURA DE LOS SISTEMAS OPERATIVOS (organización interna)

**Sistemas monolíticos:** Es muy común: no existe estructura, El s. O. Es una colección de procedimientos que se pueden llamar entre sí.

**Sistemas con capas:** Consiste en organizar el S.O. Como una jerarquía de capas:

Estructura en Capas
5- Operador
4- Programas de usuario
3- Control de E/S
2- Comunicación Operador – Proceso
1- Administración de la Memoria y del Disco
0- Asignación del Procesador y Multiprogramación

**Máquinas virtuales:** existe un elemento central → **monitor de la maquina virtual**. Pueden ejecutar cualquier S.O., cada máquina con distintos S.O. y soportan periféricos virtuales.

- **Modelo cliente-servidor:** una tendencia es mover el código a capas superiores y mantener un “Núcleo mínimo”. El núcleo controla la comunicación entre los clientes y los servidores. Además se fracciona el S.O. donde cada uno controla una parte y se adapta para su uso en los sistemas distribuidos.

#### 1.5 TENDENCIAS

- multiprocesamiento.
- Distribución del control
- Soporte del paralelismo masivo y soporte de la ejecución concurrente.
- maquinas virtuales y el almacenamiento virtual.
- Compatibilidad.

#### 1.6 HARDWARE Aspectos

**Compaginación del almacenamiento:** Objetivo: acelerar el acceso al almacenamiento primario (**banco de almacenamiento primario**)

**Registro de relocalización:** Permite relocalizar los programas. Almacena la dirección base de un programa

**interrupciones y escrutinio:**

Interrupciones: permite obtener la atención de una unidad. Salva el “estado” antes de la interrupción.

Escrutinio: permite verificar el estado de otra unidad independiente.

**Utilización del “buffer”:** es un área de almacenamiento primario.

- “Entradas de buffer simple”:
  - No puede haber simultaneidad entre operaciones → Afecta la performance.
- “Entradas de buffer doble”:
  - Permite la sobre posición de operaciones de e / s → Mejora la performance.
  - Es la técnica de “buffer biestable (o en flip flop)”.

**Dispositivos periféricos:**

- almacenamiento de grandes cantidades de información fuera del almacenamiento principal.

**Protección del almacenamiento:**

- Limita el n° de direcciones que un programa puede referenciar.
- Se implementa mediante los “registros de limites” o “claves de protección del almacenamiento”

**Temporizadores y relojes:**

- “Temporizador de intervalos”: previene que usuario monopolice el procesador en Sist. Multiusuario.
- genera una interrupción

**Operaciones en línea y fuera de línea; procesadores satélite:**

- “Operación en línea”: conectados al procesador.

- “operación fuera de línea:” conectados a unidades de control

#### **Canales de entrada / salida:**

- Se dedican al manejo de la e / s con independencia del procesador principal y tienen acceso directo al almacenamiento principal
- Principales tipos de canales:
  - Selectores.
  - Multiplexores de bytes.
  - Multiplexores de bloques.

**Robo de ciclo:** Significa que en la competencia entre el procesador y los canales para acceder a un determinado banco de almacenamiento primario (memoria principal), se da prioridad a los canales:

**Estado de problema, estado supervisor, instrucciones privilegiadas:** Son los distintos “estados de ejecución”.

- “estado de problema o de usuario:” acceso a un subconjunto de instrucciones
- “estado supervisor o de núcleo:” acceso a todas las instrucciones
- “instrucciones privilegiadas”: son aquellas a las que no se tiene acceso en estado de problema.

#### **Almacenamiento virtual:**

- Los sistemas de almacenamiento virtual permiten a los programas referenciar direcciones que no necesitan corresponder con las direcciones reales disponibles en el almacenamiento primario.
- “Direcciones virtuales” de los programas en ejecución: son traducidas dinámicamente por el hardware a las “direcciones reales”
- Se utilizan técnicas de:
  - “paginación”: bloques de datos de tamaño fijo
  - “segmentación”: identifica las unidades lógicas

**Multiprocesamiento:** Varios procesadores comparten un almacenamiento primario común y un solo s. O. Y es necesario “secuencializar” el acceso.

#### **Acceso directo a la memoria (dma):**

- Requiere una sola interrupción al procesador por cada bloque de caracteres transferidos durante la operación de e / s y mejora el rendimiento.
- Es como si el procesador, en vez de interrumpido fuera retrasado.
- “canal dma”: es el responsable del robo de ciclos y de la operación de los disp de e / s.

