

# Bloque II. Tecnología Básica

TÉCNICO AUXILIAR INFORMÁTICO

## Índice

Tema 1: Informática básica .....	2
Tema 2: Periféricos. ....	8
Tema 3: Tipos abstractos y estructuras de datos .....	11
Tema 4: Sistemas operativos.....	15
Tema 5: Sistemas de gestión de bases de datos. ....	22

## Bloque 2 - Tema 1: INFORMÁTICA BÁSICA

Entendemos por computador a una máquina o dispositivo capaz de realizar de manera automática y siguiendo un programa o secuencia de instrucciones, un determinado procesamiento sobre un conjunto de datos de entrada para obtener unos datos de salida.

Tradicionalmente se ha dividido en 5 generaciones:

**PRIMERA Generación:** tubos de vacío **TERCERA Generación:** circuito integrado  
**SEGUNDA Generación:** transistores. **CUARTA Generación:** microprocesadores  
**QUINTA Generación:** Se basa en investigaciones sobre la computación paralela y distribuida, la optimización de recursos, la miniaturización, el uso del lenguaje natural o la aplicación de IA. Se basan en la arquitectura propuesta por Von Neumann, según la cual tanto los datos como los programas son almacenados en memoria antes de ser utilizados.

### REPRESENTACIÓN Y COMUNICACIÓN DE LA INFORMACIÓN

<b>BIT:</b>	Unidad <b>mínima</b> de información. (cero o uno)	<b>PALABRA:</b> Conjunto de bits que constituyen la unidad de procesamiento en el ordenador. 32 bits = 4GB 64 bits = 16 exabytes
<b>BYTE:</b>	Conjunto de 8 bits, elemento mínimo de memoria direccionable ( <b>octeto</b> ).	<b>NIBBLE:</b> Medio byte (4 bits). Número mínimo para una cifra decimal.
<b>CÚBIT:</b>	Sistema cuántico con <b>dos estados</b> propios ( <b>superposición cuántica</b> = pueden representar unos y ceros al mismo tiempo).	

**Unidades de capacidad** (o tamaño de la memoria): cantidad de información que se puede almacenar. La unidad utilizada es el byte (1 byte = 8 bits) y múltiplos del byte.

En el **Sistema Internacional** (SI) siempre a **potencias de 10**.

$10^{-12}$ pico (p)	$10^6$ mega (M)	$10^{21}$ zetta (Z)	Ejemplos: $10^3$ bytes = 1.000 bytes = 1 Kilobyte (KB) $10^6$ bytes = 1.000 KB = 1 Megabyte (MB) $10^9$ bytes = 1.000 MB = 1 Gigabyte (GB) $10^{12}$ bytes = 1.000 GB = 1 Terabyte (TB)
$10^{-9}$ nano (n)	$10^9$ giga (G)	$10^{24}$ yotta (Y)	
$10^{-6}$ micro ( $\mu$ )	$10^{12}$ tera (T)	$10^{27}$ ronna (R)	
$10^{-3}$ mili (m)	$10^{15}$ peta (P)	$10^{30}$ quetta (Q)	
$10^3$ kilo (K)	$10^{18}$ exa (E)	Nemotécnica: PEZ Y RQ $10^{30}$	

En informática, se indica en **potencias de 2**; con los prefijos definidos en la norma **ISO/IEC 80000-13**.

<b>Byte</b>	1 B = 8 bits	$2^{60}$	1 Exa Byte = 1024 PB
$2^{10}$	1 KB = 1024 byte	$2^{70}$	1 Zetta Byte = 1024 EB
$2^{20}$	1 MB = 1024 KB	$2^{80}$	1 Yotta Byte = 1024 ZB
$2^{30}$	1 GB = 1024 MB	$2^{90}$	1 Bronto Byte = 1024 YB
$2^{40}$	1 TB = 1024 GB	$2^{100}$	1 Geop Byte = 1024 BB
$2^{50}$	1 Peta Byte = 1024 TB		Nemotécnica: PEZ Y BG $2^{100}$

### SISTEMAS DE NUMERACIÓN:

	BINARIO				Oct	Hex.
	Q3	Q2	Q1	Q0		
0	0	0	0	0	0	0
1	0	0	0	1	1	1
2	0	0	1	0	2	2
3	0	0	1	1	3	3
4	0	1	0	0	4	4
5	0	1	0	1	5	5
6	0	1	1	0	6	6
7	0	1	1	1	7	7
8	1	0	0	0	(10)	8
9	1	0	0	1	(11)	9
10	1	0	1	0	(12)	A
11	1	0	1	1	(13)	B
12	1	1	0	0	(14)	C
13	1	1	0	1	(15)	D
14	1	1	1	0	(16)	E
15	1	1	1	1	(17)	F
16					(20)	10

#### Conversión A sistema decimal:

De binario a decimal:  $1101_2 = 1*2^3 + 1*2^2 + 0*2^1 + 1*2^0 = 8 + 4 + 0 + 1 = 13_{10}$

De octal a decimal:  $63_8 = 6*8^1 + 3*8^0 = 48 + 3 = 51_{10}$

De hexadecimal a decimal:  $3AB_{16} = 3*16^2 + A*16^1 + B*16^0 = 768 + 10*16 + 11 = 939_{10}$

Potencias	0	1	2	3	4	5	6	7	8	9	10
de 2	0	2	4	8	16	32	64	128	256	512	1.024
de 8	0	8	64	512	4.096	32.768					
de 16	0	16	256	4.096	65.536	1.048.576					

#### Conversión DESDE sistema decimal:

**a BINARIO** se obtiene efectuando divisiones enteras (sin obtener decimales) por dos.

**a OCTAL** Primero a binario y luego en grupos de 3 de derecha a izquierda.

$$77_{10} = 1001101_2 = 115_8$$

**a HEXA** Primero a binario y luego en grupos de 4 de derecha a izquierda.

$$77_{10} = 1001101_2 = 4D_{16}$$

**REPRESENTACIÓN DE TEXTOS:** Con n bits se podrían codificar  $m = 2^n$  símbolos distintos. Es decir, **con 2 bits** ( $n=2$ ) podemos hacer  $2^2 = 4$  combinaciones (se podrían codificar 4 símbolos distintos  $m=4$ ).

**Código EBCDIC:** creado por IBM utiliza  $n=8$  bits  
 $m = 256$  caracteres

**Código ASCII:** creado por ANSI 7 bits con 128 caracteres  
95 imprimibles y 33 no imprim. izq.

**ASCII extendido:** para usar los 8 bits (1 byte) se añade un 0 a izq.  
01111111

**Unicode:** es un estándar de codificación de caracteres (un total de 137.994) y los principios en los que se basó su diseño son la **universalidad, la eficiencia y la no ambigüedad**. Se puede usar en UTF:

- **UTF-8:** Usa de 1 a 4 bytes por carácter (longitud variable).  
Los 128 primeros son los ASCII

- **UTF-16:** Utiliza 1 o 2 palabras de 16 bits (2 o 4 bytes) por carácter (longitud variable).

- **UTF-32:** Utiliza 32 bits (4 bytes) para codificar un símbolo (longitud fija).

Nota: El ASCII Art emplea caracteres imprimibles de la tabla de ASCII para crear imágenes.

#### FUNCIONES LÓGICAS BÁSICAS:

FUNCIONES LÓGICAS					
A	B	AND &	OR 	XOR ^	NOT
0	0	0	0	0	
0	1	0	1	1	1
1	0	0	1	1	0
1	1	1	1	0	

#### ELEMENTOS CONSTITUTIVOS DE UN SISTEMA DE INFORMACIÓN

Un sistema de información (SI) es un **conjunto de componentes** interrelacionados que recogen, procesan y distribuyen datos e información.

Los componentes del SI son: HW, SW, BD, comunicaciones, personas y procedimientos.

ESTRUCTURA		
Dimensión VERTICAL:		
El nivel <b>OPERACIONAL</b> : decisiones operativas (rutinas).	El nivel <b>TÁCTICO</b> : decisiones tácticas a corto plazo.	El nivel <b>ESTRATÉGICO</b> : decisiones estratégicas a medio y largo plazo.
Dimensión HORIZONTAL:		
cada nivel se divide en subsistemas (especialmente en el operativo y táctico).		

#### ARQUITECTURA DE ORDENADORES

**1. Organización del procesador:** la función principal de un procesador (CPU) es **ejecutar instrucciones**, los procesadores **operan según una onda cuadrada periódica** con una determinada frecuencia conocida como **señal de reloj**.

COMPONENTES DEL PROCESADOR (CPU)		
<b>Unidad aritmética y lógica o ALU:</b> circuito que hace un conjunto de operaciones aritméticas y lógicas con los datos almacenados dentro del procesador.	<b>Unidad de control o UC:</b> circuito que controla el funcionamiento de todos los componentes del procesador. Controla también las operaciones de la ALU.	<b>Conjunto de registros:</b> unidad de almacenamiento temporal de datos e instrucciones dentro de procesador.

MICROPROCESADOR	
El microprocesador <b>suele estar integrado en el mismo chip</b> del procesador.	
<b>Unidades de ejecución SIMD:</b> unidades especializadas en la ejecución de instrucciones ED vectoriales (Ej.: multimedia).	<b>Memoria caché:</b> prácticamente todos los procesadores incorporan algunos niveles de memoria caché.
<b>Unidad de gestión de memoria (MMU):</b> gestiona el espacio de direcciones virtuales, traduciendo las direcciones de memoria virtual a física.	<b>Unidad de punto flotante o (FPU):</b> unidad especializada en hacer operaciones en punto flotante; puede funcionar de manera autónoma, ya que dispone de un conjunto de registros propio.
INSTRUCCIONES	
Son autocontenidas (incluyen toda la información necesaria para su ejecución).	
<b>Código de operación:</b> especifica la operación que hace la instrucción.	<b>Operando fuente:</b> para hacer la operación pueden ser necesarios uno o más operandos fuente
<b>Operando destino:</b> almacena el resultado de la operación realizada. Uno de los operandos fuente se puede utilizar como operando destino.	<b>Dirección de la instrucción siguiente:</b> especifica dónde está la instrucción siguiente que se debe ejecutar.

TIPOS DE ARQUITECTURA DEL JUEGO DE INSTRUCCIONES		
<b>Pila:</b> los operandos son implícitos y se encuentran en la pila.	<b>Acumulador:</b> uno de los operandos se encuentra de manera implícita en un registro denominado <i>acumulador</i> .	<b>Registros:</b> los operandos se encuentran siempre de manera explícita. Se divide en <b>tres subtipos</b> :
<b>Registro-registro (o load-store):</b> Solo pueden acceder a la memoria instrucciones de carga (load) y almacenamiento (store).	<b>Registro-memoria:</b> Cualquier instrucción puede acceder a la memoria con uno de sus operandos.	<b>Memoria-memoria:</b> Memoria-memoria. Cualquier instrucción puede acceder a la memoria con todos sus operandos.

Ciclo de ejecución de instrucciones	
Son autocontenidas, es decir, incluyen toda la información necesaria para su ejecución.	
<b>Fase 1:</b> Lectura de la instrucción.	<b>Fase 2:</b> Ejecución de la instrucción y almacenamiento del operando de destino.
<b>Fase 2:</b> Lectura de los operandos fuente.	<b>Fase 4:</b> Comprobación de interrupciones.

La **segmentación de las instrucciones (pipeline)** consiste en **dividir** el ciclo de ejecución de **las instrucciones en** un conjunto de **etapas con el fin de ejecutar simultáneamente diferentes etapas** lo cual permite **aumentar el rendimiento** del procesador. Estas etapas pueden coincidir o no con las fases del ciclo de ejecución de las instrucciones.

REGISTROS	
Son elementos de memoria de acceso rápido. Se implementan utilizando memoria RAM estática.	
<b>Registros de propósito general:</b> son los que suelen utilizarse como operandos en las instrucciones del ensamblador. También tenemos el <b>registro "acumulador"</b> , donde se almacenan temporalmente los datos que serán tratados por la ALU.	<b>Registros de acceso a memoria:</b> <ul style="list-style-type: none"> <li>- <b>Memory Address Register (MAR):</b> registro de direcciones de memoria (<b>dirección de memoria a la que queremos acceder</b>).</li> <li>- <b>Memory Buffer Register (MBR):</b> registro de datos de memoria; registro donde la memoria deposita el <b>dato leído o el dato que queremos escribir</b>.</li> </ul>
<b>Registros de instrucción:</b> <ul style="list-style-type: none"> <li>- <b>Program counter (PC):</b> registro contador del programa, contiene la dirección de la instrucción <b>siguiente que hay que ejecutar</b>.</li> </ul>	<ul style="list-style-type: none"> <li>- <b>Instruction register (IR):</b> registro de instrucción contiene la instrucción <b>se está ejecutando</b>.</li> </ul>
<b>Registros de estado y de control:</b> <ul style="list-style-type: none"> <li>- <b>Bit de cero:</b> si el resultado obtenido es 0.</li> <li>- <b>Bit de transporte:</b> si en el último bit que operamos en una operación aritmética se produce transporte; también puede deberse a una operación de desplazamiento.</li> <li>- <b>Bit de desbordamiento:</b> si la última operación ha producido un resultado que no se puede representar en el formato que estamos utilizando.</li> <li>- <b>Bit de signo:</b> si el resultado obtenido es negativo.</li> <li>- <b>Bit de interrupción:</b> indica si las interrupciones están habilitadas o inhibidas.</li> </ul>	
<ul style="list-style-type: none"> <li>- <b>Bit de modo de operación:</b> indica si la instrucción se ejecuta en modo supervisor o en modo usuario.</li> <li>- <b>Nivel de ejecución:</b> indica el nivel de privilegio de un programa en ejecución. Un programa puede desalojar el programa que se ejecuta actualmente si su nivel de privilegio es superior.</li> <li>- Los <b>registros de control</b> son los que dependen más de la organización del procesador.</li> </ul>	

## 2. Arquitectura Von Neumann

Es una arquitectura de computadoras basada en la descrita en 1945 por el matemático y físico John von Neumann. Este describe que **la CPU está compuesta por la UC, la ALU y los registros**.

