

# Introducción

## ⓘ Connection with the curriculum

RA: 1 - CE: a, b, c, d

## ¿Qué es la informática?

- La disciplina que estudia el tratamiento automático de la información.
  - Informática = **INFOR**mación auto**MÁTICA**

## Partes de un Sistema Informático

- Parte física (hardware)
- Parte lógica (software)
- Personal y usuarios
- Documentación

## Representación de la información

- El procesador(cerebro de la computadora) interpreta las instrucciones en binario (1s i 0s).

## Sistemas de representación

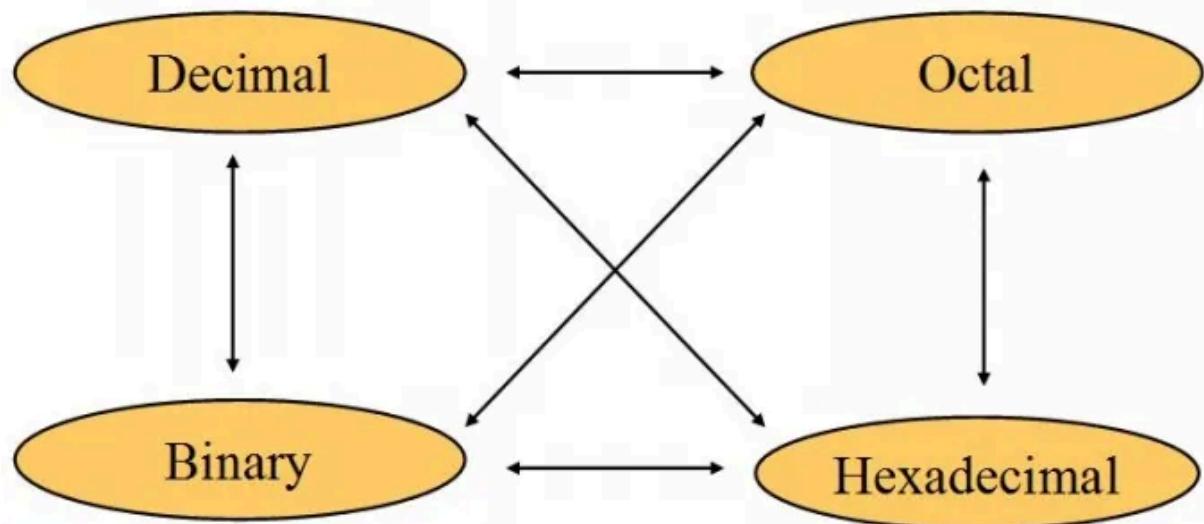
- Sistema binario (0,1)
- Sistema octal (0-7)
- Sistema decimal (0-9)
- Sistema hexadecimal (0-F)

Binari	Octal	Decimal	Hexadecim
0000	000	0	0
0001	001	1	1
0010	002	2	2
0011	003	3	3
0100	004	4	4
0101	005	5	5
0110	006	6	6
0111	007	7	7
1000	010	8	8
1001	011	9	9
1010	012	10	A
1011	013	11	B
1100	014	12	C
1101	015	13	D
1110	016	14	E
1111	017	15	F
10000	020	16	10
10001	021	17	11
10010	022	18	12
10011	023	19	13
10100	024	20	14

## Teorema fundamental de la numeración

$$\begin{aligned}
 N &= d_{n-1} \dots d_1 d_0, d_{-1} \dots d_{-k} \\
 &= d_{n-1} \cdot 10^{n-1} + \dots + d_1 \cdot 10^1 + d_0 \cdot 10^0 + d_{-1} \cdot 10^{-1} + \dots + d_{-k} \cdot 10^{-k} \\
 &= \sum_{i=-k}^{n-1} d_i \cdot 10^i
 \end{aligned}$$

$$\begin{aligned}
 3278_{10} &= 3 \times 10^3 + 2 \times 10^2 + 7 \times 10^1 + 8 \times 10^0 \\
 3278_{10} &= 3 \times \underline{1000} + 2 \times \underline{100} + 7 \times 10 + 8 \times 1
 \end{aligned}$$



## Binario a Decimal

$$101010111_2 = 1 \times 2^8 + 0 \times 2^7 + 1 \times 2^6 + 0 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$$

$$101010111_2 = 1 \times 256 + 0 \times 128 + 1 \times 64 + 0 \times 32 + 1 \times 16 + 0 \times 8 + 1 \times 4 + 1 \times 2 + 1 \times 1 = 343_{10}$$

$$101010111_2 = 256 + 0 + 64 + 0 + 16 + 0 + 4 + 2 + 1$$

$$101010111_2 = 343_{10}$$

## Decimal a Binario

### Método de divisiones sucesivas

#### Conversión de $46_{10}$ a binario

- $46 \div 2 = 23, r = 0$
- $23 \div 2 = 11, r = 1$
- $11 \div 2 = 5, r = 1$
- $5 \div 2 = 2, r = 1$
- $2 \div 2 = 1, r = 0$
- $1 \div 2 = 0, r = 1$

$$R = 101110_2$$

### Método de restas

#### Conversión de $347_{10}$ a binario

Potencia de 2	Valor	¿Cabe en el resto?	Operación	Bit
$2^8$	256	Sí	$347 - 256 = 91$	1
$2^7$	128	No	—	0
$2^6$	64	Sí	$91 - 64 = 27$	1
$2^5$	32	No	—	0
$2^4$	16	Sí	$27 - 16 = 11$	1
$2^3$	8	Sí	$11 - 8 = 3$	1
$2^2$	4	No	—	0
$2^1$	2	Sí	$3 - 2 = 1$	1
$2^0$	1	Sí	$1 - 1 = 0$	1

$$347_{10} = 101011011_2$$



Tip

[Binary Code Game](#)

## Binario a Octal (viceversa)

$1274_8$  a binario

1	2	7	4
001	010	111	100

$$1274_8 = 1010111100_2$$

$1010111100$  a un sistema octal

001	010	111	100
1	2	7	4

$$1010111100_2 = 1274_8$$

## Binario a Hexadecimal (viceversa)

001010111100 binario = 2BC<sub>16</sub>

## Hexadecimal a Octal (Intermedio)

$$\begin{array}{r} 1 \quad F \quad 4 \\ 0001 \quad 1111 \quad 0100 \\ \rightarrow \quad 1F4_{16} = 111110100_2 \\ 111110100_2 = 764_8 \end{array}$$

## Hexadecimal/Octal a Decimal

Se realiza de la misma manera que se hace de [binario a decimal](#) mediante el TFN pero con base 8 o 16

### Ejemplo de conversión de octal a decimal

Para convertir el número octal **745** a decimal:

**Multiplicación por la potencia de 8 según la posición:**

$$745_8 = 7 \times 8^2 + 4 \times 8^1 + 5 \times 8^0$$

$$745_8 = 448 + 32 + 5 = 485$$

$$745_8 = 485_{10}$$

### Ejemplo de conversión de hexadecimal a decimal

Para convertir el número hexadecimal **3FA** a decimal:

**Multiplicación por la potencia de 16 según la posición:**

$$3FA_{16} = 3 \times 16^2 + 15 \times 16^1 + 10 \times 16^0$$

$$3FA_{16} = 768 + 240 + 10 = 1018$$

$$3FA_{16} = 1018_{10}$$



Cambio de base I

# Medidas de la Información

## Medidas de la Información

### Tipos de Medidas

- Existen dos tipos de medidas para la capacidad de información:
  1. **Sistema Internacional (SI)**: Basado en unidades de 1000 en 1000.
  2. **ISO (Organización Internacional de Normalización)**: Basado en unidades de 1024 en 1024.

Aunque la medida más utilizada en la nomenclatura es la del Sistema Internacional, los equipos internamente utilizan la norma ISO. Para simplificar, se usará el Sistema Internacional.

### Medidas de la Información según el Sistema Internacional (SI)

Factor	Valor	Símbolo	Nombre
$10^0$	1	B	Byte
$10^3$	1.000	kB	Kilobyte
$10^6$	1.000.000	MB	Megabyte
$10^9$	1.000.000.000	GB	Gigabyte
$10^{12}$	1.000.000.000.000	TB	Terabyte
$10^{15}$	1.000.000.000.000.000	PB	Petabyte
$10^{18}$	1.000.000.000.000.000.000	EB	Exabyte
$10^{21}$	1.000.000.000.000.000.000.000	ZB	Zettabyte
$10^{24}$	1.000.000.000.000.000.000.000.000	YB	Yottabyte

### Medidas de la Información según la Norma ISO/IEC 80000-13

#### Note

Intento de separar la nomenclatura para que no hubiese confusión. Poca aceptación. En alguna distribución de Linux

Factor	Valor	Símbolo	Nombre
$2^0$	1	B	Byte
$2^{10}$	1.024	KiB	Kibibyte
$2^{20}$	1.048.576	MiB	Mebibyte
$2^{30}$	1.073.741.824	GiB	Gibibyte
$2^{40}$	1.099.511.627.776	TiB	Tebibyte
$2^{50}$	1.125.899.906.842.624	PiB	Pebibyte
$2^{60}$	1.152.921.504.606.846.976	EiB	Exbibyte
$2^{70}$	1.180.591.620.717.411.303.424	ZiB	Zebibyte
$2^{80}$	1.208.925.819.614.629.174.706.176	YiB	Yobibyte

```
[root@tecmint:~/vnstat-2.6]# vnstat -5
enp0s3  / 5 minute

      time     rx |   tx | total | avg. rate
-----+-----+-----+-----+
2020-09-25
 03:50    11.21 KiB |   3.96 KiB |  15.16 KiB |    414 bit/s
 03:55     7.84 KiB |   4.75 KiB |  12.59 KiB |    343 bit/s
 04:00     4.16 KiB |   4.16 KiB |   8.32 KiB |    227 bit/s
 04:05    14.82 KiB |  15.14 KiB |  29.96 KiB |    817 bit/s
 04:10     4.22 KiB |   4.22 KiB |   8.44 KiB |    230 bit/s
 04:15     3.97 KiB |   3.90 KiB |   7.87 KiB |    214 bit/s
 04:20    10.76 KiB |   9.19 KiB |  19.95 KiB |    544 bit/s
 04:25   927.72 KiB |  84.45 KiB |   0.99 MiB |  27.64 kbit/s
 04:30     3.87 KiB |   3.73 KiB |   7.61 KiB |    207 bit/s
 04:35    20.20 KiB |  26.50 KiB |  46.70 KiB |    1.27 kbit/s
 04:40     4.34 KiB |   4.29 KiB |   8.63 KiB |    235 bit/s
 04:45   169.15 KiB |  73.65 KiB | 242.80 KiB |    6.63 kbit/s
 04:50     4.48 KiB |   4.40 KiB |   8.88 KiB |    242 bit/s
```

## Conversión de Unidades

Para convertir las unidades de medida, se utilizan las siguientes operaciones:

- **Multiplicación por 1000** en cada salto hacia unidades superiores (por ejemplo, de GB a TB).
- **División por 1000** en cada salto hacia unidades inferiores (por ejemplo, de TB a GB).
- Para convertir de bits a bytes o viceversa, se utiliza:
  - **Multiplicación por 8** para pasar de bytes a bits.

- **División por 8** para pasar de bits a bytes.

### Ejemplo de Conversión:

- Convertir las siguientes unidades:
  - 1 TB, 3.000 MB y 1.000.000 B

Aplicando las conversiones, se obtiene la siguiente tabla de referencia:

b	B	KB	MB	GB	TB
8.000.000.000.000	1.000.000.000.000	1.000.000.000	1.000.000	1.000	<b>1</b>
24.000.000.000	24.000.000.000	3.000.000	<b>3.000</b>	3	0,0033
8.000.000	<b>1.000.000</b>	1.000	1.0011	0.001	0

Disco de 256GB (SI) ¿Cuánta capacidad tiene en GiB realmente (ISO)?

## Medidas de Velocidad de Transferencia

Para medir la cantidad de información que puede ser enviada o recibida por unidad de tiempo, se diferencian las siguientes medidas:

### Transferencia de datos

- **Transferencia en red:** Utiliza el bit como base y los múltiplos son de 1000.
- **Transferencia local:** Utiliza el byte como base y los múltiplos son de 1024 (como en el almacenamiento).
- **Nota:** No se aplica de manera estricta, ya que algunos fabricantes de discos duros y memorias USB usan el sistema decimal (1000) para indicar la velocidad de transferencia.

Factor	Símbolo	Nombre	Factor	Símbolo	Nombre
$10^0$	b/s	Bit por segundo	-	$2^0$	B/s
$10^3$	Kb/s	Kilobit por segundo	-	$2^{10}$	KB/s
$10^6$	Mb/s	Megabit por segundo	-	$2^{20}$	MB/s
$10^9$	Gb/s	Gigabit por segundo	-	$2^{30}$	GB/s
$10^{12}$	Tb/s	Terabit por segundo	-	$2^{40}$	TB/s

Factor	Símbolo	Nombre	Factor	Símbolo	Nombre
$10^{15}$	Pb/s	Petabit por segundo	-	$2^{50}$	PB/s
$10^{18}$	Eb/s	Exabit por segundo	-	$2^{60}$	EB/s
$10^{21}$	Zb/s	Zettabit por segundo	-	$2^{70}$	ZB/s

¿Cuánto tardará, en minutos, descargar un ADLS de 20Mbps una película de 1500MB? ¿Y una conexión de fibra de 300Mb/s(en segundos)?

Tienes que subir tu video de tu última escapada, 200GB, a Youtube para compartirlo con tus amigas

¿Cuánto tardará, en minutos u horas, si tienes una conexión de ADSL de 10Mbps? ¿cuánto tardarías si tuvieras una conexión de fibra de 600Mbps?

## Tasa de Transferencia Según el Tipo de Transmisión

Tipo de Transmisión	Tasa de Transferencia
ADSL	~ 2,5 Mbps (hasta 20 Mbps)
Fibra	~ 100 Mbps (300, 600, según contrato)
4G	~ 300 Mbps
5G	~ 10 Gbps
USB 1.0	~ 200 KB/s
USB 3.2	~ 1,2 GB/s
ATA	~ 100 MB/s
SATA 3.2	~ 2 GB/s
PCI-E v4 x16	~ 63 GB/s

Tienes una foto que has hecho con tu iPhone en RAW. ¿Cuánto tardarías en transferirla a tu ordenador si utilizas un USB 3.2 gen 2 (1.2 GB/s) si la foto pesa 25MB? ¿Y las fotos de del pasado verano en la playa (25GB)?



Financiado por  
la Unión Europea



MINISTERIO  
DE EDUCACIÓN  
Y FORMACIÓN PROFESIONAL



Plan de Recuperación,  
Transformación  
y Resiliencia



FP<sub>CV</sub>  
FP CV  
Formación Profesional  
Conselleria d'Educació, Universitats i Esports



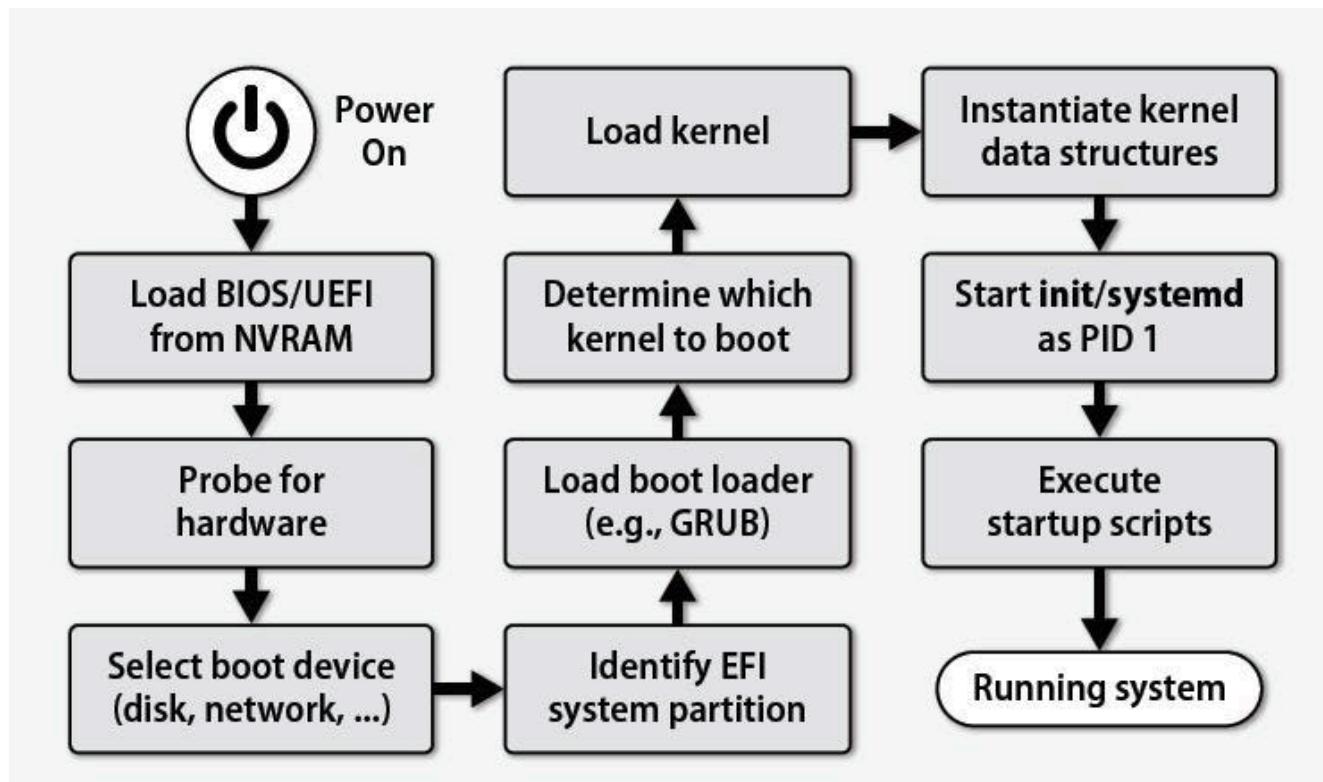
Fondos Next Generation  
en la Comunitat Valenciana



## ⚠ actividad

- Medidas de información

# POST



## Arranque de un Ordenador: POST (Power-On Self Test)

El **POST** (Power-On Self Test) es la primera fase del arranque de un ordenador. Se trata de una serie de pruebas que el sistema realiza automáticamente para verificar que los componentes de hardware esenciales funcionan correctamente antes de cargar el sistema operativo. A continuación, se describen los pasos principales del POST:

### 1. Comprobación del Hardware:

- Verifica la integridad de la memoria RAM.
- Revisa la presencia y funcionamiento de la CPU.
- Detecta dispositivos de almacenamiento (discos duros, SSD, unidades ópticas).
- Identifica y revisa la tarjeta gráfica y otros dispositivos de expansión.

### 2. Notificación de Errores:

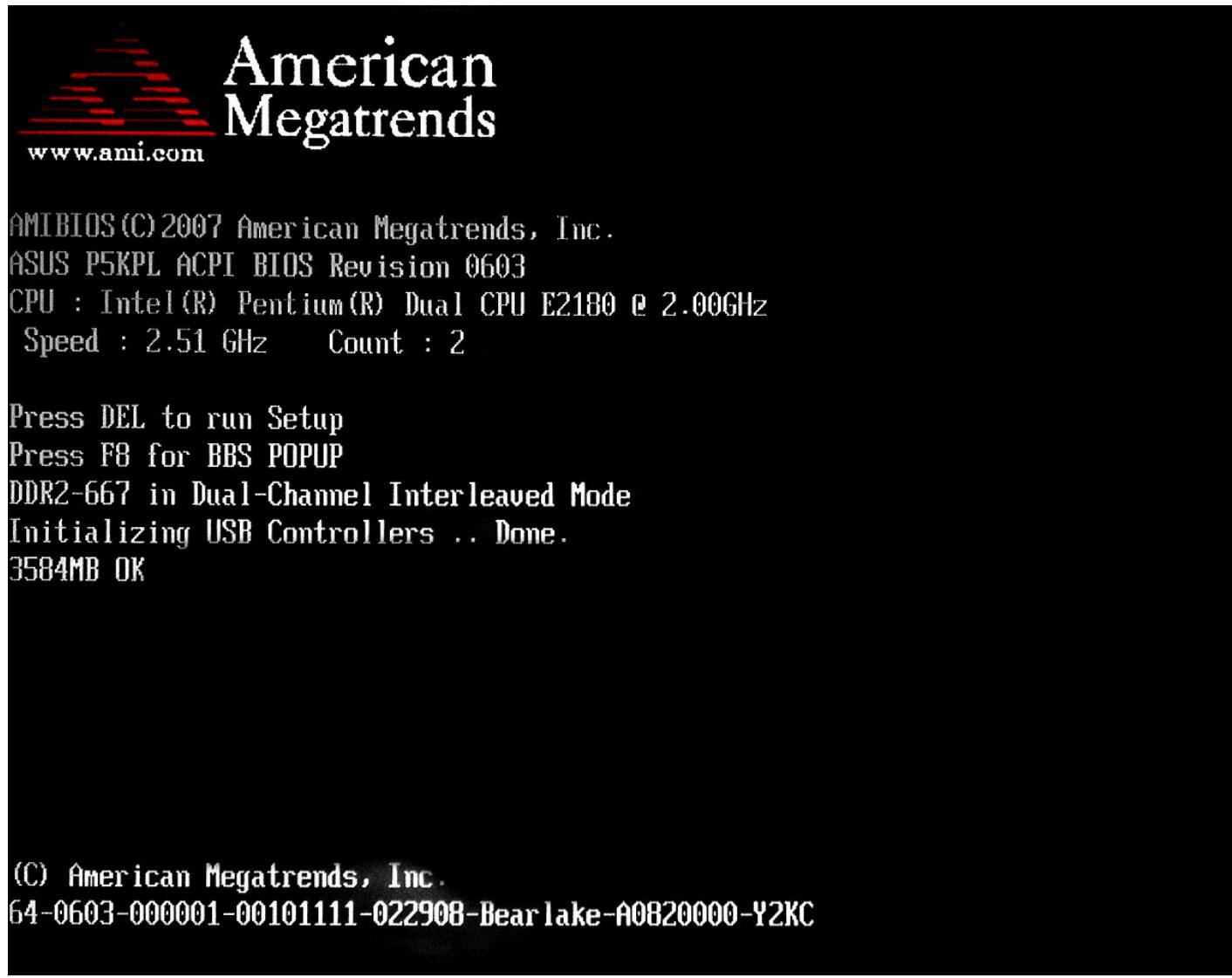
- Si se detecta algún problema, el POST emite señales acústicas (pitidos) o muestra códigos de error en la pantalla. Cada combinación de pitidos indica un fallo específico (ej.: memoria defectuosa, fallo en la CPU, etc.).

### 3. Inicialización del Hardware:

- Configura los dispositivos básicos para que puedan interactuar correctamente.
- Revisa la tabla de configuración del BIOS para confirmar que todos los dispositivos están en las ranuras o puertos correctos.

#### 4. Entrega de Control al Bootloader:

- Si todo el hardware funciona correctamente, el POST entrega el control al **bootloader** (programa encargado de cargar el sistema operativo).
- El bootloader comienza el proceso de carga del sistema operativo desde el disco duro u otro medio de arranque.



Ampliación

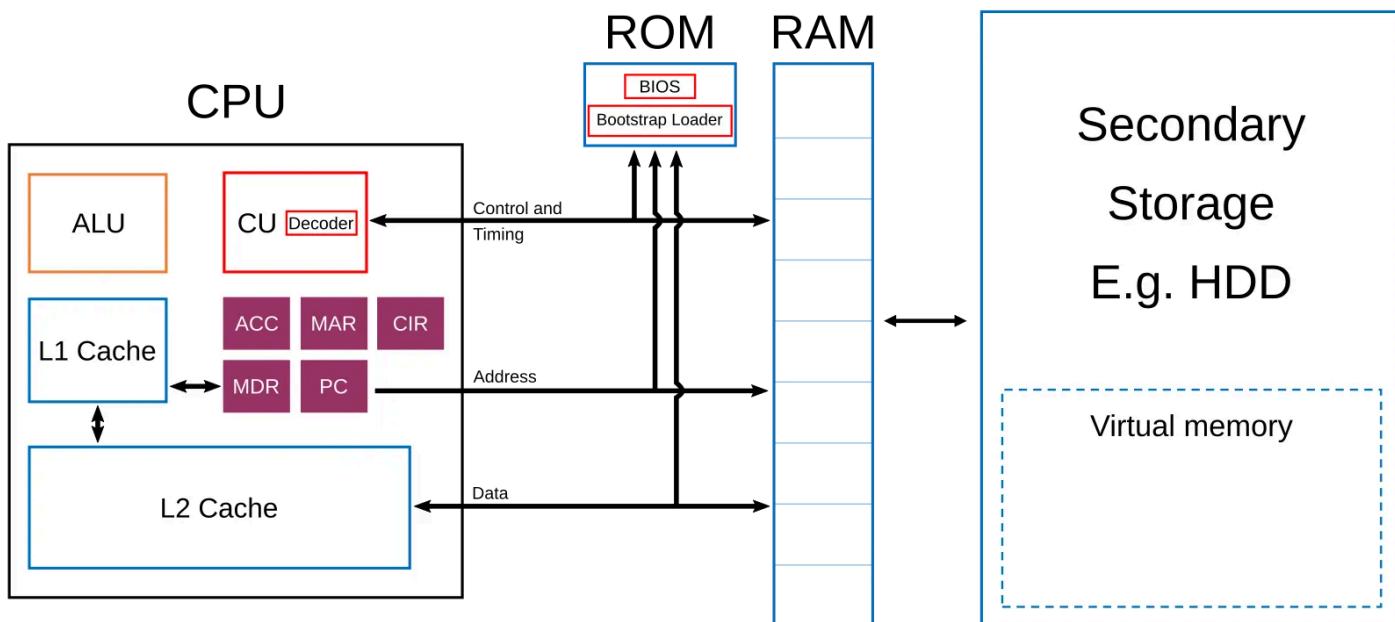
[WHY POST - EN](#)

# Arquitectura

## Arquitectura

La **arquitectura de von Neumann** es un modelo de diseño para computadores propuesto por el matemático John von Neumann en 1945. Se ha convertido en la base de casi todos los sistemas informáticos modernos.

### Computer Systems - Von Neumann Architecture



William Lau - Creative Commons - Attribution - Share-Alike - 4.0

### Componentes principales:

#### 1. Unidad Central de Procesamiento (CPU):

- Compuesta por la **Unidad de Control** (que coordina las operaciones) y la **Unidad Aritmético-Lógica (ALU)** (que realiza cálculos matemáticos y operaciones lógicas).

#### 2. Memoria:

- Almacena tanto datos como instrucciones (programas) en el mismo espacio de memoria. Esto es lo que distingue la arquitectura de von Neumann de otras, como la Harvard, que separa las instrucciones y los datos.

#### 3. Dispositivos de Entrada y Salida (E/S):

- Permiten la interacción del sistema con el exterior, facilitando la entrada de datos y la salida de resultados procesados.

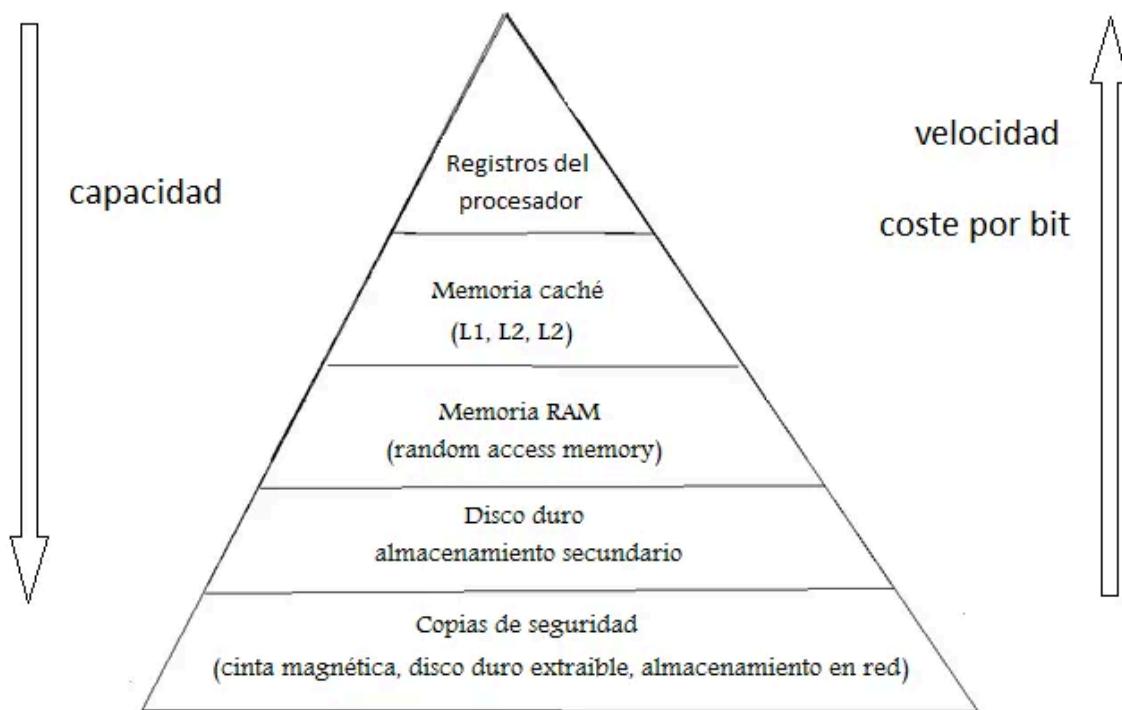
#### 4. Bus de Datos:

- Un sistema de comunicación que transporta datos, direcciones y señales de control entre los diferentes componentes.

### ⚠️ Otras arquitecturas

Busca otras arquitecturas alternativas a Von Neumann.

## JERARQUÍA DE MEMORIA DEL COMPUTADOR



## Funcionamiento básico:

### • Ciclo de búsqueda y ejecución:

1. La CPU busca una instrucción en la memoria.
2. Decodifica la instrucción.
3. Ejecuta la instrucción, ya sea realizando cálculos o moviendo datos.
4. Repite el ciclo con la siguiente instrucción.

## Características clave:

- **Almacenamiento de programas:** En lugar de tener instrucciones fijas, los programas pueden ser modificados y almacenados junto con los datos en la memoria.



MINISTERIO  
DE EDUCACIÓN  
Y FORMACIÓN PROFESIONAL



Plan de Recuperación,  
Transformación  
y Resiliencia



- **Secuencialidad:** Las instrucciones son ejecutadas una tras otra de manera secuencial, lo que permite controlarlas mediante un contador de programa.

## Ventajas:

- Simplicidad en el diseño y operación del sistema.
- Flexibilidad al permitir que el mismo hardware pueda ejecutar diferentes programas.

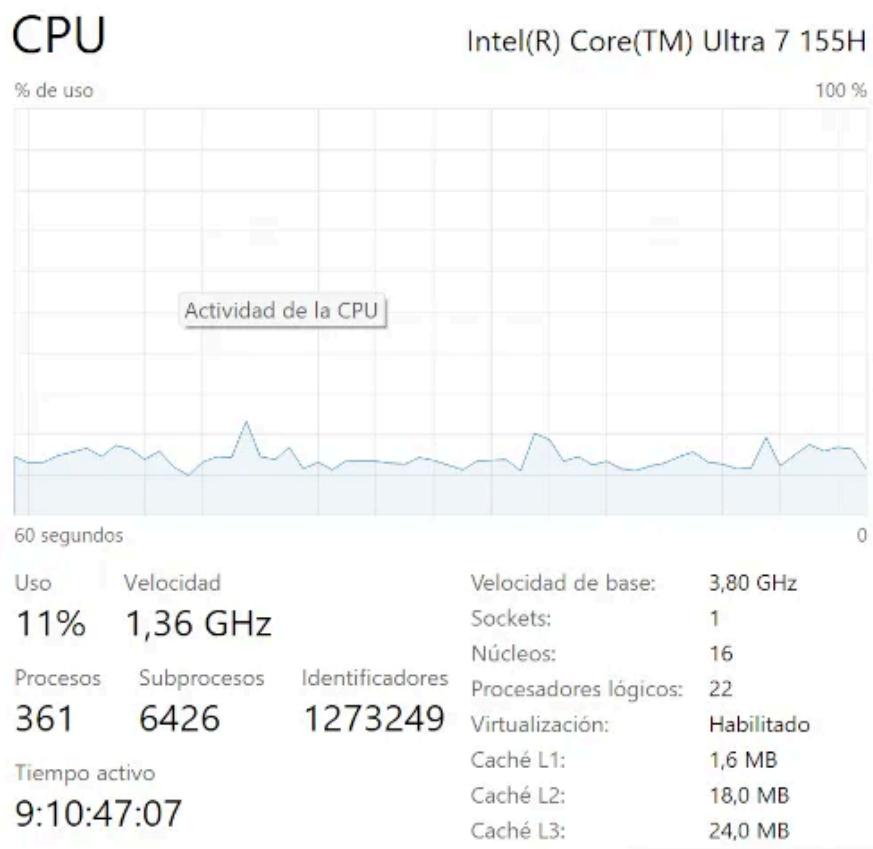
## Desventajas:

- **Cuello de botella de von Neumann:** Al compartir el mismo bus para acceder tanto a los datos como a las instrucciones, puede haber una limitación en el rendimiento, ya que la CPU puede quedar esperando mientras accede a la memoria.

# Procesador

## 1. Frecuencia (La Velocidad del Reloj)

- Se mide en **Gigahercios (GHz)**, que son miles de millones de ciclos por segundo.
- **En cristiano:** Es la velocidad a la que el procesador hace sus cálculos. Un procesador de 3.5 GHz completa 3,500 millones de operaciones cada segundo.
- **Ojo:** Aunque una frecuencia más alta suele significar más rapidez, no es el único factor que determina el rendimiento general de la CPU.



Monitor CPU de Windows 11. (ctrl + mayusculas + esc)

### Note

Un hercio (símbolo: Hz) es la unidad de medida de la frecuencia en el Sistema Internacional de Unidades (SI).

## 2. Núcleos y Hilos (Cerebros y Brazos)

- **Núcleos (Cores):** Piensa en ellos como **“cerebros” independientes** dentro del procesador. Cuantos más núcleos, más tareas pesadas puede hacer a la vez, lo que es ideal para la multitarea.

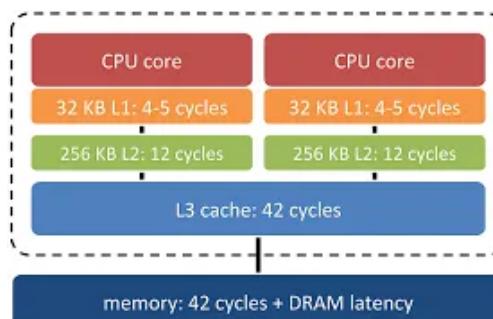
- **Hilos (Threads):** Son como los “**brazos**” de cada cerebro. Permiten gestionar las tareas dentro de un núcleo. Gracias a tecnologías como **Hyper-Threading** (Intel) o **SMT** (AMD), un solo núcleo puede manejar dos hilos a la vez, duplicando su capacidad para ciertas tareas.
- **¡Importante! Arquitecturas Híbridas:** Los procesadores modernos mezclan dos tipos de núcleos para optimizar el rendimiento y el consumo:
  - **Núcleos de Rendimiento (P-Cores):** Son los potentes, diseñados para las tareas más exigentes como juegos o edición de vídeo. En los procesadores Intel, solo los P-Cores suelen tener la capacidad de manejar dos hilos (Hyper-Threading).
  - **Núcleos de Eficiencia (E-Cores):** Son los ahorradores. Se encargan de las tareas en segundo plano (como el antivirus u ofimática) consumiendo muy poca energía.
  - Esta filosofía la usan casi todos los fabricantes: **Intel** (P-Cores/E-Cores), **AMD** (Zen 4/Zen 4c), **Apple** (Performance/Efficiency Cores) y **ARM** (big/LITTLE).

### ⚠ FYI

- Solo los P-core tienen hilos en intel.
- AMD en todos los núcleos tiene hilos.
- ARM(APPLE) no tienen hilos pero si procesos en paralelo.

## 3. Memoria Caché (La “Mesa de Trabajo” de la CPU)

- Es una **memoria súper rápida integrada en la propia CPU**. Su función es guardar los datos e instrucciones que se usan con más frecuencia para tenerlos a mano y no tener que ir a buscarlos a la memoria RAM, que es mucho más lenta.
- **Se organiza en niveles:**
  - **L1:** La más pequeña y veloz, pegada al núcleo.
  - **L2:** Un poco más grande pero algo más lenta.
  - **L3:** La más grande, compartida entre todos los núcleos.

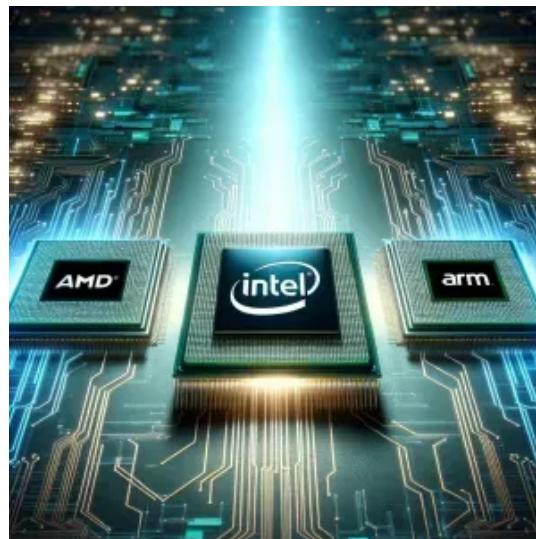


## 4. Arquitectura (El “Idioma” del Procesador)

Cuando hablamos de arquitecturas como x86 o ARM, en realidad nos referimos a su “**Conjunto de Instrucciones**”, que es como el diccionario de órdenes que el procesador entiende. Aquí es donde surgen dos filosofías de diseño totalmente opuestas: **CISC** y **RISC**.

