

# Tema 1. Conceptos fundamentales de los Sistemas Operativos

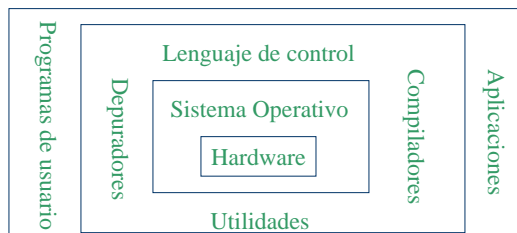
1. Introducción a los Sistemas Operativos.
  1. Concepto de Sistema Operativo. Niveles del software.
  2. Funciones principales de un Sistema Operativo.
2. Evolución y clasificación de los Sistemas Operativos.
  1. Evolución histórica de los Sistemas Operativos.
  2. Clasificación de los Sistemas Operativos.
3. Estructuras de los Sistemas Operativos.
  1. Sistemas monolíticos.
  2. Sistemas en estratos.
  3. Máquinas virtuales.
  4. Modelo cliente-servidor.
  5. Estructura orientada al objeto.

## 1. Introducción a los Sistemas Operativos (I)

### 1.1. Concepto de Sistema Operativo. Niveles del software.

**Definición:** Un Sistema Operativo es un programa o conjunto de programas que actúa como interfase entre el usuario o programador y la máquina física (el hardware).

**Niveles del software.**



# 1. Introducción a los Sistemas Operativos (II)

## 1.2. Funciones principales de un Sistema Operativo. (I)

1. **Inicializar la máquina:** preparar el ordenador para su funcionamiento.
  - a) *Inicialización total* (Initial Program Loading (IPL), Bootstrapping).
  - b) *Inicialización parcial*.
2. **Servir de máquina extendida (virtual):** ocultar los detalles del hardware al usuario y proporcionar un entorno más cómodo. Objetivos:
  - a) *Seguridad:* el S. O. debe evitar que la ejecución de los programas se interfieran unos entre otros.Modos de operación del Hardware:
  - ♦ Modo usuario (estado no privilegiado), |
  - ♦ Modo supervisor (estado privilegiado). ↓

Interrupción, llamada al sistema, llamada al supervisor, trap  
(**Interfase interna**)

# 1. Introducción a los Sistemas Operativos (III)

## 1.2. Funciones principales de un Sistema Operativo. (II)

- b) *Abstracción:* los S.O. construyen recursos (virtuales) de alto nivel a partir de los recursos de más bajo nivel (físicos). La máquina física se transforma en una *máquina virtual*. Con el lenguaje de comandos del S. O. (shell, **Interfase externa**) se invocan a esos servicios.
3. **Administrar los recursos para su funcionamiento:** el S.O. es el responsable de:
  - a) *asignar* a un programa todos los recursos que necesite. Para ello, debe ser justo en el reparto y en el tiempo asignado, impidiendo que no se favorezca a determinados programas;
  - b) *controlar* el uso correcto de los recursos de forma que los programas no se interfieran.

# 1. Introducción a los Sistemas Operativos (IV)

## Característica de un Sistema Operativo

Un S.O. debe ser:

- ♦ **Determinista:** el mismo programa ejecutado con los mismos datos debe dar los mismos resultados en cualquier momento y en cualquier ejecución;
- ♦ **In determinista:** el S. O. debe responder a circunstancias que pueden ocurrir en un orden impredecible.

# 2. Evolución y clasificación de los Sistemas Operativos (I)

## 2.1. Evolución histórica de los Sistemas Operativos. (I)

### Primera generación (1945-1955)

*Hardware:* Tubos de vacío y paneles de control.

*Sistema Operativo:* No existe.

### Segunda Generación (1955-1965)

*Hardware:* Transistores (tarjetas perforadas).

*Sistema Operativo:*

- I. S. O. de trabajo único:** sólo se dedica a “preparar” el sistema para la llegada de trabajos.
- II. S. O. por lotes (batch):** automatiza la secuencia de operaciones involucradas en la ejecución de un programa usando ordenes que lee, interpreta y ejecuta el monitor de lotes (porción de código residente en memoria).