**UC (UNIDAD DE CONTROL):** Contiene la instrucción que se está ejecutando en ese momento. Se encarga de generar las señales de control para la ejecución de la instrucción. Contiene un registro de instrucciones y un contador de programa, una memoria para almacenar tanto datos como instrucciones, almacenamiento externo, y mecanismos de E/S.

Tipos de unidades de control	
<b>UC CABLEADA:</b> <b>conjunto de señales de salida</b> que son las señales de control. En <b>máquinas RISC</b> (más rápida y costosa).	<b>UC MICROPROGRAMADA:</b> conjunto de microinstrucciones, necesita entorno de desarrollo. En <b>máquinas CISC</b> (más simple y económica).

**ALU (UNIDAD ARITMÉTICO LÓGICA):** Ejecuta las operaciones aritméticas que le señala la instrucción residente en la unidad de control.

ALU: operaciones que puede hacer	
es un circuito combinacional capaz de realizar operaciones aritméticas y lógicas	
<b>Números enteros:</b> Se representan en binario, normalmente en 32 o 64 bits.	<b>Números reales:</b> en punto fijo y en punto flotante
Notación en punto flotante <b>según la IEEE:</b>	
<ul style="list-style-type: none"> <li>- Precisión simple: números binarios en 32 bits</li> <li>- Doble precisión: 64 bits</li> <li>- Cuádruple precisión: 128 bits</li> </ul>	

**REGISTROS DE LA CPU:** Almacenan temporalmente (en la CPU) resultados y datos.

<b>Registro Contador de Programa (PC):</b> La <b>siguiente instrucción</b> .	<b>Registro de instrucción (IR):</b> La <b>que se está ejecutando</b> en ese momento.
<b>Registro de estado.</b>	<b>Registro de direcciones de memoria (MAR).</b>
<b>Registro de datos de memoria (MBR).</b>	

**MP (MEMORIA PRINCIPAL):** Almacena las instrucciones de los programas con celda de i bit (0 y 1).

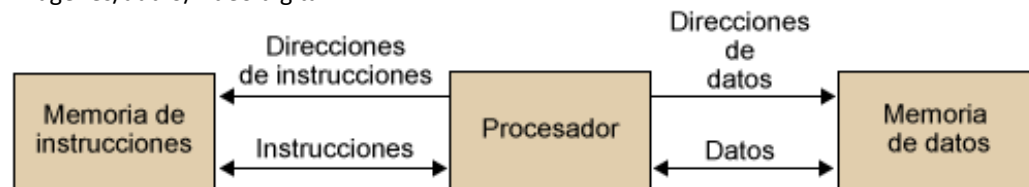
**ENTRADA/SALIDA:** dispositivos de entrada y salida

**BUSES:** líneas eléctricas para la comunicación entre los distintos elementos del computador.

De <b>DATOS</b> (lectura / escritura)	De <b>DIRECCIÓN</b> (origen / destino)	De <b>CONTROL</b> (acceso / uso)
---------------------------------------	--	----------------------------------

El diseño de una arquitectura von Neumann es más simple que la arquitectura Harvard.

**3. Arquitectura Harvard:** es más moderna, y se distingue del anterior por la división de la memoria en una memoria de instrucciones y una memoria de datos, de manera que el procesador puede acceder separada y simultáneamente a las dos memorias. No se utiliza habitualmente en computadores de propósito general, se utiliza en computadores para aplicaciones específicas, como IoT, móviles, procesamiento imágenes/audio/video digital.



**4. Taxonomía de Flynn:** se basa en el flujo que siguen los datos dentro de la máquina y de las instrucciones sobre esos datos.

<b>SISD</b> (Single Instruction stream, Single Data stream): un flujo de instrucciones sobre un flujo único de datos. Concepto de arquitectura serie de Von Neumann	<b>SIMD</b> (Single Instruction, Multiple Data): un único flujo de instrucciones que operan sobre múltiples flujos de datos. Usado en máquinas vectoriales
<b>MISD</b> (Multiple Instruction, Single Data): múltiples instrucciones sobre un único flujo de datos.	<b>MIMD</b> (Multiple Instruction, Multiple Data): múltiples instrucciones operan sobre múltiples datos.
<b>MPP</b> (Massively Parallel Processor) o <b>MMD</b> (Multiprocesador de Memoria Distribuida): distribuida en submódulos de memoria en cada uno de los procesadores individuales. Además, se añade un submódulo de E/S a los procesadores.	

**5. Arquitectura CISC vs RISC:** el de diseño de la arquitectura del procesador pueden ser:

<b>Computadores CISC</b> (Complex Instruction):	<b>Computadores RISC</b> (Reduced Instruction):
- Instrucción es de longitud variable	- Instrucción es de tamaño fijo y corto (fácil y rápido)
- Gran juego de instrucciones	- Arquitectura tipo load-store o registro-registro
- La unidad de control es microprogramada	- Pocos ciclos de reloj
- Dificulta el procesamiento simultáneo de instrucciones.	- La unidad de control es cableada
	- Segmentación simultánea de instrucciones

Los procesadores actuales no son completamente CISC o RISC. Son híbridos.

**Arquitectura ARM:** arquitectura avanzada para microprocesadores RISC. Bajo consumo energético. Uso generalizado en dispositivos móviles.

## COMPONENTES INTERNOS DE LOS EQUIPOS MICROINFORMÁTICOS

**PLACA BASE:** dónde se insertan los demás componentes del PC, proporcionando las líneas eléctricas necesarias para las transferencias de datos.



**Chipset:** conjunto de circuitos integrados para coordinar la transferencia de datos entre los distintos componentes del ordenador.

**Puente norte (Northbridge):** gestiona la interconexión entre el microprocesador, la memoria RAM y la unidad de procesamiento gráfico.

**Puente sur (Southbridge):** gestiona la interconexión entre los periféricos y los dispositivos de almacenamiento.

Por requisitos de rendimiento, el Northbridge fue absorbido en la propia CPU y conectado punto a punto con el Southbridge. Esta arquitectura es la denominada, según Intel, PCH (Platform Controller Hub) existente en la actualidad. El Southbridge cambia de nombre a ICH (I/O Controller Hub).

**BIOS:** almacenado en un módulo de memoria de tipo ROM. Se encarga de verificar (chequear) el funcionamiento del hardware. Ahora es UEFI.

**Zócalo del Procesador:**

**Slot:** zócalo de microprocesador inserción vertical. **Socket:** zócalo cuadrangular o rectangular.

Tipos de sockets:

<b>LGA:</b> pines incorporados en la propia estructura del socket. LGA 1200 (2020), LGA 1700 (2021). Número: cantidad de pines.	<b>PGA:</b> pines incorporados al procesador directamente. Socket AM4 (PGA 1331), TR4, etc	<b>BGA:</b> no se utilizan pines.
---	--	-----------------------------------



Para insertar los procesadores se ha creado un concepto llamado ZIF (Fuerza de inserción nula).

**Ranuras de Memoria RAM:** Pueden ser de tipo SIMM, DIMM o SODIMM.

#### **Ranuras de Expansión:**

- ISA (Arquitectura estándar industrial): permiten insertar ranuras ISA.
- PCI (Interconexión de componentes periféricos): se utilizan para conectar tarjetas PCI.
- PCI Express o PCIe (Interconexión de componentes periféricos rápida).
- AGP (Puerto gráfico acelerado): es un puerto rápido para tarjetas gráficas.
- M.2: para conectar discos SSD. y U.2: similar a M.2, pero ocupa más espacio en la placa base.
- SATA (serial ATA): utilizado para conexión de discos duros de 2,5" o 3,5".

**Reloj de Tiempo Real (Rtc):** sincronizar las señales del sistema.

**Pila CMOS:** alimentación eléctrica del RTC y almacena algunos datos del sistema, como la hora, la fecha y algunas configuraciones esenciales del sistema.

**Conectores de Entrada y Salida:** Puertos serie, USB, IDE, SATA, RJ45, VGA, HDMI, etc.

#### **MICROPROCESADORES:**

**Proceso de fabricación:** especifica cuál es el tamaño de los transistores integrados en el procesador cuando se fabrica. Este valor se expresa en nanómetros (nm).

**Ley de Moore:** El número de transistores de un chip se duplica cada dos años.

**Numero de nucleos (cores):** gracias a la reducción de las litografías, se pueden integrar cada vez más núcleos dentro de los procesadores.

**Ley de Amdahl:** La mejora obtenida en el rendimiento de un sistema debido a la alteración de uno de sus componentes está limitada por la fracción de tiempo que se utiliza dicho componente.

**Número de hilos (threads):** el multithreading en un procesador (Hyper-Threading en Intel o SMT -Simultaneous Multi-Threading- en AMD) significa que cada uno de los núcleos físicos es capaz de realizar dos tareas, de manera simultánea (conurrencia). Permite simular dos núcleos lógicos para cada núcleo físico (un procesador de cuatro núcleos físicos con multithreading tendría ocho hilos de proceso, y sería capaz de ejecutar ocho órdenes al mismo tiempo).

#### **Tipos de procesadores familia Intel:**

- Core: i3, i5, i7, i9 y X. Destinado a equipos de sobremesa y portátiles.

- Xeon: E, W y D. Destinado a servidores.

- Xeon escalable: Bronze, Platinum, Gold y Silver. Destinado a servidores de alto rendimiento.

- Celeron: G, J y N. Destinado a equipos con pocas prestaciones y dispositivos móviles.

- Atom: C y P. Destinado a dispositivos móviles e IoT.

Generaciones y nombre en clave de los últimos procesadores Intel:

8ª gen.: Coffee Lake	11ª gen.: Tiger Lake	14ª gen.: Raptor Lake Refresh
9ª gen.: Coffee Lake Refresh	12ª gen.: Alder Lake	15ª gen.: Arrow Lake
10ª gen.: Comet Lake	13ª gen.: Raptor Lake	

**Chips M1 y M2:** En mayo de 2021 Apple lanzó el chip M1, específicamente para Mac.

	M1 (2021)	M2 (2022)	M3 (2023)	M4 (2024)
CPU	8 núcleos	8 núcleos	8 núcleos	10 núcleos
GPU	8 núcleos	10 núcleos	10 núcleos	10 núcleos
Transistores	16 billones	20 billones	25 billones	28 billones
Proceso de fabricación	5nm	5nm	3nm	3nm
Rendimiento	2,6 TFLOPS	3,6 TFLOPS	4,1 TFLOPS	4,6 TFLOPS

#### **RENDIMIENTO: Sistemas de medida**

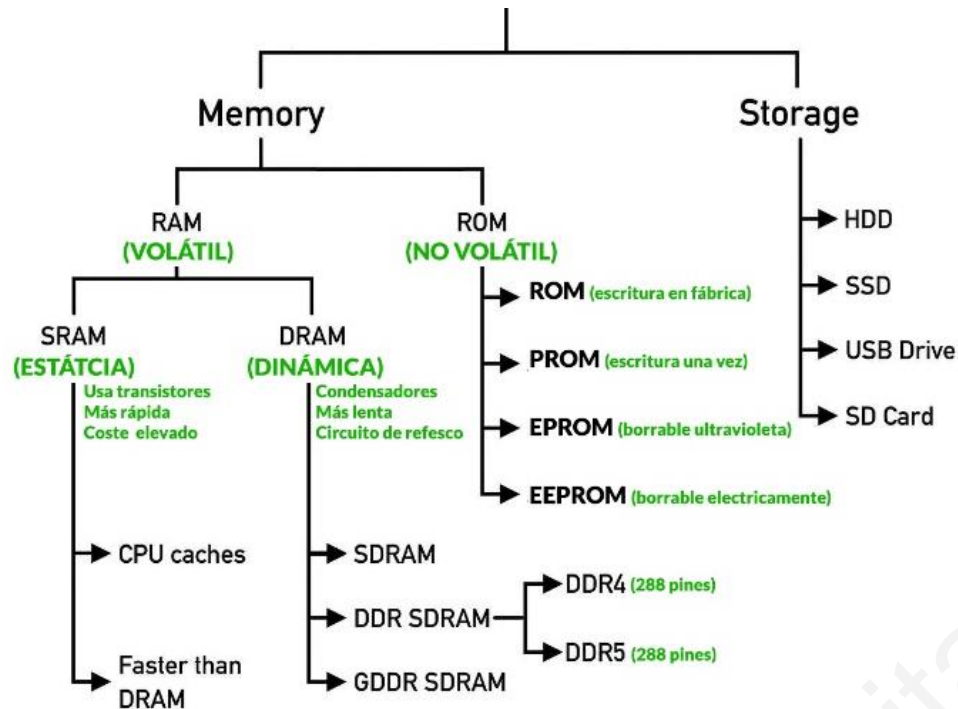
**MIPS:** cuántos millones de instrucciones por segundo es capaz de ejecutar un procesador.