### CISC vs RISC

Característica	CISC (Complex Instruction Set Computer)	RISC (Reduced Instruction Set Computer)
<b>Enfoque / Filosofía</b>	<b>El Enfoque “Navaja Suiza”.</b> Realizar tareas complejas con la menor cantidad de instrucciones posible.	<b>El Enfoque “Caja de Herramientas”.</b> Usar instrucciones muy simples, donde cada una hace una sola cosa de forma ultrarrápida.
<b>Tipo de Instrucciones</b>	<b>Potentes y complejas.</b> Una sola instrucción puede ejecutar varias operaciones a la vez.	<b>Simples y reducidas.</b> Se necesitan varias instrucciones para realizar una tarea compleja.
<b>Ejemplos de Uso</b>	La arquitectura <b>x86</b> , utilizada por los principales fabricantes de PC como <b>Intel</b> y <b>AMD</b> .	La arquitectura <b>ARM</b> , estándar en <i>smartphones</i> y tabletas.
<b>Ventajas</b>	- Se necesitan menos líneas de código para programar una tarea.	- Las instrucciones se ejecutan muy rápido (generalmente en un ciclo de reloj). - Consumo de energía y generación de calor muy bajos.
<b>Desventajas</b>	- Las instrucciones tardan varios ciclos de reloj en completarse. - Mayor consumo de energía y generación de calor.	- Se necesitan más instrucciones (y más memoria) para completar tareas complejas.



### actividad

- Busca los últimos procesadores de Intel, AMD y Apple. ¿Qué novedades han presentado?
- ¿Qué es x86 y x64? ¿Qué diferencias hay entre ellos?

## 5. Proceso de Fabricación (El Tamaño Sí Importa)

- Se mide en **nanómetros (nm)** y se refiere al tamaño de los transistores del chip.
- **La regla de oro:** Cuanto **más pequeño es el número (ej. 7 nm o 5 nm), mejor**. Significa que caben más transistores en el mismo espacio, lo que se traduce en un procesador más potente, que consume *menos energía y genera menos calor*.

## 6. TDP (El Calor que Genera)

- El **TDP (Thermal Design Power)** se mide en vatios (W) e indica el calor que el procesador genera cuando está funcionando a máxima carga.
- **¿Por qué es importante?** Un TDP más alto significa que el procesador consume más energía y necesitará un sistema de refrigeración más potente (un mejor disipador y ventilador) para no sobrecalentarse.

## 7. GPU Integrada (Gráficos “de serie”)

- Muchos procesadores incluyen una **Unidad de Procesamiento Gráfico (iGPU)**, es decir, una tarjeta gráfica integrada en el propio chip del procesador.
- **¿Para qué sirve?** Está pensada para tareas del día a día: navegar por internet, ofimática, reproducción de vídeo en alta definición o juegos poco exigentes.
- **Ejemplos:**
  - **Intel:** Sus procesadores con gráficos integrados suelen identificarse como **Intel UHD Graphics** o **Intel Iris Xe**, dependiendo de la gama. Los modelos con sufijo “F” (ej. Intel Core i5-12400F) **no** incluyen iGPU.
  - **AMD:** Los procesadores con gráficos integrados reciben el nombre de **APU (Accelerated Processing Unit)**. Ejemplos: **Ryzen 5 5600G** o **Ryzen 7 5700G**.

## 8. Otros Factores Clave

- **Compatibilidad con RAM:** La CPU determina el tipo y la velocidad de memoria RAM que puedes instalar en tu sistema (por ejemplo, **DDR4** o **DDR5**).
- **Overclocking:** Es la técnica para **aumentar la velocidad de reloj** de un procesador por encima de sus valores de fábrica. Es algo para usuarios avanzados, ya que genera más calor y requiere una refrigeración especial .
- **Integración:** Los procesadores modernos integran funciones que antes estaban en la placa base, como el controlador de memoria. Esto reduce la latencia y aumenta el ancho de banda, haciendo el sistema más

rápido y eficiente.

- **Fabricantes:** Los dos gigantes del mercado de PC son **Intel** y **AMD**. AMD ha destacado en los últimos años por su excelente relación rendimiento/precio con los procesadores Ryzen, mientras que Intel sigue siendo un competidor muy fuerte, especialmente en tareas que dependen de un solo núcleo .

## Ampliación

- [Understanding Intel Processor Names - URL](#)
- [Understanding AMD Processor Names - URL](#)
- [AMD and Intel processors naming](#)
- [Hiper Threading](#)

# Placa Base

La placa base es un circuito impreso con distintas formas estándar, que permiten que el resto de los componentes se conecten entre ellos

## Ampliación

[Circuito impreso \(es\)](#)

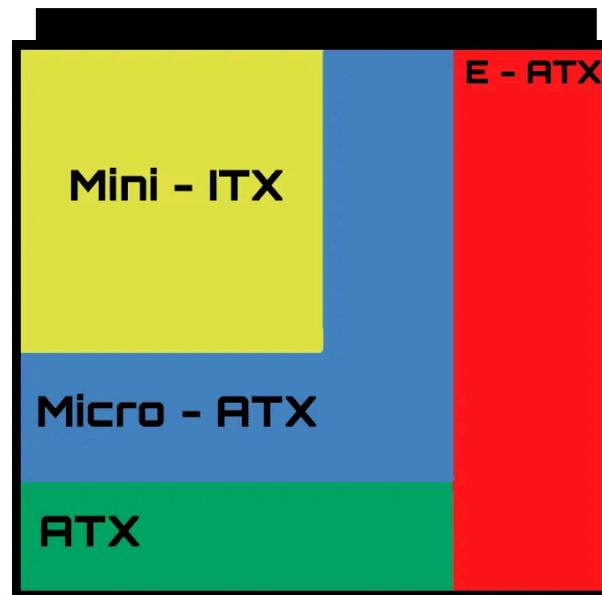
## 1. Factor de forma de una placa base

Se refiere a sus dimensiones físicas, el diseño de sus conectores y la disposición de sus componentes.

Determina qué tipo de carcasa o gabinete puede alojar la placa, así como el tipo de componentes que se pueden instalar. Los principales factores de forma para placas base son:

### Dimensiones de cada FORMATO ATX

ATX	305 x 244 mm
Mini ITX	284 x 208 mm
E - ATX	305 x 330 mm
Micro - ATX	244 x 244 mm

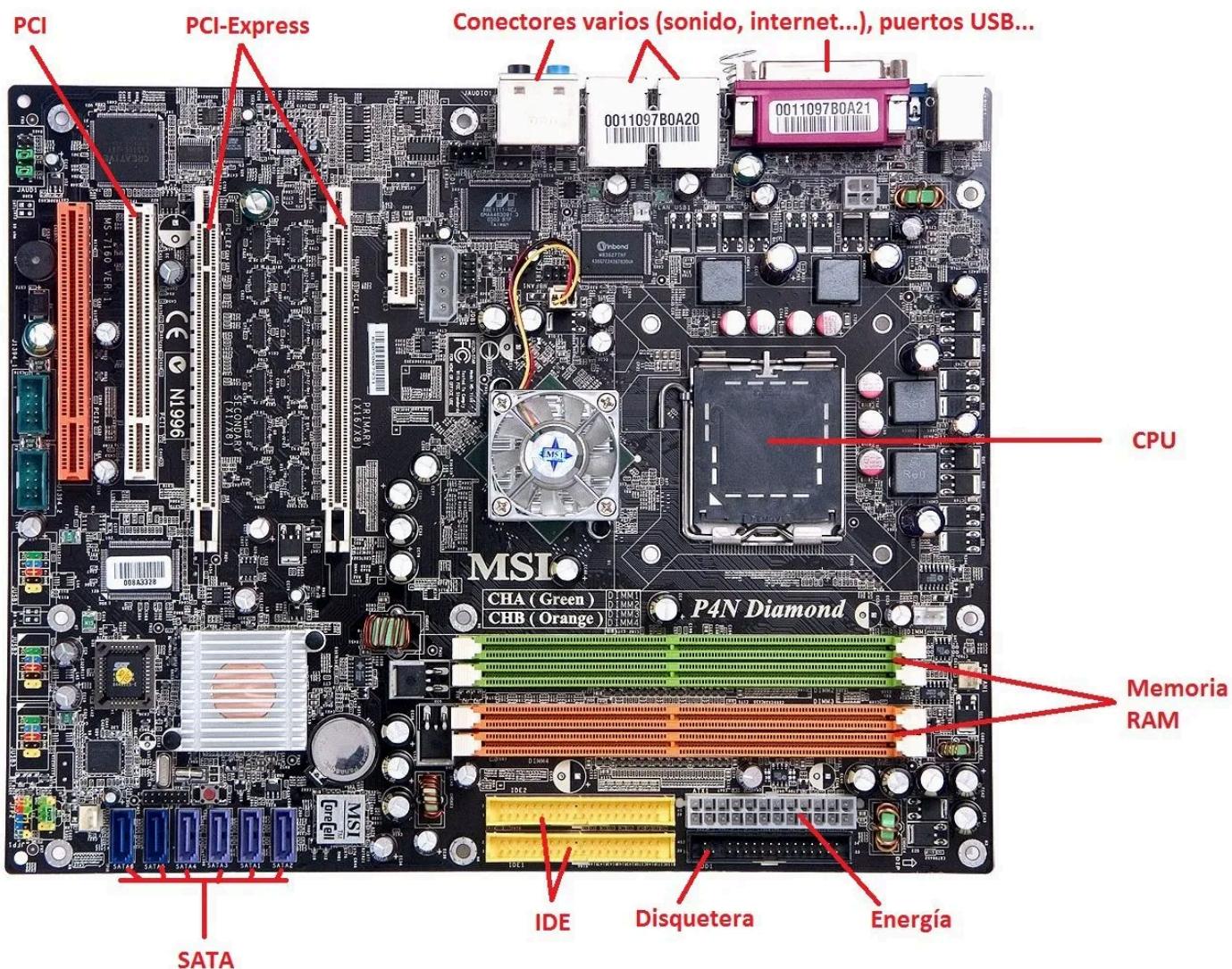


Existen otros modelos como:

FLEX ATX: 229 x 191 mm

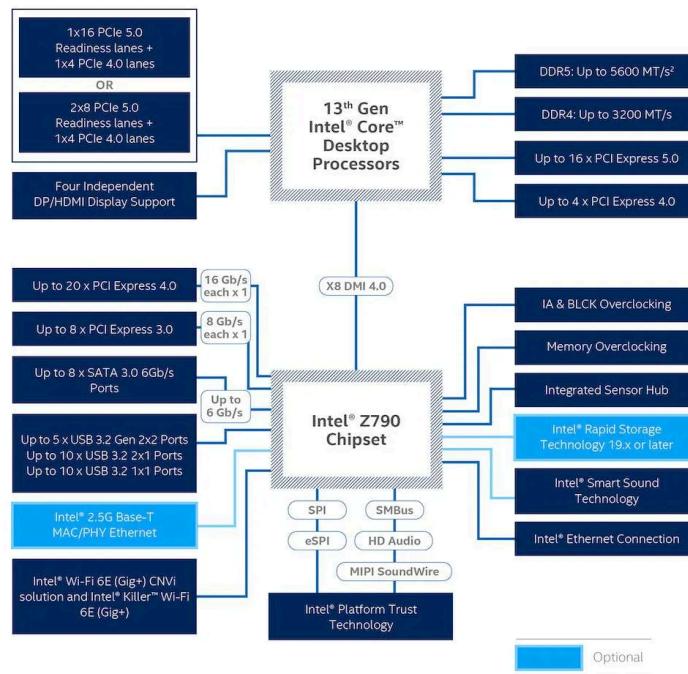
WTX: 356 x 425 mm

## 2. Componentes de la placa base



## Chipset

Chips en la placa base que actúa como un puente de comunicación entre el procesador (CPU) y otros componentes del ordenador, como la memoria RAM, almacenamiento, tarjetas gráficas, puertos USB y otros dispositivos periféricos.



## Socket

El zócalo donde se coloca el procesador, permitiendo que la CPU se comunique con el resto de los componentes del sistema a través de la placa base.

Cada procesador está diseñado para un tipo específico de socket.

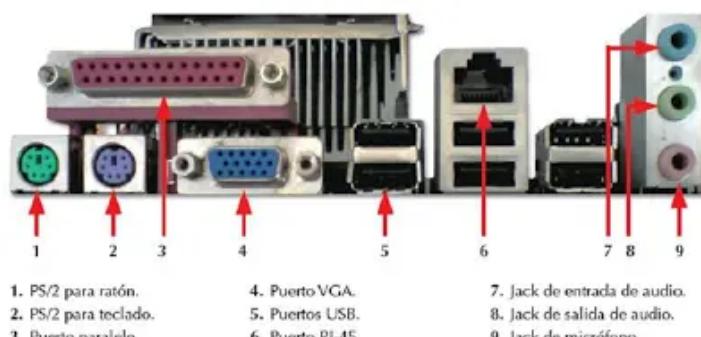
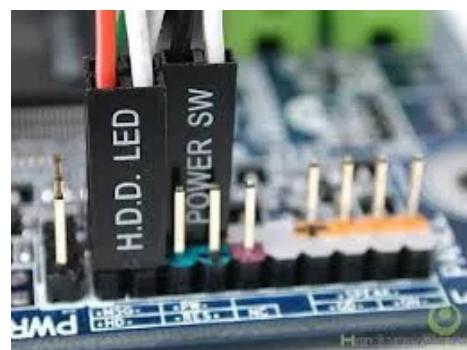
- Intel: LGA 1200, LGA 1700, etc. (el número representa la cantidad de pines del socket)
- AMD : AM4(PGA), AM5(PGA), TR4(LGA), etc.

## Refrigeración de la CPU



- Refrigeración líquida, Pasta térmica y ventilador

## Conectores frontales y posteriores



1. PS/2 para ratón.  
2. PS/2 para teclado.  
3. Puerto paralelo.  
4. Puerto VGA.  
5. Puerto USB.  
6. Puerto RJ-45.  
7. Jack de entrada de audio.  
8. Jack de salida de audio.  
9. Jack de micrófono.





### actividad

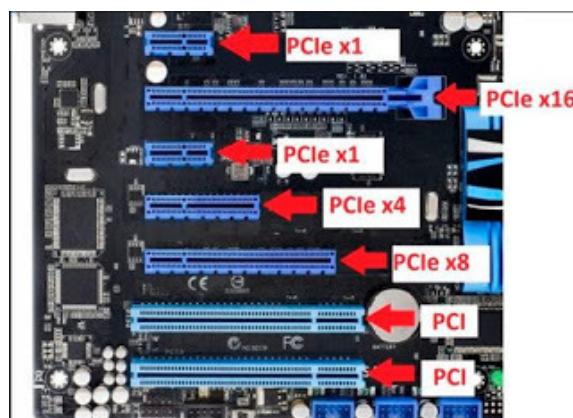
- No hay salida de HDMI, ¿A qué puede deberse?
- Busca SS10 en un USB de color rojo.
  - Haz una lista de los diferentes conectores USB-A y sus velocidades.

## PCI-Express

**PCI-Express (PCIe)** es una interfaz de conexión de alta velocidad utilizada para conectar componentes como tarjetas gráficas, SSD y tarjetas de red a la placa base.

### Características

- **Arquitectura basada en carriles:** Cada carril tiene líneas para enviar y recibir datos. Las configuraciones más comunes son x1, x4, x8 y x16.
- **Versiones y velocidad:** Con cada versión, aumenta el ancho de banda. PCIe 4.0 ofrece 2 GB/s por carril, mientras que PCIe 5.0 alcanza 4 GB/s.
- **Compatibilidad:** PCIe es retrocompatible, aunque los dispositivos funcionarán a la velocidad de la versión más baja.
- **Uso en GPUs y SSDs:** Tarjetas gráficas usan PCIe x16 para mayor ancho de banda, y los SSD NVMe aprovechan PCIe para alta velocidad.

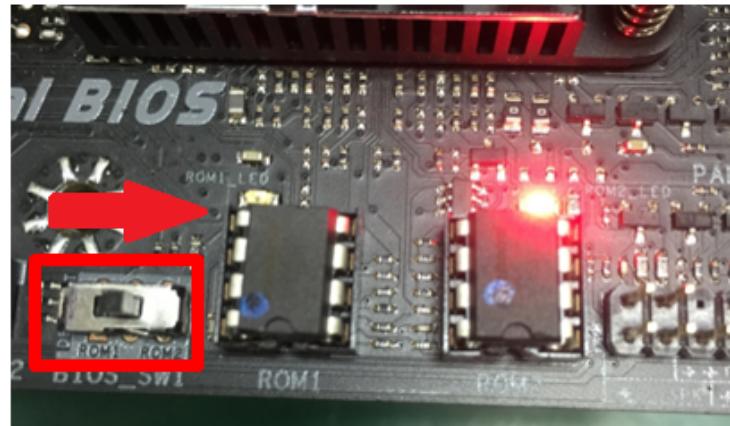


### ⚠ actividad

- Lista los PCIe más comunes y sus velocidades.
- Busca una placa base con PCIe 5.0 y otra con PCIe 4.0. y compara sus precios.

## Otros componentes

- Pila botón para BIOS/UEFI
- BIOS/UEFI



El BIOS (Basic Input/Output System) y UEFI (Unified Extensible Firmware Interface) son dos tipos de firmware que se encuentran en las placas base y se ejecutan cuando se enciende el ordenador para inicializar el hardware y arrancar el sistema operativo

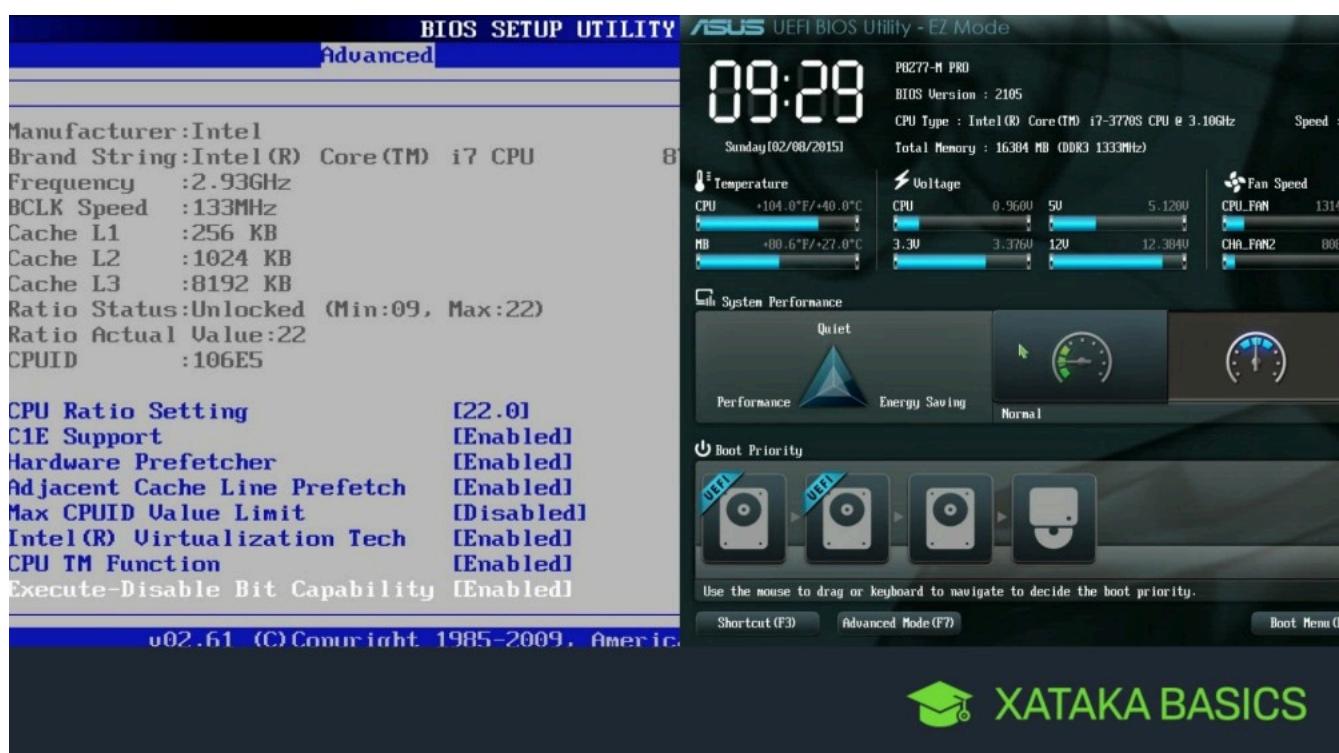


- Limitaciones BIOS:

- Soporta solo discos de hasta **2 TB** de tamaño.
- Trabaja en **modo de 16 bits**, lo que limita su velocidad y capacidades.
- Sólo puede arrancar desde discos con **MBR** (Master Boot Record).

- **Ventajas UEFI:**

- Soporta discos de más de **2 TB** y hasta **9 ZB**.
- Soporta el **arranque rápido**.
- Trabaja en **modo de 32 o 64 bits**, lo que mejora el rendimiento y la capacidad de manejo del hardware.
- Puede arrancar desde discos con **GPT** (GUID Partition Table).
- Ofrece mejores características de seguridad, como **Secure Boot**, que impide la ejecución de software malicioso durante el arranque.
- Tiene una interfaz gráfica, a menudo con soporte para ratón y más opciones configurables.



## Diferencias entre BIOS y UEFI en cuanto a tipos de memoria

- **BIOS:** Utiliza ROM o EEPROM para el código y RAM CMOS para las configuraciones. La memoria limitada restringe la cantidad de funciones disponibles.
- **UEFI:** Se almacena en memoria flash, con mayor capacidad para actualizaciones rápidas, funciones avanzadas e interfaz gráfica. Además, soporta sistemas de archivos, facilitando el acceso a discos durante el arranque.

# Caja y Fuente de Alimentación

- Chasis que engloba y protege físicamente el resto de componentes
- Condiciona al resto de componentes





## Fuente de Alimentación (Power Supply Unit - PSU)

La **fuente de alimentación** es un componente esencial en un sistema informático encargado de suministrar energía eléctrica a todos los dispositivos y componentes del ordenador. Su función principal es convertir la corriente alterna (CA) de la red eléctrica en corriente continua (CC) con los voltajes adecuados para alimentar el hardware del equipo (placa base, discos duros, tarjetas gráficas, etc.).

### Características de una Fuente de Alimentación

#### 1. Conversión de Corriente:

- Convierte la corriente alterna (AC) que se recibe de la red eléctrica en corriente continua (DC) que utilizan los componentes del ordenador.
- La conversión debe ser estable y eficiente para evitar fluctuaciones que puedan dañar el equipo.

#### 2. Voltajes de Salida:

- La fuente de alimentación proporciona diferentes voltajes según la necesidad de los componentes: 3.3V, 5V y 12V. Cada uno de estos se usa para distintos tipos de dispositivos, como el procesador, la memoria RAM y los discos duros.

#### 3. Potencia (Wattios):

- La capacidad de la fuente se mide en **wattios (W)** y determina cuánta energía puede suministrar al sistema. Una mayor potencia es necesaria en equipos con múltiples dispositivos o tarjetas gráficas de alto rendimiento.

#### 4. Conectores:

- Incluye diversos conectores para alimentar la placa base, CPU, discos duros, tarjetas gráficas, ventiladores y otros periféricos.
- Los conectores más comunes son el ATX de 24 pines, EPS de 8 pines para la CPU, SATA para discos duros y PCIe para tarjetas gráficas.

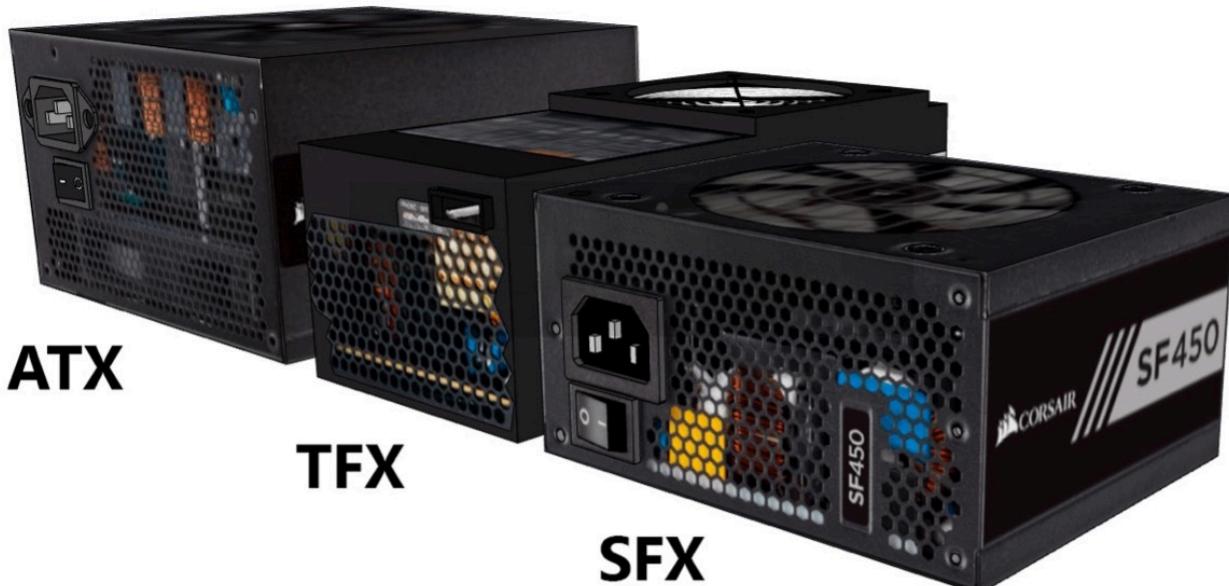


#### 5. Eficiencia Energética:

- La eficiencia de una fuente de alimentación se mide por la cantidad de energía convertida en corriente útil frente a la energía que se pierde en forma de calor.
- Las certificaciones como **80 PLUS** garantizan que la fuente tiene un mínimo de eficiencia (del 80% o superior) en diferentes niveles de carga.

Eficiencia	80 PLUS	80 PLUS BRONZE	80 PLUS SILVER	80 PLUS GOLD	80 PLUS PLATINUM	80 PLUS TITANIUM
Carga	White	Bronze	Silver	Gold	Platinum	Titanium
20%	80%	82%	85%	87%	90%	94%
50%	80%	85%	88%	90%	92%	96%
100%	80%	82%	85%	87%	97%	91%

### Tipos de Fuentes de Alimentación



### 1. ATX (Advanced Technology eXtended):

- Es el tipo más común para ordenadores de sobremesa. Proporciona todos los conectores y voltajes necesarios para alimentar la mayoría de componentes de un PC estándar.

### 2. SFX (Small Form Factor):

- Utilizadas en equipos compactos o de pequeñas dimensiones, tienen un tamaño reducido pero siguen ofreciendo potencia similar a las ATX.

### 3. Fuentes de Alimentación Modulares y Semi-Modulares:

- Permiten conectar solo los cables necesarios, lo que ayuda a mantener el interior del equipo más ordenado y facilita la ventilación.

## Funciones de Protección

Las fuentes de alimentación modernas incluyen varios mecanismos de protección para evitar daños a los componentes:

- **Protección contra sobretensiones (OVP):** Apaga la fuente si el voltaje supera el límite establecido.
- **Protección contra sobrecorriente (OCP):** Evita que una línea reciba más corriente de la que puede soportar.
- **Protección contra cortocircuitos (SCP):** Apaga el sistema si se detecta un cortocircuito.



MINISTERIO  
DE EDUCACIÓN  
Y FORMACIÓN PROFESIONAL



Plan de Recuperación,  
Transformación  
y Resiliencia



## Ampliación

[Fuentes de alimentación \(es\)](#)

### actividad

Practicum I - Placas Base y Procesadores

# Memoria RAM

La **memoria RAM** (Random Access Memory) es una memoria de acceso rápido y volátil que almacena temporalmente datos e instrucciones que el procesador necesita mientras se ejecutan programas o aplicaciones.

## Características clave

- Volátil:** Pierde su contenido al apagar el ordenador.
- Acceso rápido:** Es mucho más veloz que las unidades de almacenamiento, como discos duros o SSD.
- Función principal:** Almacena datos e instrucciones temporalmente para que el procesador acceda a ellos rápidamente, mejorando el rendimiento del sistema.

## Factor de forma

El factor de forma de la RAM DDR se refiere a su tamaño físico y disposición de pines, siendo comúnmente **DIMM** para ordenadores de sobremesa y **SO-DIMM** para portátiles y dispositivos compactos.



## Tipos de RAM

## DRAM (Dynamic RAM)

- Descripción:** La memoria RAM dinámica almacena los datos en condensadores, que deben refrescarse constantemente para mantener la información.
- Uso:** Es el tipo de RAM más común en ordenadores y dispositivos móviles.
- Tipos derivados:**
  - SDRAM (Synchronous DRAM):** Sincronizada con el reloj del sistema, lo que mejora su eficiencia.
  - DDR (Double Data Rate SDRAM):** Evolución de la SDRAM que transfiere datos en ambos flancos del ciclo de reloj, duplicando la tasa de transferencia.

### 1.1. DDR (Double Data Rate)

Tipo de RAM	Año de lanzamiento	Frecuencia de reloj (MHz)	Transferencia de datos (MT/s)	Velocidad de transferencia	Voltaje	Nº de pines
DDR	2000	100 – 200	200 – 400	1,6 – 3,2 GB/s	2,5 V	184
DDR2	2003	133 – 266	400 – 1066	3,2 – 8,5 GB/s	1,8 V	240
DDR3	2007	400 – 1066	800 – 2133	6,4 – 17 GB/s	1,5 V	240
DDR4	2014	800 – 1600	1600 – 3200	12,8 – 25,6 GB/s	1,2 V	288
DDR5	2020	1200 – 2100	4800 – 8400+	38,4 – 67,2 GB/s (o más)	1,1 V	288

### Velocidades: MHz vs MT/s

9.200 MT/s significa **9.200 millones de transferencias por segundo** (*MegaTransfers per Second*).

- ◆ **MT/s (MegaTransfers por segundo)** mide la **velocidad efectiva de transferencia de datos** de la memoria, **no la frecuencia real del reloj**. En las memorias **DDR (Double Data Rate)**, se realizan **dos transferencias por ciclo de reloj** (una en cada flanco), por lo que la velocidad en MT/s es **el doble de la frecuencia base**.

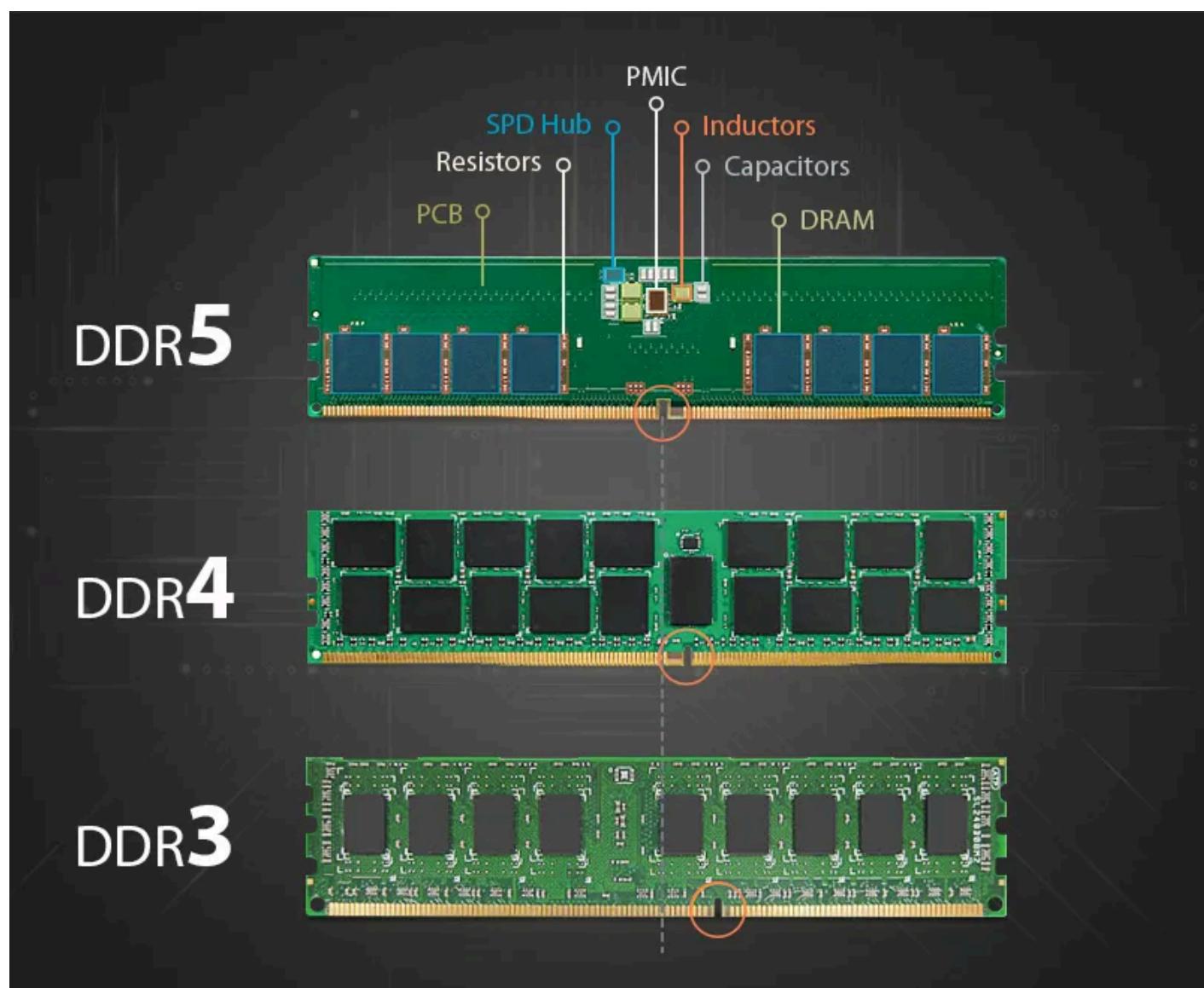
## Ejemplo práctico:

Si una memoria DDR5 trabaja a **4.600 MHz** de frecuencia real:

$$4.600 \text{ MHz} \times 2 = 9.200 \text{ MT/s}$$

Eso significa que puede realizar **9.200 millones de transferencias de datos por segundo**.

Concepto	Significado	Ejemplo
<b>MHz (frecuencia)</b>	Ciclos de reloj por segundo	4.600 MHz
<b>MT/s (transferencias)</b>	Operaciones de transferencia de datos por segundo	9.200 MT/s
<b>Relación en DDR</b>	$1 \text{ MHz} = 2 \text{ MT/s}$	DDR = "Double Data Rate"



## 1.2. LPDDR (Low Power DDR)

- **Descripción:** Variante de la memoria DDR diseñada para dispositivos móviles y portátiles.
- **Ventaja:** Optimizada para consumir menos energía, ideal para mejorar la duración de la batería.
- **Versiones:** Existen LPDDR2, LPDDR3, LPDDR4 y la más reciente **LPDDR5**, cada una con mejoras en velocidad y eficiencia energética.

## 1.3. GDDR (Graphics DDR)

- **Descripción:** Versión especializada de DDR diseñada específicamente para **tarjetas gráficas**.
- **Uso:** Utilizada en GPUs debido a sus altos requisitos de ancho de banda.
- **Versiones:** Se actualiza con versiones como **GDDR5** o **GDDR6**, ofreciendo mayor rendimiento gráfico.

## SRAM (Static RAM)

- **Descripción:** No necesita refrescarse constantemente como la DRAM, por lo que es más rápida, pero también más cara y ocupa más espacio.
- **Uso:** Se emplea principalmente en la CMOS de la BIOS y en las memorias caché de CPU y GPU.

#### Comparación de Transistores por Celda de Bit

Tipo de Memoria	Componentes por Bit	Función del Componente Adicional
<b>SRAM</b>	<b>6 Transistores</b>	Los transistores (formando un <i>flip-flop</i> biestable) mantienen el estado del bit (0 o 1) sin necesidad de refresco constante.
<b>DRAM</b>	<b>1 Transistor y 1 Condensador</b>	El transistor actúa como interruptor y el condensador almacena la carga eléctrica (el estado del bit), la cual debe ser refrescada periódicamente.

#### ¿Por que no RAM/SSD/HDD con SRAM?

- No se usa la SRAM como memoria principal debido a su alto costo y tamaño.

## Canales

Los **canales de memoria RAM** son las “autopistas” que conectan el procesador con la memoria, permitiendo el intercambio de datos. Más canales significan mayor ancho de banda y, por tanto, mejor rendimiento, especialmente en tareas intensivas como juegos o edición de vídeo.

- **Single Channel:** Un módulo de RAM activo; menor rendimiento.
- **Dual Channel:** Dos módulos trabajando en paralelo; duplica el ancho de banda.
- **Triple Channel:** Tres módulos funcionando de forma coordinada; mejora el rendimiento respecto a Dual Channel, aunque fue un estándar limitado a algunas plataformas (como Intel X58).
- **Quad Channel:** Cuatro módulos; maximiza el ancho de banda, usado en estaciones de trabajo y servidores.

Hoy en día, **Dual Channel** y **Quad Channel** son los más comunes, mientras que el **Triple Channel** ha caído en desuso.