## 2. Evolución y clasificación de los Sistemas Operativos (II)

### 2.1. Evolución histórica de los Sistemas Operativos. (II)

Tercera generación (1965-1980)

Hardware: Circuitos integrados.

Sistema Operativo:

I. S. O. de **multiprogramación**: más de un programa residente en memoria principal al mismo tiempo. (*grado de multiprogramación y overhead*)

Multiprogramación  $\Rightarrow$  Multiproceso  $\nRightarrow$  Multiprogramación



## 2. Evolución y clasificación de los Sistemas Operativos (III)

### 2.1. Evolución histórica de los Sistemas Operativos. (III)

II. S. O. de **multiprocesamiento**: más de un procesador.

III. S. O. **mutiusuario o multiacceso**: permite acceder a varios usuarios a un mismo ordenador mediante *terminales interactivos* (*tiempo de respuesta*).

Multiprogramación  $\nRightarrow$  Multiusuario

IV. S. O. de **tiempo compartido**: multiprogramación, multiusuario e interactivo. El usuario tiene la impresión de que es él el único que trabaja con la máquina.

*Sesión de trabajo*: desde que el usuario se conecta a través de un terminal, hasta que se desconecta del sistema.

## 2. Evolución y clasificación de los Sistemas Operativos (IV)

### 2.1. Evolución histórica de los Sistemas Operativos. (IV)

#### Cuarta generación (1980-1990)

*Hardware:* Microprocesador (computador personal).

*Sistema Operativo:* (para comunicar varias máquinas)

**I. S. O. de red:** el usuario es conciente de la existencia de varias máquinas e indica de forma explícita con cual desea trabajar.

**II.S.O. distribuido:** el usuario no es conciente de las máquinas conectadas entre si (*transparencia*).

## 2. Evolución y clasificación de los Sistemas Operativos (V)

### 2.2. Clasificación de los Sistemas Operativos. (I)

#### Según su entorno de funcionamiento:

- A. Número de usuarios que puede soportar: {  
B. Propósito para el que fueron diseñados: {

♦ S. O. monousuarios

♦ S. O. multiusuarios

♦ S. O. de propósito general

♦ S. O. de propósito específico

✓ S. O. de tiempo real: proveen una respuesta dentro de unos intervalos de tiempo bien definidos a priori.

✓ S. O. de tolerancia a fallos: capaces de detectar errores (hardware y software) y darles solución. Características de diseño:

- ☐ Modularidad,
- ☐ Muerte súbita,
- ☐ Tolerancia a fallos de elementos críticos,
- ☐ Mantenimiento on-line

✓ S. O. virtuales: ejecutan a otros S. O. concurrentemente en la misma máquina.

## 2. Evolución y clasificación de los Sistemas Operativos (VI)

### 2.2. Clasificación de los Sistemas Operativos. (II)

Según su entorno de funcionamiento:

- C. Modo en el que se trabaja con ellos:
  - ♦ *S. O. off-line (batch)*
  - ♦ *S. O. on-line (interactivo)*
- D. Hardware del ordenador:
  - 1. Número de procesadores
    - ♦ *S. O. monoprocesador*
    - ♦ *S. O. multiprocesador*
  - 2. Situación de la memoria principal:
    - ♦ *S. O. centralizado*
    - ♦ *S. O. distribuido*

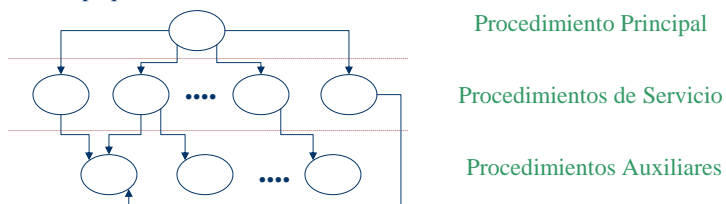
## 3. Estructura de los Sistemas Operativos (I)

### 3.1. Sistemas monolíticos.

**Descripción:** sin estructura definida. Se componen de un conjunto de procedimientos, donde cada uno de ellos puede llamar a todos los demás.