**MFLOPS:** mide cuantos millones de instrucciones en coma flotante por segundo.

**SPEC:** ejecuta en el procesador un conjunto de programas.

**Productividad (throughput):** número de tareas realizadas por unidad de tiempo.

**MEMORIA INTERNA:** formada típicamente por dos niveles fundamentales: memoria caché y memoria principal. El elemento básico en toda memoria es la celda. Una celda permite almacenar un bit, un valor 0 o 1



**Memoria FLASH:** el borrado es eléctrico, el proceso de borrar y grabar es muy rápido.

**Memoria caché:** se sitúa entre la memoria principal y el procesador, diseñadas para reducir el tiempo de acceso a la memoria, utilizando tecnología SRAM. Los procesadores modernos utilizan diferentes niveles de memoria caché, lo que se conoce como memoria caché de primer nivel, segundo nivel y tercer nivel.

- **Caché L1** es el primer lugar donde el microprocesador buscará información. Es la caché más pequeña y más rápida.

- **Caché L2** suele ser de mayor tamaño que la caché L1, pero es algo más lenta. Sin embargo, es la que mayor impacto tiene en el rendimiento.

- **Caché L3** es de mucho más tamaño que las anteriores, y generalmente se comparte entre todos los núcleos del procesador.

La organización de la memoria caché puede ser:

- **Directa:** un bloque de la memoria principal solo puede estar en una única línea de la memoria.

- **Totalmente asociativa:** un bloque de la memoria principal puede estar en cualquier línea.

- **Asociativa por conjuntos:** un bloque de la memoria principal puede estar en un subconjunto de las líneas de la memoria caché, pero dentro del subconjunto puede encontrarse en cualquier posición.

#### Algoritmos de reemplazo (tras un fallo y se tiene que llevar a la memoria caché)

<b>FIFO</b> (first in first out): para elegir la línea se utiliza una cola	<b>LRU</b> (least recently used): este algoritmo elige la línea que hace más tiempo que no se utiliza.
<b>LFU</b> (least frequently used). En este algoritmo se elige la línea que hemos utilizado menos veces.	<b>Aleatorio:</b> reemplazar al azar.

#### Proceso de arranque de un computador

**Paso 1. Encendido del PC (power on):** Conectar la fuente de alimentación y suministrar corriente.

**Paso 2. BIOS (UEFI):** El microprocesador ejecuta la BIOS, la cual toma el control y ejecuta el POST.

**Paso 3. POST:** El POST (Power On Self Test) hace un chequeo de todos los componentes.

**Paso 4. Inicio adaptador de vídeo** y habilitación de dispositivos

**Paso 5. Carga del gestor de arranque** (boot manager) y cede el control al sistema operativo



## Bloque 2 - Tema 2: PERIFÉRICOS



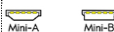


Se entenderá por periférico a todo conjunto de dispositivos que permitan realizar **operaciones complementarias de entrada/salida (E/S)** al proceso de datos que realiza la CPU. Están compuestos por una **parte mecánica (relés, conmutadores, etc.)** y una **parte eléctrica** (circuitos electrónicos) **dominada controlador E/S**.

### PUERTOS O CONECTORES:

Puerto	Norma	Puerto	Norma
SERIE / COM	RS-232	PARALELO / LPT	IEEE 1284

Puerto USB (Universal Serial Bus): estándar desarrollado por distintos fabricantes				
	Velocidad		Velocidad	Nombre comercial
USB 1.0	1,5 Mb/s	USB 3.0	4,8 Gb/s	3.2 Gen 1
USB 1.1	12 Mb/s	USB 3.1	10 Gb/s	3.2 Gen 2
USB 2.0	480 Mb/s	USB 3.2	20 Gb/s	3.2 Gen 2x2
		USB 4.0	40 Gb/s	USB Tipo C

USB On-The-Go o **USB OTG** es una extensión de **USB 2.0** que permite que los dispositivos **actúen en modo host o servidor** (se le puedan conectar otros dispositivos) o en modo dispositivo.

Conector	Puerto		Conector	Puerto	
USB Tipo A	USB 1.0 - 3.1		USB Tipo B	USB 1.0 - 3.1	
Mini USB	USB 1.0 - 2.0		Micro USB	USB 1.0 - 3.1	
USB Tipo C	USB 3.2 y 4.0 Thunderbolt 3 y 4	<b>24 pines</b> 40 Gb/s			

Puerto THUNDERBOLT (antes Light Peak): estándar para la transmisión de datos y energía.					
Thunderbolt 1	10 Gb/s	DisplayPort.	Thunderbolt 3	40 Gb/s (bidireccional)	USB-C
Thunderbolt 2	20 Gb/s	DisplayPort.	Thunderbolt 4	40 Gb/s (2 monitores 4K)	USB-C

Puerto FIREWIRE: norma IEEE 1394 (cámaras y videocámaras)			
Firewire 400	400 Mb/s	Firewire s1600	1,6 Gb/s
Firewire 800	800 Mb/s	Firewire s3200	3,2 Gb/s

Puerto DVI (Digital Visual Interface): para monitores y proyectores		
DVI-D: señal digital	DVI-A: señal analógica	DVI-I: señales digital y analógica

Puerto HDMI (High Definition Multimedia Interface): audio y video sin comprimir			
HDMI 1.0	4,9 Gb/s Full HD a 60Hz	HDMI 2.0	18 Gb/s - 4K a 60Hz
HDMI 1.4	4K a 24Hz	HDMI 2.1	48 Gb/s - 10K a 60Hz
DisplayPort (DP): creado por VESA, libre de licencias, video y audio (opcional).			
DP 1.0	10,8 Gb/s y 1440p	DP 1.4	32,4 Gb/s y 8K
DP 1.2	21,6 Gb/s y 4K	DP 2.0	80 Gb/s y 16K
DP 1.3	32,4 Gb/s y 8K		

- **SATA**: bus serie dedicado para la transmisión de datos de dispositivos de almacenamiento.
- **Puerto AGP** (Accelerated Graphics Port): para conexión de una tarjeta gráfica.
- **PCI** (Peripheral Component Interconnect): permite conectar periféricos a la placa base.

PCI Express (Peripheral Component Interconnect Express) o PCIe			
1.0	2,5 Gb/s	5.0	32 Gb/s
2.0	5 Gb/s	6.0	64 Gb/s
3.0	8 Gb/s	7.0	128 Gb/s
4.0	16 Gb/s	bus serie de alta de hasta 16 carriles	

- **Ethernet**: para la conexión a una red LAN. Conector RJ-45.

**Plug and play o PnP** es una funcionalidad por la que un ordenador es capaz de **detectar y configurar de forma automática** (sin intervención humana) un periférico.

**Hot plug** es una característica que permite que unos **dispositivos pueden ser conectados y desconectados** al ordenador **sin apagar** el mismo.

**ELEMENTOS DE IMPRESIÓN**: La impresora es el dispositivo que el PC emplea para presentarnos en papel información de salida, básicamente texto, gráficos y fotografías.

- **Velocidad**: se mide en número de **caracteres por segundo (cps)** o **páginas por minuto (ppm)**.
- **Resolución**: medida en **puntos por pulgada (ppp)** o dots per inch (**dpi**), puntos por pulgada.
- **Buffer de memoria**: almacena la **información en espera** de ser impresa.
- **Interfaz de conexión**: normalmente **puerto USB** y anteriormente el puerto paralelo o **LPT1**.

<b>Impresoras MATRICIALES</b> : son <b>impresoras de impacto</b> (de margarita o de agujas).	<b>Impresoras MULTIFUNCIÓN</b> : combinan capacidades de impresión y escaneo en una sola máquina.
--	---

**Impresoras de INYECCIÓN DE TINTA:** bajo costo y buena calidad. El sistema denominado DOD (Drop On Demand, **Inyección Bajo Demanda**), que produce pequeñas gotas cuando se necesitan. Método **Térmico o Piezoeléctrico**.

**Impresoras LÁSER:** destacan en rapidez y un coste más bajo por página. Hace uso de cuatro depósitos de tinta en polvo (**tóner**) con cuatro colores básicos: negro, cian, magenta y amarillo sobre un dispositivo en forma de cilindro denominado tambor.

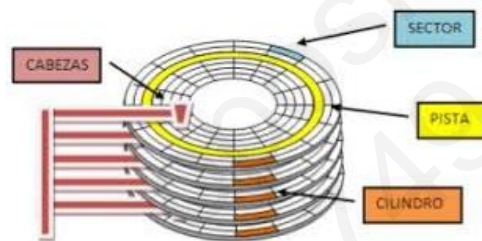
**Impresoras 3D:** en función de la tecnología destacan

Estereolitografía (SLA)	Sinterización Selectiva por Láser (SLS)	Inyección
Estereolitografía de Baja Fuerza (LFS)	Procesamiento digital de luz (DLP)	Fabricación continua de fibras (CFF)
Deposición de material fundido (FDM o FFF)		

**ELEMENTOS DE ALMACENAMIENTO:** Estos periféricos se encargan de guardar o salvar los datos.

**Los Discos Duros (Hard Disk,** abreviado con frecuencia HD o HDD) generalmente utilizan un sistema de grabación magnética digital. Actualmente incorporan platos de 3,5 pulgadas o 2,5 pulgadas para ordenadores portátiles.

- **Capacidad de almacenamiento:** cantidad de información que puede grabarse o almacenar
- **Velocidad de rotación:** velocidad a la que giran los platos. En RPM (Revoluciones Por Minuto).
- **Memoria caché** (tamaño del buffer): los datos se almacenan primeramente en el buffer.
- **Latencia:** Tiempo medio que tarda la aguja en situarse en el sector deseado.
- **Tasa de transferencia:** la cantidad de datos que se puede leer o escribir en la parte más exterior del disco o plato en un período de un segundo. Normalmente se mide en Mbit/s.



**Los componentes físicos** de un disco duro:

- Los discos (Platters)
- Las Cabezas (Heads)
- Eje: actúa como soporte
- Pistas: círculos concéntricos
- Sectores: tamaño de un sector
- Cilindros: lados opuestos del disco
- Tarjeta controladora (Firmware)

#### Procesos de codificación de la información

<b>FM</b> (Frequency Modulation – Modulación de Frecuencia)	<b>MFM</b> (Modify Frequency Modulation – Modulación de Frecuencia Modificada)	<b>RLL</b> (Run Length Limited – Longitud de recorrido limitado)
---	--	--

#### Interfaz de un disco duro:

- **IDE** (Integrated Drive Electronics, Dispositivo electrónico integrado) o **PATA** (Parallel Advanced Technology Attachment).

**SATA (Serial ATA):** Utiliza un bus serie dedicado más rápido y eficiente que IDE o PATA

<b>SATA 1.0</b>	1,5 Gb/s (150 MB/s).	<b>SATA 3.0</b>	6 Gb/s (600 MB/s).
<b>SATA 2.0</b>	3 Gb/s (300 MB/s).	<b>Mini SATA o mSATA</b>	1,5 Gb/s y 3 Gb/s.
<b>SATA Express o SATA 3.2</b> admite dispositivos tanto PCI Express como SATA			

- **SCSI** (Small Computer System Interface): preparadas para discos de gran capacidad de destinada a servidores y alto rendimiento. Hasta 80 MB/s, 16 dispositivos simultáneos.

- **SAS** (Serial Attached SCSI): Interfaz de transferencia de datos en serie, sucesor de SCSI, destinado al ámbito empresarial hasta 24 Gb/s. SATA es compatible en SAS pero no a la inversa.

**Las unidades SSD** (Solid State Drive o estado sólido) almacenan la información en memorias flash basadas en NAND interconectadas entre sí. Suelen usar SATA, SAS o PCI Express (PCIe). Por último, **M.2** es una especificación que permite conectar unidades SSD de menor tamaño (formato reducido) mediante PCIe, haciendo uso de NVMe (Non Volatile Memory Express).

**ELEMENTOS DE VISUALIZACIÓN / MONITORES:** La imagen formada en la pantalla del ordenador tiene una unidad elemental llamada píxel.

**El tamaño de punto (dot pitch)** es el espacio entre los dos puntos más cercanos medidos desde su centro. Cuanto menor sea esa distancia mayor es la nitidez.