## Frecuencias

Las **frecuencias de la RAM** indican la velocidad a la que la memoria puede leer y escribir datos. Se miden en **MHz** (megahercios) y afectan directamente al rendimiento del sistema.

## Características

1. **Frecuencia base:** Indica cuántos ciclos por segundo puede manejar la RAM. Por ejemplo, una memoria de **3200 MHz** puede realizar 3,2 mil millones de ciclos por segundo.
2. **Rendimiento:** Cuanto mayor sea la frecuencia, más rápido puede transferir datos la RAM, mejorando el rendimiento en tareas exigentes.
3. **Latencia:** A mayor frecuencia puede aumentar la latencia, aunque en general las frecuencias más altas compensan con mayor ancho de banda.
4. **Compatibilidad:** La velocidad máxima de la RAM está limitada por la **placa base** y el **procesador**.
5. **Overclocking:** Algunas memorias permiten aumentar su frecuencia por encima de la especificada, lo que puede requerir ajustes en la BIOS/UEFI.

## Ejemplo de cálculo de tasa de transferencia / ancho de banda

Supongamos una RAM DDR4 con las siguientes características:

- **Frecuencia de reloj:** 3200 MHz
- **Bus de datos:** 64 bits
- **Número de canales:** 2 (Dual Channel)

$$\text{Ancho de Banda (GB/s)} = \text{Frecuencia (MHz)} \times \text{Ancho de Bus (Bytes)} \times \text{Número de Canales}$$

- Tasa de transferencia (MB/s) =  $3200 \text{ MHz} \times 64 \text{ bits} \times 2 \text{ canales} / 8 = 51.200 \text{ MB/s} \Rightarrow 51,2 \text{ GB/s}$

Esto significa que la tasa de transferencia de esta RAM es de **51,2 GB/s**.

## Ampliación

## ⚠ Ampliación

- **CUDIMM (Consumer UDIMM)**: Nueva especificación de memoria RAM DDR5 que mejora la integridad de la señal y permite alcanzar velocidades superiores a DDR5-6400, llegando hasta 9.200 MT/s. Es compatible con plataformas DDR5 actuales y está orientada a equipos de sobremesa de alto rendimiento. Los primeros módulos han sido fabricados por V-Color, con kits de hasta 48 GB y velocidades entre 6.400 y 9.200 MT/s.

Para más información, visita [CUDIMM](#).

- **CAMM2 (Compression Attached Memory Module 2)**: Nueva especificación de memoria RAM diseñada para sistemas de sobremesa y servidores, que ocupa hasta un 60 % menos de espacio que el estándar anterior. Soporta configuración **Dual Channel** en un único módulo, reduce el consumo de energía hasta un 40 % y mejora las latencias.
- **LPCMAMM2 (Low Profile CAMM2)**: Variante de CAMM2 orientada a portátiles, con un diseño compacto. Ofrece capacidades de 64 GB, 96 GB y 128 GB según la densidad de la DRAM, y es compatible con los requisitos de eficiencia energética de la Unión Europea.

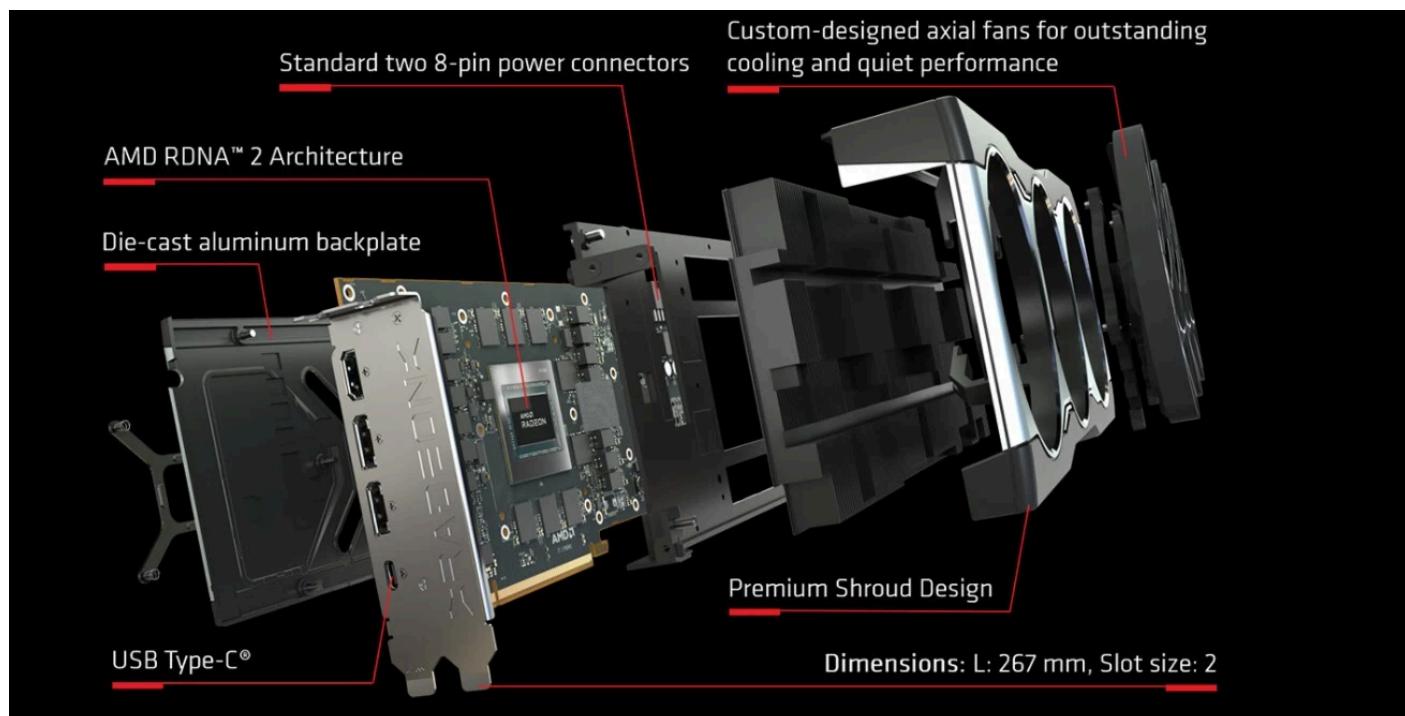
Para más información, visita [CAMM2/LPCMAMM2](#).

## 🎥 Videos

- [¿Cómo influyen la latencia y la frecuencia en la memoria RAM? \(hasta minuto 2:30\)](#).
- [¿Qué es la memoria RAM? | Explicación sencilla \(Nate Gentile\)](#)

# GPU

La GPU (**Graphics Processing Unit**) es el procesador dedicado a manejar y acelerar tareas gráficas, como el procesamiento de imágenes, videos y gráficos 3D. Aunque su función principal es renderizar gráficos, también se utiliza en otros tipos de procesamiento paralelo intensivo, como inteligencia artificial y computación científica.



## Características clave de una GPU:

- Arquitectura paralela:** A diferencia de la CPU, que está diseñada para procesar unas pocas tareas rápidamente, la GPU está optimizada para ejecutar miles de pequeñas tareas de forma simultánea, ideal para renderizado de gráficos.
- Memoria dedicada (VRAM):** La mayoría de las GPUs tienen su propia memoria RAM (VRAM), que almacena los datos gráficos para acceso rápido. Las tarjetas de gama alta suelen tener más VRAM, mejorando su rendimiento en tareas de alta demanda.
- Cores o núcleos CUDA/Stream:** Las GPUs tienen miles de núcleos más pequeños que los de una CPU, lo que les permite procesar muchas operaciones en paralelo. En Nvidia se llaman núcleos **CUDA** y en AMD **Stream Processors**.
- Uso en gaming:** En los videojuegos, las GPUs son responsables del renderizado de gráficos complejos en tiempo real, como texturas, sombras y luces.
- Uso en tareas de cálculo:** Además de gráficos, las GPUs se utilizan para tareas como **machine learning**, **minería de criptomonedas** y **simulaciones** científicas, ya que pueden manejar grandes volúmenes de datos en paralelo de manera eficiente.

**6. Ray Tracing:** Las GPUs modernas soportan tecnologías avanzadas como el **ray tracing**, que simula cómo la luz interactúa con los objetos, mejorando la calidad gráfica en juegos y aplicaciones de diseño.

Existen varios tipos de **memoria gráfica** que se utilizan en las **tarjetas gráficas** (GPU). Estas memorias están diseñadas para almacenar datos gráficos como texturas, modelos 3D, y realizar tareas de renderizado de manera rápida y eficiente.



## Principales tipos de memoria gráfica:

### 1. VRAM (Video RAM):

- Es un tipo de memoria que se encuentra en las GPUs y está dedicada al almacenamiento de datos gráficos.
- Funciona como un buffer entre la GPU y la pantalla, proporcionando acceso rápido a los datos visuales.

### 2. GDDR (Graphics Double Data Rate):

- Es una variante de la memoria DDR, diseñada específicamente para tarjetas gráficas.
- **Versiones comunes:**
  - **GDDR6:** El estándar actual en muchas GPUs modernas, con mayor rendimiento y eficiencia energética que GDDR5.

- **GDDR6X:** Utilizada en algunas GPUs de gama alta, como la Nvidia RTX 3000, ofrece mayor velocidad y rendimiento.

### 3. HBM (High Bandwidth Memory):

- Es un tipo de memoria de alto ancho de banda y baja latencia, diseñada para GPUs de alto rendimiento.
- **HBM2:** Versión mejorada que ofrece aún más ancho de banda, usada en GPUs de estaciones de trabajo y algunos modelos de gama alta como las series Radeon de AMD.

### 4. DDR/DDR2/DDR3/DDR4/DDR5/LPDDR(Low Power DDR):

- En los procesadores que incorporan la unidad gráfica, se usa la del sistema.

## Conectores de alimentación para tarjetas gráficas

### 1. Conector PCIe de 6 pines:

- Proporciona hasta **75W**.
- Usado en tarjetas gráficas de gama media.

### 2. Conector PCIe de 8 pines:

- Proporciona hasta **150W**.
- Común en tarjetas gráficas de gama alta.

### 3. Conector PCIe de 12 pines:

- Introducido por Nvidia para las RTX 3000.
- Proporciona hasta **300W**.

### 4. Conector PCIe de 16 pines (12+4 pines):

- Usado en GPUs recientes como las RTX 4000.
- Proporciona hasta **600W**.



### actividad

Practicum II

# Almacenamiento Local

## Disco Duro (HDD) vs Unidad de Estado Sólido (SSD)

Los **discos duros (HDD)** y las **unidades de estado sólido (SSD)** son dos tipos de almacenamiento que cumplen la misma función principal: **guardar y recuperar datos**. Sin embargo, sus características y funcionamiento son muy diferentes, lo que afecta directamente al rendimiento, la durabilidad y el coste. A continuación se presenta una comparativa de sus diferencias más relevantes:

Característica	Disco Duro (HDD)	Unidad de Estado Sólido (SSD)
Tecnología	Utiliza platos magnéticos giratorios y un cabezal lector/escritor que se mueve físicamente para acceder a los datos.	Utiliza memoria flash NAND, lo que permite leer y escribir datos de forma electrónica sin piezas móviles.
Velocidad de Lectura/Escritura	Menor velocidad de lectura y escritura debido a las limitaciones mecánicas.	Velocidad significativamente superior debido a la ausencia de partes móviles y tiempos de acceso más rápidos.
Tiempo de Arranque	Más lento (segundos) debido al tiempo necesario para que los platos giren a la velocidad adecuada.	Más rápido (milisegundos), ideal para tiempos de arranque casi instantáneos.
Durabilidad y Resistencia	Mayor riesgo de daños físicos por golpes y vibraciones debido a sus partes móviles.	Resistente a golpes y vibraciones, ya que no tiene partes móviles.
Ruido y Temperatura	Produce ruido durante su funcionamiento y genera más calor.	Sin ruido y genera menos calor durante su operación.
Capacidad de Almacenamiento	Disponible en mayores capacidades (hasta 20 TB o más), ideal para almacenar grandes cantidades de datos.	Menor capacidad a un precio accesible (generalmente de 500 GB a 2 TB en el mercado común).
Precio	Más económico por GB de almacenamiento.	Más caro por GB de almacenamiento debido a la tecnología avanzada.
Consumo de Energía	Mayor consumo de energía debido al motor y a las piezas móviles.	Menor consumo de energía, ideal para portátiles y dispositivos móviles.

Característica	Disco Duro (HDD)	Unidad de Estado Sólido (SSD)
<b>Tiempo de Vida Útil</b>	Larga vida útil con múltiples ciclos de lectura y escritura, pero susceptible al desgaste mecánico.	Vida útil limitada por el número de ciclos de escritura, aunque mejora con cada generación de NAND.
<b>Uso Ideal</b>	Almacenamiento de datos grandes a bajo coste, como archivos multimedia, copias de seguridad o servidores de archivos.	Sistemas operativos, aplicaciones y juegos, donde la velocidad y tiempos de carga son cruciales.
<b>Peso y Tamaño</b>	Mayor tamaño y peso, especialmente en discos de alta capacidad.	Más ligero y compacto, adecuado para equipos portátiles y ultrabooks.

## Red

### 1. Discos duros externos:

- Dispositivos de almacenamiento portátiles que se conectan al sistema mediante puertos como USB o Thunderbolt. Son ideales para hacer copias de seguridad, transferir datos y aumentar la capacidad de almacenamiento.

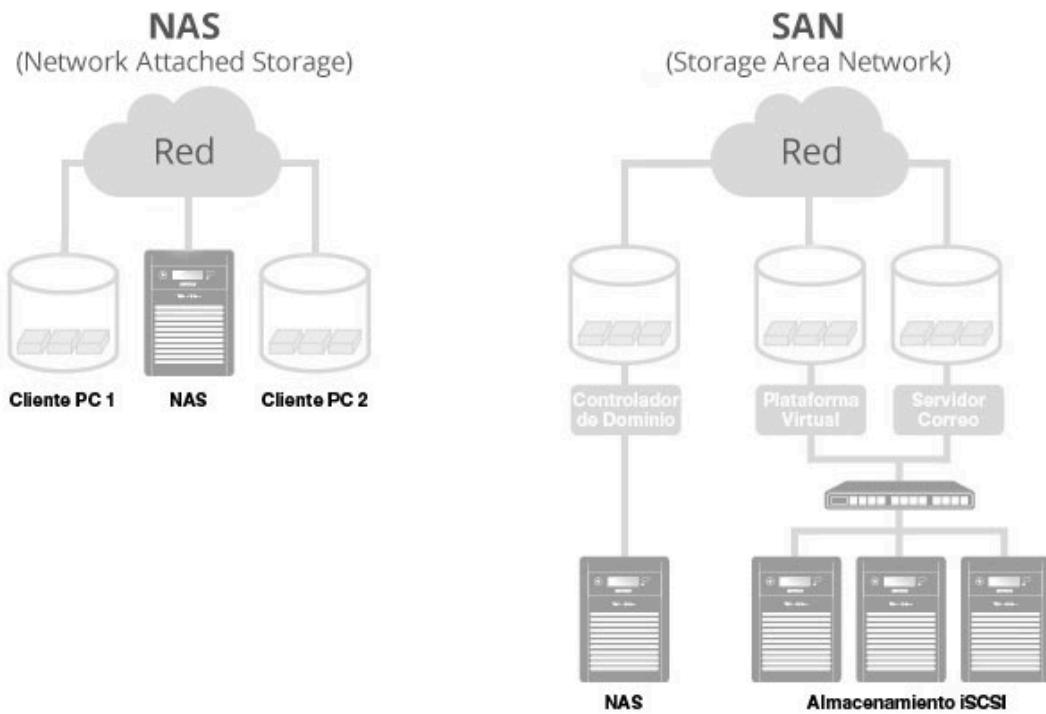
### 2. Sistemas NAS (Network Attached Storage):

- Almacenamiento conectado a la red. Un NAS permite a varios dispositivos acceder a archivos centralizados a través de una red. Marcas comunes como Synology o QNAP ofrecen soluciones para uso doméstico o empresarial.



### 3. Sistemas SAN (Storage Area Network):

- Una SAN es una red de alta velocidad que conecta múltiples dispositivos de almacenamiento con servidores, proporcionando acceso a bloques de datos. Se utiliza en entornos empresariales para grandes volúmenes de datos.



### 4. Dispositivos ópticos y cinta:

- Incluye medios como CD, DVD, Blu-ray y cintas magnéticas, que se utilizan para almacenamiento de datos a largo plazo o copias de seguridad. Aunque menos comunes hoy en día, siguen siendo una opción



Financiado por  
la Unión Europea



MINISTERIO  
DE EDUCACIÓN  
Y FORMACIÓN PROFESIONAL



Plan de Recuperación,  
Transformación  
y Resiliencia



GENERALITAT  
VALENCIANA  
Consejería de Educación,  
Universidades y Empleo



FP CV  
Formación Profesional  
Conselleria d'Ensenyament

GVA NEXT

Fondos Next Generation  
en la Comunitat Valenciana



viable para almacenamiento offline y seguro.

## 88 Ampliació

### NAS VS SAN

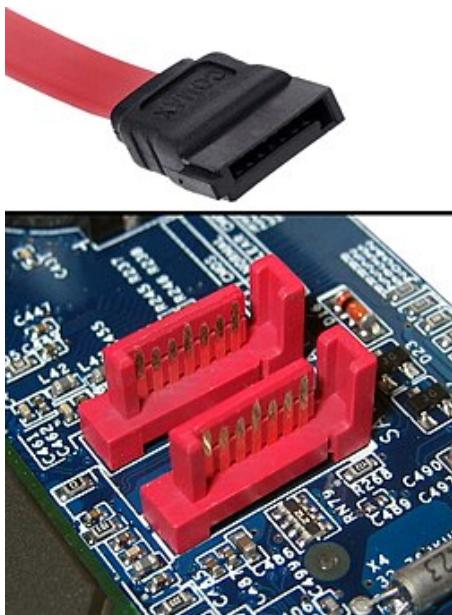
# Interfaces y Conectores

## Interfaces de almacenamiento

Una **interfaz** es el punto de conexión y comunicación entre diferentes componentes de un sistema informático, como dispositivos de hardware o periféricos, que permite el intercambio de información y la interoperabilidad. En un PC, las interfaces pueden ser **físicas (puertos USB, SATA, PCIe)** o **lógicas (drivers, protocolos)** y facilitan la interacción y el funcionamiento conjunto de los diferentes elementos del sistema.

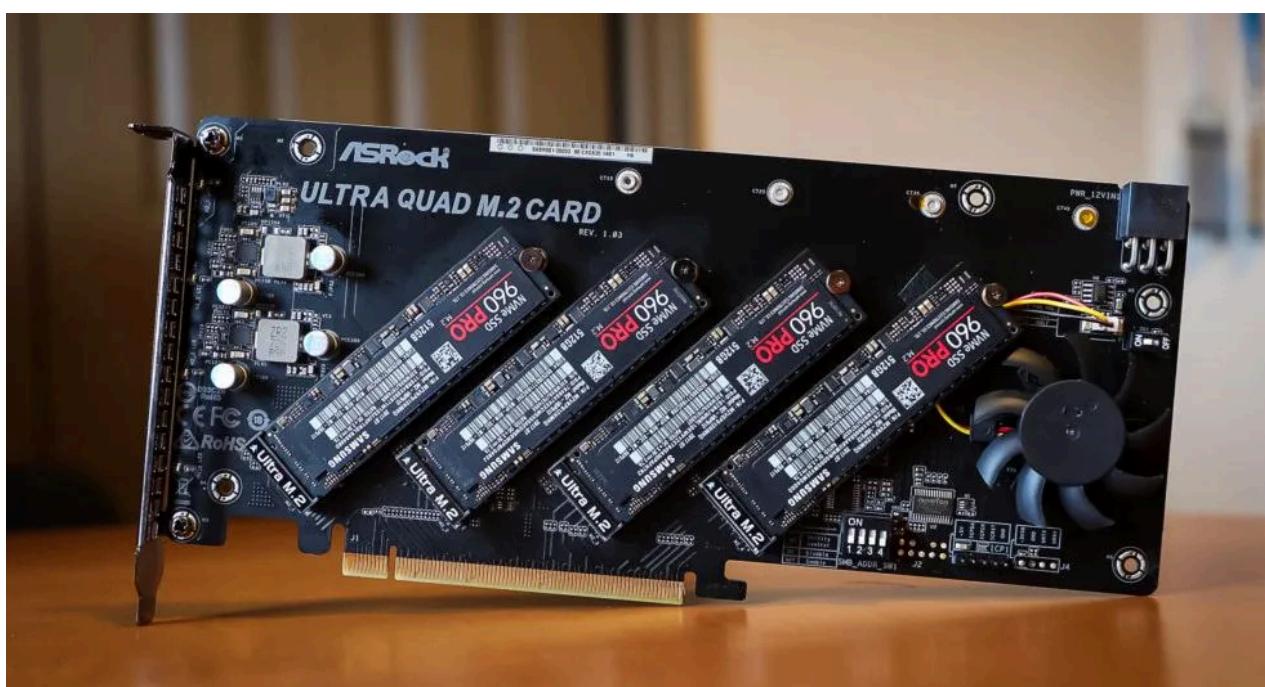
### 1. SATA (Serial ATA):

- Utilizada principalmente para conectar discos duros (HDD), unidades de estado sólido (SSD) y unidades ópticas (CD/DVD).
- SATA III: Hasta 6.0 Gbps (600 MB/s), que es el estándar más común hoy en día.



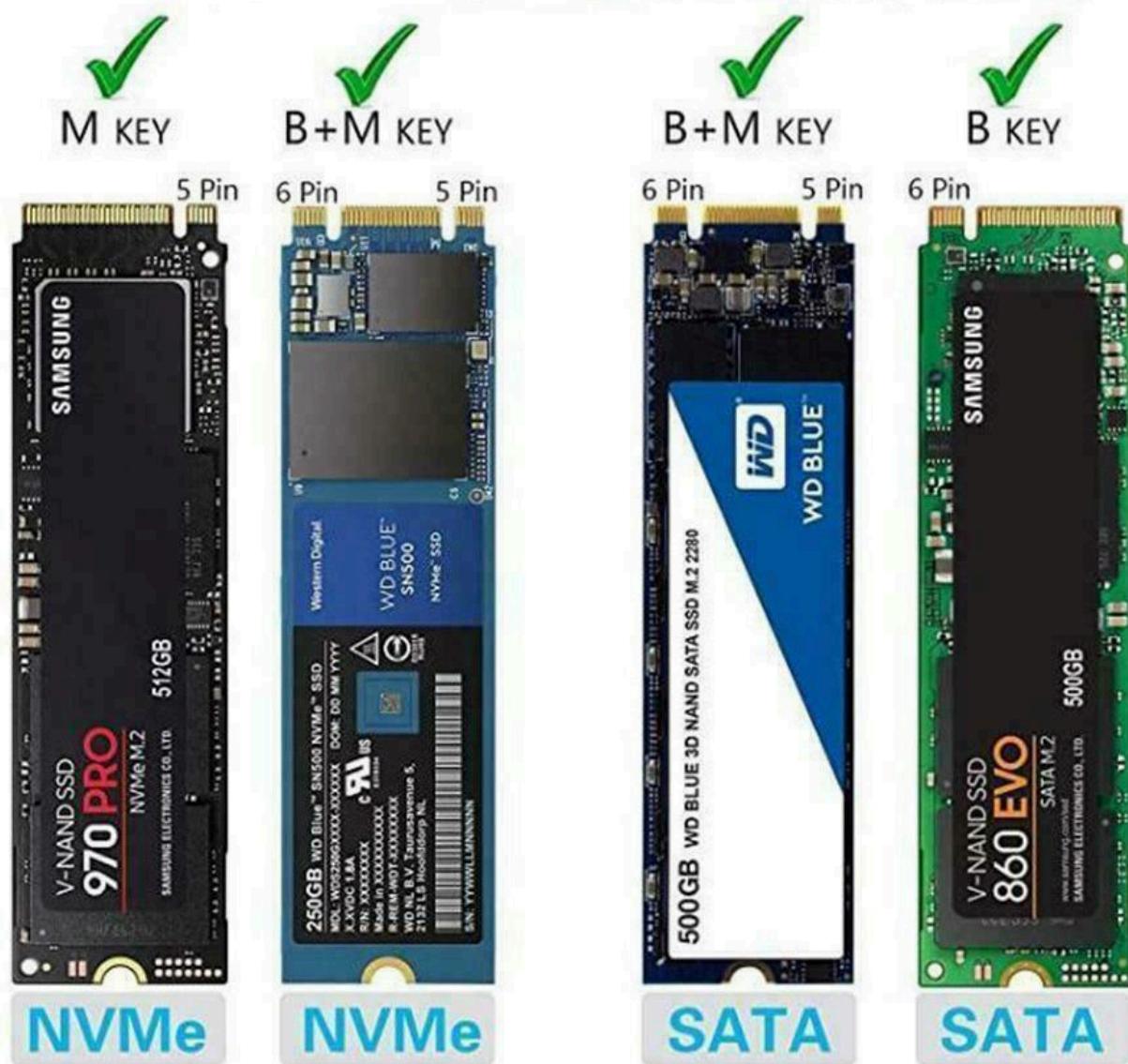
### 2. PCI-Express (PCIe):

- Interfaz de alta velocidad para conectar tarjetas gráficas (GPU), SSDs NVMe, tarjetas de red y otros periféricos de alto rendimiento.



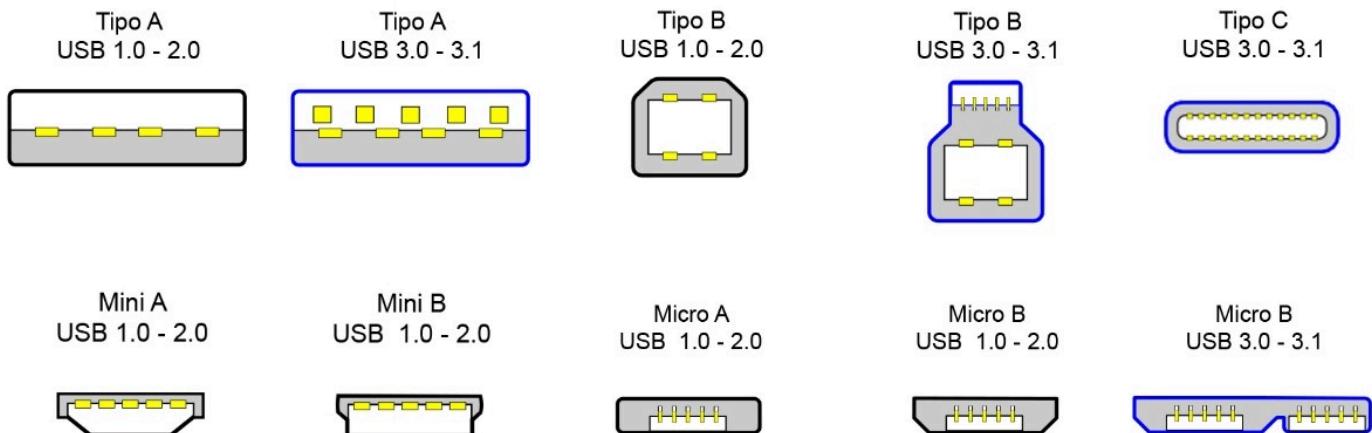
- **M.2:**

- Los SSDs con formato NVMe (Non-Volatile Memory Express) que usan M.2 alcanzan velocidades de hasta 32 Gbps (4 GB/s) por PCIe.
- Tamaños físicos: Los módulos M.2 vienen en varios tamaños, como 2242, 2260, 2280, donde los dos primeros dígitos indican el ancho (22 mm) y los últimos el largo (42, 60, 80 mm). El tamaño más común es 2280.
- Tipos de conectores: Hay varios tipos de conectores M.2, los más comunes son B key y M key, o una combinación de ambos (B+M). Estos definen cuántas pistas PCIe están disponibles y si el dispositivo soporta SATA, PCIe o ambos.



#### 4. USB (Universal Serial Bus):

- Se utiliza para una amplia variedad de dispositivos como ratones, teclados, discos duros externos, memorias USB, impresoras, etc. Tiene varias versiones: USB 2.0, USB 3.0, USB 3.1, USB-C.
- Listado de los principales tipos de conectores USB



## Interfaces de vídeo con sus velocidades (últimas versiones)

### 1. HDMI (High-Definition Multimedia Interface):

- Interfaz para la transmisión de audio y vídeo de alta definición, usada en monitores, televisores y proyectores.
- **HDMI 2.1:** Soporta un ancho de banda de hasta **48 Gbps**, con resoluciones de hasta **8K a 60 Hz** o **4K a 120 Hz**.

### 2. DisplayPort:

- Similar a HDMI, pero diseñada específicamente para monitores de alta resolución y frecuencia de refresco.
- **DisplayPort 2.0:** Ofrece un ancho de banda de hasta **77,4 Gbps**, con resoluciones de hasta **16K a 60 Hz** o **8K a 120 Hz**.

### 3. Thunderbolt:

- Interfaz de alta velocidad que combina la tecnología PCIe y DisplayPort, utilizada para la transferencia de datos, vídeo y energía a través de un solo cable.
- **Thunderbolt 4:** Ofrece un ancho de banda de hasta **40 Gbps**, soportando conexiones de múltiples pantallas 4K o una sola pantalla 8K.

### 4. DVI (Digital Visual Interface):

- Usada para conectar monitores, especialmente en equipos más antiguos.

- **DVI Dual-Link:** Soporta resoluciones de hasta **2560 x 1600 píxeles** con una velocidad máxima de **7,92 Gbps**.

## 5. VGA (Video Graphics Array):

- Conector analógico utilizado para monitores en ordenadores más antiguos.
- Resoluciones de hasta **2048 x 1536 píxeles** a **85 Hz** en sus versiones más avanzadas.

# Otros

## 1. Audio Jack (3.5mm):

- Para conectar auriculares, altavoces o micrófonos.

## 2. PS/2:

- Conector más antiguo utilizado para teclados y ratones.

## 3. Ethernet (RJ-45):

- Puerto para conectarse a redes mediante cable Ethernet, proporcionando acceso a Internet o a redes locales.
- **Ethernet 10G (10 Gigabit):** Soporta velocidades de hasta **10 Gbps**.

### actividad

- Busca información sobre GPMI. ¿conector o interfaz?
- Practicum III

# RAID

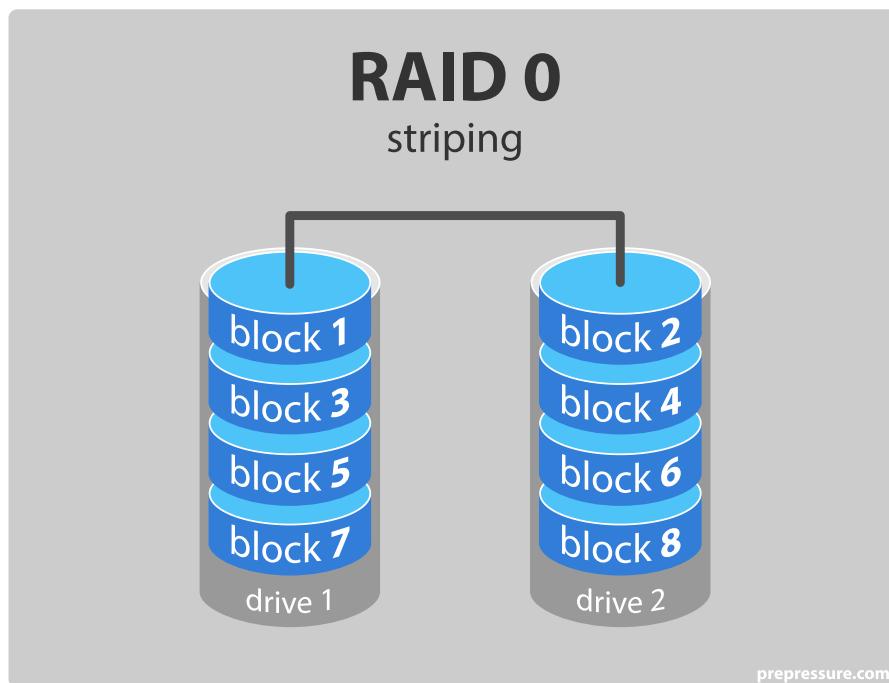
RAID (Redundant Array of Independent Disks) es una tecnología que permite combinar varios discos duros en una única unidad lógica para mejorar la **tolerancia a fallos**, el **rendimiento**, o ambas. RAID puede configurarse a través de hardware dedicado o mediante software, y se usa comúnmente en servidores, estaciones de trabajo y sistemas de almacenamiento masivo.

## Principales niveles de RAID y ejemplos típicos:

### 1. RAID 0 (Striping)

RAID 0 distribuye los datos equitativamente entre dos o más discos sin redundancia. Ofrece **mejores velocidades de lectura y escritura**, pero si uno de los discos falla, se pierden todos los datos, ya que no hay tolerancia a fallos.

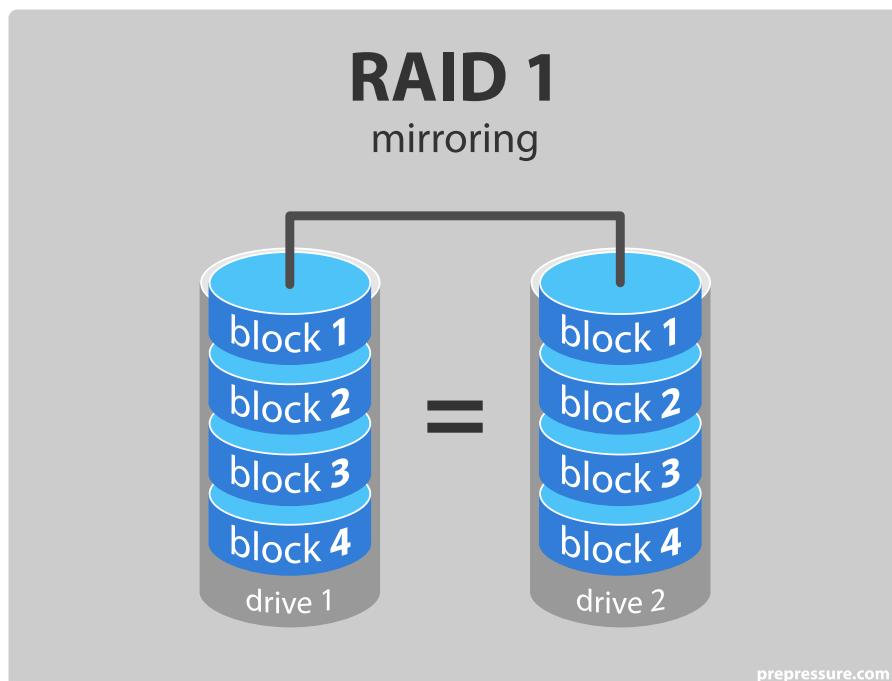
- **Ventaja:** Velocidad de acceso incrementada.
- **Desventaja:** No hay redundancia; si falla un disco, los datos se pierden.
- **Uso típico:** Gaming, edición de video, cuando la velocidad es prioritaria y la pérdida de datos no es crítica.



### 2. RAID 1 (Mirroring)

RAID 1 duplica los datos en dos o más discos. Si uno de los discos falla, los datos pueden recuperarse de los discos restantes. Este nivel de RAID ofrece una **alta disponibilidad** de los datos, pero el espacio de almacenamiento es la mitad del total, ya que todo se duplica.

- **Ventaja:** Alta tolerancia a fallos; si un disco falla, los datos aún están seguros.
- **Desventaja:** No mejora el rendimiento; solo duplica los datos.
- **Uso típico:** Servidores donde la seguridad de los datos es esencial (por ejemplo, bases de datos).



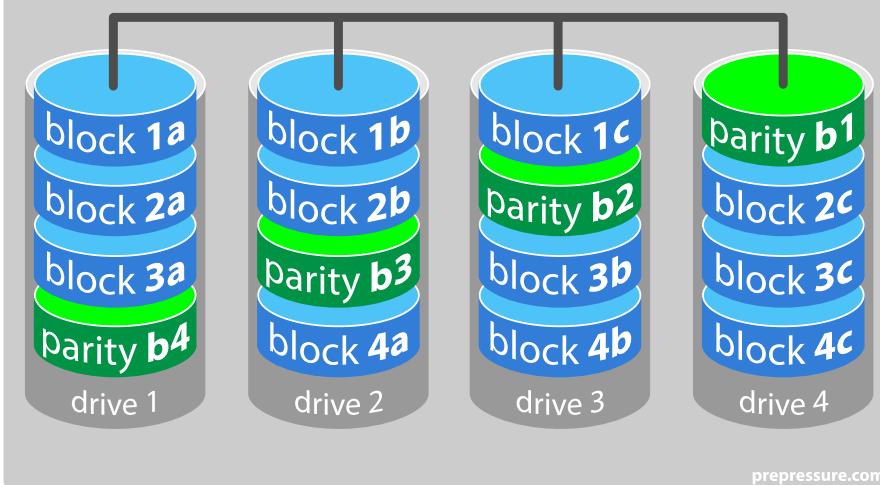
### 3. RAID 5 (Striping with Parity)

RAID 5 utiliza **striping** junto con **paridad**, distribuyendo los datos y la información de paridad entre tres o más discos. Si un disco falla, los datos se pueden reconstruir utilizando la información de paridad almacenada en los discos restantes. RAID 5 proporciona un equilibrio entre **rendimiento, capacidad y redundancia**.

- **Ventaja:** Tolerancia a fallos; mayor rendimiento de lectura y escritura.
- **Desventaja:** Se pierde el equivalente al espacio de un disco para almacenar la paridad.
- **Uso típico:** Servidores de archivo y sistemas NAS donde se requieren tanto velocidad como redundancia.