**Canalización:** Es la técnica de explotar ciertos tipos de paralelismo durante el procesamiento de instrucciones.

#### **Jerarquía de almacenamiento:**

- Almacenamiento “cache”: memoria veloz que aumenta la velocidad de ejecución de los programas:
- Almacenamiento primario: memoria principal.
- Almacenamiento secundario: discos, cintas, etc.

## **1.7 SOFTWARE**

Consiste en los programas de instrucciones y datos que definen los algoritmos necesarios para la resolución de problemas.

#### **Programación en lenguaje de maquina:**

- “Lenguaje de maquina”:
  - Lenguaje que un computador puede comprender directamente y que es “dependiente de la maquina”
  - Muy poco usado.

#### **Ensambladores y macroprocesadores:**

- Los “lenguajes ensambladores”:
  - Incrementar la velocidad de programación.
  - Reducir los errores de codificación.
- Los “macroprocesadores”:
  - Acelerar la codificación de un programa ensamblador.
  - Una “macroinstrucción” indica la ejecución de varias instrucciones en lenguaje ensamblador.

**Compiladores:** Es un “Lenguajes de alto nivel” que permiten el desarrollo de programas “independientes de la maquina”.

- “compiladores”: traducen los lenguajes de alto nivel al lenguaje de maquina.
- “traductores”: denominación para “compiladores” y “ensambladores”.

#### **Sistemas de control de entrada / salida (iocs: input / output control system):**

- Libera al programador de aplicaciones de la complejidad de la administración de la e / s:

#### **Utilización del spool (operación simultanea de periféricos en línea):**

- Evita la demora en la ejecución de programas como consecuencia del uso de periféricos lentos.

#### **Lenguajes orientados hacia el procedimiento versus lenguajes orientados hacia el problema:**

- O. Hacia el procedimiento: resolver gran variedad de problemas:
- O. Hacia el problema: resolver determinados tipos de problemas:

#### **Compiladores rápidos y sucios versus compiladores optimizadores:**

- C. Rápidos y sucios: producen rápidamente un programa objeto que puede ser ineficiente respecto de almacenamiento y velocidad de ejecución:
- C. Optimizadores: mayor lentitud pero altamente eficiente en almacenamiento y ejecución

#### **Interpretores:**

- No producen un programa objeto.
- Ejecutan un programa fuente.

#### **Cargadores absolutos y de relocación:**

- Los programas se ejecutan en el almacenamiento principal.
- “asignación”: es la asociación de instrucciones y datos
- “cargador”: coloca las instrucciones y datos
- Cargador absoluto”: instrucciones y datos en las localizaciones específicas
- “cargador de relocación”: Un programa en varios lugares

#### **Cargadores de enlace y editores de enlace:**

- “cargador de enlace”: en el momento de carga, combina cualesquiera programas requeridos y los carga directamente en el almacenamiento primario.
- “editor de enlace”: ejecuta la combinación de programas mencionada y además crea una imagen de carga a memoria

### **1.8 MEMORIA FIJA**

- La “microprogramación dinámica”: permite cargar fácilmente los nuevos “microprogramas” (“microcodigo”) dentro del “almacenamiento de control”
- Los “microprogramas” están formados por “microinstrucciones” son:
  - Más elemental.
  - Función más dispersa.
  - El almacenamiento de control debe ser mucho mas rápido que el almacenamiento primario.

#### **Microcodigos vertical y horizontal:**

El microcodigo permite mejorar el rendimiento en la ejecución de un sistema computacional.

El criterio es colocar en la memoria fija (en vez de en el software) las secuencias de instrucciones utilizadas con mas frecuencia.

- M. Vertical:
  - Similar lenguaje de maquina.
  - Especifica el movimiento de uno o varios datos entre registros.
- M. Horizontal:
  - Es más poderoso pero más complejo que el m. Vertical.
  - Las microinstrucciones requieren muchos más bits.

**Emulación:** es la técnica que hace que una máquina aparente ser otra, donde el conjunto de instrucciones de lenguaje de maquina que va a ser emulada se microprograma en la “maquina anfitriona”.

**Microdiagnosticos:** efectúa la detección y corrección de errores más amplia a un nivel mas fino.

**Computadores personalizados:** Moldean el sistema computacional según las necesidades del usuario.

#### **Microprogramacion y sistemas operativos:**

- Las Funciones implementadas frecuentemente en microcodigo son:
  - Manejo de interrupciones.
  - Mantenimiento de estructuras de datos.
  - Primitivas de sincronización
  - “intercambio de contexto”,