**La resolución** indica el número total de píxeles representados en una pantalla (horizontal y vertical). Una resolución de 1920x1080 píxeles se abrevia en solo 1080p (1080 líneas de resolución vertical), la "p" en 1080p hace referencia a la exploración PROGRESIVA o escaneo progresivo

**La tasa de refresco:** 400 Hz equivalen a 400 barridos por segundo.

**Las dimensiones** están representadas en pulgadas (inch). 1" = 2,54 centímetros.

#### Resumen formatos de resolución

<b>SD</b> (Standar Definition)	640 x 480 (Estándar VGA)	<b>UHD</b> (Ultra High Definition)	2160p, 4K
<b>HD</b> (High Definition)	1280 x 720 (720p)	<b>UHD 8K</b>	4320p, 8K
<b>FHD</b> (Full HD)	1920 x 1080 (1080p, 2K)	<b>UHD 16K</b>	8640p, 16K

<b>Monitor de rayos catódicos (CRT)</b>	<b>Monitor LED</b> (Light-Emitting Diode)
<b>Monitor de cristal líquido (LCD):</b> Liquid Crystal Display	<b>Monitor OLED</b> (Organic Light-Emitting Diode): más precisas con una calidad de brillo superior y mejor contraste
<b>Monitor de plasma</b>	<b>Monitor AMOLED</b> (Active-Matrix Organic Light-Emitting Diode): OLED con matriz activa con negros serán reales.
<b>Monitor TFT LCD:</b> variante del LCD para aumentar su calidad.	<b>Monitor QLED</b> (Quantum Dot LED): son capaces de activar un píxel de manera individual
<b>Pantallas táctiles</b>	
<b>Resistivas:</b> dos capas de película plástica.	<b>Infrarrojas:</b> uso de emisores y receptores de rayos infrarrojos. <b>De onda acústica:</b> el dedo absorbe una parte de la potencia acústica.
<b>Capacitivas:</b> panel de vidrio recubierto con un material conductor transparente. Aprovecha de la capacidad del cuerpo humano de conducir electricidad	

**ELEMENTOS DE ENTRADA / TECLADO:** definimos una distribución de teclado como la ubicación de las teclas.

- **QWERTY** o distribución Scholes: la separación de teclas que más se utilizan juntas, España, Reino Unido, países nórdicos y América son los países y regiones donde se utiliza esta distribución.
- **QWERTZ:** cambia la Z por la Y respecto a QWERTY. Alemania y los países del este de Europa.
- **AZERTY:** Francia y las regiones francófonas son las zonas que hacen uso de esta distribución.
- **Dvorak:** objetivos ergonómicos y aumentar la velocidad de escritura. Estándar ANSI.

**ELEMENTOS DE DIGITALIZACIÓN / ESCÁNER:** en general, se puede decir que existen tres tipos:

- **Escáneres de cama plana:** permiten escanear colocando el documento de cara al panel de vidrio.
- **Escáneres de mano:** deben desplazarse de forma manual por el documento.
- **Escáneres con alimentador de documentos:** de manera similar a las máquinas de fax.
- **Escáneres de tambor:** dirigido a la industria gráfica. Gran calidad, costosos y lentos.
- **Escáneres cenital:** para copias digitales de libros o documentos viejos sin deteriorarlos.

Características:

- **Resolución:** puntos por pulgada, dpi o ppp.
- **Resolución interpolada:** mediante la estimación matemática (interpolación) calculada.
- **Velocidad de captura:** páginas por minuto (ppm).
- **Profundidad del color:** es el número de colores de cada píxel de la imagen en bits.
- **Interfaces de E/S:** USB, SCSI, LPT, ethernet.
- **Controlador:** TWAIN, ISIS, SANE o WIA.
- **OCR** (Optical Character Recognition) es un software de reconocimiento de texto.

<b>Sensores CCD</b> (Charge-Coupled Devices): utilizan una luz de xenón que emite un haz de luz, la cual rebota y mediante un espejo. Entornos profesionales.	<b>Sensores CIS</b> (Contact Image Sensor): el haz de luz se redirecciona hacia una fila de sensores acompañados de LED. Uso doméstico.
---	---

#### Modelos de color

<b>RGB:</b> rojo (R), verde (G) y azul (B) Valores del 0 al 255.	<b>CMYK:</b> cian (C), magenta (M), amarillo (Y) y negro (K). Valores 0% a 100%.	<b>HSL:</b> matiz (H) en grados, saturación (S) y luminosidad (L) en porcentaje.
<b>HSV:</b> matiz (H), saturación (S) y valor (V).	<b>Escala de grises:</b> se mide solo por el componente luminosidad, en valores de 0 a 255.	

**Norma Técnica de Interoperabilidad (NTI) Digitalización de documentos:** El Esquema Nacional de Interoperabilidad (ENI) se define en el artículo 156 de la Ley 40/2015 del RJSP las condiciones necesarias para garantizar el adecuado nivel de interoperabilidad técnica, semántica y organizativa de los sistemas y aplicaciones empleados por las Administraciones públicas.

**Definimos digitalización** como el **proceso tecnológico** que permite **convertir un documento en soporte electrónico** que contienen la **imagen codificada, fiel e íntegra** del documento.

El **nivel de resolución mínimo** para imágenes electrónicas será de **200ppp**.

**Proceso de digitalización:** Digitalización > optimización > asignación de metadatos (obligatorios y opcionales) > firma (si procede).

## Bloque 2 - Tema 3: TIPOS ABSTRACTOS Y ESTRUCTURAS DE DATOS. ORGANIZACIONES DE FICHEROS. ALGORITMOS. FORMATOS DE INFORMACIÓN Y FICHEROS.

### TIPOS ABSTRACTOS Y ESTRUCTURAS DE DATOS

Un **Tipo de Datos (TD)** es el conjunto de valores que puede tomar una variable. Se clasifican en dos:

**Dato elemental:** almacenados en cualquier soporte de información

Numéricos (0, 1, 2, 3, ..., 9)    Alfabéticos (a, b, c, ..., x, y, z)    Especiales (¿, ?, =, /, &, %, etc.)

**Tipos compuestos:**

Arrays, listas, árboles, grafos, registros y ficheros.

Un **Tipo Abstracto de Datos (TAD)** representa una abstracción donde se destacan los detalles de la especificación (el qué) y se ocultan los detalles de la implementación (el cómo). Podemos definir un TDA **Lista**, **Árbol**, **Pila** o **Cola**, independientemente de las estructuras de datos usadas para su implementación.

Una **Estructura de Datos (ED)** es una forma de organizar un conjunto de datos elementales con el objetivo de facilitar la manipulación de estos datos como un todo o individualmente, es decir, es la implementación de un TAD. Clasificación de estructuras de datos:

#### 1ª CLASIFICACIÓN: según cómo se ubiquen en la memoria del ordenador

**Contiguas:** representarse en el hardware del ordenador, situando sus datos en áreas adyacentes de memoria.  
Ejemplo: cadenas, arrays, vectores, registros.

**Enlazadas:** los datos se relacionan unos con otros mediante punteros. La localización del dato no es inmediata.

#### 2ª CLASIFICACIÓN: Según la variabilidad de su tamaño durante la ejecución de un programa

**Estáticas:** el tamaño ocupado en memoria se define con anterioridad.  
Ejemplo: array o matriz.

**Dinámicas:** pueden crecer en tamaño.  
Ejemplo: lista o árbol.

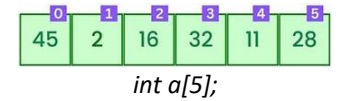
#### 3ª CLASIFICACIÓN: Según los tipos de datos de contengan:

**Homogéneas:** únicamente se utiliza un tipo de datos.

**Heterogéneas:** sus elementos puedan ser de tipos de datos distintos.

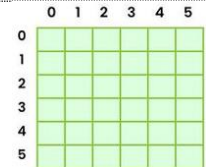
### Estructura de Datos

**Vectores o Array unidimensional:** Colección de elementos homogéneos (de igual longitud y tipo). El índice comienza en 0 y termina en tamaño-1.



`int a[5];`

**Matriz o Arrays bidimensional:** estructurados en filas (i) y columnas (j). También pueden existir los arrays multidimensionales.  
`int b[5][5][2];`



`int b[5][5];`

**Registros:** son agrupaciones heterogéneas, contiguas y estáticas. Los elementos contenidos en la agrupación pueden ser de distintos tipos (incluso estructurados) y se denominan **Campos**.

```
//Definición registro
struct Nombre {
    Tipo1 Campo1; Tipo2 Campo2; ...
};

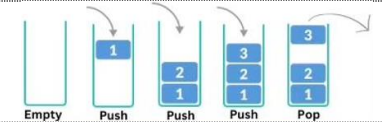
//Declaración variable
NombreVar : Nombre

//Asignación
NombreVar.Campo1 = valor;
```

**Lista (TAD):** un conjunto lógico de elementos homogéneos (nodos) entre los que existe una relación lineal a través de **punteros**. Pueden ser: simple, doble, múltiple o por saltos, según la cantidad de enlaces por nodo.



**Pila (TAD):** colección de elementos homogéneos en la que sólo se pueden añadir (Push) y eliminar (Pop) por el principio, siguiendo la filosofía **LIFO** (Last In First Out).

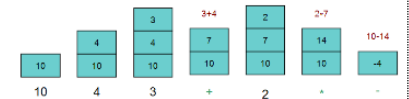


**Notación polaca inversa (RPN) o postfija:** es otra forma de escribir expresiones matemáticas según una Pila.

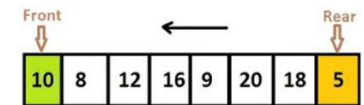
Ejemplo:

Notación normal / infija :  $10 - (4 + 3) * 2$

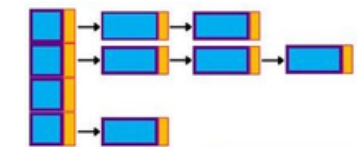
Notación postfija :  $10 \ 4 \ 3 + 2 * -$



**Cola (TAD):** colección de elementos homogéneos en la que sólo se pueden añadir elementos por el final y se eliminan por el principio (Front), siguiendo la filosofía **FIFO** (First In First Out).



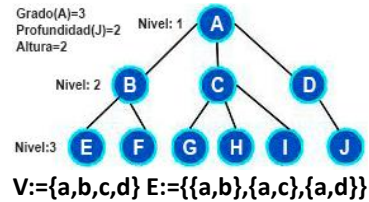
**Tabla Hash:** es un contenedor asociativo (tipo Diccionario). Cada una de estas entradas tiene asociada una clave única. Dispersión: asociar el con la clave. Resolución de colisiones: Hashing ABIERTO (lista) o CERRADO (vector).





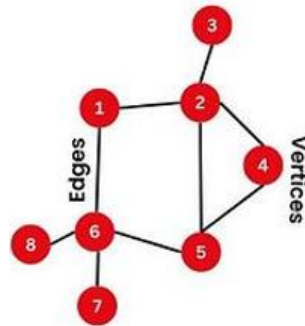
**Árboles:** ED jerárquica cuya recorrido se puede realizar en amplitud y en profundidad. **Es un grafo conexo, sin ciclos y de grado mayor a 1.**

- **Grado:** número máximo de hijos
- **Grado (nodo):** número de hijos
- **Profundidad:** longitud entre raíz y nodo.
- **Altura:** profundidad máxima de nodos.
- **Lleno:** todo nodo tiene dos hijos. **Grado=2**



**Grafos:** conjunto de **vértices** o nodos **unidos por** enlaces llamados **aristas** o arcos.

- **Orden:** número de vértices.
- **Tamaño:** número de aristas
- Dirigido: arcos son ordenados con ->
- **Grado(u):** número de aristas que contiene.
- **Ponderado:** Si tiene asociado peso numérico.
- **Pseudografo:** Si tiene bucles.
- **Conexo:** cada par de vértices está conectado por un camino que los une.
- Recorrido: en ANCHURA y en PROFUNDIDAD.



#### Clasificación de los árboles según el grado

**Árboles BINARIOS:** de grado = 2 (máximo dos hijos). Recorrido: en profundidad y en anchura.

- **Preorden (R-IZQ-DCHA):** A, E, B, F, G, C, H, I, J, D
- **Inorden (IZQ-R-DCHA):** E, B, F, G, A, C, H, I, J, D
- **Postorden (IZQ-DCHA-R):** E, B, F, G, C, H, I, J, D, A

**Árbol binario de búsqueda (ABB):** todos los elementos almacenados en el subárbol izquierdo son menores que los almacenados en el subárbol derecho.

**Árbol AVL:** (Adelson-Velskii y Landis) es un ABB equilibrado en altura.

Un árbol está equilibrado: si para cada uno de sus nodos las alturas de sus dos subárboles difieren como mucho en 1.

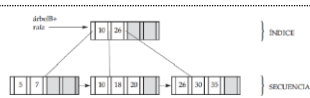


**Árboles MULTICAMINO:** de grado > 2.

**Árbol B:** árbol multicamino, de un orden determinado. Crecen "hacia arriba" y decrecen "hacia abajo".

**Árbol B+:** formado por índice y secuencia.

**Árbol B\*:** mejora la eficiencia del acceso directo y la sobrecarga.



Requiere que los nodos no raíz estén por lo menos a 2/3 de ocupación en lugar de 1/2.