## RAID 5

striping with parity across drives



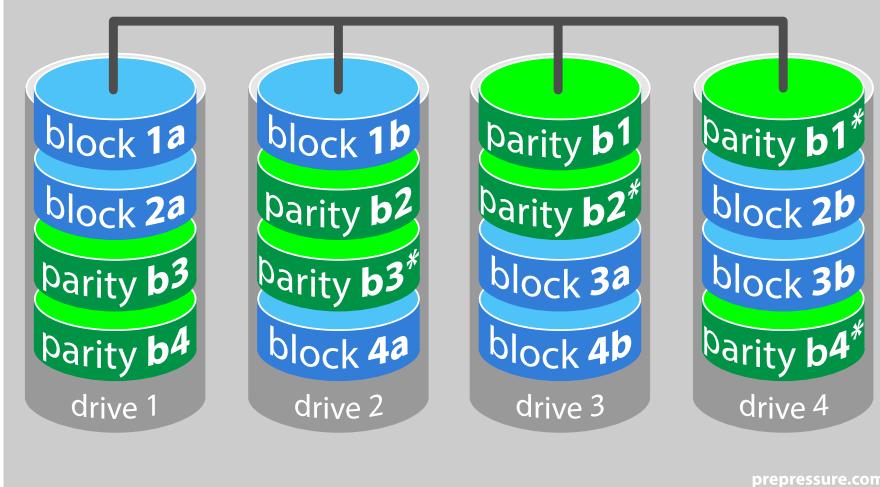
## 4. RAID 6 (Striping with Double Parity)

RAID 6 es similar a RAID 5, pero almacena **dos bloques de paridad** en lugar de uno, lo que permite la **falla de hasta dos discos** sin pérdida de datos. Esto mejora la seguridad, pero a costa de un poco de rendimiento adicional.

- **Ventaja:** Mayor tolerancia a fallos que RAID 5; puede soportar la pérdida de dos discos.
- **Desventaja:** Menor rendimiento en escritura y mayor sobrecarga de almacenamiento.
- **Uso típico:** Sistemas empresariales críticos donde la redundancia es crucial.

## RAID 6

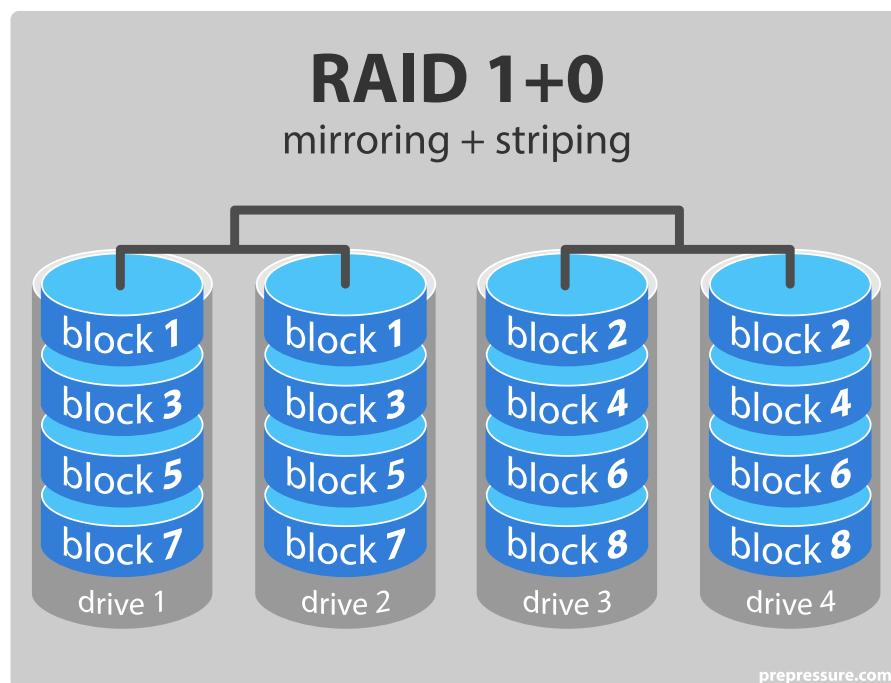
striping with dual parity across drives



## 5. RAID 10 (1+0, Mirroring + Striping)

RAID 10 combina las características de RAID 1 y RAID 0. Primero, los datos se duplican (RAID 1) y luego se dividen (RAID 0). Este nivel proporciona un **equilibrio óptimo entre redundancia y rendimiento**, pero requiere al menos cuatro discos y pierde el 50% del espacio de almacenamiento total debido a la duplicación.

- **Ventaja:** Alta velocidad y redundancia; puede soportar múltiples fallos de discos.
- **Desventaja:** Alto coste debido a la duplicación de datos.
- **Uso típico:** Bases de datos transaccionales, aplicaciones que requieren alto rendimiento y disponibilidad.



## Cálculo de paridad en RAID 5

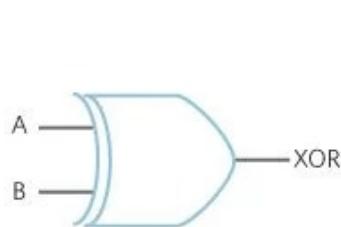
Supongamos que tenemos tres discos en un RAID 5 (Disco A, Disco B y Disco C). Los datos se distribuyen de la siguiente manera:

- **Bloque de datos en Disco A:** 1010
- **Bloque de datos en Disco B:** 1100
- **Paridad almacenada en Disco C:** 0110 (resultado de 1010 XOR 1100)

Si el **Disco A** falla, se puede calcular el contenido perdido aplicando XOR a los bloques del Disco B y la paridad en el Disco C:

- **Recuperación de Disco A:** 1100 XOR 0110 = 1010 (datos originales del Disco A)

$$X = A \oplus B$$



A	B	XOR
0	0	0
0	1	1
1	0	1
1	1	0

## ⓘ Ejercicio

### 1. Calcula los bloques de paridad

Disco 0	Disco 1	Disco 2	Disco 3
1A2	3C4	5E6	<b>780</b>
F00	A1B	C3D	<b>926</b>

### 2. Simulación de fallo de disco:

- 1A2 XOR 5E6 XOR 780 = **3C4** (1A2 XOR 5E6 XOR 780 = 3C4)
- F00 XOR C3D XOR 926 = **A1B** (F00 XOR C3D XOR 926 = A1B)

#### Detalle

- **0001 1010 0010** (1A2)  
**0101 1110 0110** (5E6)  
**0111 1000 0000** (780)  
-----  
**0011 1100 0100 = 3C4**

# Otros Componentes

## Monitor

### Tipos

#### 1. Monitores LCD (Liquid Crystal Display):

- Son los más comunes y utilizan tecnología de cristal líquido.
- Son ligeros y consumen menos energía en comparación con los monitores CRT.
- Ofrecen buena calidad de imagen y son ideales para el uso diario.

#### 2. Monitores LED (Light Emitting Diode):

- Son un tipo de monitor LCD que utiliza diodos emisores de luz para iluminar la pantalla.
- Proporcionan colores más brillantes y mejor contraste.
- Son más delgados y eficientes energéticamente.

#### 3. Monitores OLED (Organic Light Emitting Diode):

- Son un tipo de monitor que utiliza diodos orgánicos emisores de luz en cada píxel, eliminando la necesidad de una retroiluminación.
- Ofrecen negros más profundos y un contraste infinito, ya que cada píxel puede apagarse completamente.
- Son más delgados, ligeros y proporcionan colores más vivos, manteniendo la calidad de imagen desde cualquier ángulo de visión.

### Resolución

La resolución se refiere a la cantidad de píxeles que puede mostrar un monitor, y se mide en píxeles (horizontal x vertical). Algunas resoluciones comunes son:

- **HD (1280 x 720)**: Buena para uso general.
- **Full HD (1920 x 1080)**: Ideal para juegos y películas, ofreciendo una imagen clara y nítida.
- **4K (3840 x 2160)**: Alta definición que proporciona detalles impresionantes, aunque requiere un hardware más potente.

## Estudio científico

El artículo de PC Gamer informa sobre una nueva investigación sobre la visión humana (realizada por investigadores de la Universidad de Cambridge y Meta Reality Labs) que valida el valor de los monitores de PC de muy alta resolución, hasta 16K. El resumen de los puntos clave es el siguiente:

- Validación de Alta Resolución: La investigación sugiere que los monitores de alta densidad de píxeles (DPI), incluso más allá del 4K o 8K, son beneficiosos y no un simple truco publicitario.
- Nueva Acuidad Visual: El estudio, publicado en Nature, determinó que la agudeza visual humana es mayor de lo que se creía anteriormente, alcanzando los 94 píxeles por grado (ppd) para la visión acromática foveal (la parte central y más nítida del campo visual) (Abel et al., 2020).
- Factores Clave: La utilidad de una alta resolución depende totalmente de la distancia de visualización y el tamaño de la pantalla.
- Calculadora de Resolución: Los investigadores han creado una calculadora de resolución de pantalla que permite a los usuarios determinar si pueden percibir la diferencia de píxeles en su monitor o televisor actual o propuesto, basándose en la distancia a la que se sientan.
- Ejemplos de TV (Larga Distancia):
  - Para un televisor de 50 pulgadas visto a tres metros de distancia, el límite de la visión humana es de alrededor de 1440p. Las resoluciones 4K u 8K serían un desperdicio en estas condiciones.
  - Para notar la diferencia con 4K en un televisor, se necesitaría una pantalla de al menos 60 pulgadas a esa misma distancia (tres metros), o acercarse a dos metros para una pantalla de 50 pulgadas.
- Ejemplos de Monitores de PC (Corta Distancia):
  - Para los usuarios de PC, que suelen sentarse a unos 40 cm de la pantalla, el beneficio de la alta resolución es evidente.
  - El estudio indica que, a una distancia de visualización típica de PC, los usuarios podrían llegar hasta la resolución de 16K con pantallas de 30 a 40 pulgadas y aun así notar la diferencia en la calidad de la imagen. Referencias Abel, J. T., Ouillette, P., Williams, C. L., Blau, J., Cheng, J., Yao, K., Lee, W. Y., Cornish, T. C., Balis, U. G. J., & McClintock, D. S. (2020). Display Characteristics and Their Impact on Digital Pathology: A Current Review of Pathologists' Future "Microscope". Journal of Pathology Informatics, 11(1), 23. [https://doi.org/10.4103/jpi.jpi\\_38\\_20](https://doi.org/10.4103/jpi.jpi_38_20)

## Tasa de Refresco

La tasa de refresco, medida en Hertz (Hz), indica cuántas veces por segundo se actualiza la imagen en la pantalla. Una mayor tasa de refresco puede ofrecer una experiencia más fluida, especialmente en juegos.

- **60 Hz**: Adecuado para uso general.
- **144 Hz o más**: Recomendado para jugadores que buscan una experiencia más fluida.

## Conexiones

- **HDMI**: Conexión digital que transmite audio y video de alta calidad.
  - Soporta resoluciones hasta 4K y HDR.
- **DisplayPort**: Similar a HDMI, pero con soporte para múltiples monitores.

- Ideal para configuraciones de juegos y estaciones de trabajo.
- **DVI:** Conexión digital que puede transmitir video de alta calidad, pero sin audio.
- **VGA:** Conexión analógica más antigua, que está en desuso.

## Altavoces

- **Altavoces estéreo (2.0):** Dos altavoces (izquierda y derecha) para sonido básico.
- **Altavoces 2.1:** Dos altavoces y un subwoofer para mejorar los graves.
- **Altavoces 5.1 o superiores:** Sistemas envolventes para una experiencia más inmersiva, ideales para cine o juegos.

La conexión de los altavoces puede ser a través de cables (jack de 3.5 mm, RCA), óptica o inalámbrica (Bluetooth).

## Problemas y monitoreo

El **monitoreo de un PC** implica la supervisión del rendimiento y el estado de varios componentes para asegurar su buen funcionamiento. A continuación se presentan los principales aspectos que se monitorean:

1. **Temperatura del sistema:** Controlar las temperaturas de la CPU, GPU y otros componentes para evitar el sobrecalentamiento.
2. **Uso de la CPU:** Verificar cuánta carga tiene el procesador durante tareas intensivas o inactivas.
3. **Memoria RAM:** Monitorear la cantidad de RAM utilizada en tiempo real, asegurándose de que no haya sobrecargas.
4. **Uso del disco:** Controlar el rendimiento y el estado de los discos duros o SSD, como el espacio disponible y la velocidad de lectura/escritura.
5. **Velocidades de los ventiladores:** Asegurarse de que los ventiladores funcionen correctamente para mantener las temperaturas bajo control.
6. **Consumo de energía:** Verificar cuánta energía está consumiendo el sistema.

El monitoreo se puede realizar mediante software especializado, como **HWMonitor**, **MSI Afterburner** o **Task Manager** (Administrador de tareas en Windows), que permiten visualizar todos estos datos en tiempo real. Esto es crucial para mantener la estabilidad del sistema, prevenir fallos y optimizar el rendimiento en tareas intensivas, como juegos o edición de vídeo.

## Administrador de Sistemas



MINISTERIO  
DE EDUCACIÓN  
Y FORMACIÓN PROFESIONAL



Plan de Recuperación,  
Transformación  
y Resiliencia



FPCV  
FP Profesional  
Conselleria d'Educació, Universitats i Esports

GVA NEXT  
Fondos Next Generation  
en la Comunitat Valenciana



## Software

## Enlace

Open Hardware Monitor <https://openhardwaremonitor.org/>

AIDA64 <https://www.aida64.com/>

HWiINFO <https://www.hwinfo.com/>

NETDATA <https://www.netdata.cloud>

## Ampliación

[¿Cómo funciona un PC y qué hace cada pieza? | Componentes del ordenador explicados \(Nate Gentile\)](#)

# Tipos de Redes y Topologías

 Connection with the curriculum

RA: 1 - CE: e, f, g

## 1. Tipos de redes

Las redes se clasifican según el **ámbito de aplicación** y el **medio de transmisión** utilizado.

### Clasificación según el ámbito de aplicación:

#### 1. PAN (Personal Area Network):

- Redes de corto alcance, utilizadas para conectar dispositivos personales como teléfonos móviles, ordenadores y periféricos.
- Ejemplos: Bluetooth, NFC.

#### 2. LAN (Local Area Network):

- Redes de área local, usadas en oficinas, escuelas o viviendas para conectar ordenadores y compartir recursos.
- Se pueden segmentar en subredes.
- Medio de transmisión: Cableado (cobre, fibra óptica) y sin cables (Wi-Fi).

#### 3. MAN (Metropolitan Area Network):

- Redes que conectan varias LAN dentro de una ciudad o campus universitario.
- Medio de transmisión: Fibra óptica, WiMAX.

#### 4. WAN (Wide Area Network):

- Redes que abarcan grandes distancias, como países o continentes.
- Conectan múltiples LAN y MAN utilizando enlaces de larga distancia (satélites, redes telefónicas).
- Ejemplo: Internet.

### Clasificación según el nivel de privacidad:

- **Públicas:** Acceso abierto para cualquier usuario, como las redes de Wi-Fi abiertas en lugares públicos.
- **Privadas:** Redes protegidas por contraseñas, sólo accesibles para usuarios autorizados.
- **Semipúblicas:** Acceso controlado, como redes de empresas con invitaciones externas.

## Clasificación según el tipo de conexión:

- **Punto a punto (Point-to-Point):** Comunicación directa entre dos dispositivos.
- **Difusión (Broadcast):** Un dispositivo envía datos a todos los demás dispositivos de la red.
- **Punto a multipunto (Point-to-Multipoint):** Un nodo se comunica con múltiples nodos receptores.

## 2. Topologías de red

Las topologías describen cómo se conectan físicamente y lógicamente los dispositivos en una red.

### 1. Topología estrella:

- Todos los dispositivos están conectados a un nodo central.
- Ventaja: Alta tolerancia a fallos (excepto en el nodo central).
- Desventaja: Si el nodo central falla, toda la red deja de funcionar.

### 2. Topología anillo:

- Los dispositivos están conectados en un bucle cerrado. Cada uno se conecta al siguiente y el último al primero.
- Ventaja: Flujo de datos ordenado y predecible.
- Desventaja: Si un dispositivo falla, la red se ve interrumpida.

### 3. Topología bus:

- Todos los dispositivos comparten un único canal de comunicación.
- Ventaja: Fácil de implementar y económica.
- Desventaja: El tráfico de la red puede generar colisiones, disminuyendo la velocidad.

### 4. Topología malla:

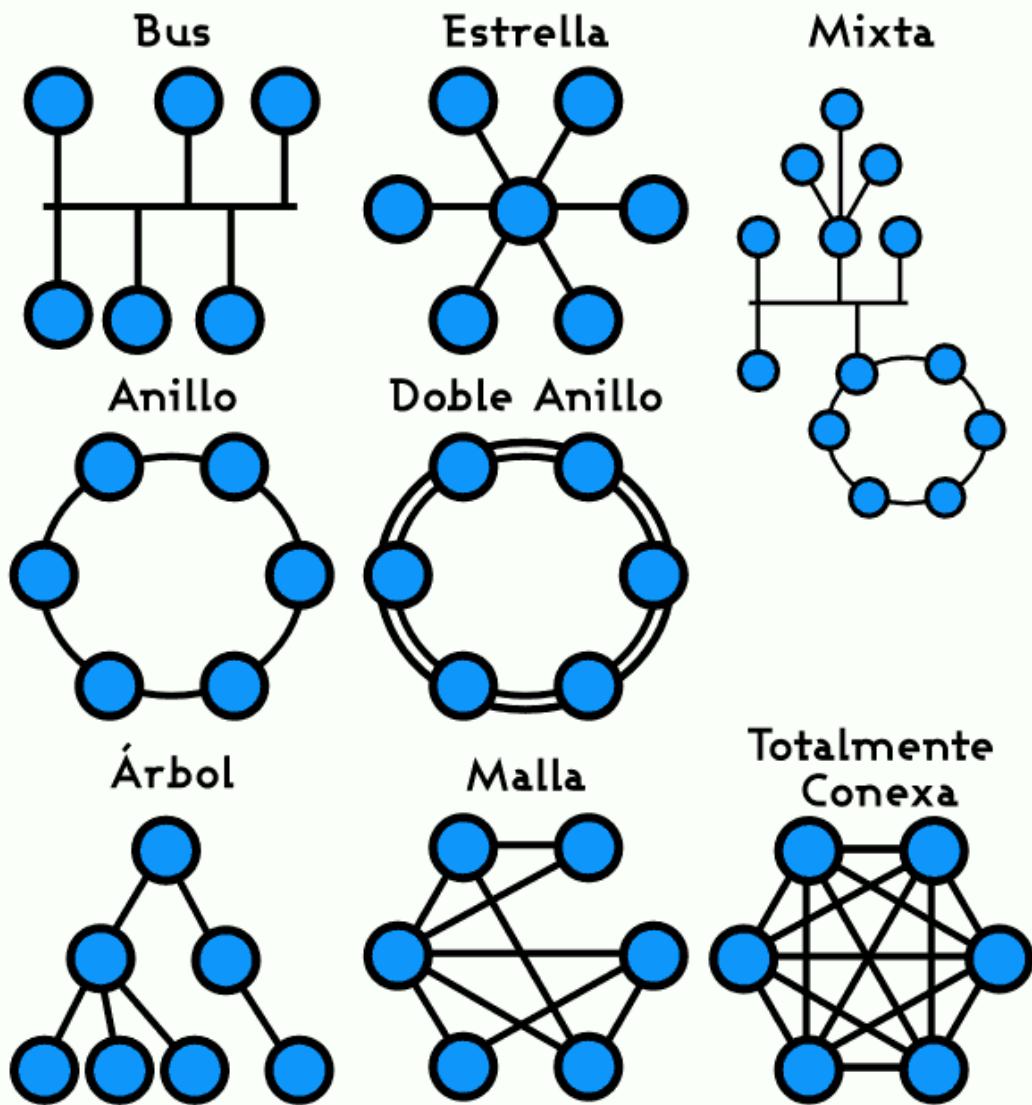
- Todos los dispositivos están interconectados entre sí.
- Ventaja: Alta tolerancia a fallos y mayor redundancia.
- Desventaja: Requiere gran cantidad de cables y conexiones, lo que aumenta el costo.

### 5. Topología árbol:

- Estructura jerárquica que combina la topología en estrella y la topología en bus.
- Ventaja: Organización jerárquica y fácil escalabilidad.
- Desventaja: Si un enlace central falla, la conexión de los nodos descendientes también se ve afectada.

### 6. Topología híbrida:

- Combinación de dos o más topologías. Ofrece flexibilidad para redes complejas.



# Componentes y Protocolos

## 1. Componentes de una red informática

### 1. Dispositivos de interconexión:

- **Switch:** Dispositivo de capa 2 que distribuye los paquetes de datos dentro de la misma red según direcciones MAC.
- **Router:** Opera en la capa 3 y conecta diferentes redes, dirigiendo el tráfico entre ellas mediante direcciones IP.
- **Hub:** Funciona en la capa 1, retransmitiendo paquetes a todos los dispositivos conectados sin filtrar.
- **Bridge:** Conecta dos segmentos de red y filtra el tráfico según direcciones MAC para mejorar el rendimiento.
- **Patch Panel:** Panel de conexiones que organiza y distribuye cables de red en instalaciones estructuradas. No procesa datos, solo facilita la conexión física.

### 2. Servidores:

- Proporcionan servicios a la red, como almacenamiento, correo electrónico, bases de datos.
- Ejemplos: Servidor web, servidor de archivos.

### 3. Clientes:

- Dispositivos que acceden a los recursos proporcionados por los servidores.
- Ejemplos: Ordenadores, teléfonos móviles, tabletas.

### 4. Medios de transmisión:

- **Cableado de cobre (UTP/FTP):** Utilizado en redes LAN.

U/FTP Pantalla de aluminio en los pares



F/FTP Pantalla de aluminio en los pares  
y en el cable



S/FTP Pantalla de aluminio en los pares  
y malla de aluminio en el cable



F/UTP Pantalla de aluminio en el cable



SF/FTP Pantalla y malla de aluminio  
en el cable



**Figura:** Apantallamiento de cables

**Figura:** RJ45 para  
cable UTP sin  
pantalla



**Figura:** RJ45 para  
cables con blindaje

- **Fibra óptica (monomodo, multimodo):** Transmisión de datos a alta velocidad y largas distancias.
- **Inalámbricos (Wi-Fi, Bluetooth, WiMAX):** Comunicaciones sin cables para LAN y MAN.

### 1. Periféricos de red:

- Dispositivos como impresoras en red, escáneres y unidades de almacenamiento en red (NAS).

## 2. Protocolos de red

Los protocolos definen cómo se comunican los dispositivos en una red.

### 1. Protocolos de capa de red:

- **IP (Internet Protocol):** Direccionamiento y enrutamiento de datos.
- **ICMP (Internet Control Message Protocol):** Informes de errores y control de la red.

### 2. Protocolos de capa de transporte:

- **TCP (Transmission Control Protocol):** Protocolo orientado a la conexión para garantizar la entrega de datos.
- **UDP (User Datagram Protocol):** Protocolo sin conexión, más rápido pero sin garantía de entrega.

### 3. Protocolos de capa de aplicación:

- **HTTP (HyperText Transfer Protocol):** Transferencia de páginas web.
- **FTP (File Transfer Protocol):** Transferencia de archivos.

- **SMTP (Simple Mail Transfer Protocol):** Envío de correo electrónico.
- **DNS (Domain Name System):** Resolución de nombres de dominio a direcciones IP.

#### 4. Protocolos de seguridad:

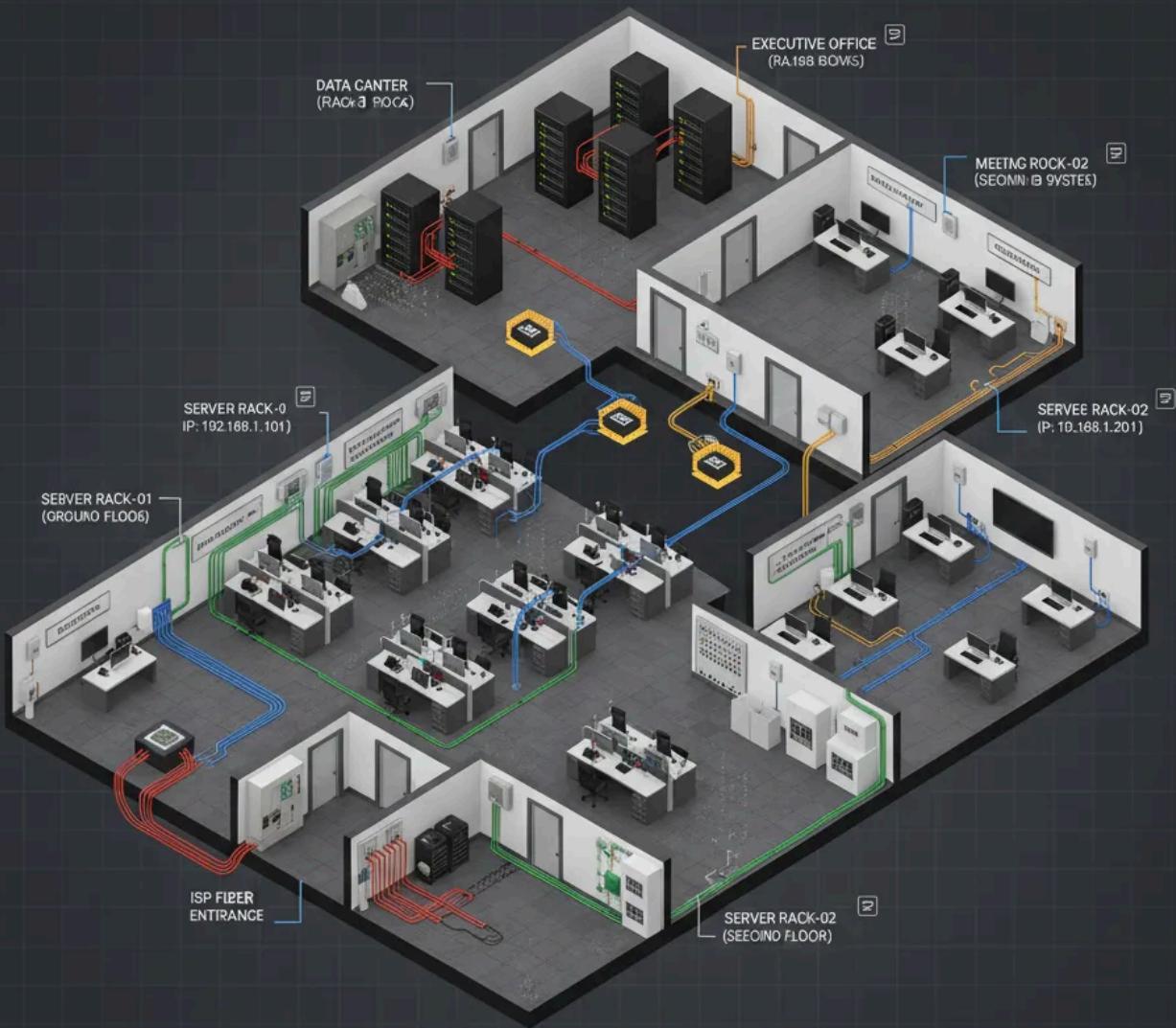
- **WEP, WPA, WPA2, WPA3:** Protocolos de cifrado para proteger redes inalámbricas.

## 3. Mapas físicos y lógicos

### 1. Mapas físicos:

- Representan la disposición de los dispositivos y el cableado físico de la red.
- Ejemplo: Un diagrama que muestra la ubicación de los ordenadores, servidores y cables en un edificio.

## GLOBALNET OFFICE - PHYSICAL NETWORK MAP



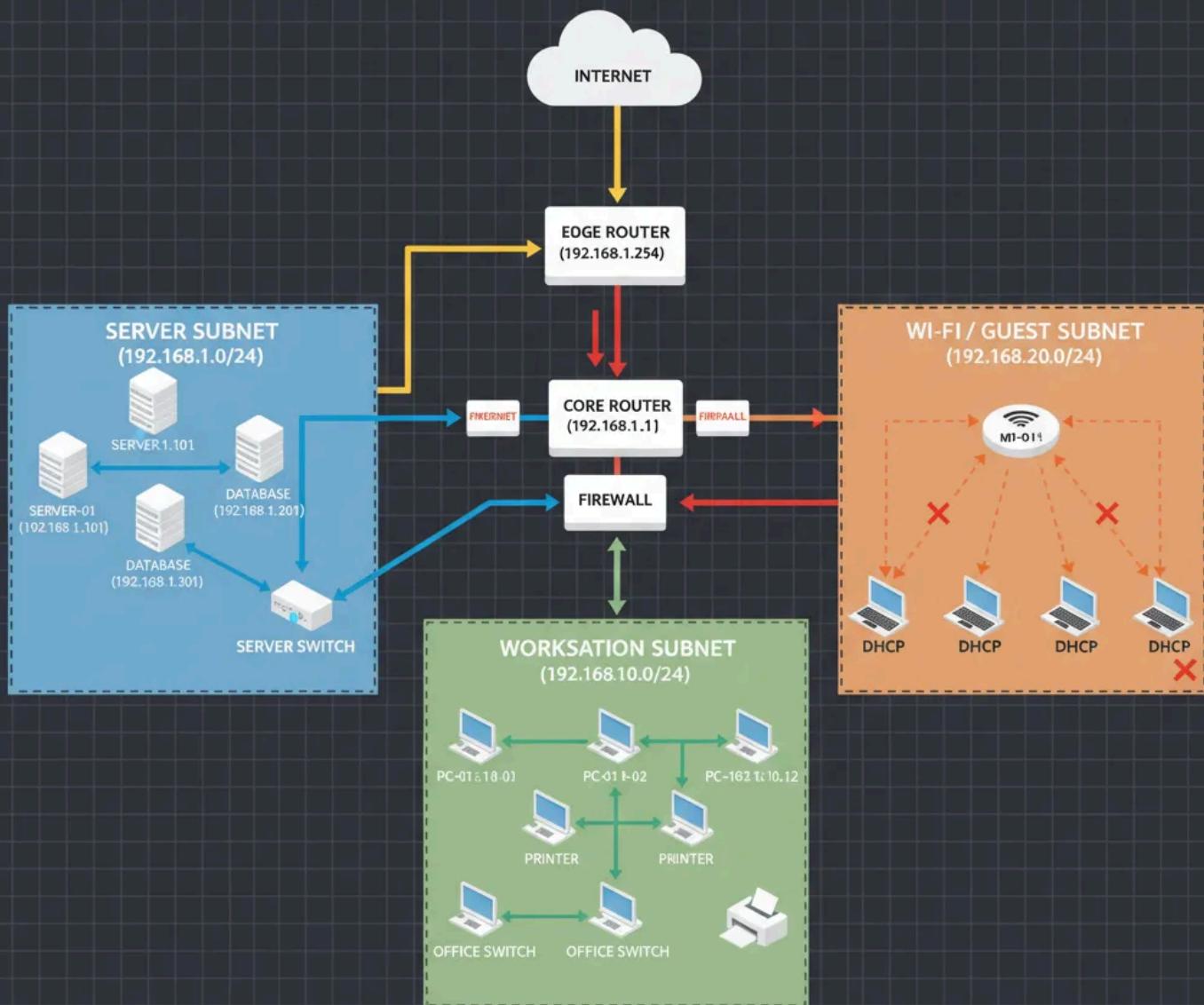
### LEGEND

	BIEKENM&ATION-101		WORKSATON-02
	Single-mode Fiber		Single-mode Fiber
	Piso/xtiole Fiber		Multi-Mode Fiber

### 1. Mapas lógicos:

- Representan cómo se comunican los dispositivos entre sí, incluyendo subredes, direcciones IP y ruteo.
- Ejemplo: Un diagrama con las direcciones IP de cada dispositivo y rutas de tráfico configuradas.

## GLOBALNET OFFICE – LOGICAL NETWORK MAP



### Legend

—	EOEVTSOMNE
—	OOTUBAL
—	CIOMAEOT



MDNIGRIERA



# Precauciones básicas y montaje

 Connection with the curriculum

RA: 1 - CE: h

## 1. Precauciones básicas en el montaje y mantenimiento de ordenadores

Cuando se trabaja con ordenadores para montaje, mantenimiento o expansión, es esencial seguir una serie de precauciones básicas para garantizar la seguridad tanto del usuario como de los componentes del sistema.

### Regla de oro de la seguridad:

"Desconectar la alimentación del ordenador antes de realizar cualquier operación o ampliar el equipo."

Los ordenadores funcionan con tensiones de  $\pm 12$  y  $\pm 5$  voltios, que son relativamente inofensivas, pero para conseguir estas tensiones se convierten los 220 voltios de la red eléctrica, lo cual implica un riesgo considerable.

### Electricidad estática:

Aunque una pequeña cantidad de corriente liberada puede ser inofensiva para los seres humanos, puede ser mortal para los componentes informáticos. Los componentes CMOS son especialmente sensibles a estos daños. Incluso una carga ligeramente superior a 5 voltios es suficiente para dañar componentes, provocando el fallo de la placa base.

- Recomendación:** Utilizar guantes o pulsera antiestática conectada a tierra o tocar regularmente el chasis del ordenador para descargar cualquier electricidad estática antes de tocar los componentes electrónicos.

### Cortes:

La placa u otros componentes puede ser una fuente de cortes o pinchazos.

### Ácaros:

Posibles alergias con la suciedad dentro del ordenador.

## 2. Buenas prácticas durante el montaje y mantenimiento

### 1. Evitar utilizar la fuerza:

- Todas las conexiones en el ordenador deben ser fácilmente insertadas y separadas sin esfuerzo. La mayoría de los conectores son unidireccionales y están diseñados para acoplarse en una sola dirección.
- Si observas que se necesita un esfuerzo excesivo para conectar algo, revisa la dirección y la polaridad.

## 2. Trabajar lentamente:

- Los errores o las inexactitudes menores pueden convertirse en problemas difíciles de resolver en el futuro. Tómate tu tiempo y reflexiona dos veces cada paso antes de actuar.

## 3. Preparación previa:

- Asegúrate de tener los manuales y documentos a mano antes de empezar a trabajar para comprobar la compatibilidad de los componentes con el sistema.

## 4. Atención con los cables:

- Trabaja en un lugar con suficiente espacio para colocar los cables. Asegúrate de que los cables sean suficientemente largos para evitar roturas y desconexiones accidentales.

## 5. No comer ni beber cerca del ordenador:

- Un derrame accidental podría dañar componentes críticos como la placa base o la unidad de procesamiento.

## 6. Utilizar el tornillo adecuado:

- Los tornillos deben coincidir con los agujeros correspondientes. Si un tornillo no entra fácilmente, comprueba si el hilo corresponde al agujero o si la longitud del tornillo es la adecuada.

## 7. Almacenar piezas pequeñas por separado:

- Guardar tornillos y otras piezas pequeñas en contenedores preparados para evitar que caigan dentro del ordenador. Si una pieza cae dentro, detén inmediatamente el trabajo y recuérdala.

## 3. Registro de cambios y configuraciones

Cuando realices cambios en la configuración del ordenador, registra siempre la configuración original antes de hacer cualquier cambio. Esto te permitirá volver fácilmente a la configuración anterior en caso de error.

## 4. Comprobaciones finales

Una vez completado el montaje o la modificación:

- Comprueba que todos los componentes están bien colocados y conectados.



MINISTERIO  
DE EDUCACIÓN  
Y FORMACIÓN PROFESIONAL



Plan de Recuperación,  
Transformación  
y Resiliencia



- Realiza pruebas de funcionamiento para asegurarte de que el ordenador funciona correctamente antes de cerrar la carcasa o instalar el ordenador en su ubicación final.

## Instalación paso a paso

### 1. Instalar la CPU:

- Abrir el zócalo de la CPU en la placa base.
- Alinear las muescas de la CPU con las del zócalo.
- Colocar suavemente la CPU en el zócalo y asegurarla.

### 2. Instalar el enfriador de la CPU:

- Aplicar pasta térmica si es necesario.
- Conectar el enfriador de la CPU según las instrucciones del fabricante.

### 3. Instalar la RAM:

- Abrir las ranuras de RAM en la placa base.
- Alinear las muescas de la RAM con las de las ranuras.
- Presionar hacia abajo en ambos extremos de la RAM hasta que encaje en su lugar.

### 4. Instalar la placa base:

- Colocar la placa base en la caja.
- Fijar la placa base a la caja con tornillos.

### 5. Instalar el almacenamiento:

- Conectar el dispositivo de almacenamiento (HDD/SSD) al puerto apropiado (IDE/SATA/M.2) en la placa base.
- Asegurar el dispositivo de almacenamiento en la caja con tornillos.

### 6. Instalar la fuente de alimentación:

- Insertar la fuente de alimentación en la caja.
- Conectar los cables de alimentación necesarios a la placa base, CPU, tarjeta gráfica y unidades.

### 7. Conectar los ventiladores de la caja y los conectores del panel frontal:



MINISTERIO  
DE EDUCACIÓN  
Y FORMACIÓN PROFESIONAL



Plan de Recuperación,  
Transformación  
y Resiliencia



- Conectar los conectores de alimentación de los ventiladores de la caja a cualquier cabezal de ventilador en la placa base.
- Conectar los conectores del panel frontal de la caja a la placa base.

## 8. Instalar las tarjetas de expansión:

- Insertar la tarjeta en la ranura PCI Express apropiada en la placa base.
- Asegurar la tarjeta a la caja con tornillos.

## 9. Comprobar las conexiones:

- Verificar todas las conexiones para asegurarse de que todo esté correctamente asentado y conectado.