**Proceso para requerir un servicio del S. O.:** el programa de usuario coloca los parámetros necesarios en los lugares previamente definidos, y seguidamente ejecuta una llamada al supervisor (se cambia de modo usuario a modo supervisor), transfiriendo el control al S. O. que examina dichos parámetros y determina el procedimiento que debe invocar. Cuando se complete la llamada devuelve el control al programa de usuario.

S. O. con una pequeña estructura de niveles:



## 3. Estructura de los Sistemas Operativos (II)

### 3.2. Sistemas en estratos.

**Descripción:** se organiza en una jerarquía de estratos, estando construido cada uno de ellos sobre el otro que tiene menor jerarquía que él.

**Ejemplos:**

A. **THE** (Dijkstra, 1968): sistema de procesamiento por lotes.

Operador de THE
Programas de usuario
Administración de E/S
Comunicación entre operador y procesos
Administración de memoria y tambor
Distribución del procesador y multiprogramación

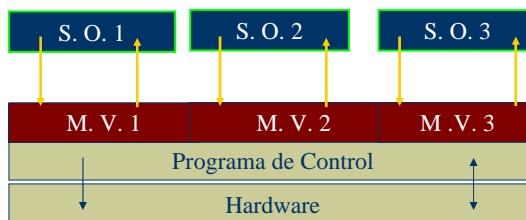
B. **MULTICS**: anillos concéntricos alrededor del hardware, donde los internos son más privilegiados que los externos.

## 3. Estructura de los Sistemas Operativos (III)

### 3.3. Máquinas virtuales.

**Descripción:** crea ilusiones (máquinas virtuales) de la máquina real, permitiendo que en cada máquina virtual se ejecute un S. O. distinto.

El programa de control es el que se ejecuta directamente sobre el propio hardware y ofrece al nivel inmediatamente superior varias máquinas virtuales.

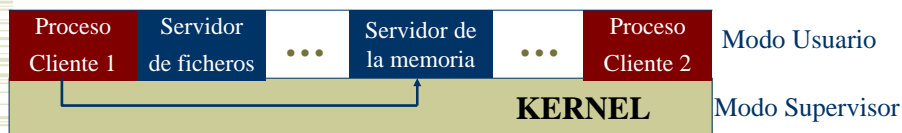


### 3. Estructura de los Sistemas Operativos (IV)

#### 3.4. Modelo Cliente-Servidor.

**Descripción:** su objetivo es minimizar el kernel desplazando el código de todos sus servicios a estratos lo más superiores posibles. Para ello, la mayoría de sus funciones se implementan como procesos de usuario, denominados *procesos servidores*, de forma que cuando un proceso de usuario llamado *proceso cliente*, necesita un servicio del S.O. lo que hace es enviar un mensaje al proceso servidor correspondiente, que realiza el trabajo y devuelve la respuesta.

El kernel lo único que hace es implementar la comunicación entre clientes y servidores y entre servidores y el hardware.



### 3. Estructura de los Sistemas Operativos (V)

#### 3.5. Estructura orientada al objeto.

**Descripción:** se basan en una colección de objetos, donde las funciones del sistema son un tipo de objeto (ficheros, dispositivos, etc). La interacción entre dichos objetos viene determinada por las *capacidades* que cada uno tenga para actuar con el otro.

El kernel es el responsable del mantenimiento de las definiciones de los tipos de objetos soportados y del control de los privilegios de acceso a los mismos. Cuando un programa desee realizar una operación sobre un objeto determinado, deberá ejecutar una llamada al sistema, indicando qué derechos tiene para poder utilizarlo y qué operación intenta llevar a cabo. Como resultado de dicha llamada, el sistema validará la petición y, si puede ser aceptada, permitirá la realización de dicha operación.