**ALGORITMOS:** secuencia ordenada y finita de operaciones elementales que devuelve un resultado (Tiempo) a partir de los datos de entrada. **T(n)**

**Eficiencia de un algoritmo.** ¿cuál es mejor? según la eficiencia empíricamente o teóricamente.

**La complejidad algorítmica** representa la cantidad de recursos que necesita un algoritmo para resolver un problema ¿Cómo se calcula? **Usando la notación asintótica O(n)**

Funciones de complejidad en tiempo más usuales (mayor a menor eficiencia):

$O(n) = 1$ : constante	$O(n) = n \log n$ : quasilínea
$O(n) = \log n$ : logarítmica	$O(n) = n^2$ : cuadrática
$O(n) = n$ : lineal	$O(n) = k^n$ : exponencial

**Recursividad:** divide el problema original en varios más pequeños que son del mismo tipo que

el inicial, procediendo seguidamente a encontrar la solución de estos subproblemas. Ejemplo con Fibonacci 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, ... se obtiene sumando los dos términos anteriores.

**ALGORITMOS DE BÚSQUEDA:** para la localización de un elemento dentro de la estructura de datos.

**SECUENCIAL:** Esta búsqueda es aplicable a una tabla organizada como un **vector**.

**BINARIA:** Se compara el argumento con la clave del elemento medio de la tabla. Si son iguales, la búsqueda termina con éxito; en caso contrario se busca de manera similar en la mitad superior o inferior de la tabla.

**INTERPOLACIÓN:** Suponiendo que las claves están distribuidas de manera uniforme entre dos si k está presente en el árbol la comparamos con valores k(low) y k(high).

**Árboles binarios de búsqueda:** Para determinar la clave situada en la raíz, r. Si coincide la búsqueda finaliza con éxito, si  $k < r$  es evidente que k, de estar presente, será un descendiente del hijo izquierdo de la raíz, y si es mayor será un descendiente de hijo derecho.



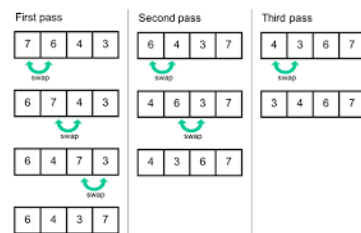
## ALGORITMOS DE ORDENACIÓN:

**BURBUJA** (bubblesort): sencillo algoritmo de ordenamiento.

Consiste en recorrer la lista comparando y ordenando los elementos adyacentes de dos en dos, **intercambiándolos de posición** si están en el orden equivocado.

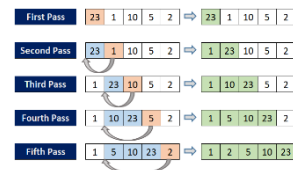
Máximo: N-1

Complejidad:  $O(n^2)$  cuadrática.



**INSERCIÓN** (insertion sort): Denominado así porque **en el i-ésimo recorrido se inserta el i-ésimo** elemento,  $A[i]$ , en el lugar correcto, entre  $A[0]$ ,  $A[1]$ , ...,  $A[i-1]$ , los cuales fueron ordenados previamente.

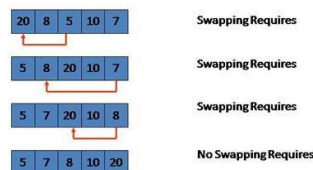
Complejidad:  $O(n^2)$  cuadrática.



**SELECCIÓN** (selection sort): Buscar el mínimo elemento entre una posición  $i$  y el final de la lista e **intercambiar el mínimo** con el elemento de la posición  $i$ .

Comparaciones:  $N(N-1)/2$

Complejidad:  $O(n^2)$  cuadrática.



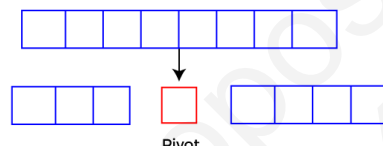
**Quicksort**: este algoritmo (inestable) se basa

en el concepto del **pivote**.

Elegir un elemento del conjunto de elementos a ordenar, al que llamaremos pivote.

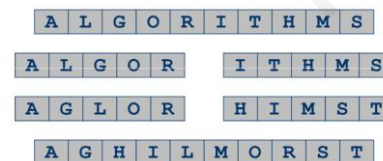
Resituarse los demás elementos de la lista a cada lado del pivote y repetir el proceso.

Complejidad:  $O(n^2)$  cuadrática.



**Mergesort**: Se le considera de ordenamiento externo, dedicado a grandes cantidades de información como el almacenamiento persistente. **Divide la lista en dos partes** y ordena cada una de ellas recursivamente, luego mezcla las dos sublistas.

Complejidad:  $O(n \log n)$



**Heapsort**: en ordenar todos los elementos **en un montículo** y luego ir extrayéndolos en orden.

Complejidad:  $O(n \log n)$

**Otros**: Shell sort, Bucket sort o Binsort, Counting sort, Radix sort, Radix sort y Cubesort

## ALGORITMOS TEORÍA DE GRAFOS:

**Kruskal**: para **encontrar un árbol recubridor mínimo** en un grafo conexo, **no dirigido y ponderado**. Empezando por cualquier arista.

**Dijkstra** (algoritmo de caminos mínimos): para la determinación del **camino más corto** **dado un vértice origen** al resto de vértices en un grafo con pesos en cada arista.

**Prim**: para **encontrar un árbol recubridor mínimo** en un grafo conexo, **no dirigido y cuyas aristas están etiquetadas**. Empezando por cualquier vértice

**Bellman-Ford**: es **como el algoritmo de Dijkstra**, pero en el que el peso de las **aristas** puede ser **negativo**

**búsqueda A\***: busca el camino más corto **entre dos vértices** guiado por una función heurística.

**Floyd-Warshall**: encuentra el camino más corto **entre todos los pares de nodos** de un grafo.

## ORGANIZACIONES DE FICHEROS

- **Organización secuencial**: almacenar los registros en posiciones físicas contiguas de memoria:

Organización secuencial indexada, encadenada o indexada encadenada.

- **Organización directa (aleatoria)**: el almacenamiento se hace a través de un identificador o clave que indicará la posición del registro en el fichero (también es conocida como dispersa hash).

- **Organización en montículos** (ORGANIZACIÓN ESPECIAL): se puede insertar registros en cualquier posición del fichero en el que haya espacio suficiente, no existiendo ninguna ordenación de los registros. Usado en bases de datos, un archivo por tabla.

## FORMATOS DE INFORMACIÓN Y FICHEROS:

### Formatos de imágenes:

Compresión SIN pérdida:		Compresión CON pérdida:	
<b>.bmp</b>	Windows Bitmap (Microsoft).	<b>.jpeg</b>	Joint Photographic Experts Group, imágenes de <b>16 millones de colores</b> .
<b>.gif</b>	Graphics Interchange Format, crea una tabla de <b>256 colores</b> .	<b>.jpg</b>	
<b>.png</b>	Portable Network Graphics, profundidad de <b>8 bits, 24 y 48 bits</b>	<b>.jp2</b>	JPEG 2000. Trató de sustituir el formato JPEG.
<b>.svg</b>		<b>.svg</b>	Scalable Vector Graphics, es un formato de gráficos vectoriales bidimensionales, en lenguaje de marcado XML.
<b>.raw</b>	Imagen RAW.		
<b>.tif</b>	Tagged Image File Format ( <b>con o sin pérdida</b> ), normalmente usado sin compresión y sin pérdidas. Pueden utilizar un algoritmo de compresión sin pérdidas llamado LZW.		
<b>Formatos propios de distintos programas de edición:</b>			
<b>.psp</b>	Paint Shop Pro.	<b>.psd</b>	Photoshop.
<b>.xcf</b>	GIMP	<b>.ai</b>	Adobe Illustrator.
<b>.dwg</b>	AutoCAD	<b>.cdr</b>	Corel Draw

Algoritmos de compresión con pérdidas: TDC

Sin pérdidas: LZW

Formatos de documentos:			
<b>.doc</b>	Archivos que contienen textos con formato (Microsoft).	<b>.odt</b>	Open Office Writer puede abrir los tres tipos de extensiones.
<b>.docx</b>			
<b>.ppt</b>	Presentación en diapositivas.	<b>.odp</b>	Open Office Impress puede abrir los dos.
<b>.xls</b>	Libros u hojas de cálculo.	<b>.ods</b>	Open Office Calc puede abrir los dos tipos de extensiones.
<b>.xlsx</b>			

Formatos de intercambio de datos:	
<b>JSON</b>	Basado en JS aunque es completamente independiente del lenguaje. <b>Clave – Valor</b> . Ejemplo: {"coches":["Ford", "BMW", "Renault"], null}}
<b>BSON</b>	Es una serialización codificada en binario de documentos similares a JSON. Almacena datos en binario. Es ligero, transitable y eficiente.
<b>CSV</b>	Valores separados por coma. Se utilizan con .csv y .txt.
<b>XML</b>	Lenguaje de Etiquetado Extensible desarrollado por W3C.
<b>YAML</b>	Destinado facilitar el mapeo de estructuras de datos más complejas (como listas y arreglos asociativos) en un documento de texto plano legible para un ser humano, usado en el despliegue de contenedores de aplicaciones ( <b>docker</b> ).

Formatos y códecs video:			
Formatos		Códecs	
<b>AVI</b>	Creado por Microsoft.	<b>AV1 (AOM Video 1):</b>	NUEVO códec de código abierto desarrollado por la Alliance for Open Media (AOMedia).
<b>WMV</b>			
<b>FLV (Flash)</b>	Creado por Adobe.	<b>dirac-research</b>	Extensión .drc
<b>MKV</b>	Creado por Matroska.	<b>OpenH264</b>	Creado por Cisco.
<b>MOV</b>	Creado por Apple.		
<b>MPEG-4</b>	abreviado como MP4, utiliza los códecs H.264 y H.265.	<b>H.264 y H.265</b>	Estándares del ITU-T.
<b>Ogg</b>	Fundación Xiph.Org	<b>libtheora</b>	Extensión .ogg (Fundación Xiph.Org).
<b>WebM</b>	Creado por Google: formato abierto para video y audio. Soporta VP9 y AV1 para video y Opus para audio.	<b>libvpx (VP9)</b>	Creado por Google.

Formatos de audio: AAC, FLAC, Vorbis, AC3, G.711 - G.729, MP2, MP3 y WMA (Microsoft).

Códecs de audio: AAC, FLAC, Vorbis y Opus.

## Bloque 2 - Tema 4: SISTEMAS OPERATIVOS

**Sistema Operativo:** programa que controla la ejecución de los programas de aplicación y que actúa como **interfaz entre el usuario de un computador y el hardware** del mismo.

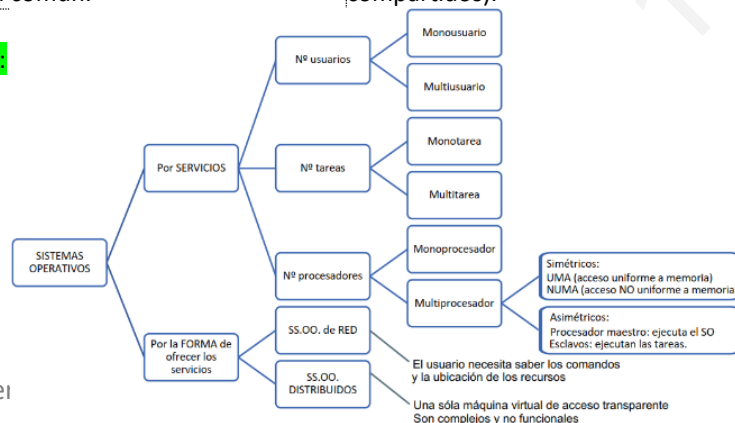
### ESTRUCTURA DE UN SISTEMA OPERATIVO

<b>MONOLÍTICA:</b> Un único programa compuesto por un conjunto de rutinas entrelazadas. módulos compilados separadamente con un gran acoplamiento. Los <b>servicios se colocan en PILA</b> y ejecución en TRAP. Ventajas: <b>eficiencia y rapidez.</b> Desventajas: nula flexibilidad.	<b>JERÁRQUICA o CAPAS:</b> Cada capa sólo tiene acceso a la inmediatamente superior e inferior protegidas de accesos indebidos desde capas externas.  Usada por SSOO actuales como THE, Multics y Unix.
<b>CLIENTE-SERVIDOR (microkernel):</b> El núcleo establece la comunicación entre los clientes y los servidores. Los procesos pueden ser tanto servidores como clientes, el núcleo provee solamente funciones muy básicas de memoria, entrada/salida... Ventajas: <b>flexibilidad.</b>	<b>MÁQUINA VIRTUAL:</b> Presentan una interfaz a cada proceso, e integran distintos SSOO dando la sensación de ser varias máquinas diferentes. <b>Núcleo = monitor virtual</b> (multiprogramación).  Usada por Citrix XenServer, VMWare o VirtualBox.

### Componentes del SO

<b>Administrador de procesos:</b> sincronización, de comunicación de procesos.	<b>Administrador de archivos:</b> Gestiona el almacenamiento en disco.
<b>Administrador de memoria principal:</b> asigna y recupera espacio no necesitado.	<b>Sistema de protección:</b> se ocupa de la autorización y evita accesos no permitidos.
<b>Administrador del sistema de E/S:</b> encargado del hardware específico y de presentar una interfaz general común.	<b>Sistema de comunicación:</b> encargado de gestionar los accesos a la red (recursos compartidos).