## 10. Gestión de cables:

- Organizar y asegurar los cables para mejorar el flujo de aire y la estética.

## 11. Encender:

- Conectar los periféricos, el monitor y encender el PC.

### actividad

- Practicum IV
- Resumen tema en A6 - a mano
- T1.1 - INTRODUCCIÓN A LOS SISTEMAS INFORMÁTICOS

# Introducción

## ⓘ Connexió amb el currículum

RA: 2 - CE: a, b, c

## 1. Introducción a los sistemas operativos: componentes, software y licencias

### Elementos funcionales de un sistema informático

Los componentes funcionales de un sistema informático son:

- Los componentes físicos: **hardware**.
- Los componentes lógicos: **software**.
- El personal y usuarios.
- Documentación.

Un **Sistema Operativo (SO)** se incluye en la categoría de **SOFTWARE**.

### Definición y Tipos de Software

Se denomina **software** al conjunto de los programas, procedimientos, reglas, documentación y datos asociados, que forman parte de las operaciones de un sistema de computación.

#### Tipos de software (dependiendo de su función):

Tipo de Software	Función principal
<b>Software de sistema</b>	Desvincular al usuario y al programador de los detalles del hardware del sistema informático.
<b>Software de aplicación</b>	Permite a los usuarios llevar a cabo tareas específicas.

**Tipos de software de sistema:** El software base o de sistema actúa como intermediario entre el usuario y el hardware. En esta categoría se incluye:

- **Sistemas Operativos** (de Escritorio, Móviles, de Servidores, Virtuales/Hypervisores).
- BIOS.

- Firmware de equipos y dispositivos.
- Drivers de dispositivos.
- Aplicaciones de diagnóstico y software de mejora de equipos.

**Tipos de software de aplicación:** Son programas diseñados para realizar tareas específicas y pueden clasificarse según su propósito o funcionalidad. Los principales tipos son:

- Aplicaciones de productividad, comunicación, multimedia, gestión empresarial, diseño y cad, educativas, entretenimiento y juegos, navegación y mapas, seguridad, financieras y contabilidad, científicas y de ingeniería, web y navegadores, utilidades y mantenimiento del sistema, móviles.

## Tipos de Software y Licencias

Además del software de sistema y de aplicación, se distingue por el tipo de acceso y distribución:

Tipo de Software	Acceso y Características
<b>Software libre</b>	Se proporciona el <b>código fuente</b> y libertad de copia, uso y modificación.
<b>Software privativo o propietario</b>	Se concede únicamente derechos sobre su uso. Los fabricantes suelen ofrecer soporte técnico y actualizaciones. El usuario no dispone del código fuente.

Una **licencia** es un contrato entre el desarrollador y el usuario que define los derechos y deberes de ambas partes.

## Licencias de Software Libre

En las licencias de software libre, los derechos del usuario son muy abiertos y permisivos. El usuario dispone de **cuatro libertades**:

1. Libertad para ejecutar con cualquier propósito.
2. Libertad para estudiarlo y adaptarlo a sus necesidades.
3. Libertad para distribuir copias.
4. Libertad para modificarlo y mejorarlo (código fuente).

El software libre puede ser gratuito o no, ya que estas libertades no implican que no se pueda comercializar.

### Ejemplos de licencias de software libre:

- **SW con copyleft (Ej. GNU GPL):** El autor conserva los derechos de autor, pero permite la redistribución y modificación bajo esa misma licencia de software libre (efecto vírico). Ejemplo: el kernel de Linux.

- **SW sin copyleft (Ej. BSD, APACHE):** Es una licencia permisiva que no impone obligaciones sobre las obras derivadas. Ejemplo: Android usa Apache.
- **SW de dominio público:** No tiene derechos de autor. Es un código sin copyleft que puede derivar en privativo.

## Formas de Distribución (Software No Libre)

Son formas de distribución para dar a conocer el producto, pero en la mayoría de los casos, el código fuente no está disponible, por lo que no son software libre:

- **Shareware:** O no tiene todas las funcionalidades habilitadas, o su uso se limita temporalmente. Si después de probarlo el usuario decide que le gusta, tiene que comprarlo.
- **Freeware:** Software gratuito y sin limitaciones temporales.

## Licencias de Software Propietario

En estas licencias se especifican plazos y modos concretos de instalación y uso. Normalmente, no se puede copiar ni distribuir el software, y el usuario puede utilizarlo sin ser el propietario.

- **Tipos genéricos:** EULA (end user license agreement) o CLUF (contrato de licencia de usuario final).  
[EULA/ALUF/CLUF](#)
- **Licencias de software propietario usadas por Microsoft:**
  - **OEM:** Se otorga al fabricante de hardware para que se distribuya con él en un único uso. Está ligada a un equipo físico concreto.
  - **Retail (ALUF - Acuerdo de Licencia de Usuario Final):** Es la licencia que se compra directamente del desarrollador y se puede instalar en cualquier hardware, normalmente en una sola máquina a la vez.
  - **VLM (Licencias de Volumen):** Permite la instalación en un número de ordenadores con una única licencia, de forma más sencilla y económica. Destinada a corporaciones y empresas.

## 2. Evolución histórica

Periodo	Características principales
<b>Años 40-50 (Primeros Ordenadores)</b>	Se interactuaba directamente con el hardware mediante <a href="#">tarjetas perforadas</a> . No existía sistema operativo ni monitores. La supervisión del proceso se realizaba manualmente por ingenieros, físicos y matemáticos.

Periodo	Características principales
<b>Años 50-70 (Primeros SO)</b>	General Motors crea el primer SO para IBM, basado en un programa monitor que gestiona una cola de trabajos. <b>Evolución del hardware y del SO:</b> Sistemas por lotes multiprogramación (aprovechan tiempos muertos del procesador), Sistemas de tiempo compartido (simulan uso simultáneo/multiusuario asignando <i>quantum</i> de procesador), y Sistemas multiprocesador.
<b>Años 70 - Actualidad (Ordenadores Personales)</b>	Inversión en <b>usabilidad</b> que acerca los ordenadores a usuarios no expertos. Mejora de la interacción mediante dispositivos de E/S más avanzados. Se desarrollan interfaces más usables y aplicaciones de propósito general (juegos, ofimática, etc.).

### 3. Características, funciones y tipos de un SO

El sistema operativo es un software con características particulares porque debe administrar todos los recursos del sistema entre los usuarios y el resto del software. El SO es software y consume recursos, más que la mayoría de las aplicaciones de usuario.

#### Características de los SO

- Adaptabilidad:** Debe actualizarse para adaptarse al nuevo software y hardware, que evolucionan en paralelo.
- Facilidad de uso:** Es primordial, pero implica mayor gasto en recursos (ej., el sistema gráfico de ventanas). Algunos SO ganan eficiencia restringiendo su manejabilidad.
- Eficiencia:** Debe atender todas las peticiones de recursos limitados (RAM, núcleos, acceso a discos, colas de impresión, etc.), teniendo en cuenta la importancia de las solicitudes y quién las realiza. Esta tarea es compleja y crítica.

#### Funciones del Sistema Operativo

Las funciones de un SO incluyen:

Función	Descripción
<b>Gestión de procesos</b>	Determina qué proceso ejecutará el procesador en cada ciclo de reloj, reparte el tiempo de ejecución, y crea, destruye, arranca, para y comunica procesos.

Función	Descripción
Gestión memoria	Reparte la memoria entre los procesos, realizando funciones de asignación, liberación y control de acceso a la misma.
Gestión E/S	Controla las transferencias de información entre periféricos y CPU.
Gestión ficheros	Gestionar el almacenamiento permanente y la seguridad de los ficheros y archivos.
Seguridad	Incluye mecanismos para permitir o denegar el acceso a recursos (ficheros, red, etc.). Debe garantizar <b>el servicio y disponibilidad</b> , la <b>confidencialidad, protección e integridad</b> del sistema y de los datos.
Control de errores	Gestionar todo tipo de errores, informando al usuario y salvaguardando de forma prioritaria la seguridad del sistema y los datos.
Interfaz con el usuario	La interfaz gráfica o textual permite la comunicación, gestión, administración y explotación del sistema operativo por parte del usuario.

## Tipos de Sistemas Operativos (Clasificación)

Los SO se distinguen según los servicios ofrecidos:

Criterio	Tipos	Descripción
Número de procesos simultáneos	Monotarea	Solo lanza un proceso cuando el anterior ha terminado. El usuario solo puede ejecutar un proceso a la vez.
	Multitarea	Atiende a varios procesos de forma simultánea. Hay varios programas cargados en memoria listos para ser ejecutados.
Número de usuarios	Monousuario	Solo atiende a un usuario, que dispone de todos los recursos.
	Multiusuario	Da servicio a más de un usuario a la vez. Los recursos deben ser gestionados y repartidos.
Número de procesadores	Monoproceso	Solo tiene una CPU, pero puede simular multitarea ( <i>quantum</i> ).

Criterio	Tipos	Descripción
	Multiproceso	Soporta varios procesos simultáneos mediante varios procesadores.
<b>Forma de ofrecer los servicios</b>	Centralizados	Un ordenador central realiza las tareas de computación, y los usuarios se conectan por terminales ligeros.
	Distribuidos	Una red de ordenadores independientes se presenta como un único sistema, repartiendo los cálculos entre los equipos.
<b>Disponibilidad</b>	SO propietarios	Código fuente, derechos y permisos de uso son de una empresa.
	SO libres	Libertad de uso, modificación, distribución y mejora del programa.
<b>Tipo de procesamiento</b>	Sistemas de tiempo real	Diseñados para tareas críticas (ej. control de sistemas de un coche o avión). Deben responder correctamente en un tiempo muy limitado, siendo la fiabilidad fundamental.
	SO interactivos o de tiempo compartido	Sistemas en los que no prima la limitación de tiempos de respuesta, como Windows o Ubuntu.
<b>Interfaz empleado</b>	Textuales	No hay entorno gráfico; se introducen comandos en consola. Requiere mayor conocimiento de sintaxis, pero la explotación puede ser más potente.
	Gráficos	Existe un entorno gráfico con ventanas que facilita el manejo, pero consume muchos más recursos a nivel de procesador.
<b>Servicios (Adicional)</b>	<b>SO en red</b>	Gestionan la red (usuarios y recursos) de forma centralizada a través de un servidor con software específico (SOR). Ejemplos: Windows Server, Linux/UNIX server.
	<b>SO cliente o de escritorio</b>	Procesan la información, gestionan procesos, memoria y E/S de un equipo. Orientados a uso como cliente de una red u ordenador independiente. Ejemplos: Windows 10, Ubuntu.



Financiado por  
la Unión Europea



MINISTERIO  
DE EDUCACIÓN  
Y FORMACIÓN PROFESIONAL



Plan de Recuperación,  
Transformación  
y Resiliencia



GENERALITAT  
VALENCIANA  
Consejería de Educación,  
Universidades y Empleo



FP CV  
Formación Profesional  
Conselleria d'Ensenyament



GVA NEXT  
Fondos Next Generation  
en la Comunitat Valenciana



[enlaces](#)

[OS stats](#)

# Boot

## 1. Proceso de arranque de un SO. Gestores de arranque

El arranque del sistema operativo tiene dos fases: Arranque del hardware y Arranque del software.

### Componentes de Arranque

- **BIOS (Basic Input Output System):** Circuito integrado con firmware que arranca el sistema y controla parámetros del hardware a bajo nivel.
- **POST (Power-On Self Test):** Programa dentro del firmware de la BIOS que comprueba el hardware.
- Tras el POST, la BIOS transfiere el control al medio de almacenamiento que contenga el **GESTOR DE ARRANQUE**.

#### Particiones y Sistemas de Ficheros

Una **partición** es una división lógica del disco que funciona como si fuese un disco duro independiente. En cada partición se puede instalar un único sistema de archivos (formateo).

- **Ventajas de usar particiones:** Organización de la información, eficiencia del disco, posibilidad de tener varios SOs en el mismo disco, y seguridad (el fallo de una partición no implica errores en las demás).
- **Sistemas de ficheros:** ext, ext2, ext3, ext4 (para Linux); FAT, FAT32, NTFS (para Windows).
- **Estructura lógica de un disco/partición:** Sector de arranque, Tabla de asignación de ficheros (FAT, índice), Directorio raíz, y la Zona de datos.

### Tipos de Particiones (Modelo MBR)

- **Física/Primaria:** Solo 4 por disco duro. Pueden contener sistemas operativos.
- **Extendida/Secundaria:** Solo puede haber una por disco. No almacena datos; contiene particiones lógicas. No contiene sistema de ficheros.
- **Lógica:** Creadas dentro de la partición extendida (máximo 23). Se pueden instalar sistemas operativos en ellas.

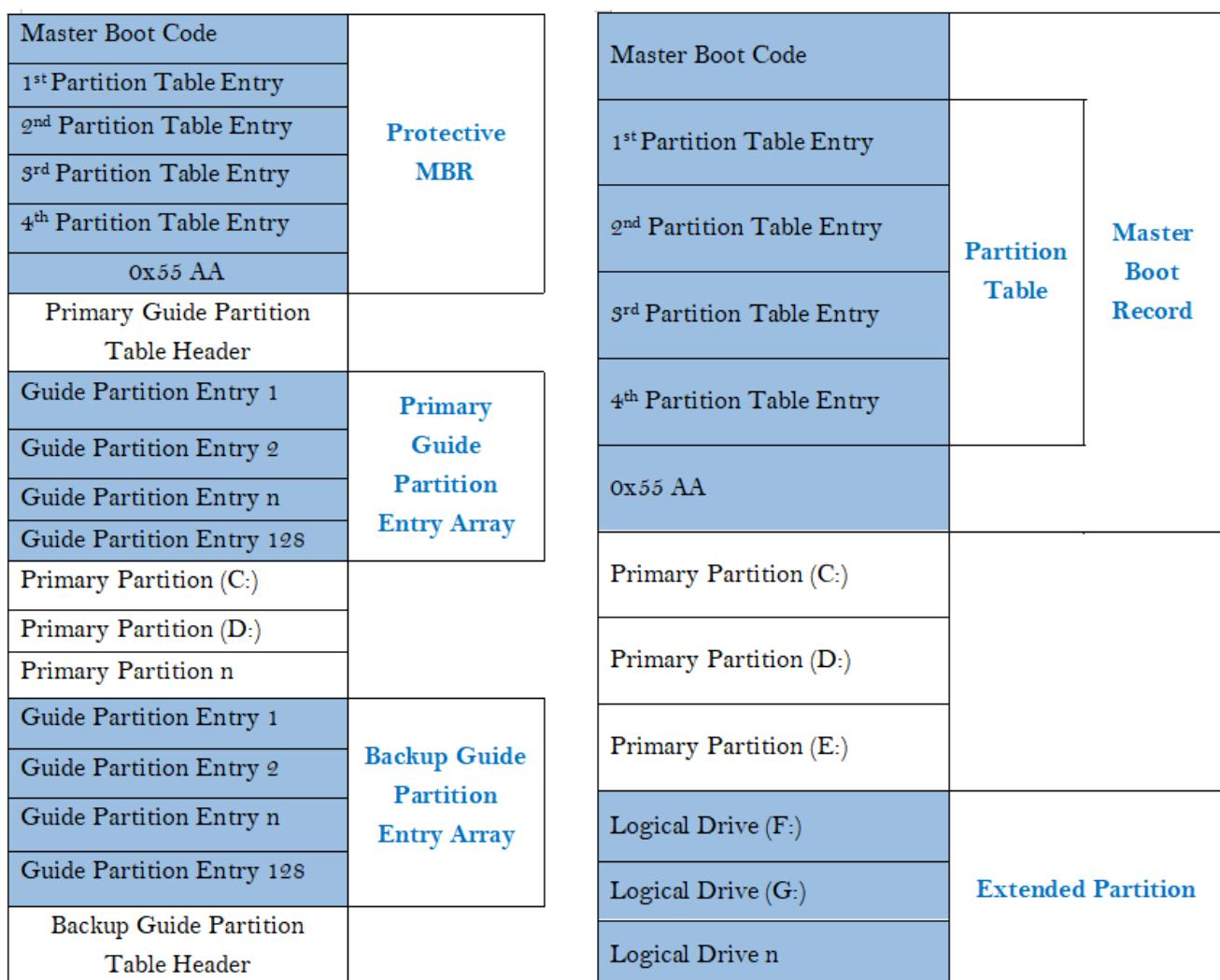
### Sistemas de Particionado (MBR vs GPT)

Los dos estándares más usados son MBR y GPT:

Estándar	MBR (Master Boot Record)	GPT (Globally Unique IDentifier Partition Table)
Uso	Obsoleto. Usado en ordenadores con BIOS.	Sustituye a MBR. Usado en ordenadores con <b>UEFI</b> .

<b>Estándar</b>	MBR (Master Boot Record)	GPT (Globally Unique IDentifier Partition Table)
<b>Capacidad</b>	Limitada a 2 TB.	Mejora la capacidad a teóricamente 9,4 ZB.
<b>Particiones</b>	Utiliza particiones físicas, extendidas y lógicas.	Solo existen particiones primarias (hasta 128 en SO Microsoft).
<b>Seguridad</b>	Baja.	Alta: La información de la tabla de particiones está al inicio y con una copia fragmentada y repartida por todo el disco para recuperación. Comprueba el núcleo de los SOs.

## GPT VS. MBR Structure



## UEFI y BIOS

El SO es el **primer programa que se carga en la memoria RAM**.

### Proceso de carga inicial:

1. La BIOS/UEFI comprueba que todo está correcto.
  2. Se busca el **cargador de arranque (bootloader)** del SO.
  3. El bootloader se encuentra en el sector de arranque del disco duro o SSD.
    - En sistemas con **BIOS**, se encuentra en el Master Boot Record (MBR).
    - En sistemas con **UEFI**, se encuentra en la partición EFI del disco.
- **UEFI** tiene un interfaz más amigable (admite ratón) y arranca más rápido que la BIOS.
  - En **BIOS**, el sector de arranque está en el MBR. En **UEFI**, esta información está en la GPT.
  - **Modos de arranque UEFI:**
    - **Heredado (BIOS):** Establece compatibilidad hacia atrás con discos con esquema MBR.
    - **UEFI:** Modo recomendado. Al instalar el SO en este modo, se crean particiones GPT.

### UBIOS

Busca información sobre UBIOS, un firmware que combina características de BIOS y UEFI.

## Nomenclatura de Particiones

- **Windows:** Asigna letras (C; D; E; etc.). Las letras A y B se reservaban tradicionalmente para unidades lectoras extraíbles (disquetes, cintas) por herencia de MS-DOS.
- **Linux (MBR):** Asigna una carpeta dentro de /dev (sda, sdb). Las particiones primarias/físicas van del 1 al 4 (sda1), y las lógicas a partir del 5 (sda5). Si existe una partición lógica (ej. sda5), implica que debe haber una partición extendida.
- **Linux (GPT):** Las particiones solo son primarias (sda1, sda2...).

## Gestores de Arranque (Bootloaders)

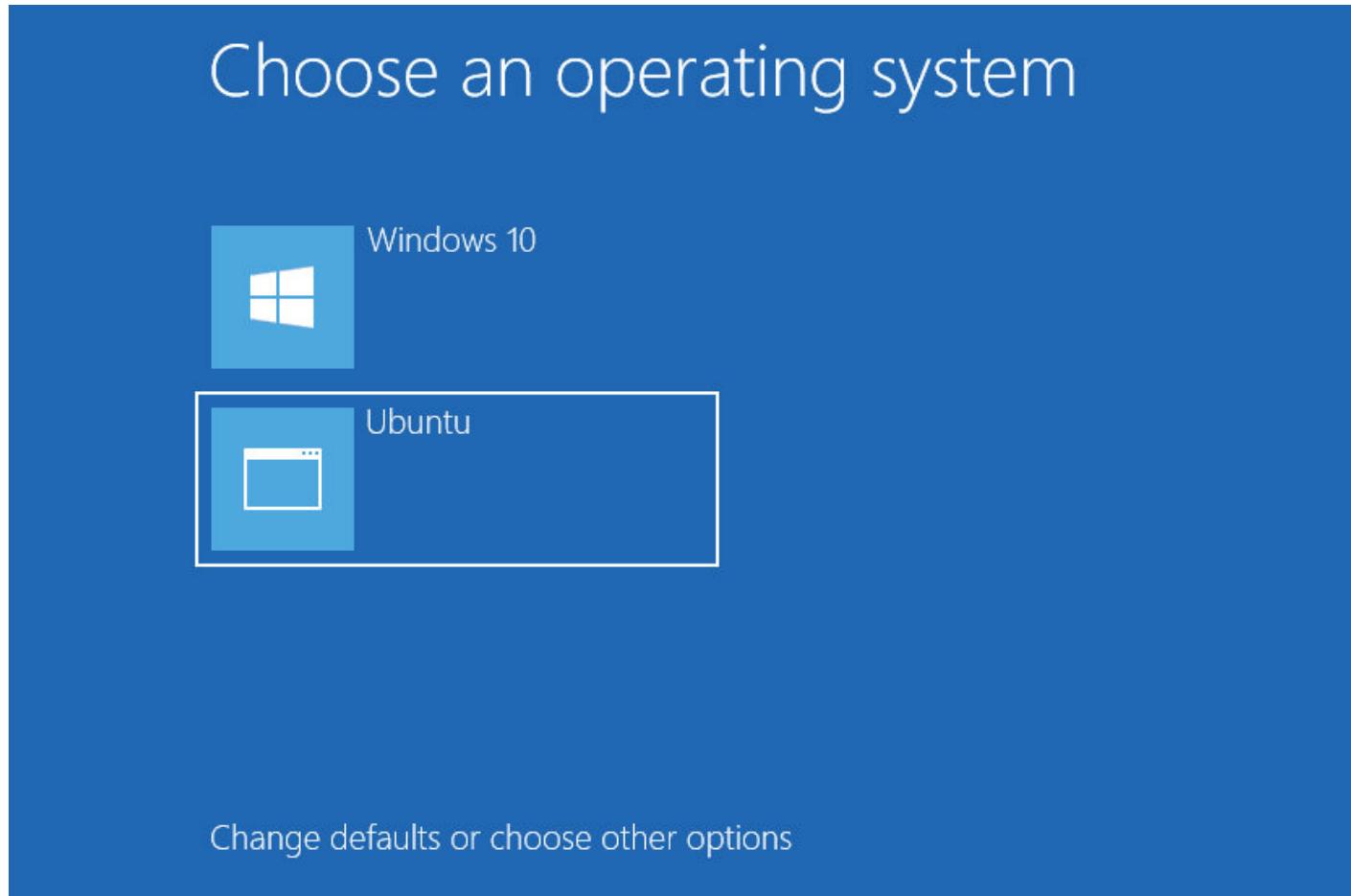
Un gestor de arranque localiza la partición activa con el sistema operativo y le transfiere el control. Si hay más de un SO, muestra un menú.

### 1. BOOTMGR (Windows Boot Manager)

Es el cargador de arranque utilizado por Microsoft Windows desde Windows Vista y Server 2012.

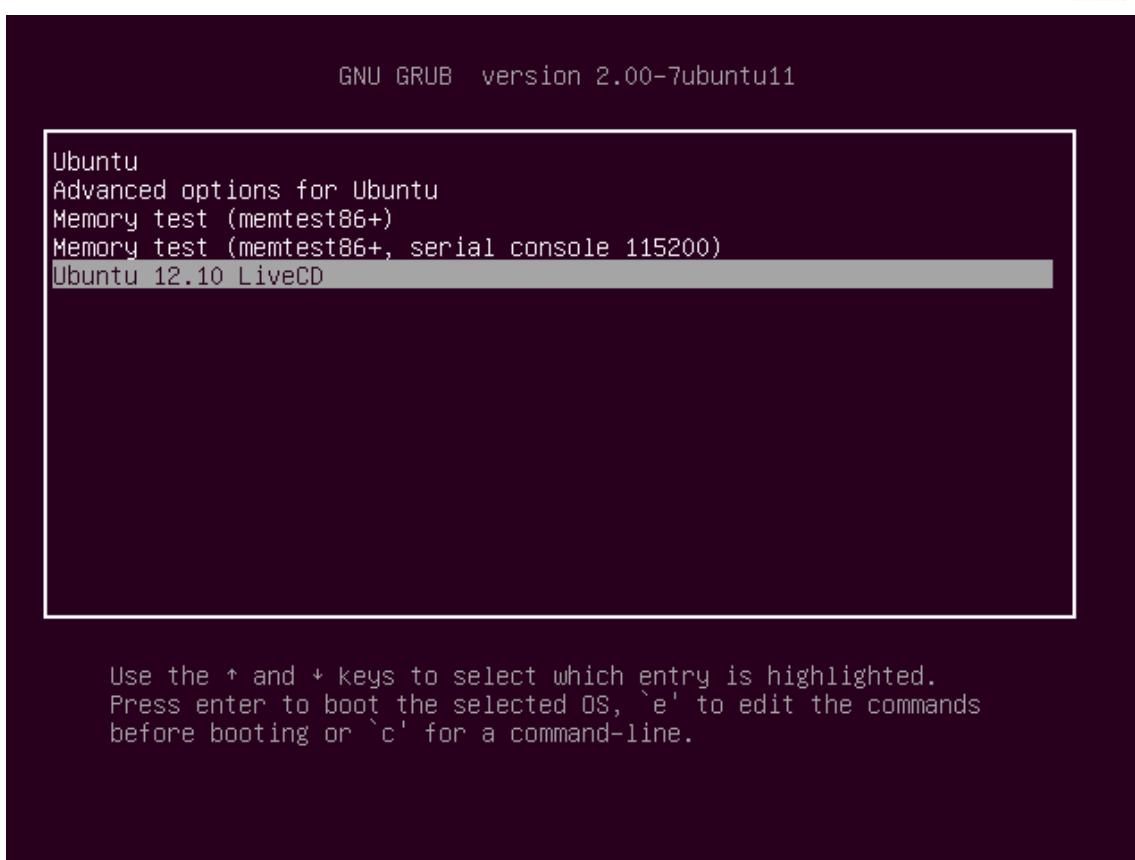
- **Inconveniente:** Da muchos problemas arrancando sistemas operativos distintos de Microsoft.

- Utiliza un almacén de datos de configuración de arranque, **BCD (Boot Configuration Data)**, que puede modificarse.
- La modificación de opciones se realiza mediante el comando BCDEDIT.

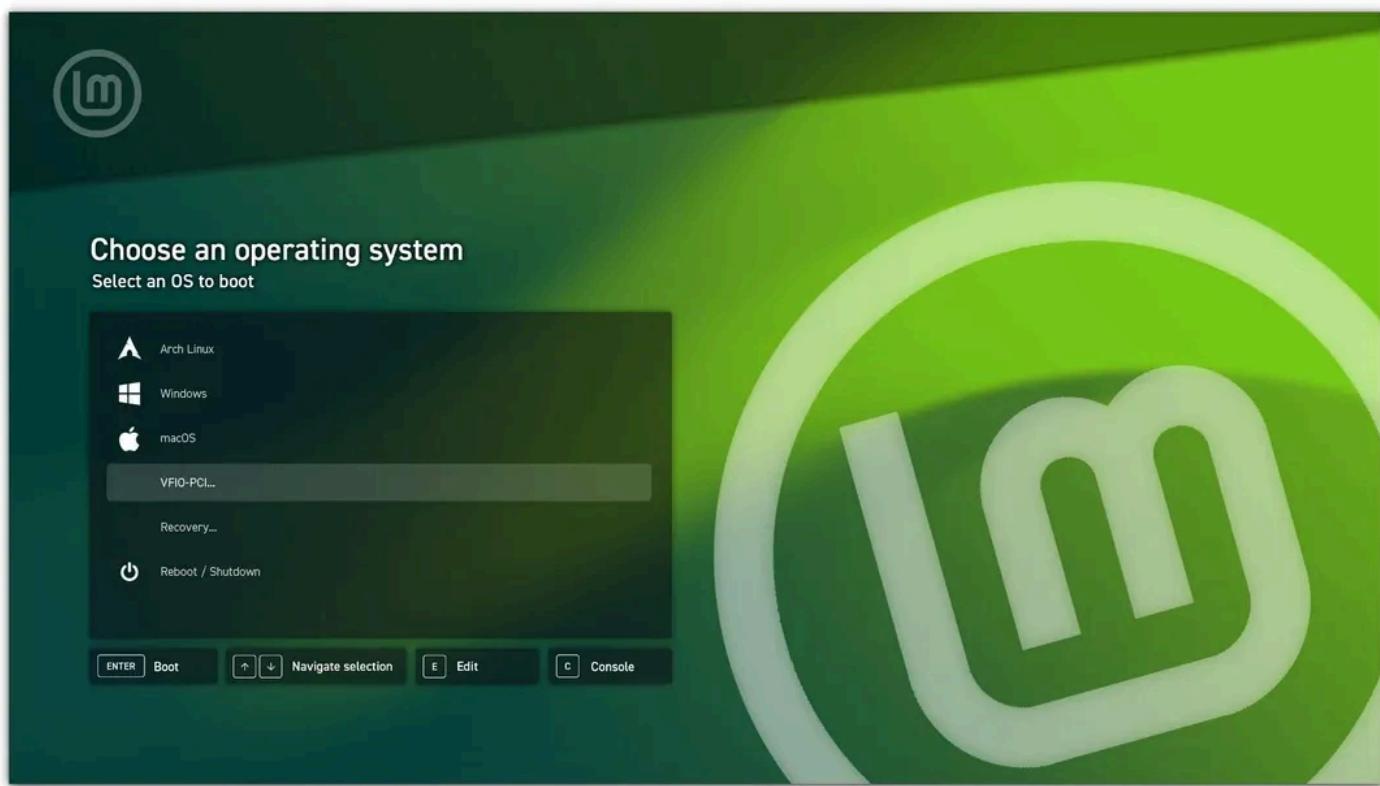


## 2. GRUB (GNU Grand Unified Bootloader)

Viene preinstalado en la mayoría de distribuciones GNU/Linux.



- Es muy potente y flexible; puede instalarse y **reconocer cualquier sistema operativo**, por lo que es el que se emplea en la práctica.
- Puede arrancar sistemas operativos basados en Linux, Windows, BSD y más.
- Muestra un menú gráfico para seleccionar las opciones de arranque. Si se selecciona un SO "no soportado" (no Linux), transfiere el control a su gestor de arranque propio.
- **Archivos de configuración de GRUB 2 en Ubuntu:**
  - /etc/default/grub: Gestiona el menú gráfico (tiempo de espera, selección por defecto, password). Se edita con privilegios de root.
  - /boot/grub/grub.cfg: Archivo principal de configuración (formado por scripts), *no debe ser modificado directamente*.
- Para modificar las opciones de GRUB, se edita /etc/default/grub y se ejecuta update-grub.



## Instalación en Sistemas Mixtos (Windows y Linux)

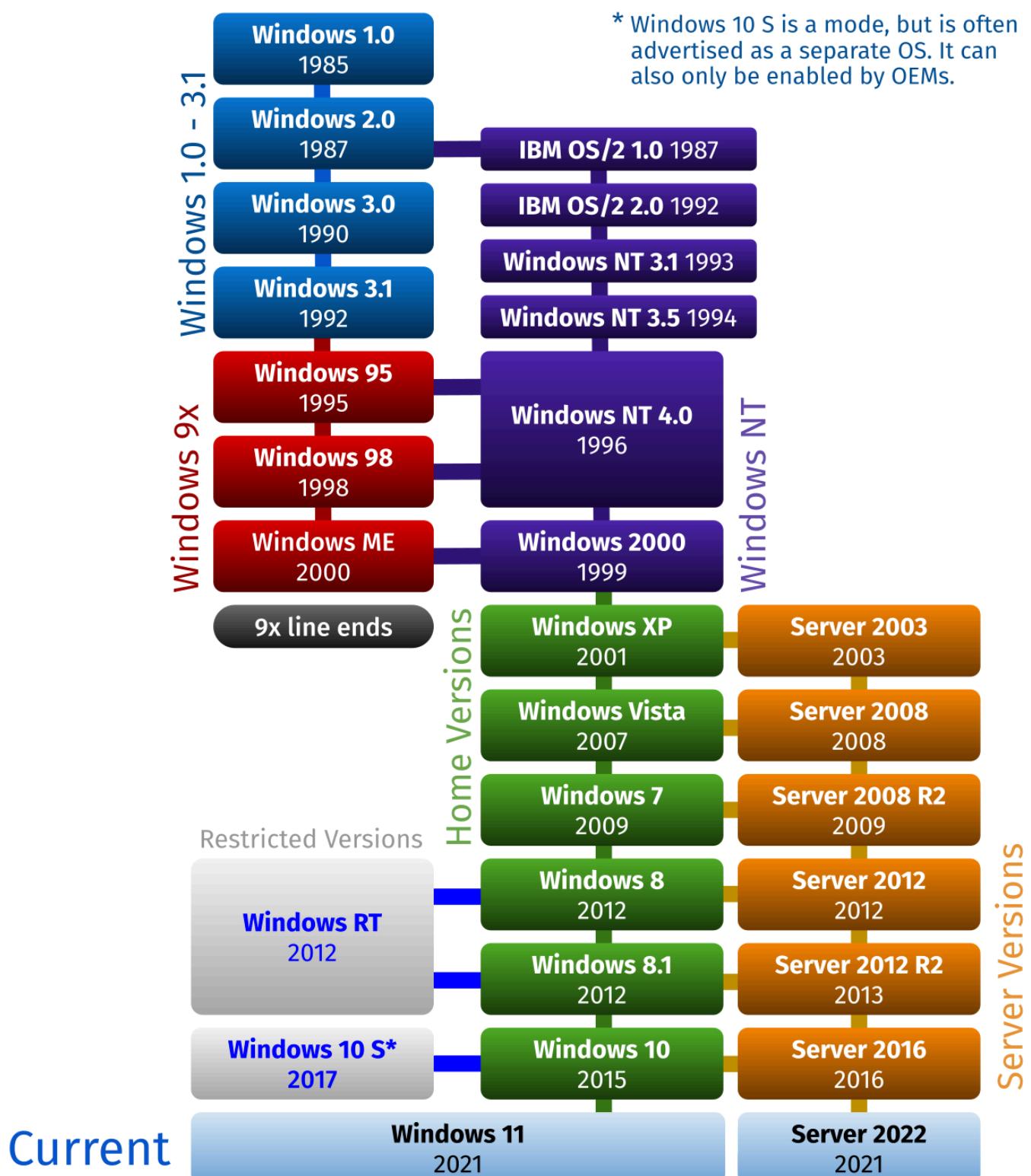
- **Windows:** Requiere una partición primaria para la unidad C: y crea automáticamente una partición para el sistema (arranque).
- **Linux:** Requiere un mínimo de 2 particiones (lógicas o primarias): el **directorio raíz** y la **memoria SWAP** (partición de intercambio). Es recomendable separar el directorio /home en otra partición para poder reinstalar el sistema sin perder datos.
- **Orden de instalación mixta:** Primero se instala **Windows** (situado en el disco principal) y después **Linux**, ya que su cargador de arranque (GRUB2) es capaz de iniciar Windows, mientras que BOOTMGR no lo hace bien con otros SOs. GRUB2 sobrescribiría a BOOTMGR.

# MS Windows

## Gestión de actualizaciones

- “Windows Update” es la herramienta donde se definen las directrices a seguir.
- Es recomendable mantener el sistema al día siempre que contemos con actualizaciones de tipo crítico.

## Árbol de versiones MS Windows.



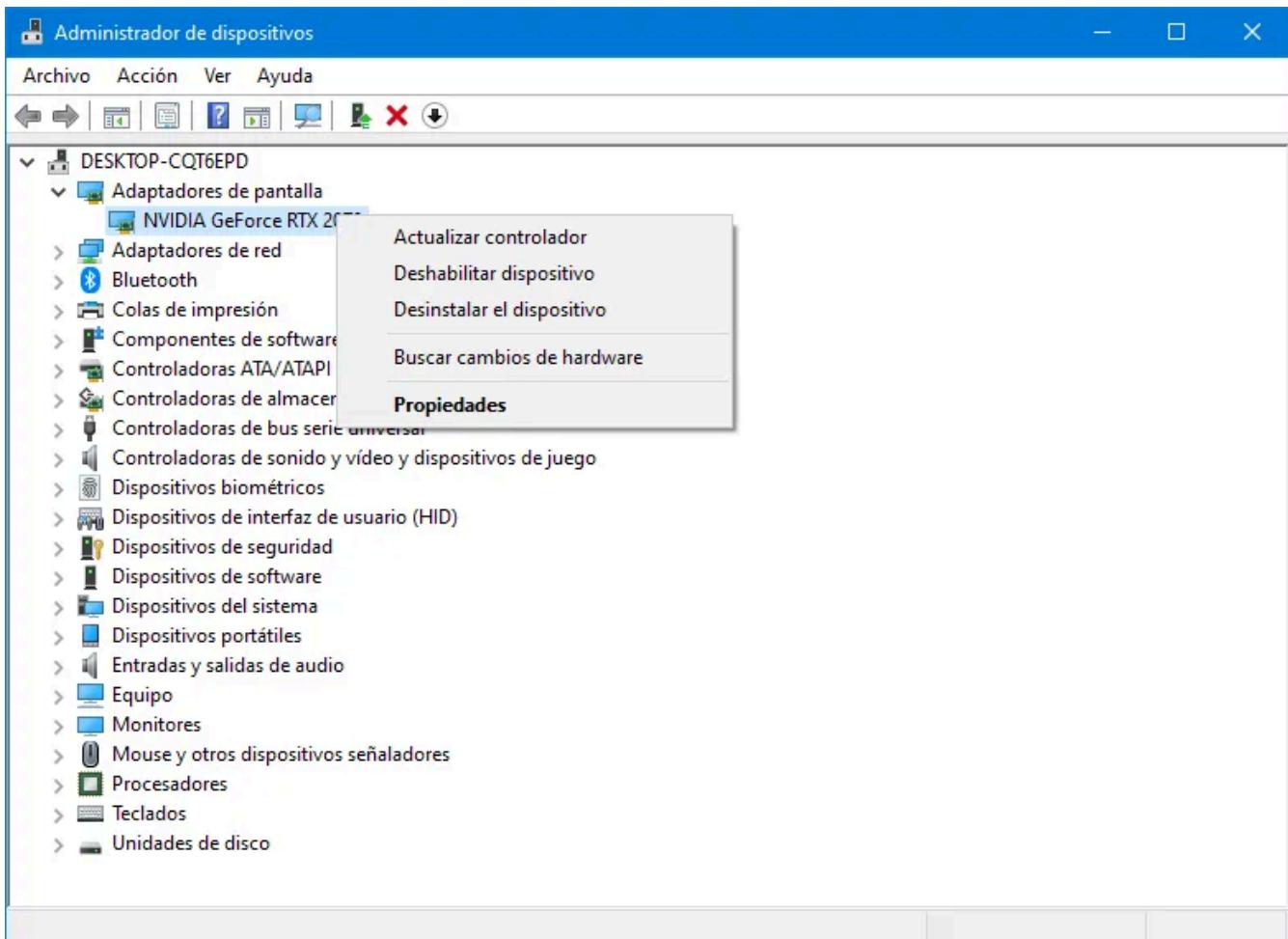
## Actualización de Controladores

1. Abre el **Administrador de Dispositivos** (clic derecho en Inicio).
2. Localiza y haz clic derecho en el dispositivo, selecciona **Actualizar controlador**.

3. Elige una de las opciones:

- **Buscar controladores automáticamente:** Windows buscará la última versión disponible.
- **Buscar en mi equipo:** Descarga primero los controladores desde la página web del fabricante.

4. Reinicia si es necesario.



## Tiny11

Tiny11 es una versión ligera de Windows 11 diseñada para mejorar el rendimiento en hardware antiguo o con recursos limitados.

- El uso de Tiny10/11 o cualquier versión modificada de Windows 10/11 constituye una violación directa del EULA de Microsoft.
- Se usa a nivel educativo para equipos con bajos recursos.

# GNU/Linux

**GNU** y **Linux** son componentes esenciales en muchos sistemas operativos de código abierto, pero desempeñan roles distintos:

- **GNU**: Es un proyecto iniciado en 1983 por Richard Stallman con el objetivo de desarrollar un sistema operativo libre similar a Unix. GNU proporciona\*\* herramientas y aplicaciones fundamentales, como compiladores, editores de texto y utilidades del sistema\*\*, que permiten la interacción del usuario con el sistema operativo.
- **Linux**: Es el **núcleo (kernel)** del sistema operativo, desarrollado por Linus Torvalds en 1991. El núcleo gestiona los recursos del hardware y permite la comunicación entre el software y el hardware, siendo responsable de tareas como la gestión de procesos, memoria y dispositivos.

La combinación del núcleo Linux con las herramientas y aplicaciones de GNU da lugar a un sistema operativo completo, comúnmente denominado **GNU/Linux**. Este término reconoce la contribución de ambos proyectos en la creación de un sistema operativo funcional y libre.

Es importante destacar que, aunque el término "Linux" se utiliza frecuentemente para referirse al sistema operativo completo, técnicamente se refiere solo al núcleo. Por ello, el uso de "GNU/Linux" es más preciso al referirse al sistema operativo en su totalidad.

## Distribuciones - "Distros"

Las **distribuciones Linux** son versiones del sistema operativo Linux adaptadas y personalizadas según el enfoque o las necesidades específicas de diferentes usuarios o grupos. Cada distribución incluye un conjunto específico de software, configuraciones predeterminadas y un gestor de paquetes que facilita la instalación y actualización de programas.

Las distros de Linux son desarrolladas, empaquetadas y mantenidas por empresas o por entusiastas con un objetivo concreto. Por lo general, se mantienen bajo licencia GNU, con opciones de pago.

## Tipos de usuarios vs distribuciones:

The diagram illustrates the variety of Linux distributions available, categorized by difficulty:

- Beginner-friendly:** Ubuntu, Pop!\_OS, elementary OS, Mint, Zorin, Solus.
- Intermediate:** Garuda Linux, EndeavourOS, Manjaro, MX Linux, Fedora, OpenSUSE.
- Hard mode:** Arch, Gentoo, Slackware, Linux From Scratch, Qubes OS, NixOS.

Cada distribución tiene su propia filosofía y enfoque, haciendo que Linux sea una plataforma altamente versátil y adaptativa según las necesidades del usuario o del proyecto.

#### Árbol de Distribuciones

[Distribuciones](#)

## Métodos de Instalación/actualizaciones

### 1. Tiendas de aplicaciones gráficas:

- **Centro de Software de Ubuntu:** Proporciona una interfaz gráfica para buscar e instalar aplicaciones en Ubuntu.
- **GNOME Software:** Disponible en entornos de escritorio GNOME, permite gestionar aplicaciones de manera visual.
- **Discover:** Utilizado en entornos KDE Plasma, facilita la instalación de software mediante una interfaz amigable.

### 2. Repositórios: Servidores de paquetes, accedidos a través de un gestor de paquetes.

#### 1. Gestores de paquetes tradicionales

- **APT (Advanced Package Tool):** Utilizado en distribuciones basadas en Debian, como Ubuntu y Linux Mint. Permite instalar paquetes .deb desde los repositorios oficiales mediante comandos como `sudo apt install nombre-del-paquete`.

- **YUM/DNF:** Empleados en distribuciones basadas en Red Hat, como Fedora y CentOS. Facilitan la instalación de paquetes .rpm con comandos como `sudo dnf install nombre-del-paquete`.
- **Pacman:** Propio de Arch Linux y sus derivadas, gestiona paquetes con comandos como `sudo pacman -S nombre-del-paquete`.

## 2. Gestores de paquetes universales

- **Snap:** Desarrollado por Canonical, permite instalar aplicaciones empaquetadas con todas sus dependencias, funcionando en múltiples distribuciones. Se instala con `sudo snap install nombre-del-paquete`.
- **Flatpak:** Ofrece una plataforma para distribuir aplicaciones en diversas distribuciones, ejecutándolas en entornos aislados. Se utiliza con `flatpak install nombre-del-paquete`.
- **AppImage:** Proporciona aplicaciones portátiles que no requieren instalación; basta con descargar el archivo .AppImage, otorgarle permisos de ejecución y ejecutarlo directamente.

## 3. Compilación desde el código fuente:

- Algunas aplicaciones se distribuyen en forma de código fuente. Para instalarlas, se descargan los archivos fuente, se descomprimen y se compilan utilizando comandos como `./configure`, `make` y `sudo make install`. Este método ofrece flexibilidad, pero requiere conocimientos técnicos y puede ser más complejo.

## Comandos de APT

APT es el sistema de gestión de paquetes para distribuciones basadas en Debian, y facilita la instalación y administración de software.

- `apt update`: Actualiza la lista de paquetes de los repositorios configurados en el sistema para que el gestor tenga la información más reciente.
- `apt install <paquete>`: Instala el paquete especificado y sus dependencias.
- `apt upgrade`: Actualiza todos los paquetes del sistema a las versiones más recientes.
- `apt show <paquete>`: Muestra información detallada sobre un paquete, como su versión, descripción y dependencias.
- `apt list --installed`: Lista los paquetes que están instalados en el sistema.
- `apt edit-sources`: Permite editar el archivo de fuentes de los repositorios para añadir o modificar repositorios.
- `apt-get --purge remove <paquete>`: Elimina un paquete junto con sus archivos de configuración.

## Tipos de repositorios

- **Main:** Software oficial respaldado por Canonical.
- **Restricted:** Software cerrado, drivers con soporte limitado.
- **Universe:** Mantenido por la comunidad, sin soporte oficial.
- **Multiverse:** Software con licencias restrictivas, sin soporte.

El repositorio **Main** respaldado por **Canonical** es específico de las distribuciones **basadas en Ubuntu**, ya que **Canonical** es la empresa que desarrolla y respalda **Ubuntu**.

#### aclaración

Cada distribución tiene su propio sistema de gestión de repositorios y su política de respaldo, que puede variar significativamente según la filosofía y objetivos de la distribución.

- En **Debian**, por ejemplo, existen repositorios como **Main**, **Contrib** y **Non-Free**, pero la mayoría del software en **Main** es mantenido por la comunidad sin el respaldo de una empresa como Canonical.
- En **Fedor**a, los repositorios oficiales son respaldados y mantenidos por la **comunidad Fedor**a con el soporte de **Red Hat**.
- En **openSUSE**, el repositorio **OSS (Open Source Software)** contiene software libre, mientras que la comunidad mantiene otros repositorios adicionales.

#### Enlaces

- [Vídeo introductorio Linux - es](#)
- [Testing distros on-line](#)

# Virtualización

## ⓘ Conexión con el currículo

RA: 2 - CE: d, e, f, g, h, i

Es una tecnología que permite crear una versión virtual de un componente de hardware o software, como un servidor, un sistema operativo, un dispositivo de almacenamiento o una red.

## 1. Tipos de Virtualización

- **Virtualización de Plataforma:** Su objetivo es ofrecer un sistema informático virtual.
  - **Hipervisor Tipo 1:** Se instala directamente sobre el hardware, también conocido como "bare metal".
  - **Hipervisor Tipo 2:** Se instala como una aplicación en el sistema operativo anfitrión, también conocido como hipervisor alojado.
- **Virtualización de Aplicaciones:** Ofrece un entorno virtual para aplicaciones, independientemente del hardware y sistema operativo.

### ⓘ Tip

[Virtualization - EN](#)

### 1.1 Virtualización de Plataforma

- **VMWARE:** La más utilizada en entornos empresariales.
- **Microsoft Hyper-V:** [Hyper-V](#)
- **Oracle VirtualBox:** Más utilizado en entornos educativos.
- **QEMU:** Herramienta libre.

### 1.2 Virtualización de Entornos de Aplicaciones

- **Docker:** Herramienta de creación y gestión de contenedores de aplicación.
- **Kubernetes:** Herramienta de coordinación de contenedores Docker.

## 2. Microsoft Hyper-V vs WSL vs Azure

- **Microsoft Hyper-V:** Herramienta de virtualización integrada en sistemas Pro de Microsoft.



MINISTERIO  
DE EDUCACIÓN  
Y FORMACIÓN PROFESIONAL



Plan de Recuperación,  
Transformación  
y Resiliencia



FP<sub>CV</sub>  
FP<sub>CV</sub>  
Formación Profesional  
Conselleria d'Educació, Universitats i Esports

GVA NEXT  
Fondos Next Generation  
en la Comunitat Valenciana



- **WSL (Windows Subsystem Linux):** Permite instalar y ejecutar aplicaciones Linux.
- **Azure Virtualization:** Posibilidad de implementar máquinas virtuales en la nube, con acceso mediante RDP.

#### Actividad

- Instalaciones
- Investigación prerrequisitos de plataforma y obtención de medios.

# VirtualBox - Funciones Avanzadas

## 1. Guest Additions

**Guest Additions** es un conjunto de programas que mejoran la experiencia del usuario y la eficiencia de las máquinas virtuales. Entre las mejoras que proporciona se incluyen:

- **Soporte para pantalla completa:** Permite ajustar la resolución de la pantalla al tamaño del monitor del host.
- **Carpetas compartidas:** Facilita el intercambio de archivos entre el sistema host y el sistema guest.
- **Mejora del rendimiento gráfico:** Proporciona un controlador gráfico optimizado para el sistema guest.
- **Funcionalidad “arrastrar y soltar”:** Permite mover archivos entre el host y el guest simplemente arrastrándolos.

## 2. Snapshots (instantáneas)

Las **snapshots** permiten hacer una captura del estado de la máquina virtual en un momento determinado. Es una herramienta muy útil cuando se hacen cambios importantes, como la instalación de nuevo software o la configuración del sistema:

- **Restauración rápida:** Permite volver a un estado anterior si algo sale mal durante una instalación o prueba.
- **Múltiples snapshots:** Se pueden crear varias instantáneas para tener diferentes puntos de restauración.

## 3. Soporte para USB

VirtualBox permite el **soporte para dispositivos USB** conectados al host. Esto significa que los dispositivos USB pueden ser reconocidos y utilizados por las máquinas virtuales:

- **USB 2.0 y USB 3.0:** Con el paquete de extensiones instalado, es posible utilizar dispositivos USB de alta velocidad.
- **Montaje de dispositivos automático:** VirtualBox puede montar automáticamente los dispositivos USB conectados para facilitar su uso.

## 4. Módulo de Red Avanzada

**VirtualBox** ofrece varias opciones de configuración de red para las máquinas virtuales:

- **NAT:** Permite que el guest acceda a internet a través del host.



MINISTERIO  
DE EDUCACIÓN  
Y FORMACIÓN PROFESIONAL



Plan de Recuperación,  
Transformación  
y Resiliencia



- **Bridged Networking:** Hace que el sistema guest aparezca como un dispositivo más de la red local, permitiendo la comunicación directa con otros dispositivos.
- **Internal Network:** Las máquinas virtuales pueden comunicarse entre ellas, pero no con la red externa.
- **Host-Only Networking:** Solo permite la comunicación entre el host y las máquinas virtuales, sin acceso a la red externa.

## 5. Carpetas Compartidas

Las **carpetas compartidas** permiten compartir directorios entre el sistema host y la máquina virtual. Esto es muy útil para transferir archivos sin tener que usar medios externos.

## 6. Remote Display Protocol (RDP)

VirtualBox proporciona un **servidor RDP** integrado, que permite acceder a la máquina virtual desde cualquier dispositivo compatible con el protocolo **Remote Desktop**. Esto es especialmente útil para trabajar de manera remota:

- **Control remoto completo:** Permite gestionar completamente la máquina virtual desde cualquier lugar.
- **Conexiones multiusuario:** Es posible que varios usuarios se conecten a la misma máquina virtual, compartiendo recursos y tareas.

## 7. Exportación e Importación en Formato OVA

Las máquinas virtuales creadas con VirtualBox pueden ser **exportadas** en un formato **OVA** (Open Virtual Appliance), que es compatible con otras soluciones de virtualización. Esto permite la portabilidad entre plataformas.

## 8. Integración con la Nube

VirtualBox ofrece integración con la nube, permitiendo la **creación de máquinas virtuales en la nube** y la conexión directa con Oracle Cloud Infrastructure:

- **Migración fácil:** Montar una máquina virtual en la nube para aumentar la disponibilidad y la escalabilidad.
- **Backup remoto:** Guardar copias de seguridad de las máquinas virtuales para asegurar su integridad.

## Instalación de Guest Additions



MINISTERIO  
DE EDUCACIÓN  
Y FORMACIÓN PROFESIONAL



Plan de Recuperación,  
Transformación  
y Resiliencia



Para instalar **Guest Additions** en una máquina virtual:

1. **Iniciar la máquina virtual.**
2. Seleccionar **Dispositivos > Insertar imagen de CD de Guest Additions.**
3. Seguir las instrucciones para completar la instalación.
4. **Reiniciar** la máquina virtual después de la instalación.

## Creación y Administración de Grupos

**VirtualBox** permite organizar máquinas virtuales en **grupos** para facilitar su administración, sobre todo cuando se manejan múltiples máquinas. Esta característica ayuda a realizar acciones comunes sobre varias máquinas al mismo tiempo.

### 1. Creación de Grupos

Para crear un grupo en VirtualBox:

1. **Seleccionar las máquinas virtuales** que deseas agrupar, manteniendo presionada la tecla **Ctrl**.
2. **Clic derecho** sobre cualquiera de las máquinas seleccionadas y elige la opción “**Crear grupo**”.

### 2. Administración de Grupos

Una vez que las máquinas están agrupadas, se pueden realizar diversas acciones en ellas de forma simultánea:

- **Iniciar o Detener:** Puedes iniciar o detener todas las máquinas del grupo con un solo clic.
- **Eliminar Grupo:** Selecciona el grupo y haz clic derecho para eliminarlo.
- **Configurar Grupo:** Puedes modificar configuraciones compartidas por las máquinas del grupo.

## Oracle VirtualBox Administrador

Archivo Grupo Ayuda



## Herramientas

Nueva Añadir Descargar Iniciar

## Sistemes Informàtics



w10



Apagada



lubuntu



Apagada

## Entornos de Desarrollo



Debian



Apagada



Suse



Apagada



ubuntu



Apagada

## General

Nombre: w10  
Sistema operativo: Windows 10 (64-bit)  
Grupos: Sistemes Informàtics

## Sistema

Memoria base: 2048 MB  
Orden de arranque: Disquete, Óptica, Disco duro  
Aceleración: Paganación anidada, Paravirtualización Hyper-V

## General

Nombre: lubuntu  
Sistema operativo: Ubuntu (64-bit)  
Grupos: Sistemes Informàtics

## Sistema

Memoria base: 8640 MB  
Procesadores: 8  
Orden de arranque: Disquete, Óptica, Disco duro  
Aceleración: Paganación anidada, Paravirtualización KVM

## Previsualización



## Previsualización



actividad

Virtualbox Advanced

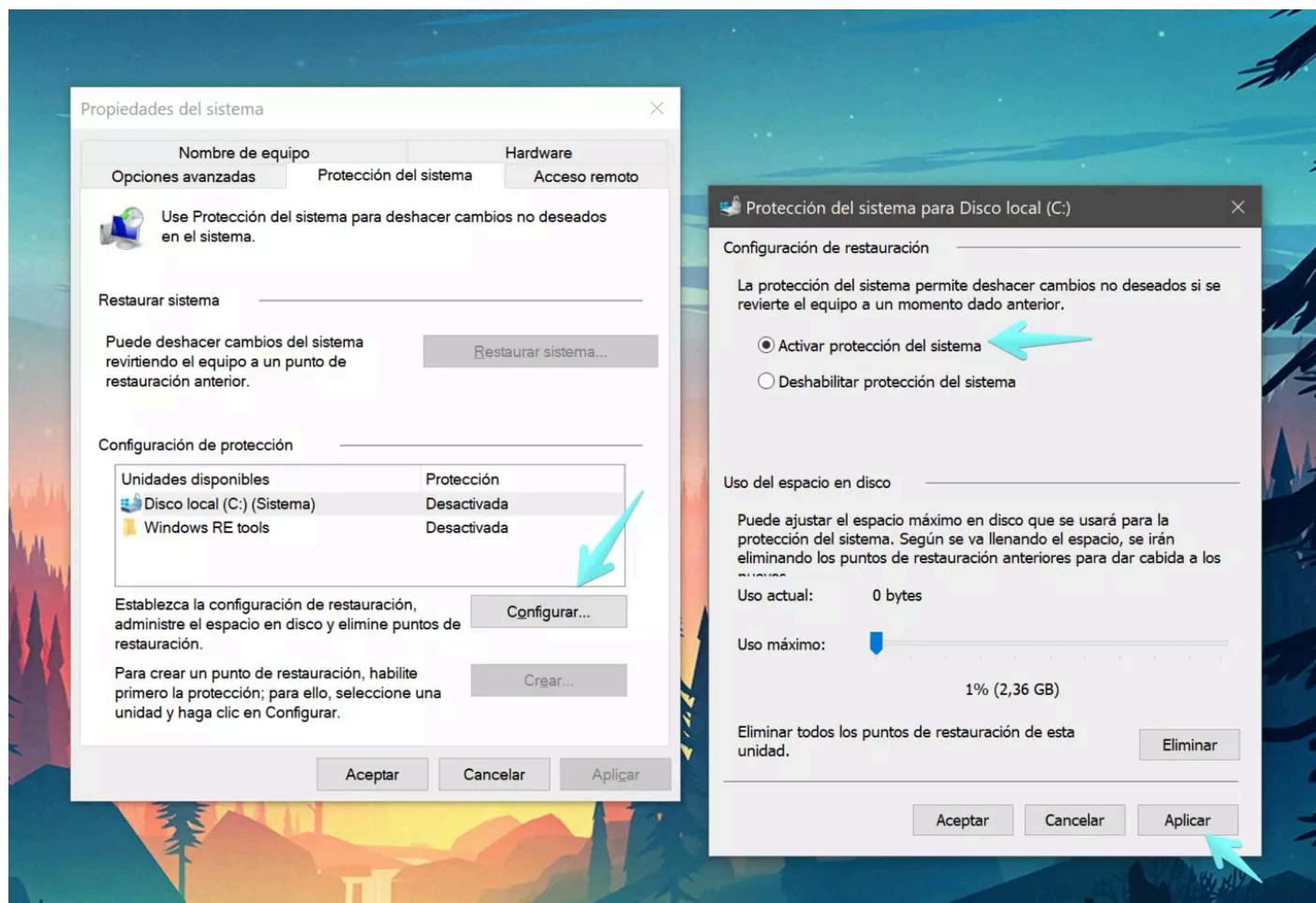
# Opciones de recuperación

## 1. MS Windows - Opciones de recuperación

En la [página principal de Windows "support"](#) disponemos de los posibles problemas y soluciones para cada uno de ellos.

### Puntos de Restauración en Windows

Los puntos de restauración son como “fotos” del estado de tu sistema en un momento determinado. Windows guarda información sobre la configuración del sistema, los archivos del sistema y el registro. Si algo va mal, como después de instalar un programa o controlador que causa problemas, puedes usar un punto de restauración para “retroceder en el tiempo” y revertir tu sistema a ese estado anterior.



### ¿Cuándo se crean los puntos de restauración?

- Automáticamente por Windows:
  - Antes de instalar actualizaciones importantes.
  - Al instalar algunos programas.
  - Periódicamente, si no se han creado otros puntos de restauración.



MINISTERIO  
DE EDUCACIÓN  
Y FORMACIÓN PROFESIONAL



Plan de Recuperación,  
Transformación  
y Resiliencia



GVA NEXT  
Fondos Next Generation  
en la Comunitat Valenciana



- Manualmente: Puedes crear un punto de restauración en cualquier momento.

## ¿Cómo crear un punto de restauración?

1. Busca "Crear un punto de restauración" en el menú inicio.
2. En la pestaña "Protección del sistema", selecciona la unidad del sistema (generalmente C:).
3. Haz clic en "Crear".
4. Escribe una descripción para identificar el punto de restauración.
5. Haz clic en "Crear" de nuevo.

## ¿Cómo usar un punto de restauración?

1. Busca "Crear un punto de restauración" en el menú inicio.
2. Haz clic en "Restaurar sistema".
3. Sigue las instrucciones en pantalla. Puedes elegir un punto de restauración recomendado o uno diferente.
4. Confirma la restauración. El equipo se reiniciará para completar el proceso.

### Recuerda

- Restaurar un punto no afecta a tus archivos personales, como documentos, fotos o música.
- Es recomendable crear puntos de restauración antes de realizar cambios importantes en el sistema.

### actividad

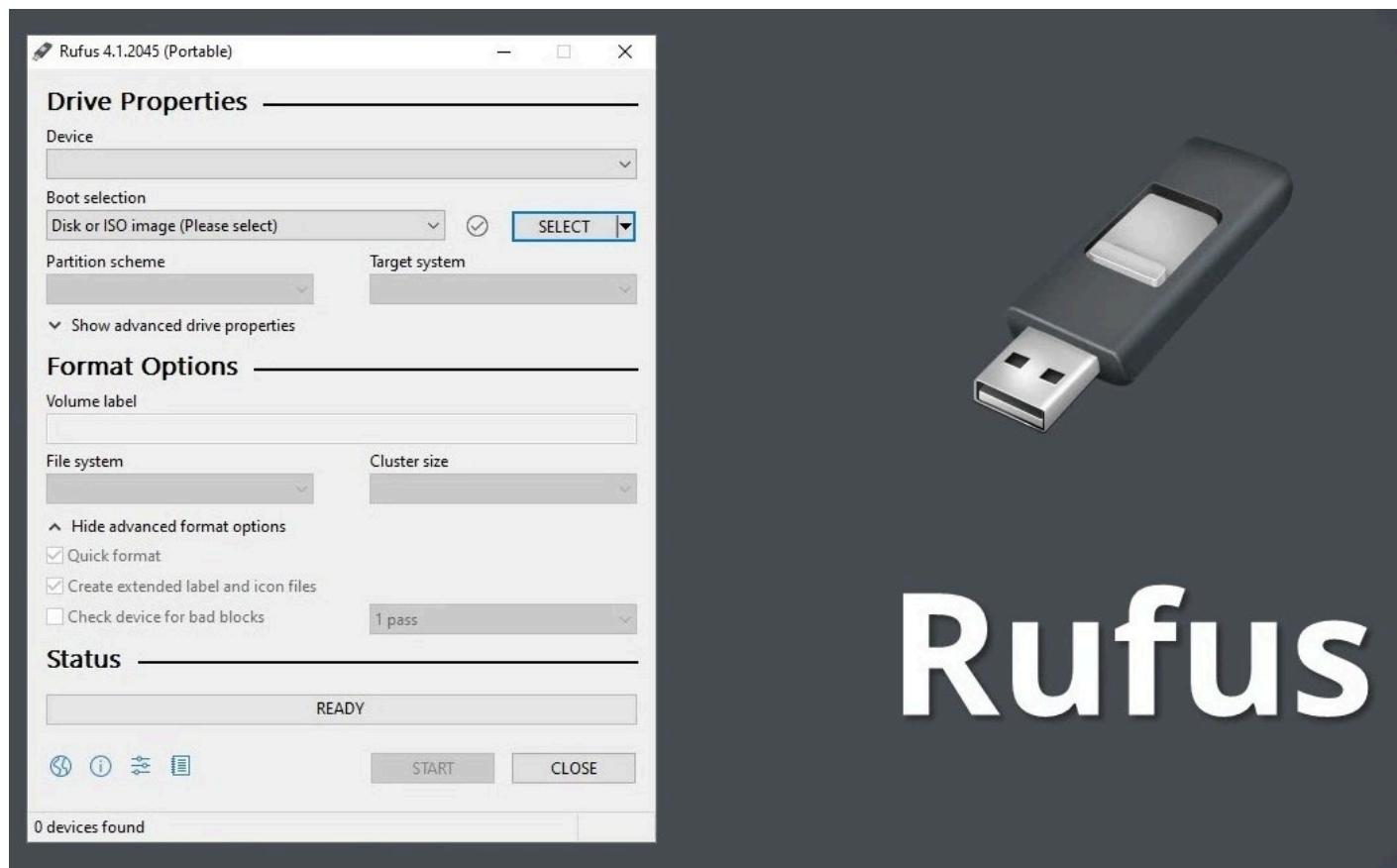
Punto de restauración

# Medios de Instalación

En el proceso de instalación o recuperación de sistemas operativos se suele utilizar USB para alojar las imágenes ISO con el sistema deseado, para ello, usamos diferentes herramientas que nos posibilitan el arranque desde estos dispositivos. También se pueden usar para la prueba de GNU/Linux, ya que facilitan una versión de prueba, LIVE, que queda residida en la memoria principal.

## 1. Rufus

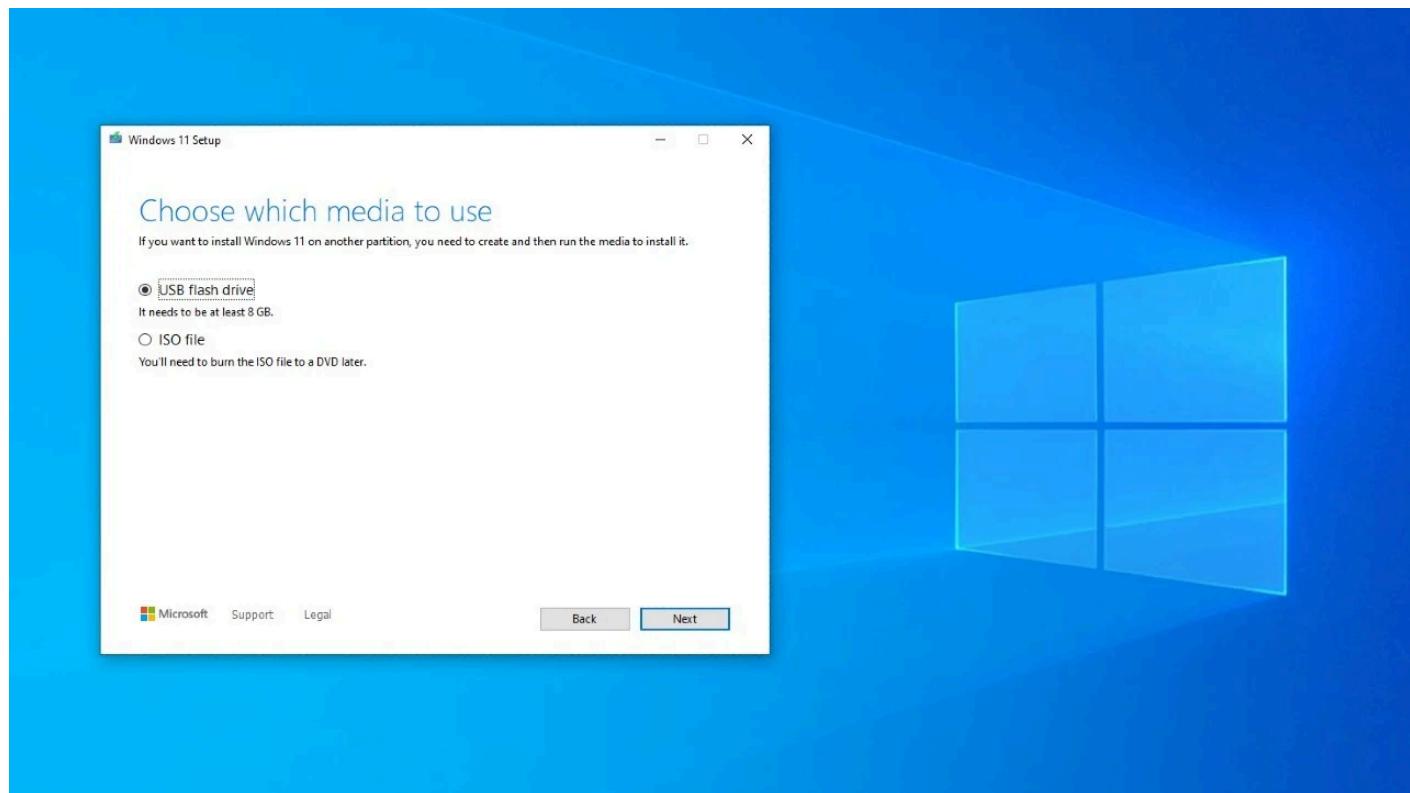
**Rufus** es una herramienta ligera y gratuita que permite crear unidades USB arrancables con sistemas operativos de forma rápida y sencilla. Es compatible con **Windows** y permite preparar medios para sistemas Windows, Linux y otros. Rufus destaca por ser fácil de usar, ofrecer múltiples opciones de personalización del proceso de creación, y ser capaz de formatear y configurar USBs con el formato necesario para arrancar desde BIOS o UEFI.



## 2. Windows Media Creation Tool

La herramienta oficial de Microsoft, **Media Creation Tool**, se utiliza para crear un USB de instalación con **Windows 10 o 11**. Solo funciona para Windows y permite descargar la ISO oficial y copiarla en una unidad

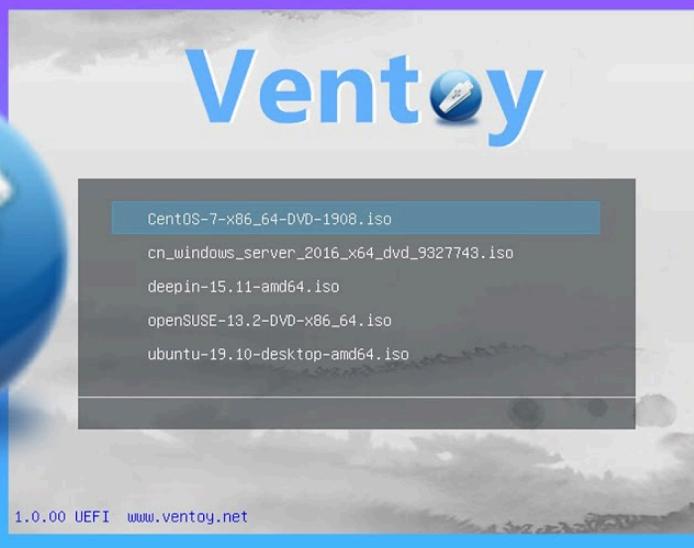
USB. La herramienta es muy simple y está diseñada para ofrecer una forma directa de crear medios de instalación, asegurando que el usuario tiene una copia oficial del sistema operativo.



### 3. Ventoy

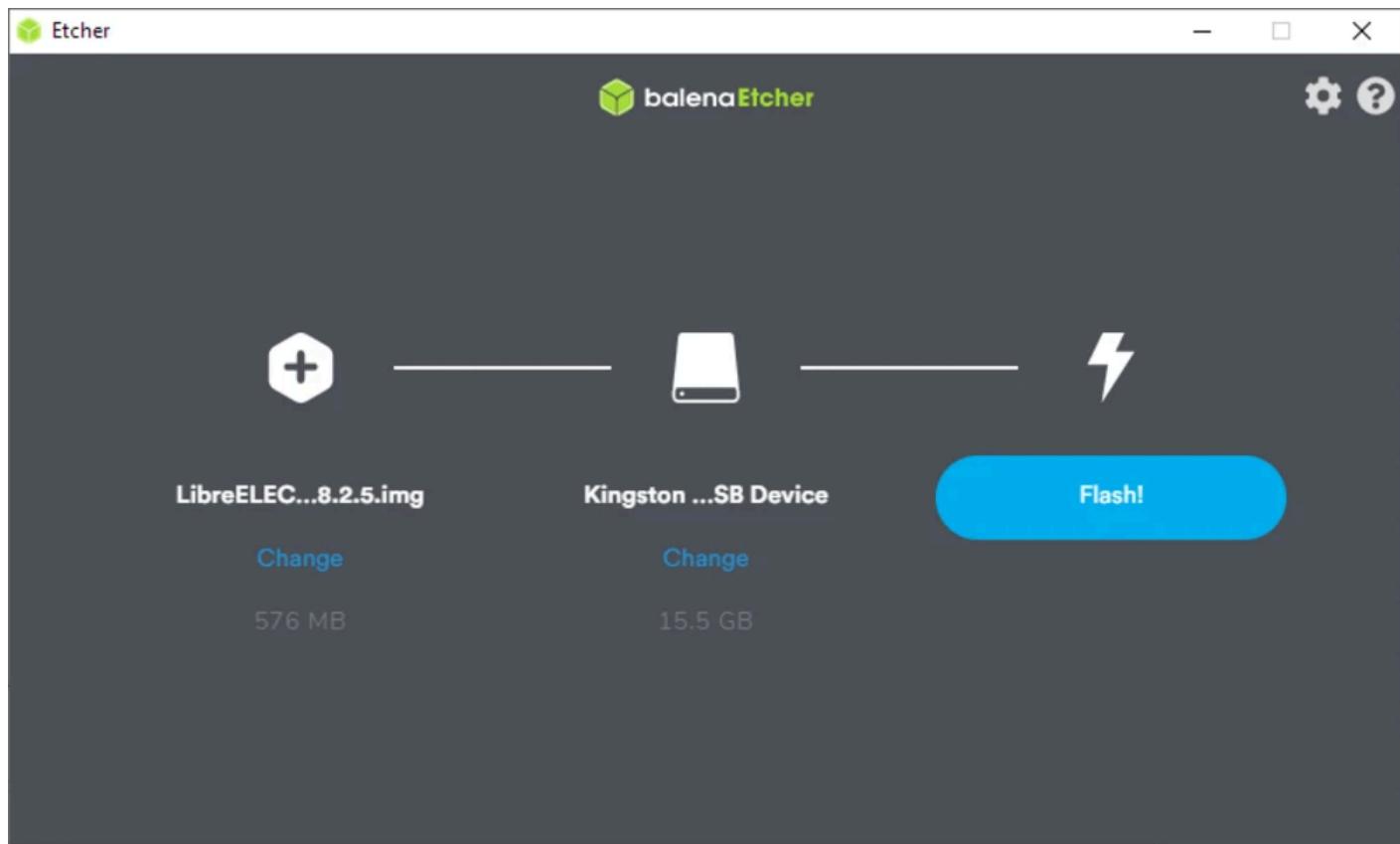
**Ventoy** es una herramienta innovadora que permite crear un USB arrancable donde puedes almacenar múltiples **ISOs**. En lugar de tener que copiar una única ISO cada vez, Ventoy prepara la unidad para que pueda arrancar desde cualquier ISO que copies en ella, ofreciendo una gran flexibilidad para quienes necesitan instalar distintos sistemas operativos o versiones. Es compatible con **Windows, Linux y BSD**, y se destaca por su simplicidad y capacidad de crear un “multiboot”.

## Una USB.... MUCHOS ISOs



## 4. Balena Etcher

**Balena Etcher** es una herramienta popular y multiplataforma (disponible para **Windows, macOS y Linux**) para crear unidades USB arrancables o grabar sistemas operativos en tarjetas SD, especialmente utilizada en proyectos como **Raspberry Pi**. Su interfaz sencilla y su proceso de tres pasos (seleccionar la imagen, seleccionar el medio y grabar) la hacen accesible para usuarios con pocos conocimientos técnicos, asegurando que se minimicen los errores durante el proceso de grabado.



## Comprobación de imágenes

Se puede realizar una comprobación de la integridad de un archivo descargado, asegurando que no haya sido modificado o corrompido durante la transferencia.