### CLASIFICACIÓN:



Todos los dei

**ADMINISTRACIÓN DE PROCESOS:** Un proceso es un programa en ejecución. Incluye los valores que tienen el contador de programa, los registros y las variables

### CREACIÓN de procesos

Inicialización del sistema: el arranque del SO	Llamada al sistema por un proceso en ejecución
Solicitud de un usuario para crear un proceso.	Inicio por lotes: aplicable a los mainframes.

### TERMINACIÓN de procesos

Terminación normal (voluntaria)	Error fatal o inesperado (involuntaria)
Terminación por error (voluntaria)	Terminado por otro proceso (involuntaria)

### JERARQUÍA de procesos

Padre	Hijos
-------	-------

### ESTADOS de un proceso

En ejecución	Bloqueado
Listo para ejecutarse	

### MODELO de procesos

La gestión del conjunto se implementa a través de la tabla de procesos que contiene una entrada para cada proceso con su estado y la información necesaria para continuar su ejecución.

**Planificación del procesador:** se refiere a la manera o técnica que se usa para decidir cuánto tiempo de ejecución y cuándo se le asigna a cada proceso del sistema.

### Se identifican tres niveles de planificación

<b>LARGO PLAZO:</b> long term scheduler. Trabajos candidatos para convertirse en procesos listos para usar recursos del sistema.	<b>MEDIO PLAZO:</b> mid term Scheduler. Procesos que se suspenden o se reanudan para lograr rendimiento.	<b>CORTO PLAZO:</b> dispatcher o short term scheduler. Procesos listos para ejecutarse en la unidad central de proceso.
--	--	---

### Algoritmos de planificación

<b>SJF (Shortest Job First):</b> el proceso <b>más corto primero</b> Planificación: no apropiativa. El mejor tiempo de respuesta promedio.	<b>Turno circular (Round Robin):</b> Consiste en darle a cada <b>proceso un intervalo de tiempo</b> de ejecución (denominado quantum) en una <b>cola circular</b> . Ejemplo: quantum $q=2$ .
<b>SRJF (Shortest Remaining Time First):</b> primero a que <b>le queda menos por ejecutar</b> . Planificación: apropiativa.	<b>Por prioridad:</b> los de <b>mayor prioridad</b> se ejecutan <b>primero</b> . Ventaja: es flexible.
<b>FCFS (First Come First Served):</b> primero en llegar primero en ejecutarse. <b>Cola tipo FIFO</b> .	<b>Colas multinivel:</b> Se asigna un algoritmo y prioridad a a cada lotes o grupo de procesos.

**En los sistemas concurrentes** los procesos compiten por los recursos del sistema, pudiendo generarse una serie de **problemas como por ejemplo el aplazamiento indefinido (inanición)** que sucede cuando un proceso es **postergado continuamente sin posibilidad de finalizar su ejecución**.

ibida su copia y/o distribución.

## ADMINISTRACIÓN DE LA MEMORIA: Técnicas

**1. Gestión simple:** en sistemas monousuario y monotarea, como MSDOS. La memoria real se utiliza para el único proceso en ejecución. Limitado a la cantidad de RAM disponible.

**2. Multiprogramación:** varios procesos al mismo tiempo en memoria física (que se divide en secciones).

**Relocalización:** este problema sucede si el programa se carga en una sección diferente de memoria real, las direcciones reales ya no coinciden. Solución «estática» recalcula o «dinámica» guarda la dirección.

**Protección:** este problema consiste en leer o escribir en áreas de otros programas cómo solución se puede ser el uso de un *registro base* y un *registro límite*.

**3. Multiprogramación en memoria real:** disponemos dos alternativas para asignarle a cada programa su partición correspondiente (particiones de tamaño fijo o particiones de tamaño variable.)

**Particiones FIJAS** que se crean cuando se enciende el equipo hasta su reinicio.

**Particiones VARIABLES** contiguas de tamaño variable y el sistema asigna los huecos libres mediante algoritmos (**primer, mejor, peor o siguiente** ajuste, incluso **ajuste rápido mediante listas enlazadas**). Se pueden **unir huecos mediante compactación**.

**4. Multiprogramación en memoria virtual:** mecanismo para ejecutar automáticamente programas más grandes que la memoria real disponible. Se suma la memoria de almacenamiento primario (memoria real) y una cantidad determinada de almacenamiento secundario (discos). Sistema de **paginación pura o swapping** con fragmentación interna.

### 4.1 Sistemas de manejo de memoria virtual.

**PAGINACIÓN pura o swapping:** ya no se requiere que los programas estén ubicados en zonas de memoria adyacente, ya que **las páginas pueden estar ubicadas en cualquier lugar de la memoria RAM creando una tabla (registro) de direcciones virtuales**.

Si las páginas son muy grandes podemos tener **sobrecarga o fragmentación interna**.

Algoritmos de reemplazo de páginas:

- Primera en entrar, primera en salir (FIFO).
- Not Recently Used (NRU).
- Least Recently Used (LRU): Búsqueda exhaustiva de la menos usada.
- Not Frequently Used (NFU): se basa en el número de referencias y no en tiempo.
- Least Frequently Used (LFU): Búsqueda exhaustiva de la página menos referenciada.
- Aleatoriamente: se elige cualquier página sin criterio alguno.

**SEGMENTACIÓN pura:** aprovecha que los programas se dividen en partes lógicas, como son la parte de datos, de código y de pila (stack) **asignando particiones de memoria a cada segmento de un programa** creando una tabla de segmentos indicando la longitud, los permisos, etc. Como problema se destaca la **fragmentación externa** quedando **zonas libres de memoria sin asignar**.

**Esquemas COMBINADOS:** la mayoría de los sistemas operativos modernos implantan esquema combinados, es decir, **combinan la paginación y la segmentación**. Se puede decir que se tiene un direccionamiento tridimensional.

**ADMINISTRACIÓN DEL SISTEMA DE E/S:** El sistema de E/S es la parte del SO que actúa como interfaz entre los dispositivos y los usuarios. Los **controladores de dispositivos** son la parte electrónica de los periféricos, los cuales pueden tener la forma de una tarjeta o un circuito impreso en la placa base.

El **acceso directo a memoria (DMA)** permite la transferencia de datos entre un periférico y la memoria sin intervención de la CPU (libera a la CPU de carga).

Los principios de software en la E/S: manejadores de interrupciones, manejadores de dispositivos, Software independiente del dispositivo y Software para usuarios.

**ADMINISTRACIÓN DE ARCHIVOS:** Un sistema de archivos (File System) es una estructura de directorios organizados el cual nos permite manipular archivos en diferentes formatos.

El **esquema básico de un sistema de almacenamiento** es: las superficies de **almacenamiento** se dividen en círculos concéntricos "**pistas**", las cuales se dividen en "**sectores**". A la unión lógica de varias pistas paralelas se le denomina "cilindro".

### ALGORITMOS de planificación de peticiones de lectura y escritura a discos

**Primero en llegar, primero en salir (FCFS o FIFO)** Primero el más cercano a la posición actual (**SSTF**)

**Por exploración** (algoritmo del elevador o **SCAN**): Por exploración circular (**C-SCAN**): para al anterior el brazo se está moviendo en todo momento desde el perímetro del disco hacia su centro y viceversa, resolviendo las peticiones. pero al llegar a la parte central, el brazo regresa al exterior sin resolver ninguna petición.

### ASIGNACIÓN del espacio de almacenamiento (gestión del espacio libre)

**Asignación contigua:** si el archivo es borrado y luego creado otro más pequeño, quedarán huecos inútiles, lo cual se denomina **fragmentación externa**. **Asignación encadenada:** se crea una lista encadenada, no es necesario que los bloques estén contiguos y **no existe fragmentación externa**.

**Asignación con índices (indexada o INODO):** se crea en el directorio un bloque de índices para cada archivo, de manera que **el acceso directo se agiliza**, a cambio de sacrificar algo de espacio. Éste es un esquema excelente **para archivos grandes** pero no para pequeños.

### Métodos de acceso en los sistemas de archivos

**Acceso secuencial:** recorrer los componentes de un archivo uno a uno hasta llegar al registro deseado. **Acceso directo:** permite acceder a cualquier sector o registro inmediatamente.

**Acceso directo indexado:** cada archivo tiene una **tabla de apuntadores**, donde cada apuntador va a la dirección de un bloque de índices. Consume muchos recursos (espacio) pero es muy rápido.



## SISTEMAS DE FICHEROS EXISTENTES

### Los sistemas de ficheros de disco más importantes son

	Máx. archivo	Máx. volumen	SSOO	Otros datos
<b>FAT16</b>	2 GB	2 GB	MS-DOS (16 bits)	
<b>FAT32</b>	4 GB	4 GB	Windows 95 y posteriores macOS y Linux	Ni compresión ni cifrado
<b>exFAT</b>	16 EB	64 ZB	Windows XP y posteriores macOS y Linux	Usado en Flash y tarjetas SD No admite compresión No admite cifrado
<b>NTFS</b>	16 TB	16 TB	Creado por Microsoft Windows NT y posteriores macOS (Sólo lectura) Linux (sólo lectura)	ED árboles B Permisos con listas ACL. <b>Compresión y cifrado</b>
<b>ReiserFS</b>	8 TB	16 TB		Ni compresión ni cifrado
<b>Reiser4</b>	8 TB		Creado por Linux	ED árboles B* <b>Compresión y cifrado</b>
<b>JFS</b>	64 GB	1 TB		Árboles B+ (en JFS) Arboles B* (en JFS2)
<b>JFS2</b>	16 TB	32 TB	Creado por AIX (de IBM) Compatible con Linux	Permisos con listas ACL. <b>Compresión</b>
<b>UFS</b> <b>UFS2</b>	8 ZB	8 ZB	Unix	Unix File System
<b>VFS</b>	Es una capa de abstracción		Windows, Linux, Mac OS	Virtual File System
<b>XFS</b>	8 EB	8 EB	Linux	Árboles B+ Alta escalabilidad
<b>ISO 9660</b>			Todas las plataformas	Discos ópticos

Respecto a los sistemas de ficheros de red destacan:

- **NFS (Network File System)**: protocolo de sistemas de ficheros en red que permite acceder a ficheros a través de la red como si el sistema de archivos fuese local. NFSv4 es la última versión.
- **AFS (Andrew File System)**: sistema de archivos distribuido.
- **CIFS (Common Internet File System)**: protocolo **desarrollado por Microsoft** que permite compartir archivos e impresoras entre nodos de una red (evolución de SMB –Server Message Block-). En sistemas Unix, macOS y GNU Linux **existe la implementación libre Samba**.

**SISTEMAS WINDOWS** La primera versión de Windows (1.0) fue dada a conocer en el año 1985 por Microsoft. Se trataba de una interfaz gráfica de usuario para el sistema operativo MS-DOS.

Versión	Uso	Características
Windows NT	Workstation y Server	capa de abstracción de hardware (HAL)
Windows 2000	Server empresarial	núcleo de Windows NT
Windows XP	Home y Professional	núcleo de Windows NT
Windows Vista	Home y Professional	Windows Media Center e interfaz Aero
Windows 7	Home y Professional	Soporta discos virtuales. Versiones de 32 bits y 64 bits
Windows 8	PCs, tablets y servers	soporte para procesadores ARM Interfaz Modern UI (táctil), desaparece el menú de inicio Y además la app OneDrive
Windows 10	Home y Professional	BitLocker, Microsoft Cortana, Microsoft Edge, Windows Hello, Vista de Tareas (escritorios virtuales) y virtualización de apps (App-V). Interfaz Continuum (táctiles o ratón).
Windows 11	Home y Professional	Requiere TPM (procesador criptográfico seguro) soporta aplicaciones Android. Interfaz Fluent Design.

### Funcionalidades destacadas:

- BitLocker**: Cifrado para protección de datos que se integra (Windows 10 y 11).
- Windows Defender**: antivirus y malware (Windows 8.1 y 10).
- Microsoft Security Essentials**: antivirus y malware para versiones más antiguas
- Microsoft Cortana**: asistente digital (disponible en Windows 10) utiliza el motor de Bing.
- Microsoft Edge**: navegador Web basado en Chromium (Windows 10 y superiores).
- Windows Hello**: inicio de sesión (facial, huella o un PIN) a partir de Windows 10.
- Microsoft Application Virtualization (App-V)**: aplicaciones Win32 como aplicaciones virtuales.
- Windows Insider**: permite instalar y usar las últimas compilaciones de Windows Insider Preview.
- WSL (Windows Subsystem for Linux)**: simula un kernel de Linux en Windows 10.
- WIP (Win. Information Protection)**: cifrar los datos de trabajo y separarlos de los datos personales.

**Microsoft Azure**: es un conjunto de servicios de computación en la nube enfocado a empresas.



**Active Directory** almacena información acerca de los objetos de la red y facilita la búsqueda y el uso de esta información para los administradores y los usuarios. Es una estructura basada en el servicio de directorio LDAP (Lightweight Directory Access Protocol).

**Dominios:** dividen el directorio en secciones dentro de un solo bosque. Usan sintaxis DNS.

**Unidades organizativas (OU):** permiten agrupar recursos, como cuentas o de usuario o PCs, en un dominio para su administración como una única Unidad.