Junto a los archivos ISO, suele haber unos códigos MD5, SHA1/256/512

### Cómo verificar con SHA1:

#### 1. Linux o macOS:

```
sha1sum nombre_del_archivo.iso
```

#### 2. Windows (PowerShell):

```
Get-FileHash -Algorithm SHA1 nombre_del_archivo.iso
```

### Cómo verificar con MD5:

#### 1. Linux o macOS:

```
md5sum nombre_del_archivo.iso
```

## 2. Windows (PowerShell):

```
Get-FileHash -Algorithm MD5 nombre_del_archivo.iso
```

- Reemplaza `nombre_del_archivo.iso` con el nombre real del archivo descargado y compara el hash generado con el proporcionado.

## Comparativa entre SHA1, MD5 y SHA512:

Algoritmo	Longitud del Hash	Seguridad Actual	Uso Recomendado
<b>SHA1</b> Secure Hash Algorithm 1	160 bits (40 dígitos)	Vulnerable (no recomendado)	Verificación de archivos no críticos
<b>MD5</b> Message Digest Algorithm 5	128 bits (32 dígitos)	Vulnerable (no recomendado)	Pruebas de integridad básicas
<b>SHA512</b> Secure Hash Algorithm 512	512 bits	Muy seguro	Instalación de sistemas operativos, archivos críticos

### Ampliación

- [Generar y comprobar código hash](#)

### Actividad

Creación de USB Bootable

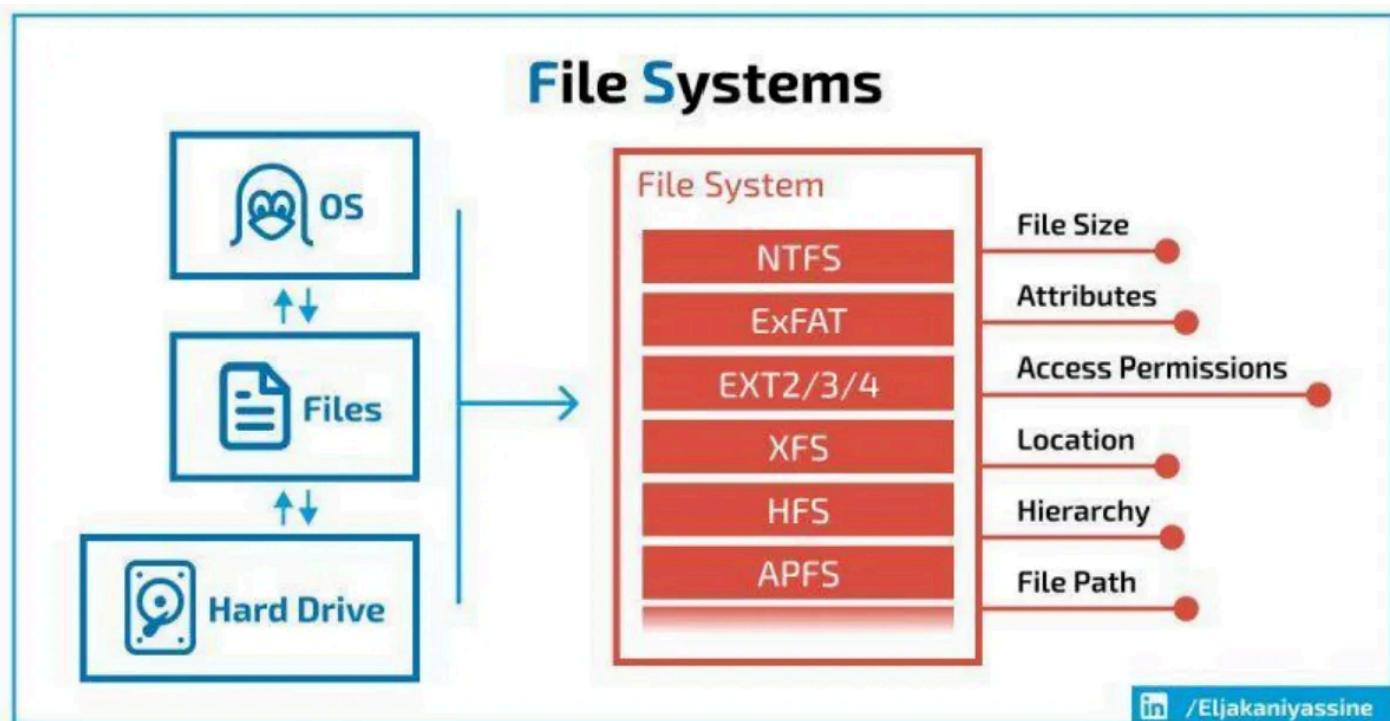
# Sistemas de ficheros

ⓘ Connexió con el currículum

RA: 3 - CE: a, b, c

## Sistemas de archivos

Un sistema de archivos es como un **índice o catálogo** que un sistema operativo utiliza para organizar y almacenar datos en un dispositivo de almacenamiento, como un disco duro, una unidad de estado sólido (SSD) o una memoria USB.



## ¿Por qué son importantes?

- **Organización:** Permiten organizar los datos de forma jerárquica en directorios y archivos, lo que facilita su localización y acceso.
- **Gestión del espacio:** Administran el espacio disponible en el dispositivo de almacenamiento, asignando espacio a los archivos y manteniendo un registro del espacio libre.
- **Acceso a los datos:** Proporcionan mecanismos para leer, escribir y modificar los datos almacenados en los archivos.
- **Seguridad:** Permiten establecer permisos de acceso a los archivos, controlando quién puede leerlos, escribirlos o ejecutarlos.

# File System Workflow



## Características de un sistema de archivos

- **Formato:** Cada sistema de archivos tiene un formato específico que define cómo se organizan los datos en el dispositivo.
- **Compatibilidad:** Los sistemas operativos suelen ser compatibles con varios sistemas de archivos, pero algunos son específicos de un sistema operativo en particular.
- **Rendimiento:** El rendimiento de un sistema de archivos puede variar en función de su diseño y de las operaciones que se realicen.
- **Fiabilidad:** Un buen sistema de archivos debe ser fiable y proteger los datos de posibles errores o fallos.

### ⚠️ Búsqueda avanzada

Haz una tabla con los diferentes tipos de sistemas de archivos (FAT32, NTFS, ext4, APFS, etc.) indicando sus características principales, ventajas y desventajas, sistemas operativos que lo soportan.

APFS, HFS+, FAT32, exFAT, NTFS, ext4, Btrfs, XFS

# Directories

## Nomenclatura y ubicación de las unidades de disco

Característica	Linux	Windows	macOS
<b>Nomenclatura</b>	/dev/sdXZ, /dev/nvmeXnYpZ	Letras: C:, D:, etc.	/dev/diskX, /dev/diskXsY
<b>Sistema de archivos</b>	Ext4, XFS, Btrfs, NTFS, FAT32	NTFS, FAT32, exFAT	APFS, HFS+, FAT32, exFAT
<b>Ubicación</b>	Montaje en directorios	Letras asignadas	Montaje automático en /Volumes
<b>Gestión</b>	Herramientas como lsblk, fdisk, GParted	"Administrador de discos", diskpart	Utilidad de Discos, diskutil CLI

### 💡 ¿Sabías que...?

En **Partitionado Tradicional** (SATA/AHCI) se divide este dispositivo en secciones lógicas (particiones), como /dev/sda1, /dev/sda2, etc. Donde todo el tráfico de entrada/salida (I/O) para todas las particiones (sda1, sda2) viaja a través del **mismo único canal y cola de comandos** del disco SATA.

Sin embargo, con **NVMe Namespaces**, cada Namespace puede tener su **propia cola de comandos independiente** y gestionar el tráfico I/O de forma separada. Esto mejora significativamente el rendimiento y la eficiencia. El SO los trata *como si fueran dos discos físicos diferentes* conectados.

## Estructura de Directorios en Windows

La estructura de directorios en Windows está organizada jerárquicamente y empieza por unidades de almacenamiento (por ejemplo, **C:**, **D:**). Estas unidades contienen todos los archivos y carpetas. A continuación, algunos de los directorios principales:

Directorio	Función
C:\	Directorio raíz del sistema, donde se encuentran todos los archivos principales.
C:\Program Files	Contiene programas instalados (aplicaciones de 64 bits).
C:\Program Files (x86)	Contiene programas de 32 bits en sistemas de 64 bits.

Directorio	Función
C:\Windows	Archivos del sistema operativo, como el kernel y bibliotecas del sistema.
C:\Users	Carpetas personales de cada usuario.
C:\Windows\System32	Archivos esenciales del sistema, incluyendo comandos y utilidades.
C:\Temp	Archivos temporales del sistema.
C:\Windows\Temp	Archivos temporales utilizados por el sistema y programas.
C:\ProgramData	Archivos de configuración de programas accesibles a todos los usuarios.
C:\Users\<Usuario>\AppData	Configuraciones locales de aplicaciones para cada usuario.



## Estructura de Directorios en Linux

En Linux, la estructura de directorios sigue el estándar **Filesystem Hierarchy Standard (FHS)** y todos los directorios parten de la raíz representada por **/**. Algunos de los directorios principales son:

## Jerarquía de Ficheros

Directorio	Función
/	Directorio raíz, contiene todos los demás directorios.
/bin	Programas esenciales del sistema (binarios), como comandos básicos ( <code>ls</code> , <code>cp</code> ).
/boot	Archivos necesarios para el arranque del sistema (núcleo de Linux, GRUB).

## Directorio Función

/dev	Archivos de dispositivos de hardware (discos duros, impresoras, etc.).
/etc	Archivos de configuración del sistema y programas instalados (configuración de red, usuarios).
/home	Directorios personales de los usuarios.
/lib	Bibliotecas compartidas y módulos necesarios para el sistema y el núcleo.
/media	Punto de montaje para medios extraíbles (discos externos, USB, CD).
/mnt	Directorio para montar temporalmente sistemas de archivos o discos.
/opt	Directorio para software adicional instalado opcionalmente.
/proc	Sistema de archivos virtual con información sobre procesos en ejecución y hardware.
/root	Directorio personal del usuario root (superusuario).
/sbin	Programas esenciales para la administración del sistema (solo para administradores).
/tmp	Directorio para archivos temporales.
/usr	Contiene aplicaciones y utilidades del usuario (subdirectorios como /usr/bin, /usr/lib, /usr/share).
/var	Archivos variables, como registros del sistema, colas de correo, etc. (/var/log).

## Rutas absolutas y relativas

### Ruta Absoluta:

- Es la ruta completa desde el directorio raíz (/ en Linux o C:\ en Windows).

### Ejemplos:

- **Linux:**

```
/home/usuario/Documentos/proyecto.txt
```

Comienza desde la raíz (/) y llega hasta proyecto.txt.

- **Windows:**



MINISTERIO  
DE EDUCACIÓN  
Y FORMACIÓN PROFESIONAL



Plan de Recuperación,  
Transformación  
y Resiliencia



C:\Users\Usuario\Documents\proyecto.txt

Comienza desde el disco (C:\) y llega hasta proyecto.txt.

## Ruta Relativa:

- Es la ruta desde la ubicación actual.

### Ejemplos:

- **Linux:**

Si estás en /home/usuario y quieres acceder a proyecto.txt en Documentos:

Documentos/proyecto.txt

- **Windows:**

Si estás en C:\Users\Usuario y quieres acceder a proyecto.txt en Documents:

Documents\proyecto.txt

# Particiones / Volúmenes

ⓘ Connexió con el currículum

RA: 3 - CE: d, g

## 1. Particionamiento de discos

El **particionamiento de discos** es el proceso de dividir un dispositivo de almacenamiento físico, como un disco duro o SSD, en múltiples secciones lógicas o particiones. Cada partición se comporta como una unidad independiente con su propio sistema de archivos, lo que permite una mejor organización y gestión de los datos.

### Usos comunes del particionamiento de discos:

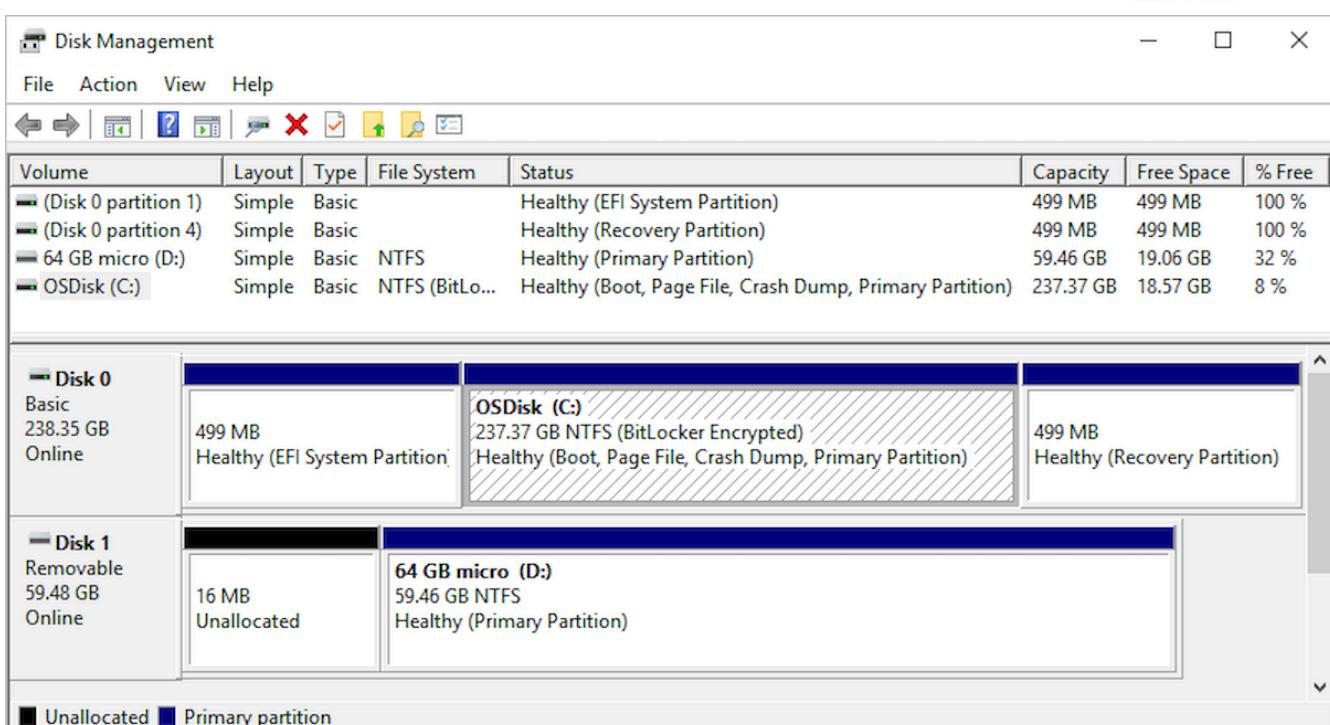
- **Instalación de sistemas operativos:** Permite instalar múltiples sistemas operativos en un solo disco físico, dedicando una partición a cada uno.
- **Copia de seguridad y recuperación:** Facilita la creación de particiones para copias de seguridad, simplificando la restauración en caso de fallos del sistema.
- **Seguridad y cifrado:** Mejora la seguridad al aislar datos sensibles en particiones separadas y permite aplicar cifrado en particiones específicas.

## Herramientas en Windows

### Escritorio

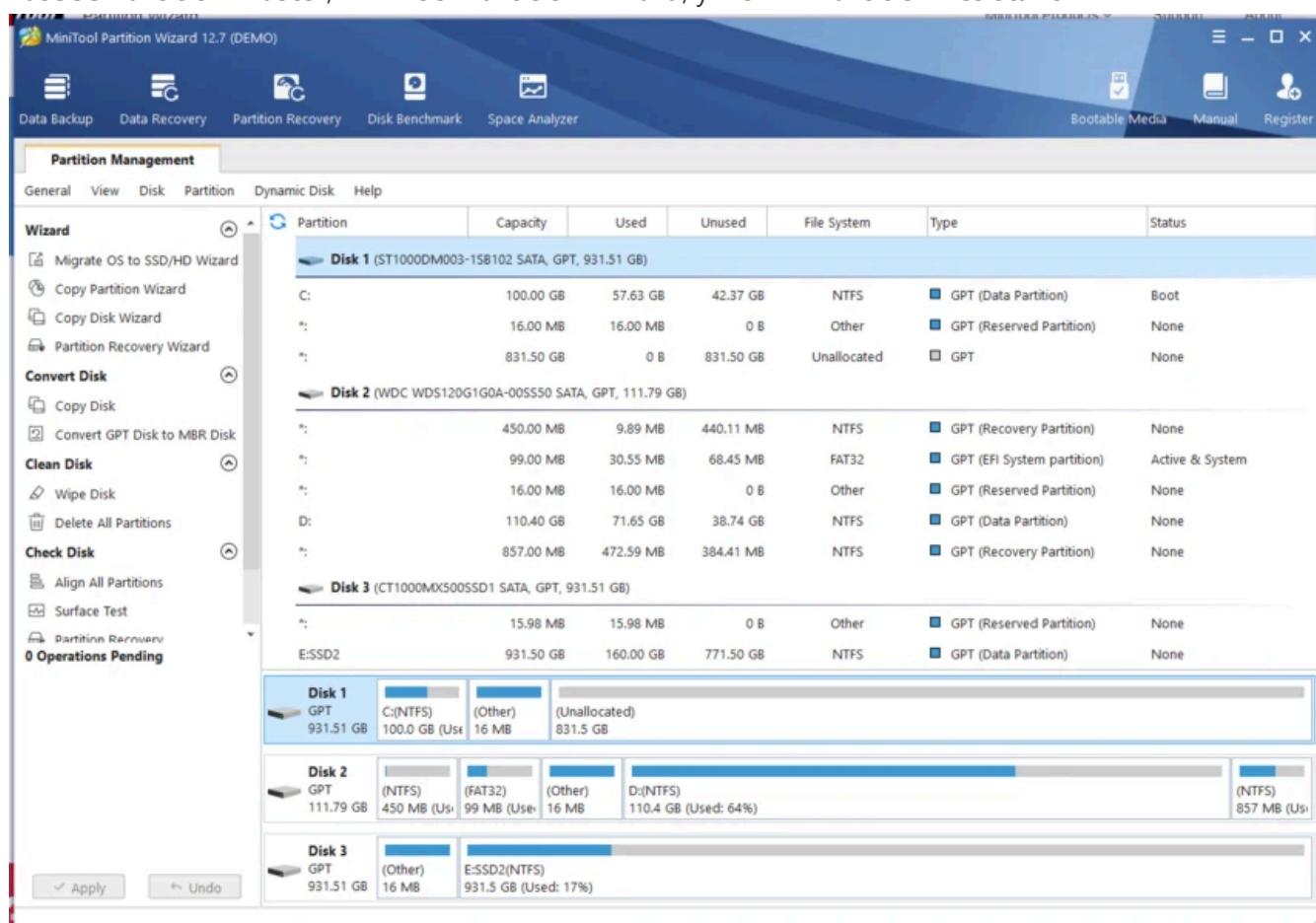
#### Administrador de Discos (Disk Management)

- Herramienta nativa de Windows



## Herramientas de Terceros

- EaseUS Partition Master, MiniTool Partition Wizard, y AOMEI Partition Assistant



## Terminal

## DiskPart

- Herramienta de línea de comandos de Discos.

```
diskpart
list disk
select disk 0
create partition primary size=10240
```

## Windows PowerShell

- También se pueden realizar algunas tareas relacionadas con particiones usando comandos en **PowerShell**:

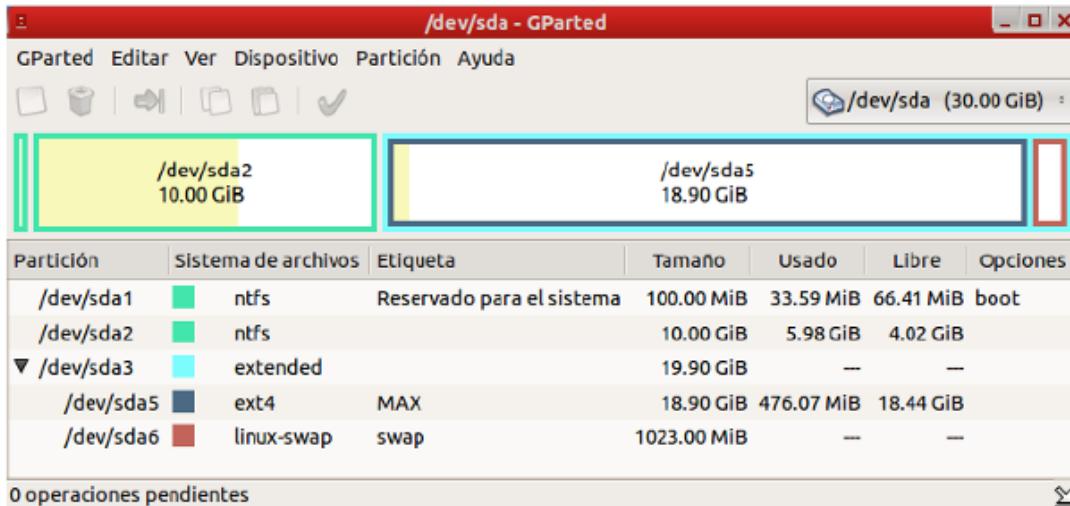
```
Get-Partition
New-Partition -DiskNumber 1 -UseMaximumSize -AssignDriveLetter
```

# Herramientas en Linux

## Escritorio

### GParted

- **GParted** es una herramienta gráfica (EXT4, FAT32, NTFS, etc.)



## Terminal

### parted

- **parted** es una herramienta más avanzada que **fdisk** y permite trabajar tanto con **MBR** como con **GPT**.
- Permite **redimensionar particiones** sin pérdida de datos, así como **crear, eliminar, y formatear**.
- Ejemplo de uso:

```
sudo parted /dev/sda
(parted) mklabel gpt
(parted) mkpart primary ext4 1MiB 20GiB
```

## 2. Agrupación de Discos / Volúmenes (Pooling)

### Windows: Volumen Simple Expandido (*Spanned Volume*)

El **Volumen Simple Expandido** es una característica de la Administración de Discos de Windows (requiere **discos dinámicos**) que permite a un usuario combinar el espacio libre no contiguo de **dos o más discos físicos** (hasta 32) en una **única unidad lógica** accesible por el sistema operativo.

- **Función:** Aumentar la capacidad de un volumen más allá del límite de un solo disco físico.
- **Mecanismo:** Los datos se escriben secuencialmente, llenando el espacio del primer disco antes de pasar al segundo, y así sucesivamente.
- **Riesgo Clave:** No ofrece redundancia; el fallo de **cualquier** disco en el volumen resulta en la pérdida total de los datos almacenados en el volumen expandido.

### Linux: LVM (*Logical Volume Manager*)

**LVM** es un subsistema de gestión de almacenamiento que proporciona una capa de abstracción sobre los discos físicos. Permite agrupar uno o más discos o particiones (*Physical Volumes - PV*) en un *Grupo de Volúmenes (VG)* para después crear particiones flexibles (*Logical Volumes - LV*) que pueden abarcar la capacidad combinada de los discos subyacentes.

- **Función:** Proporcionar una gestión de almacenamiento más flexible que las particiones tradicionales, permitiendo la creación de volúmenes lógicos que agrupan capacidad de múltiples discos físicos.
- **Mecanismo:** Abstacta el hardware, haciendo que el sistema de archivos vea el **LV** como una única partición, independientemente de cuántos **PV** físicos lo compongan.
- **Ventaja Principal:** Flexibilidad para cambiar el tamaño de los volúmenes lógicos (expandir o reducir) y moverlos, incluso mientras el sistema está en funcionamiento.

#### Actividad

Particiones y volúmenes

# Backup y automatización.

## ⓘ Connexión con el currículum

RA: 3 - CE: e, f, g

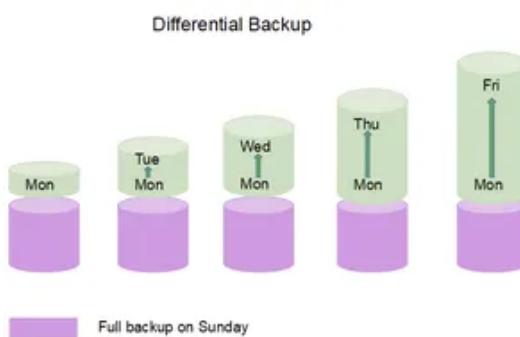
## 1. Introducción a las Copias de Seguridad

Una copia de seguridad es una copia de los datos importantes de un sistema informático que se realiza con el objetivo de:

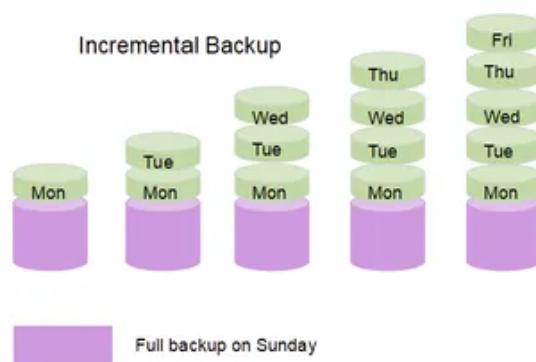
- **Recuperación de datos** (en caso de pérdida o daño)
- **Continuidad del negocio**
- **Protección contra ransomware**
  - El ransomware cifra los datos y exige un rescate para su recuperación
- **Cumplimiento legal**
  - En algunos sectores, las empresas están obligadas por ley a realizar copias de seguridad de ciertos datos

## Tipos de copias de seguridad

- **COMPLETA:** Copia la totalidad de los datos por eso tarda menos tiempo en recuperarlos, por lo contrario lleva mucho tiempo realizar una copia de seguridad, depende del tamaño de datos a respaldar.
- **DIFERENCIAL:** El lunes se hace una copia de seguridad sólo de los archivos que han cambiado desde el domingo, el martes se hace una copia de seguridad sólo de los archivos que han cambiado desde el domingo, y así sucesivamente hasta la siguiente copia de seguridad completa.



- **INCREMENTAL:** Las copias de seguridad incrementales también hacen una copia de seguridad sólo de los datos modificados, pero sólo de los datos que han cambiado desde la última copia de seguridad, ya sea completa o incremental.

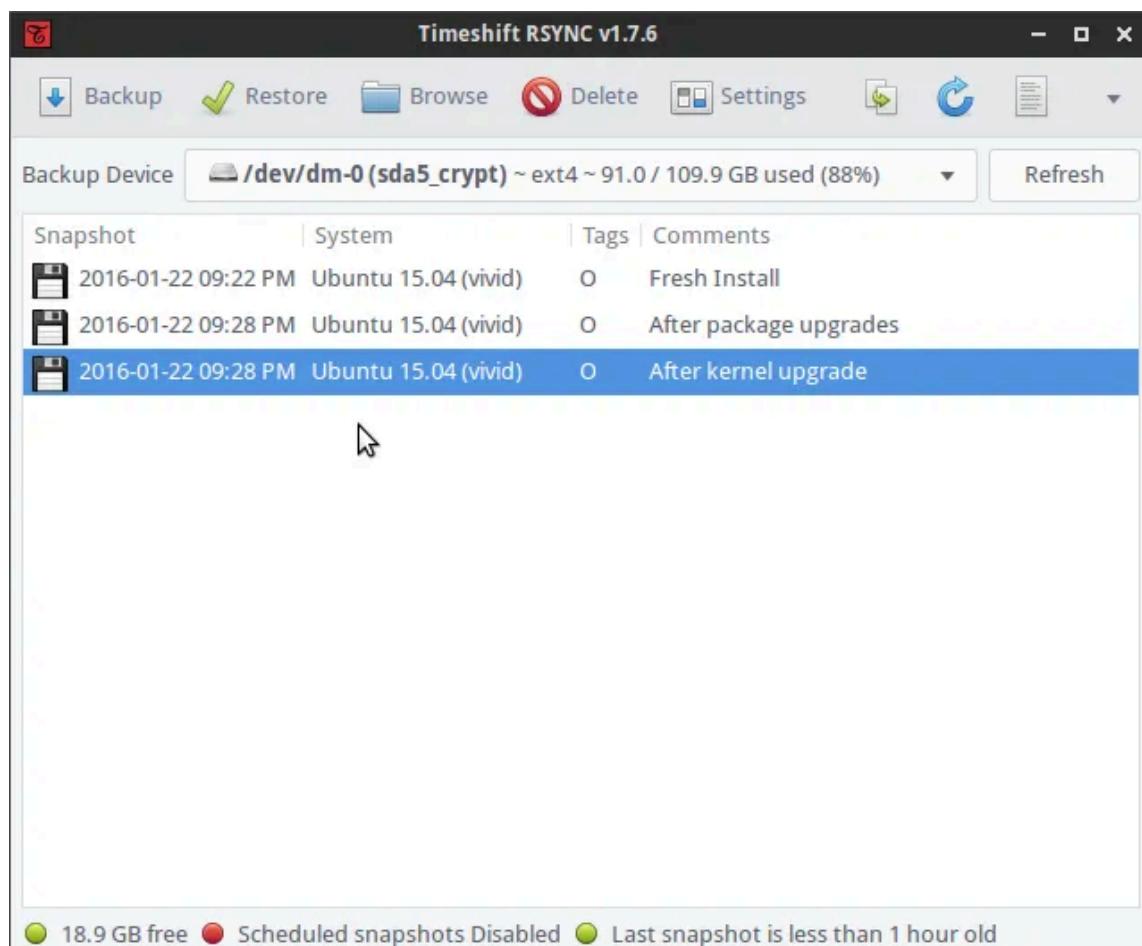


## Métodos de almacenamiento

- Almacenamiento local (discos duros externos, NAS).
- Almacenamiento en la nube (Google Drive, Dropbox, Amazon S3).

## Realización de Copias de Seguridad

- **Herramientas para realizar copias de seguridad:**
  - **Windows Backup / historial de archivos:** Herramienta nativa en sistemas Windows.
  - **Time Machine (MacOS):** Sistema de copia de seguridad automático.
  - **Duplicati y Veeam Backup:** Software de copias de seguridad para Windows, Linux y MacOS.
  - **rsync:** Comando para hacer copias de seguridad en Linux.



## 2. Planificación y Automatización de Tareas

- La planificación de tareas es un proceso que sirve para organizar y gestionar el tiempo de manera eficiente. Consiste en identificar y programar las tareas que se deben realizar

### Programación de tareas en el sistema operativo

- Windows:** El "[Programador de tareas](#)" permite automatizar tareas como la ejecución de programas, el envío de correos electrónicos o la realización de copias de seguridad.
- Linux:** **Cron** es una herramienta que permite programar la ejecución de comandos o scripts en momentos específicos. Para inicio de sesión existe la automatización del propio sistema con [aplicaciones de inicio](#)

 Tip

[crontab\\_guru](#)

## Scripts

- **Bash** (Linux): Permite automatizar tareas en sistemas Linux, como la gestión de archivos, la configuración del sistema o la ejecución de programas.
- **PowerShell** (Windows): Similar a Bash, pero para Windows. Permite automatizar tareas administrativas y de gestión del sistema.

### Actividad

Backups y automatización

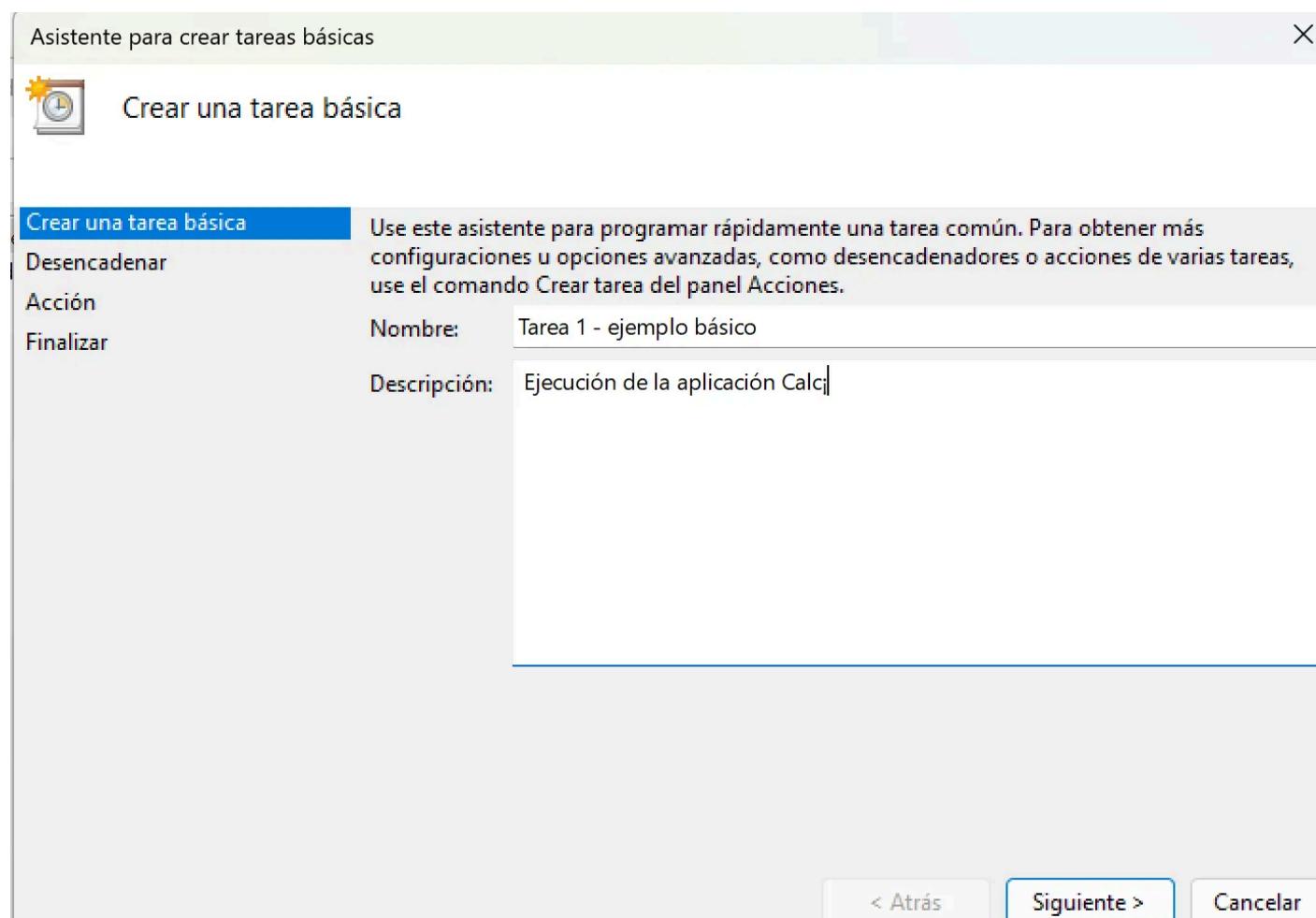
# Automatización Windows

## Gestión de Tareas Programadas en Windows

Una tarea programada es una acción que el sistema operativo realizará en alguna de las siguientes circunstancias:

- Cuando llegue un momento determinado en el futuro.
- Cuando se produzca un determinado evento.
- Con una frecuencia determinada.

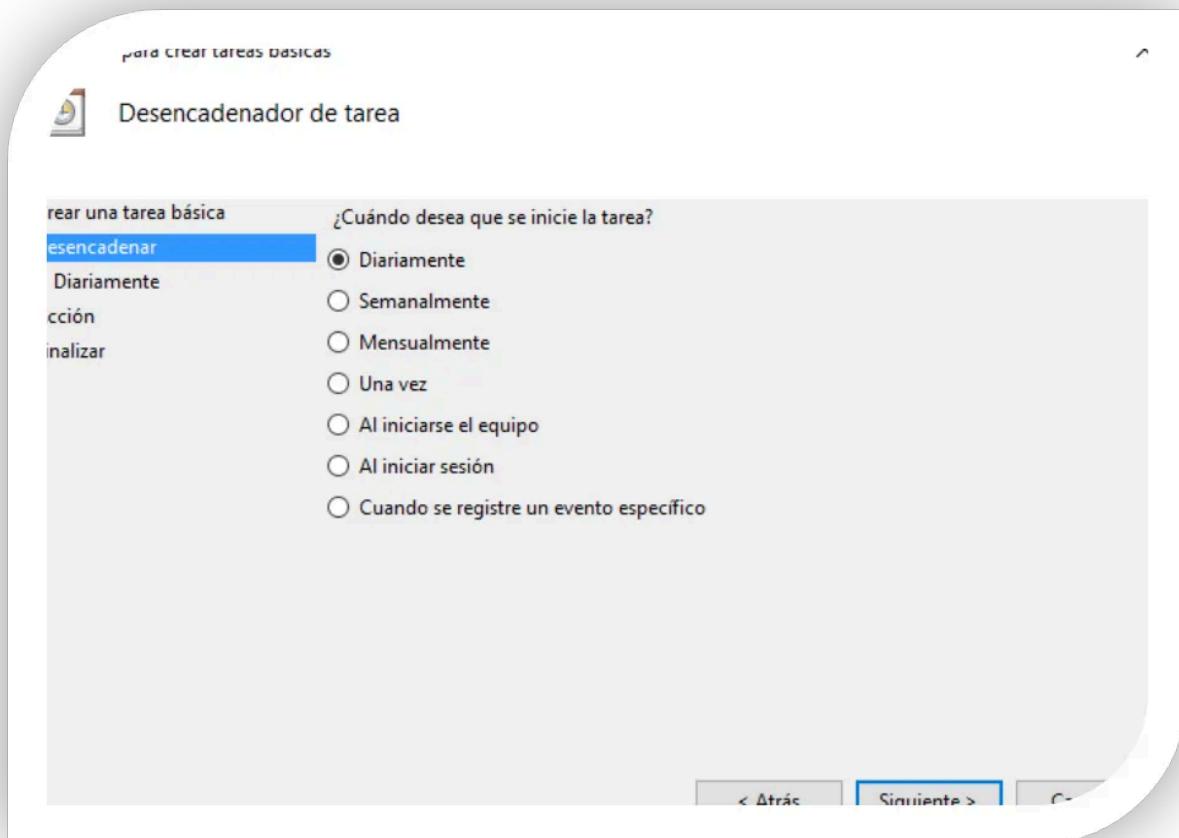
De esta manera, podremos planificar y programar aquellas acciones que deseamos que se realicen, evitando posibles olvidos. Para ello, utilizaremos el programador de tareas. Para acceder a él entraremos en **Panel de control > Herramientas administrativas > Programador de tareas** o también podemos hacerlo mediante el comando taskschd.msc.