**Bosques:** son los límites de seguridad y el nivel más alto de la estructura lógica.

**Directivas de Grupo (GPO):** permiten implementar configuraciones específicas para uno o varios usuarios y/o equipos.

Para la configuración de GPO que sólo afecten a un usuario o equipo local se puede utilizar el editor de **directivas locales gpedit.msc** o acceder desde la **consola de administración del Servicios de Dominio gpmmc.msc**

Las GPO pueden diferenciar el objeto que configuran en distintos niveles:

- **Equipo Local:** tan solo se aplican en el equipo que las tiene asignada (independiente del dominio).

- **Sitio:** se aplican a los equipos y/o usuarios de un sitio, independientemente del dominio.

- **Dominio:** se aplican a todos los equipos y/o usuarios de un dominio.

- **Unidad Organizativa (OU):** se aplican únicamente a los equipos y/o usuarios de la OU.

Un sistema tipo **Unix puede participar en una red Microsoft**, actuando incluso como controlador de dominio. Para ello hay que instalar y configurar **Samba** en un sistema tipo Unix.

#### Comandos DOS habituales:

arp	Muestra o modifica la tabla ARP	nbtstat	Envía a imprimir a impresora remota
ipconfig	Configuración IP en modo texto	nslookup	Muestra estadísticas NetBIOS
ftp/tftp	Cliente FTP/TFTP	ping	Consultas a un servicio DNS
hostname	Muestra el nombre del host	rcp	Eco ICMP a host remoto
lpq	Muestra el estado de una cola de impresión remota (lpd)	rexec rsh	Copia archivos a un equipo remoto mediante servicio RCP

#### Comandos NET (Comandos DOS para red):

netstat	Muestra estadísticas de red, conexiones y puertos abiertos	net localgroup	Agrega, muestra o modifica grupos locales
net computer	Agrega o elimina máquinas a un dominio	net name	Añade o elimina un alias
net start	Inicia un servicio. Sin parámetros, muestra los servicios en ejecución	tracert	Realiza una traza a un host remoto
net accounts	Gestiona parámetros de cuentas como la longitud mínima de la contraseña	net send	Envía mensajes a otros usuarios

net config	Gestiona la configuración de servicios	net stop	Detiene un servicio
net print	Gestiona trabajos y colas de impresión	net session	Muestra o desconecta sesiones del equipo. Sin parámetros, muestra las sesiones establecidas actualmente
net pause	Suspende la ejecución de un servicio	net file	Desbloquea un archivo para otros usuarios. Sin parámetros, muestra los archivos compartidos abiertos
net time	Sincroniza el reloj del sistema con el de un host remoto o dominio	net share	Gestiona recursos compartidos
net group	Agrega o modifica grupos globales	net statistics	Estadísticas del servicio local

**Registro de Windows:** base de datos jerárquica centralizada dispuesta para almacenar la información de configuración del sistema para usuarios, aplicaciones y dispositivos. Los archivos auxiliares de cada sección excepto HKEY\_CURRENT\_USER suelen estar en la carpeta %SystemRoot%\System32\Config.

#### Los tipos de valores más importantes usados en el registro:

HKEY_CURRENT_USER	Contiene la raíz de la información de configuración del usuario que ha iniciado sesión.
HKEY_USERS	Contiene todos los perfiles de usuario cargados activamente en el equipo
HKEY_LOCAL_MACHINE	Contiene información de configuración específica del equipo (para cualquier usuario).
HKEY_CLASSES_ROOT	Es una subclave de HKEY_LOCAL_MACHINE\Software. La información que se almacena aquí garantiza que cuando abra un archivo con el Explorador de Windows se abrirá el programa correcto.
HKEY_CURRENT_CONFIG	Contiene información acerca del perfil de hardware que utiliza el equipo local cuando se inicia el sistema.

#### Los servicios de informática en la nube se engloban en 5 categorías generales:

INFRAESTRUCTURA como servicio (IaaS)	Se alquila <b>infraestructura</b> de TI (servidores, máquinas virtuales, almacenamiento, etc.) Servicios como <b>AWS o Google Cloud Platform</b>
PLATAFORMA como servicio (PaaS)	Permite <b>desplegar aplicaciones</b> a los desarrolladores para la creación rápida de proyectos. <b>Firebase o Google APP Engine</b>
CONTENEDOR como servicio (CaaS)	permiten gestionar e implementar las aplicaciones usando el aislamiento en basados en <b>Docker en la nube</b> .
FUNCIÓN como servicio (FaaS)	Se conoce como "INFORMÁTICA SIN SERVIDOR" y permite que los desarrolladores ejecuten las aplicaciones web en respuesta a ciertos eventos, sin tener que gestionar los servidores.
SOFTWARE como servicio (SaaS)	Ofrece aplicaciones a los usuarios finales a través de Internet, normalmente, <b>con una suscripción (como Dropbox o Drive)</b> .

## SISTEMAS UNIX Y LINUX

**Unix** es un **sistema operativo portable**, multitarea y multiusuario; desarrollado en 1969 por un grupo de empleados de los laboratorios Bell de AT&T.

**GNU/Linux** es un **conjunto de sistemas operativos** libres multiplataforma, multiusuario y multitarea **basados en Unix** el núcleo o **kernel Linux**, programado por **Linus Torvalds**.

Distribución	Basado en	Gestor de paquetes	Distribución	Basado en	Gestor de paquetes
Debian		APT	Red Hat (RHEL)		RPM
Ubuntu	Debian	APT	Fedora	Red Hat Linux	RPM, YUM
Linux Mint	Ubuntu	APT	CentOS	Red Hat (RHEL)	YUM
Elementary OS	Ubuntu	APT	openSUSE	SUSE Linux	YaST
Knoppix	Debian	APT	Arch Linux		Pacman
Trisquel	Debian/Ubuntu	APT	Mandriva		RPM

El **gestor de paquetes** es una colección de herramientas que sirven **para automatizar el proceso de instalación, actualización**, configuración y eliminación de paquetes de software.

UNIX no usa el bloque 0 (bloque de arranque), que a menudo contiene código para arrancar el ordenador. El bloque 1 es el superbloque; contiene información crucial acerca de la organización del sistema de archivos, **incluido el número de i-nodos**.

### Campos genéricos de un i-nodo

UID	UID del propietario	Tiempos	Acceso y modificación
GID	GID del propietario	Nº enlaces	Nº archivos apunten a ese i-nodo
Tipo	Tipo de archivo	Entradas	Direcciones de bloques de datos
Permisos	Permisos (rxw- rxw- rxw)	Tamaño	Tamaño del archivo en bytes

**Procesos**: son las entidades activas principales en un sistema Unix/Linux. Cada proceso ejecuta un solo programa, y al principio tiene un solo hilo de control.

Linux **es un sistema de multiprogramación**, por lo que puede haber **varios procesos** independientes en ejecución **al mismo tiempo**, que podemos distinguir entre:

- Procesos interactivos.

- Procesos demonio: que se ejecutan en background (ejemplo: el proceso **cron**).

Se crea llamada al sistema **fork** crea una copia exacta del proceso original. El proceso que va a realizar la bifurcación es el proceso padre y al nuevo proceso se le conoce como proceso hijo.

El sistema fork al iniciar un proceso asigna (aunque se puede alterar mediante **setuid** o **setgid**):

- Un identificador denominado PID **valor 0 para el hijo** y un valor **distinto al padre**.

- El identificador del proceso que lo lanzó PPID.

- El usuario propietario que lo lanzó UID.

- El grupo de pertenencia GID: lo que definirá el perfil de permisos.

```
$ ps -e (-e muestra todos los procesos)
$ ps -ef (-f muestra opciones completas)
$ ps -eF (-F muestra opciones completas extra)
$ ps -eo user,pid,ttty (-o output personalizado, se indican los campos separados por coma)
$ ps -eH (muestra árbol de procesos)
$ ps axf (lo mismo en formato BSD)
$ ps -ec (el comando que se está ejecutando, sin la ruta, solo el nombre real)
$ ps -el (muestra formato largo de varias columnas, muy práctico)
$ ps L (no muestra procesos, lista todos los códigos de formatos)
$ kill PID (matar o terminar proceso)
$ killall (afectará a todos los procesos que tengan ese nombre)
$ nice (cambiar la prioridad de un proceso que por defecto es 0)
$ renice (permite alterarla en tiempo real)
$ nohup y & (ejecutar procesos en background)
$ top (monitorización en tiempo real del estado de los procesos)
```

### La estructura de directorios del sistema UNIX/Linux es:

/	Directorio raíz	/mnt	Montaje de dispositivos externos
/bin	Comandos de usuario (ejecutables)	/media	Montaje para dispositivos de medios
/boot	Archivos de arranque	/opt	Paquetes de aplicaciones estáticas
/dev	Ficheros de dispositivos	/proc	Archivos de control de procesos
/etc	Ficheros de configuración (arranque)	/root	Directorio del superusuario root
/home	Directorios de inicio de los usuarios	/sbin	Archivos ejecutables del sistema
/lib	Librerías del sistema	/tmp	Ficheros temporales
/usr	Ejecutables y documentación	/var	Archivos de log

```
$ ls -l (listado en formato largo con permisos, enlaces, usuario, grupo, tamaño y fecha modificación).
$ ls -h (imprime el tamaño de los archivos.)
$ ls -d (el nombre del subdirectorio)
$ ls -t (el listado ordenado por la fecha de última modificación)
$ ls -r (mostrando los más recientes al final)
$ ls -L (muestra los datos del archivo referenciado del link)
$ ls -1 (muestra el listado en una sola columna)
$ ls -i (el número del i-nodo)
$ ls -m (muestra los archivos en una línea y separados por comas)
$ ls -a (muestra los archivos ocultos)
$ pwd (Print Working Directory)
$ df -h (Disk Free)
$ du -h (disk usage)
```

## Métodos de COMPRESIÓN y DESCOMPRESIÓN

Comprimir <b>.tar.gz</b>	tar -czvf empaquetado.tar.gz /carpeta/a/empaquetar/	Comprimir <b>.tar</b>	tar -cvf paquete.tar /dir/a/comprimir/
Descomprimir	tar -xvzf archivo.tar.gz	Descomprimir	tar -xvf paquete.tar
Listar	tar -tzvf paquete.tar.gz	Listar	tar -tvf paquete.tar
Comprimir <b>.gz</b>	gzip -9 index.php	Comprimir <b>.zip</b>	zip archivo.zip carpeta
Descomprimir	gzip -d index.php.gz	Descomprimir	unzip archivo.zip
		Listar	unzip -l archivo.zip

Para **buscar dentro de un archivo** utilizamos el comando **grep [options] pattern [FILE]**

```
$ grep -i (no distingue entre mayúsculas y minúsculas)
$ grep -c (número de coincidencias (líneas) con el patrón buscado)
$ grep -r (búsqueda recursiva en el directorio actual)
$ grep -n (número de línea de cada coincidencia con el patrón buscado)
$ grep -v (líneas que no coinciden con el patrón buscado)
$ grep -i Autor libro.txt (busca la palabra Autor en libro.txt sin distinción con mayúsculas)
```

Para **buscar archivos** utilizamos el comando **find directorio [options] pattern**

```
$ find / -name "[0-9]*" (busca ficheros cuyo nombre comience por un dígito)
$ find / -name "[a-g]*" (ficheros cuyo nombre comiencen entre a y g)
$ find / -type f -empty (ficheros vacíos)
$ find / -perm /u=r (ficheros de solo lectura para el propietario)
$ find / -perm 777 (ficheros con permisos 777)
$ find /home -user pablo -iname "*.txt" (ficheros .txt cuyo propietario es pablo)
$ find / -atime -50 (ficheros accedidos en los últimos 50 días)
$ find / --size +100M -exec rm -f {} \ (borra los ficheros con un tamaño superior a 100MB)
$ find / -amin -60 (busca ficheros accedidos en la última hora)
$ find / -size 50M (ficheros cuyo tamaño es 50MB)
$ find / -size +50M -size -100M (ficheros cuyo tamaño es mayor a 50MB y menor a 100MB)
```

**Permisos de archivo y directorio:** cualquier archivo o directorio tiene **3 niveles de permisos**.

- Permisos del usuario propietario (UID).
- Permisos del grupo propietario (GID).
- Permisos del resto de usuarios. Comúnmente denominados "otros".

Opción	Valor en octal	Descripción
r	4	read (lectura)
w	3	write (escritura)
x	1	execution (ejecución)

La combinación de bits en cada grupo da ocho posibles combinaciones de valores, es decir, la suma de los bits.

Permisos	Valor	Descripción
- - -	= 0 (000)	No se tiene ningún permiso
- - x	= 1 (001)	Solo permiso de ejecución
- w -	= 2 (010)	Solo permiso de escritura
- w x	= 3 (011)	Permisos de escritura y ejecución
- w x	= 4 (100)	Solo permiso de lectura
r - x	= 5 (101)	Permisos de lectura y ejecución
r w -	= 6 (110)	Permisos de lectura y ejecución
r w x	= 7 (111)	Permisos de lectura y ejecución

Cuando se combinan los permisos del usuario, grupo y otros, se obtienen un número de tres Cifras: **rwX r-x r-x (755) El propietario rwX**, el grupo **rx** y otros **rx**.

Se pueden modificar los permisos con el comando **chmod** (la opción **-R**, de recursividad)

Se pueden modificar los permisos con el comando **chmod** (la opción **-R**, de recursividad)

```
$ chmod -R 755 datos/* (modificar permisos la opción -R, de recursividad)
$ umask 022 datos/* (permisos por defecto con la resta de 777-755=022)
$ chown usuario archivo (cambiar propietario).
$ chgrp grupo archivo (cambiar grupo).
```

## SISTEMAS OPERATIVOS PARA DISPOSITIVOS MÓVILES

1. **Android** es un **SO móvil basado en Linux desarrollado por Google**. Utiliza SQLite para el almacenamiento de datos. Navegador web basado en WebKit incluido.

### Arquitectura:

- **Aplicaciones** centrales del sistema: Email, SMS, contactos, etc. (todas escritas en Java).
- **Marco de trabajo** de aplicaciones: acceso completo a los APIs del framework (escritos en Java).
- **Bibliotecas**: nativas usadas por varios componentes del sistema (escritos en C/C++).
- **Runtime de Android**: cada proceso corre su propia instancia de la **máquina virtual ART** (Runtime). Antes la versión 5.0 Lollipop, la máquina virtual era Dalvik.
- **Capa de abstracción de hardware** (HAL): interfaces estándares que exponen las capacidades de hardware del dispositivo a la API de Java de nivel más alto.
- **Núcleo Linux**: base de la plataforma Android es el kernel de Linux y depende de él para los servicios base del sistema como seguridad, gestión de memoria, gestión de procesos, pila de red y modelo de controladores.

Versión	Nombre		Versión		
1.0	Apple Pie	23/09/2008	5.0 – 5.1	Lollipop	12/11/2014
1.1	Banana Bread	09/02/2009	6.0	Marshmallow	05/10/2015
1.5	Cupcake	25/04/2009	7.0	Nougat	15/06/2016
1.6	Donut	15/09/2009	8.0	Oreo	21/08/2017
2.0 – 2.1	Eclair	26/10/2009	9.0	Pie	06/08/2018
2.2	Froyo	20/05/2010	10	Android 10	03/09/2019
2.3	Gingerbread	06/12/2010	11	Android 11	08/09/2020
3	Honeycomb	22/02/2011	12	Android 12	04/10/2021
4.0	Ice Cream Sandwich	18/10/2011	13	Android 13	15/08/2022
4.1 – 4.3	Jelly Bean	09/07/2012	14	Android 14	05/10/2023
4.4	KitKat	31/10/2013	15	Vanilla Ice Cream	15/10/2024

Nota: hasta la versión 9 están nombradas alfabéticamente.

#### Características incorporadas en Android 13:

- Compatibilidad con el estándar MIDI 2.0.
- Actualización de las bibliotecas principales de Android con la versión de OpenJDK 11 con LTS
- Java 11 para desarrolladores de aplicaciones y plataformas.
- Las APIs de Camera2 admiten la captura de video de alto rango dinámico (HDR).
- Compatibilidad con el esquema de firma de APK v3.1

#### Características incorporadas en Android 14:

- Soporte para las notificaciones flash.
- Modo de reparación que aísla los datos y apps del usuario.

2. **iOS** es un SO móvil de la empresa Apple Inc. Desarrollado para el iPhone (iPhone OS), después se ha usado para el resto de dispositivos de la marca. La **versión actual es iOS 18**.

#### **Arquitectura 4 capas:**

- Core OS: núcleo del sistema. Se encarga de la seguridad, memoria, manejo de ficheros...
- Core Services: servicios principales del dispositivo: bases de datos, red...
- Media: desarrollada principalmente en Objective C con tecnologías para a ficheros multimedia.
- Cocoa Touch: capa con componentes visuales con la que interactúa el usuario.

3. **Windows Phone 10 Mobile**: En la actualidad, Microsoft (10/12/2019) **ha dejado de dar soporte**.

4. **Firefox OS**: basado en **HTML5 con núcleo Linux**, de código abierto desarrollado por Mozilla.

#### **Arquitectura 3 capas:**

- Gaia (interfaz de usuario): La interfaz de usuario de la plataforma Firefox OS.
- Gecko (motor de renderizado web): HTML, CSS y JavaScript.

- **Gonk (núcleo)**: es el SO de bajo nivel que consiste un núcleo/kernel Linux (basado en Android).

5. **Ubuntu Touch**: basado en Linux **desarrollado por Canonical**.

6. **Tizen**: SO móvil de código abierto basado en Linux.

7. **MeeGo**: Predecesor de Tizen **impulsado por Linux Foundation**.

8. **HarmonyOS**: de código abierto **desarrollado por Huawei** que es multiplataforma.

9. **Sailfish OS**: independiente (**sin vínculos con grandes organizaciones**) y de código abierto desarrollado por la empresa finlandesa Jolla tanto para tabletas como para dispositivos móviles.

10. **KaiOS**: basado en Linux, sucesor de Firefox OS (descontinuado) y derivado de B2G (Boot to Gecko). **Destinado a móviles básicos no táctiles y de bajo coste**.

11. **Otros SO móviles**: Symbian Os y BlackBerry OS.



Versiones Mac OS		
11.0	Big Sur	12 de noviembre de 2020
12.0	Monterey	25 de octubre de 2021
13.0	Ventura	24 de octubre de 2022
14.0	Sonoma	26 de septiembre de 2023
15.0	Sequoia	16 de septiembre 2024



## Bloque 2 - Tema 5: SGBD RELACIONALES, ORIENTADOS A OBJETOS Y NOSQL

El SGBD es un conjunto de programas, procedimientos y lenguajes, que garantiza la confidencialidad y la seguridad al proporcionar a los usuarios y administradores las herramientas necesarias.

### Funciones:

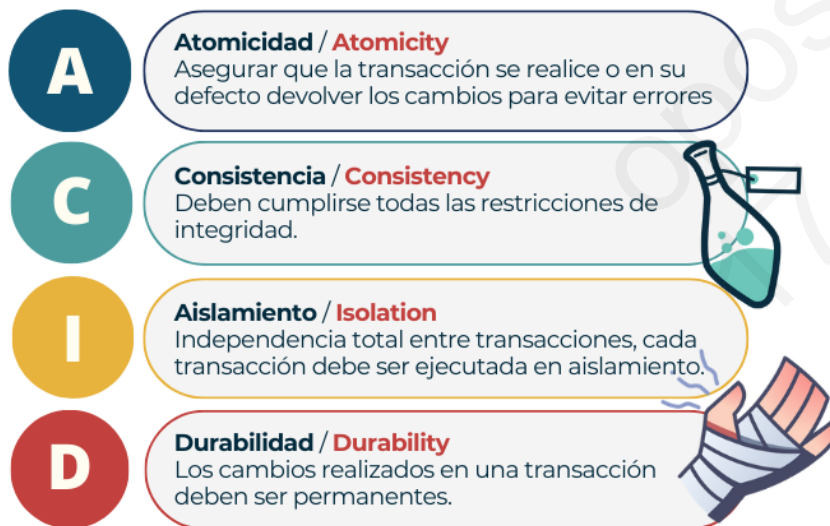
- Contribuye a una creación de base de datos más **eficiente y consistente**.
- Definen la estructura y relaciones de los datos. -> **DDL**
- Facilita la manipulación de los datos -> **DML**
- Controla los movimientos de acceso a la DB mediante privilegios -> **DCL**
- Proporciona un **diccionario de datos**.
- Ayuda a mantener la **integridad y el acceso concurrente** a los datos.
- Soporta de **transacciones**.

Un SGBD sigue la arquitectura **ANSI/X3/SPARC de 3 niveles** (lógico, conceptual y físico).

### Transacciones: integridad en las actualizaciones y acceso concurrente:

Una transacción es una **secuencia de operaciones de acceso a la base de datos** que constituye una unidad lógica de ejecución. Las transacciones se inician de forma implícita al ejecutar sentencias como CREATE, ALTER, INSERT, UPDATE o de forma explícita con la sentencia SET TRANSACTION.

Para **garantizar la seguridad de las transacciones** deben cumplirse las **propiedades ACID**.



La **finalización** de una transacción debe ser explícita mediante **COMMIT, ROLLBACK O SAVEPOINT**.

**SGBD RELACIONALES** es un modelo que **almacena datos en filas y columnas** formando tablas conectadas por claves comunes. Son **escalables verticalmente**, utilizan SQL como lenguaje y para que puedan considerarse relacional **debe cumplir al menos 6** de las 12 **reglas de Codd**. Codd estableció una regla general global, llamada “Regla Cero” y doce principios más:

### 12 reglas de Codd

Regla 0. Gestión de una Base de Datos Relacional.		
<b>Regla 1.</b> Representación de la información.	<b>Regla 5.</b> Sublenguaje global de datos.	<b>Regla 9.</b> Independencia lógica de los datos.
<b>Regla 2.</b> Acceso garantizado.	<b>Regla 6.</b> Actualización de vistas.	<b>Regla 10.</b> Independencia de la integridad.
<b>Regla 3.</b> Representación sistemática de la información que falta.	<b>Regla 7.</b> Inserciones, actualizaciones y eliminaciones de alto nivel.	<b>Regla 11.</b> Independencia de la distribución.
<b>Regla 4.</b> Catálogo dinámico.	<b>Regla 8.</b> Independencia física de los datos.	<b>Regla 12.</b> Regla de la no inversión (o no subversión).

### SGBD Relacionales:

Oracle	MySQL	Microsoft SQL Server
- De pago y multiplataforma	- Soft. Libre (GNU GPL)	- De pago y nativo*
- Estabilidad y escalabilidad	- Multiplataforma	*Soporta Linux y Docker
- Estructura física: <b>DATAFILES</b>	- Multihilo y multiusuario	- Estabilidad y escalabilidad
- Estructura lógica: <b>TABLESPACE</b>	- Soporte TLS	- Permite snapshot
- Catálogo: SYSTEM	- InnoDB: almacen. Estándar	- Estructura lógica: <b>TABLESPACE</b>
- Vista: consulta ejecutada	- MyISAM: versiones antiguas	- Vista: consulta ejecutada
- Vista materializada: consulta almacenada	- <del>Escalabilidad</del>	- Vista indexada = <b>materializada</b>
<b>SQLite</b>	<b>MariaDB</b>	<b>PostgreSQL</b>
- Lenguaje C	- Derivación de MySQL	- Concurrencias MultiVersión
- Tamaño máximo de 281 TB	- Gran escalabilidad	- Multiplataforma
- Portabilidad y rendimiento.	- Extensiones para NoSQL	- Soporta JSON
<b>Apache Derby</b> (java)	<b>MaxDB</b> (SAP)	<b>HSQLDB</b> (Base de OpenOffice)
<b>Amazon Aurora:</b> compatible con MySQL y PostgreSQL creada para la nube.		

**SGBD ORIENTADO A OBJETOS** los datos se guardan en un objeto junto con sus funciones (métodos) y los atributos. Usan el **lenguaje OQL** y son de escasa aplicación.

### SGBD Orientados a objetos:

ObjectDB	db4o	Gemstone	Zope ObjectDB (ZODB)
----------	------	----------	----------------------



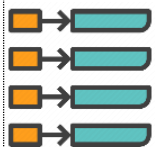
**SGBD NoSQL** Son **escalables horizontalmente**, usan un lenguaje alternativo a SQL capaz de almacenar información no estructurada en entornos distribuidos. **Basada en documentos** (no tiene ni tablas ni columnas) usa como contenido y ficheros, fotos, etc.



#### Tipos de bases de datos NoSQL

##### Clave -> valor:

Almacenan objeto binario (BLOB)



- Cassandra (Facebook)
- Redis (en ANSI C)
- DynamoDB (Amazon)
- Memcached
- Oracle NoSQL
- Aerospike y Riak

##### Grafos

Nodos, relaciones y aristas.



- Neo4j (en Java)
- AllegroGraph
- FlockDB
- InfinityGraph
- OrientDB (en Java)

##### Documentales

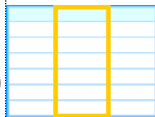
Almacenan docs (JSON o XML)



- MongoDB (BSON, C++)
- CouchDB (Apache, JSON)
- SimpleDB
- RethinkDB

##### Columnas o tabular:

Columnas en lugar de en filas.



- Bigtable (Google)
- Apache HBase (en Java)
- Cassandra (en Java)

##### IndexedDB



IndexedDB

Es una API de bajo nivel que ofrece almacenamiento persistente en el cliente, en concreto, en el navegador del usuario, de grandes cantidades de datos estructurados para **administrar bases de datos NoSQL de objetos JSON**. Permite trabajar de forma offline.

**Bases de datos de series temporales** las cuales recopilan, sintetizan y derivan información de manera eficiente a partir de datos que cambian con el tiempo y con consultas que abarcan intervalos de tiempo. La aplicación práctica de este tipo de bases de datos es: análisis de información generada por **sensores (IoT)**, flujos de clics, cotizaciones bursátiles o **telemetría**.

#### SGBD de series temporales:

AWS Timestream.	InfluxDB (código abierto).	Prometheus.
TimescaleDB	Graphite	QuestDB
OpenTSDB		