La forma más sencilla de programar una tarea es hacer clic en **Crear tarea básica** (situado en el menú de acciones a la derecha). A continuación, aparecerá un asistente que nos ayudará a crear la tarea planificada.

Indicaremos cuándo o debido a qué queremos que se ejecute la tarea. En esta pantalla, seleccionaremos una de las opciones dependiendo de cómo o cuándo queremos que se ejecute la tarea (diariamente, semanalmente, mensualmente, una vez, inicio de sesión, fin de sesión, o vinculado a un evento particular).

Por ejemplo, vamos a establecer que la tarea se ejecute diariamente a una hora determinada.



- para crear tareas básicas

Diariamente

[Crear una tarea básica](#)      Inicio:      Sincronizar zonas horarias

[Desencadenar](#)

**Diariamente**      Repetir cada:  días

[Acción](#)

[Iniciar un programa](#)

[Finalizar](#)

[\*\*< Atrás\*\*](#) [\*\*Siguiente >\*\*](#) [\*\*Cancelar\*\*](#)

A continuación, indicaremos la **acción a realizar**. En este caso podremos indicar la ejecución de un programa, enviar un correo electrónico, mostrar un mensaje, etc.

- para crear tareas básicas

Acción

Crear una tarea básica

Desencadenar

Acción

Finalizar

¿Qué acción desea que realice la tarea?

Iniciar un programa

Enviar un correo electrónico (desusado)

Mostrar un mensaje (desusado)

< Atrás Siguiente > Cerrar

Como ejemplo, vamos a crear una tarea sencilla: abrir la calculadora de Windows a una hora determinada todos los días.



## Iniciar un programa

rear una tarea básica

esencadenar

Una vez

cción

Iniciar un programa

nalizar

Programa o script:

c:\windows\system32\calc.exe

Examinar...

Agregar argumentos (opcional):

Iniciar en (opcional):

< Atrás

Siguiente >

Ca...

#### Asistente para crear tareas básicas



##### Resumen

###### Crear una tarea básica

Desencadenar

Nombre:

Tarea 1 - Ejemplo básico

Diariamente

Descripción:

Estoy probando la creación de tarea básica

Acción

Iniciar un programa

Finalizar

Desencadenador:

Diariamente; A las 21:58 todos los días

Acción:

Iniciar un programa; C:\Windows\System32\calc.exe

Abrir el diálogo Propiedades para esta tarea al hacer clic en Finalizar

Al hacer clic en Finalizar, la nueva tarea se creará y se agregará a su programación de Windows.

< Atrás

Finalizar

Cancelar

Accediendo a la tarea creada podemos modificarla y añadir acciones o condiciones. También es posible consultar las ejecuciones (o intentos fallidos) de cada tarea. Para ello, primero debemos habilitar la opción de historial (en Windows 11).



Las opciones que se encuentran en las pestañas **General**, **Desencadenadores**, **Acciones**, **Condiciones** y **Configuración** se pueden configurar directamente si creamos una tarea desde la opción **Crear Tarea** (no utilizando "Crear tarea básica").

Nivel	Fecha y...	Id. de...	Categoría de la...	Código operativo	Id. de correlo...
Inf...	16/05/2...	102	Task completed	(2)	fcf5e82d-32...
Inf...	16/05/2...	201	Action complet...	(2)	fcf5e82d-32...
Inf...	16/05/2...	110	Task triggered ...	Información	fcf5e82d-32...

- En la pestaña **General**, podemos modificar la descripción de la tarea y establecer las opciones de seguridad adecuadas.
- En la pestaña **Desencadenadores**, podemos ver y modificar los datos relativos a los criterios que deben cumplirse para iniciar la ejecución de la tarea.

Desencadenador	Detalles	Estado
Diariamente	A las 17:03 todos los días	Habilitado

- La siguiente pestaña contiene las **acciones** asociadas a la tarea programada. Si hubiese más de una, se ejecutarían en orden secuencial.

Acción	Detalles
Iniciar un programa	c:\windows\system32\calc.exe

- En la pestaña **Condiciones**, se indican los criterios que deben cumplirse para que el desencadenador se active (por ejemplo, que el equipo esté conectado a corriente alterna).

General Desencadenadores Acciones Condiciones Configuración Historial

Especifique las condiciones que, junto con los desencadenadores, determinarán si se debe ejecutar la tarea. Si alguna de las condiciones especificadas no se cumple, no se ejecutará la tarea. Para cambiar estas condiciones, abra las páginas de propiedades de la tarea con el comando Propiedades.

Inactivo

Iniciar la tarea solo si el equipo está inactivo durante: 10 minutos  
 Esperar a que esté inactivo durante: 1 hora  
 Detener si el equipo deja de estar inactivo  
 Reiniciar si el estado de inactividad se reanuda

Energía

Iniciar la tarea solo si el equipo está conectado a la corriente alterna  
 Detener si el equipo empieza a usar la batería  
 Activar el equipo para ejecutar esta tarea

Red

Iniciar solo si la siguiente conexión de red está disponible:  
 Cualquier conexión

- En la pestaña **Configuración**, se indican algunas cuestiones complementarias sobre cómo debe comportarse el sistema en relación con la tarea (por ejemplo, que la tarea se detenga si se ejecuta durante más de 3 días).

General Desencadenadores Acciones Condiciones Configuración Historial

Especifique otros valores que afectan a la tarea. Para cambiar la configuración, abra las páginas de propiedades con el comando Propiedades.

Permitir que la tarea se ejecute a petición

Ejecutar tarea lo antes posible si no hubo inicio programado

Si la tarea no se ejecuta, reiniciarla cada: 1 minuto  
Intentar el reinicio un máximo de: 3 veces

Detener la tarea si se ejecuta durante más de: 3 días

Detener tarea en ejecución si no finaliza cuando se solicite

Eliminar tareas no reprogramadas después de: 30 días

Aplicar la siguiente regla si la tarea ya está en ejecución:

No iniciar una instancia nueva

# Automatización Linux

## 1. Crea un script que abra la carpeta y hazlo ejecutable:

- Puedes usar botón derecho sobre el escritorio y ejecutar "abrir en un terminal"

```
mkdir ~/Desktop/TAREAS_DEL_DÍA  
nano ~/Desktop/TAREAS_DEL_DÍA/abrir_tareas.sh
```

- y escribe el siguiente script:

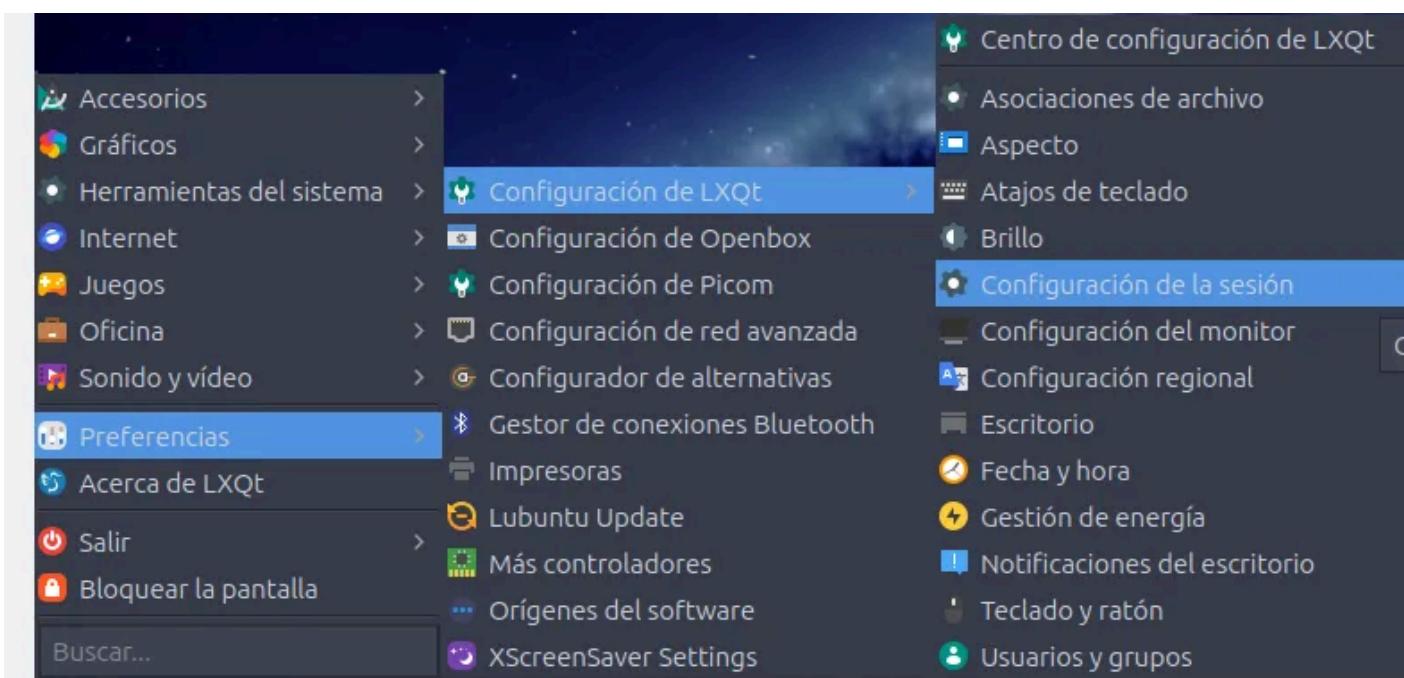
```
#!/bin/bash  
pcmanfm-qt ~/Desktop/TAREAS_DEL_DÍA  
#pcmanfm-qt es el comando para abrir el explorador de archivos predeterminado de lubuntu.  
Este puede variar según distribución.
```

- Guarda el archivo y hazlo ejecutable:

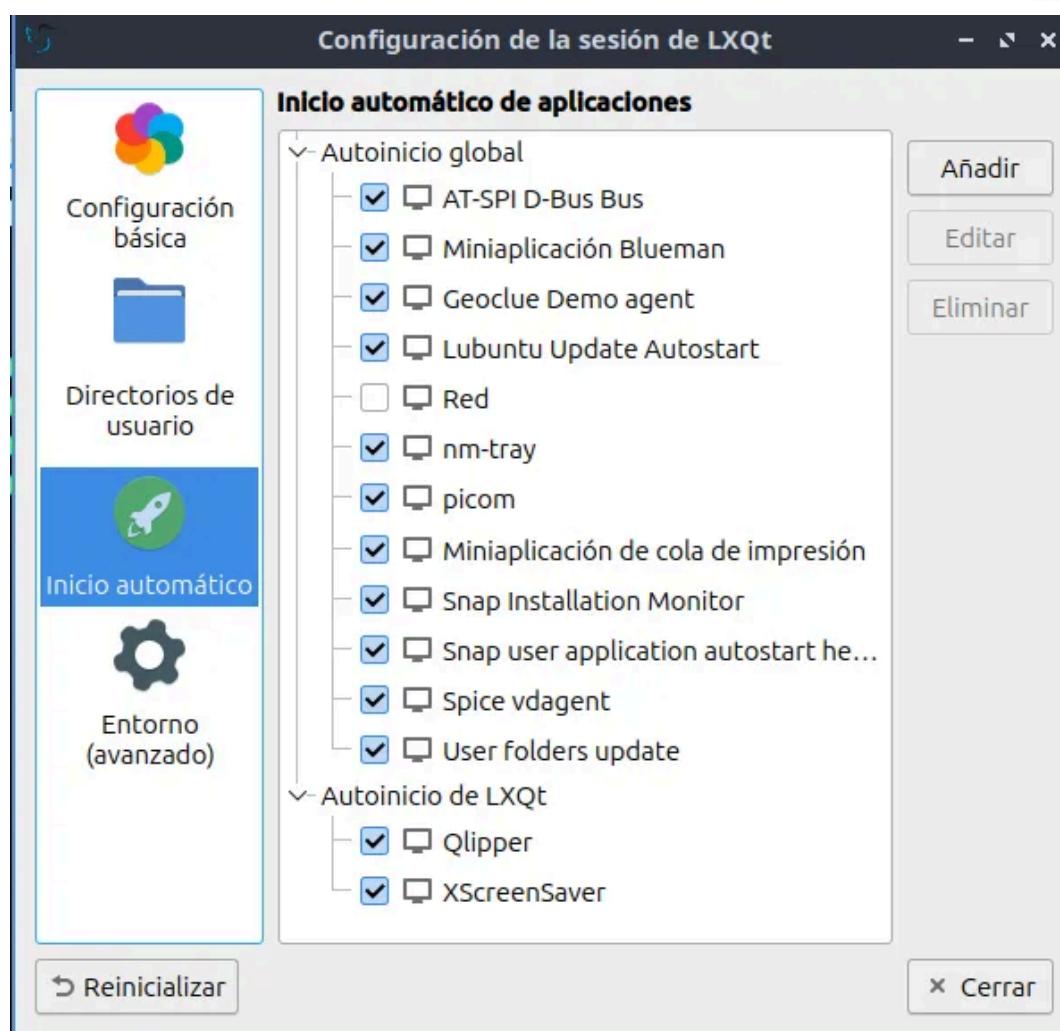
- Para salir en nano: **ctrl+x**
- Para confirmar en nano: **'s** o **S`**
- Para hacer ejecutable: **chmod +x ~/Desktop/TAREAS\_DEL\_DÍA/abrir\_tareas.sh**

## 2. Agrega el script a las aplicaciones de inicio:

- En el menú de aplicaciones de tu sistema, busca "Aplicaciones al inicio" o "Aplicaciones de inicio".



- Haz clic en **Añadir**.



- En **Nombre**, escribe "Abrir Tareas del Día".
- En **Comando**, selecciona el script (/ruta/donde/guardaste/abrir\_tareas.sh).
- Haz clic en **Añadir** o **Guardar**.

# Automatización con n8n

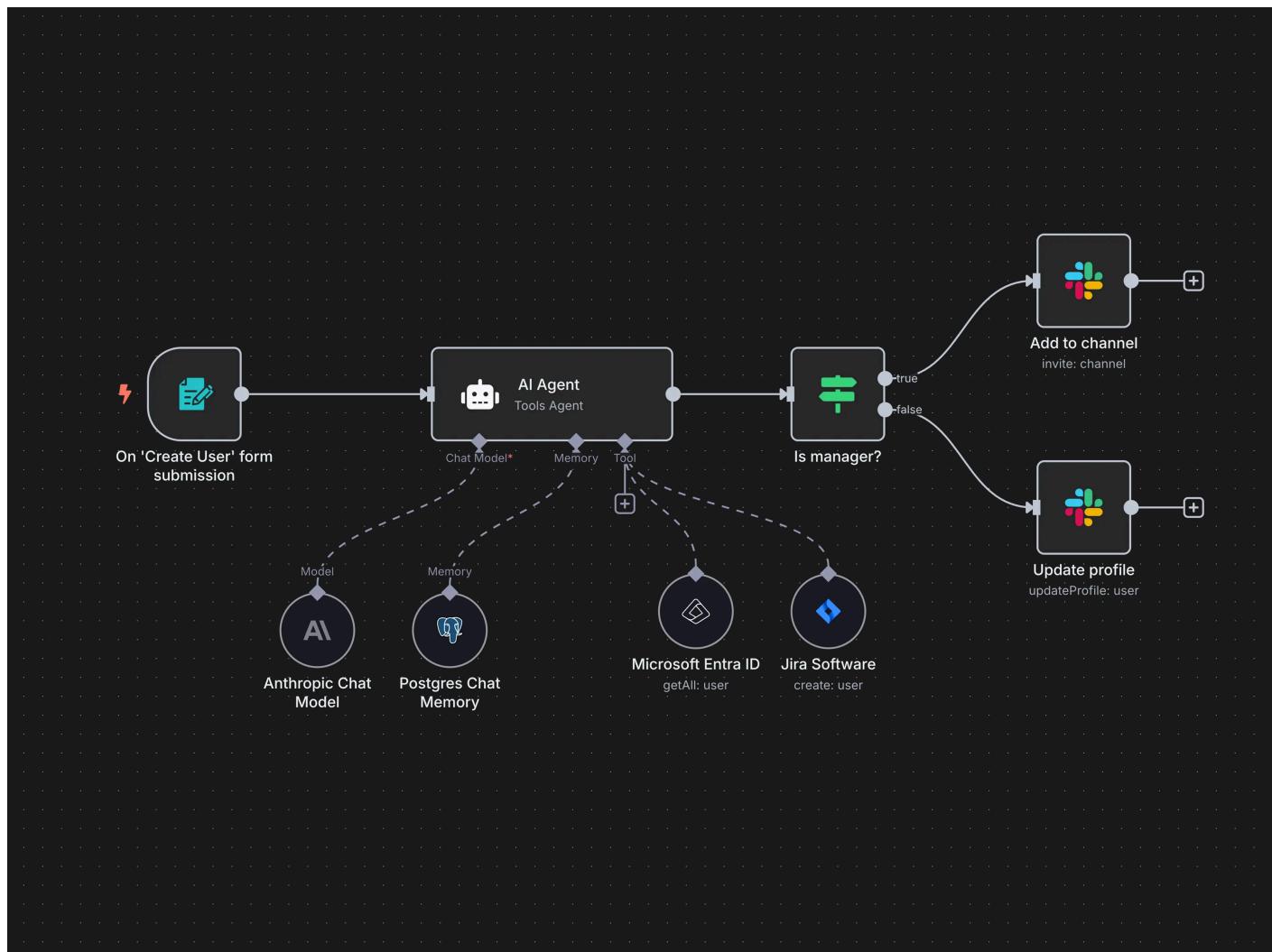
## ¿Qué es n8n?



- Plataforma de automatización basada en nodos.
- Open Source y autohospedable.
- Similar a Zapier o Make, pero libre y local.

## ¿Cómo funciona?

- Trabaja con **nodos conectados**.
- Ejecuta flujos de trabajo (*workflows*).
- Permite integrar APIs, scripts y servicios.



# Instalación Paso a Paso (Docker Compose)

## 1. Requisitos Previos

Asegúrate de tener instalados los siguientes programas en tu sistema:

- **Docker:** El motor de contenedores.

```
// comprobamos la versión
docker --version

// en Windows, si no está instalado, usar dentro de PowerShell:
winget install docker

// actualizar a la última versión
winget upgrade docker
```

- **Docker Compose:** Para definir y ejecutar aplicaciones Docker de múltiples contenedores (simplifica mucho el proceso).



MINISTERIO  
DE EDUCACIÓN  
Y FORMACIÓN PROFESIONAL



Plan de Recuperación,  
Transformación  
y Resiliencia



## 2. Crear el Archivo de Configuración (`docker-compose.yml`)

Crea un nuevo directorio para tu proyecto de n8n (ej. `n8n-project`) y dentro de él, crea un archivo llamado [`docker-compose.yml`](#).

## 3. Iniciar n8n

Abre tu terminal, navega hasta el directorio donde guardaste el archivo `docker-compose.yml` y ejecuta el siguiente comando:

```
docker-compose up -d
```

- `up`: Crea e inicia los contenedores.
- `-d`: Ejecuta los contenedores en modo *detached* (en segundo plano).

## 4. Verificar y Acceder

### 1. Verifica que los contenedores estén corriendo:

```
docker-compose ps
```

Deberías ver algo como Up en la columna de State para los servicios n8n y postgres.

### 2. Accede a n8n: Abre tu navegador web y navega a: <http://localhost:5678>

Se te pedirá el usuario y la contraseña que configuraste en `N8N_BASIC_AUTH_USER` y `N8N_BASIC_AUTH_PASSWORD`.

## 5. Comandos Útiles

- **Para detener los contenedores (sin eliminar los datos):**

```
docker-compose stop
```

- **Para detener y eliminar los contenedores (sin eliminar los datos persistentes):**

```
docker-compose down
```

- **Para ver los logs de los contenedores (útil para solucionar problemas):**



MINISTERIO  
DE EDUCACIÓN  
Y FORMACIÓN PROFESIONAL



Plan de Recuperación,  
Transformación  
y Resiliencia



GVA NEXT  
Fondos Next Generation  
en la Comunitat Valenciana



```
docker-compose logs -f
```

- **Ejecutar n8nn en Docker:**

```
docker run -it -p 5678:5678 n8nio/n8n
```

Más info: <https://docs.n8n.io/hosting/installation/docker/>

## Primeros Pasos en n8n

### Crear tu primer Workflow

[Primer Workflow en n8n](#)

### Persistencia de datos (opcional)

Por defecto, n8n no guarda los workflows si lo cierras. Para guardarlos:

1. Crea una carpeta en tu equipo, por ejemplo: C:\n8ndata o /home/tusuuario/n8ndata
2. Inicia n8n con esta variable de entorno:

```
N8N_STORAGE_BASE_PATH= ./n8n-data n8n start
```

## Recursos Oficiales

- Documentación oficial: <https://docs.n8n.io>
- Comunidad y foro: <https://community.n8n.io>
- Plantillas predefinidas: <https://n8n.io/workflows>
- [Cómo conectar cualquier API con n8n](#)
- [Nodos más usados en n8n - noobs](#)
- [Ejemplos n8n](#)

# Command Line Interfaces

Las Interfaces de Línea de Comandos (CLI), también conocidas como interfaces basadas en texto o consolas, permiten a los usuarios interactuar con un sistema informático a través de comandos de texto. Estos comandos se introducen típicamente a través de una terminal o un símbolo del sistema.

Las interfaces CLI pueden parecer arcaicas en la era de las elegantes interfaces gráficas de usuario (GUI) y las emergentes tecnologías de interfaces naturales de usuario (NUI), pero siguen siendo herramientas indispensables para los usuarios avanzados y profesionales que buscan un control y una eficiencia incomparables.



Las interfaces CLI siguen siendo una herramienta fundamental en la informática, especialmente para tareas que requieren un control preciso y automatización.



## Pros y Contras de CLI

- **Pros:**

- Bueno para hacer cosas repetitivas y automatizarlas.
- No consume mucha potencia de la computadora.

- Ofrece un control exacto sobre cómo se configura el sistema.
- Necesario para servidores donde el acceso remoto es común y usar una GUI podría no funcionar bien.

● **Contras:**

- Más difícil de aprender para los nuevos usuarios.
- Hay que escribir comandos, lo que puede parecer más difícil que hacer clic en cosas.
- No muestra tanta información en la pantalla como las GUI.
- Los comandos pueden ser complejos a veces, lo que puede llevar a errores si no se escriben correctamente.

 Tip

[Manz.dev desarrollador que destaca el terminal](#)

# Terminal Windows

## Introducción al terminal de PowerShell

### Cómo abrir el terminal de PowerShell

1. Desde el Menú de Inicio
2. Desde el Explorador de Archivos
3. Con acceso rápido (Presiona **Win + X**)

### Cómo actualizar PowerShell a la última versión

```
winget install --id Microsoft.PowerShell --source winget
```

### Verificar la Versión de PowerShell

```
$PSVersionTable.PSVersion
```

### Estructura y uso de cmdlets

Los comandos de PowerShell, conocidos como **cmdlets**, tienen una estructura específica que facilita su uso y comprensión. Aquí te explico cómo se estructuran:

1. **Verbo-Sustantivo**: Cada cmdlet sigue la convención de nombre Verbo-Sustantivo. Por ejemplo, Get-Process (obtener proceso) o Stop-Service (detener servicio). Esto ayuda a entender rápidamente qué hace el cmdlet.
2. **Parámetros**: Los cmdlets pueden aceptar parámetros que modifican su comportamiento. Los parámetros se añaden después del cmdlet, separados por espacios. Por ejemplo, Get-Process -Name "notepad" obtiene información sobre el proceso de Notepad.
3. **Pipelines**: PowerShell permite encadenar cmdlets usando el operador | (pipe). Esto pasa la salida de un cmdlet como entrada a otro. Por ejemplo, Get-Process | Where-Object {\$\_\_.CPU -gt 100} filtra los procesos que usan más de 100 unidades de CPU.
4. **Alias**: PowerShell también soporta alias, que son nombres cortos para cmdlets. Por ejemplo, ls es un alias para Get-ChildItem.

5. **Ayuda:** Puedes obtener ayuda sobre cualquier cmdlet usando Get-Help. Por ejemplo, Get-Help Get-Process muestra información detallada sobre el cmdlet Get-Process.

```
Get-Process -Name "notepad" | Stop-Process  
#Este comando obtiene el proceso de Notepad y luego lo detiene.
```

## Tabla de Comandos de Consola

### 🧭 Navegación y Exploración de Directorios

Tarea	CMD (MS-DOS)	PowerShell	Alias de PowerShell
<b>Mostrar directorio actual</b>	cd	Get-Location	gl, pwd
Ejemplo	cd	Get-Location	pwd
<b>Cambiar directorio</b>	cd [ruta]	Set-Location [ruta]	sl, cd
Ejemplo	cd C:\Users\Admin	Set-Location C:\Datos	cd \
<b>Listar contenido del directorio</b>	dir	Get-ChildItem	gci, ls, dir
Ejemplo	dir /p (página a página)	Get-ChildItem -Force (incluir ocultos)	ls

### 📁 Gestión de Directorios

Tarea	CMD (MS-DOS)	PowerShell	Alias de PowerShell
<b>Crear nuevo directorio</b>	mkdir [nombre]	New-Item -ItemType Directory -Name [nombre]	md, ni
Ejemplo	mkdir ProyectosFP	New-Item -ItemType Directory -Name 'Backup Mayo'	md Clientes

Tarea	CMD (MS-DOS)	PowerShell	Alias de PowerShell
<b>Eliminar directorio (vacío)</b>	<code>rmdir [nombre]</code>	<code>Remove-Item [nombre]</code>	<code>ri, rd</code>
Ejemplo	<code>rmdir Temp</code>	<code>Remove-Item Temp</code>	<code>rd Fotos</code>
<b>Eliminar directorio (con contenido)</b>	<code>rmdir /s [nombre]</code>	<code>Remove-Item -Recurse [nombre]</code>	<code>ri</code>
Ejemplo	<code>rmdir /s Logs</code>	<code>Remove-Item -Recurse Logs</code>	<code>ri -Recurse Logs</code>

## Gestión de Archivos

Tarea	CMD (MS-DOS)	PowerShell	Alias de PowerShell
<b>Crear archivo vacío</b>	<code>type nul &gt; [nombre.ext]</code>	<code>New-Item -ItemType File - Name [nombre.ext]</code>	<code>ni</code>
Ejemplo	<code>type nul &gt; readme.txt</code>	<code>New-Item -ItemType File - Name 'config.ini'</code>	<code>ni test.log</code>
<b>Copiar archivo</b>	<code>copy [origen] [destino]</code>	<code>Copy-Item [origen] [destino]</code>	<code>cpi, cp</code>
Ejemplo	<code>copy informe.pdf D:\</code>	<code>Copy-Item informe.pdf D:\</code>	<code>cp hosts c:\temp</code>
<b>Mover/Renombrar archivo</b>	<code>move [origen] [destino]</code>	<code>Move-Item [origen] [destino]</code>	<code>mi, mv</code>
Ejemplo	<code>move doc.txt informe.txt</code>	<code>Move-Item doc.txt informe.txt</code>	<code>mv doc.txt informe.txt</code>
<b>Eliminar archivo</b>	<code>del [nombre.ext]</code>	<code>Remove-Item [nombre.ext]</code>	<code>ri, del</code>
Ejemplo	<code>del temporal.tmp</code>	<code>Remove-Item temporal.tmp</code>	<code>del *.bak</code>

Tarea	CMD (MS-DOS)	PowerShell	Alias de PowerShell
Mostrar contenido de un archivo	<code>type [nombre.ext]</code>	<code>Get-Content [nombre.ext]</code>	<code>gc, cat</code>
Ejemplo	<code>type error.log</code>	<code>Get-Content error.log -Tail 5</code>	<code>cat error.log</code>

## 🔍 Búsqueda de Archivos y Texto

Tarea	CMD (MS-DOS)	PowerShell	Alias de PowerShell
Buscar texto en archivo	<code>find "[texto]" [archivo]</code>	<code>Select-String -Pattern "[texto]" -Path [archivo]</code>	<code>sls</code>
Ejemplo	<code>find "error" log.txt</code>	<code>Select-String -Pattern "error" -Path log.txt</code>	<code>sls "red" *.log</code>
Buscar Archivo por nombre	<i>No aplica un comando simple</i>	<code>Get-ChildItem -Filter "[patrón]"</code>	<code>gci -fi "[patrón]"</code>
Ejemplo	N/A	<code>Get-ChildItem -Filter *.pdf</code>	<code>gci -fi *.pdf</code>

¡Absolutamente! El comando **attrib** es clave para entender las propiedades básicas de los archivos en CMD, algo que deben dominar.

Aquí tienes la tabla ajustada para enfocarse en **attrib** y una breve explicación de los valores que maneja, que son los atributos básicos del sistema de archivos.

## 🔒 Atributos y Propiedades de Archivos (Windows)

Tarea	CMD (MS-DOS)	PowerShell	Alias de PowerShell
Mostrar/Modificar Atributos	<code>attrib [+/-] [R H]</code>	<code>Get-ItemProperty</code>	<code>gip</code>
Ejemplo Mostrar	<code>attrib informe.pdf</code>	<code>(Get-Item informe.pdf).Attributes</code>	<code>gip informe.pdf</code>

Tarea	CMD (MS-DOS)	PowerShell	Alias de PowerShell
Ejemplo Modificar	attrib +R +H informe.pdf	(Get-Item informe.pdf).IsReadOnly = \$true	(gip informe.pdf).IsReadOnly = \$true

- **Sintaxis:** Se usa el signo + para establecer el atributo y - para quitarlo.
- **Ejemplo:** Para hacer un archivo oculto y de solo lectura: attrib +H +R archivo.txt

# Terminal Linux

## Conceptos básicos

```
# Comprobar si WSL está instalado:  
wsl --status  
# Ejecutar WSL:  
wsl o simplemente buscarlo en el menú inicio como "WSL" (ícono pinguino)  
# Listar las distribuciones disponibles:  
wsl --list --online  
# Instalar WSL en Windows 10/11, abre PowerShell como administrador y ejecuta:  
wsl --install Ubuntu
```

### WSL - Carpetas personales

- **/home/USUARIO-LINUX/** # Carpeta personal en Linux
- **/mnt/c/Users/USUARIO-HOST/** # Carpeta personal en HOST (Windows)

### WS - Establecer Ubuntu por defecto

En caso de que WSL se cierre inmediatamente al iniciar, se debe establecer Ubuntu por defecto

- Establece Ubuntu por defecto: `wsl --set-default Ubuntu`

## Prompt de Linux

- El **prompt** muestra información útil como el usuario, la máquina y el directorio actual.
  - Ejemplo: `beatriz@ubuntu:~$`
  - `$`: Usuario sin privilegios.
  - `#`: Usuario con privilegios root.

## Comandos y Entrada/Salida Estándar

1. Los comandos en Linux son **palabras reservadas** utilizadas por el sistema operativo para ejecutar programas a través de una terminal o línea de comandos.

- **Sintaxis general:**

```
orden [-opciones] argumento1 argumento2
```

- **Agrupación de opciones:**

```
ls -l -a = ls -la
```

```
profesor@MSI:~$ ls -la
total 52
drwxr-x--- 6 profesor profesor 4096 Dec 19 09:52 .
drwxr-xr-x 3 root      root    4096 Nov  7 08:22 ..
-rw----- 1 profesor profesor  250 Dec 19 08:42 .bash_history
-rw-r--r-- 1 profesor profesor 220 Nov  7 08:22 .bash_logout
-rw-r--r-- 1 profesor profesor 3771 Nov  7 08:22 .bashrc
drwx----- 2 profesor profesor 4096 Nov  7 08:22 .cache
drwxr-xr-x 2 profesor profesor 4096 Dec 16 13:24 .landscape
-rw----- 1 profesor profesor   20 Dec 19 09:45 .lesshist
drwxr-xr-x 3 profesor profesor 4096 Dec 16 13:25 .local
-rw-r--r-- 1 profesor profesor    0 Dec 19 08:29 .motd_shown
-rw-r--r-- 1 profesor profesor  807 Nov  7 08:22 .profile
lrwxrwxrwx 1 profesor profesor   13 Dec 19 09:52 t -> test_file.txt
-rw-r--r-- 2 profesor profesor    5 Dec 19 09:51 t2
-rw-r--r-- 2 profesor profesor    5 Dec 19 09:51 test_file.txt
drwxr-xr-x 2 profesor profesor 4096 Dec 16 13:29 tests
```



1. **Diferenciación entre mayúsculas y minúsculas:** En Linux, las mayúsculas y minúsculas son diferentes, por lo que debes prestar atención al escribir los comandos.

## Entrada y Salida Estándar en Linux

Todo programa en Linux trabaja con tres tipos de flujos principales:

- **Entrada estándar** (stdin): Teclado.
- **Salida estándar** (stdout): Pantalla.
- **Salida de error** (stderr): Pantalla.

## Redirección, contatenación y ayuda

- > y >>: Redirigen la salida a un archivo.
  - Ejemplo: ls -a > fichero.txt
- <: Redirige la entrada desde un archivo.
  - Ejemplo: wc < fichero.txt

- **Tuberías:** Permiten que la salida de un comando se convierta en la entrada de otro.

- comando1 | comando2 | comando3
- Ejemplo: cat /etc/passwd | wc -l

- **Concatenación:** Ejecuta comandos de forma secuencial sin relación:

- comando1 && comando2 && comandoN
- Ejemplo: cat /etc/passwd && cat /etc/groups

### Ayuda sobre Comandos

- **man:** Muestra una página de ayuda completa.
  - man pwd
- **help:** Proporciona ayuda rápida.
  - pwd --help

# Comandos

## Comandos Básicos

Comando	Descripción	Ejemplo
<b>pwd</b>	muestra la ruta actual.	pwd
<b>cd</b>	cambia de directorio.	cd /home/usuario
<b>ls</b>	lista archivos.	ls -l
<b>mkdir</b>	crea directorios.	mkdir nueva_carpa
<b>rm</b>	borra ficheros o directorios.	rm archivo.txt
<b>mv</b>	mueve ficheros o directorios.	mv archivo.txt carpeta/
<b>cp</b>	copia ficheros o directorios.	cp archivo.txt copia.txt
<b>uname</b>	muestra información del sistema.	uname -a
<b>shutdown</b>	apaga o reinicia el equipo.	shutdown now o shutdown -r
<b>reboot</b>	reinicia el equipo.	reboot

Comando	Descripción	Ejemplo
<b>clear</b>	limpia la pantalla del terminal.	clear
<b>date</b>	muestra la fecha y hora del sistema.	date
<b>cal</b>	muestra el calendario del mes o año indicado.	cal 2024
<b>who</b>	muestra quién está conectado al sistema.	who
<b>type</b>	Para saber si un comando existe.	type ls
<b>touch</b>	Para crear un archivo vacío o actualizar la marca de tiempo de un archivo.	touch archivo.txt

# Terminal Linux avanzado

## Wildcards (Comodines)

Las **wildcards o comodines** son caracteres especiales. En una baraja de cartas, un comodín es una carta especial que puede representar a cualquier otra carta. De manera similar, en informática, una wildcard es un carácter especial que puede sustituir a cualquier otro carácter.

### 1. Múltiple: creación o manipulación de varios ficheros/directorios

```
touch file{1..100}.txt # Crea ficheros file1.txt hasta file100.txt
cp {hola.txt, adios.txt} ~ # Copia hola.txt y adios.txt al directorio del usuario
mkdir -p Nivel1/Nivel2/{Nivel3/{Nivel4/Nivel5,logs,scripts},docs,data} # Crea la estructura de
directorios compleja
- Para esto hay que instalar bash si no se tiene:
- sudo apt install bash-completion
- echo $SHELL #comprobamos el shell actual
- chsh -s /bin/bash #cambiamos el shell a bash
```

```
Nivel1/
└── Nivel2/
    ├── Nivel3/
    │   ├── Nivel4/
    │   │   └── Nivel5/
    │   ├── logs/
    │   └── scripts/
    ├── docs/
    └── data/
```

### 2. Asterisco (\*): Representa cero o más caracteres. Es útil para buscar archivos que coincidan con un patrón específico.

```
ls *.txt # Lista todos los archivos que terminan en .txt
```

### 3. Signo de interrogación (?): Representa un solo carácter. Es útil cuando se busca un archivo con un nombre específico pero con un carácter variable.

```
ls file?.txt # Lista archivos como file1.txt, file2.txt, etc.
```

MINISTERIO  
DE EDUCACIÓN  
Y FORMACIÓN PROFESIONALPlan de Recuperación,  
Transformación  
y ResilienciaGENERALITAT  
VALENCIANA  
Conselleria d'Educació,  
Universitats i EsportsFP CV  
Formación Profesional  
Conselleria d'EducacióFondos Next Generation  
en la Comunitat Valenciana

4. **Corchetes ([ ])**: Representan un rango de caracteres. Se utiliza para buscar archivos que contengan cualquiera de los caracteres especificados dentro de los corchetes.

```
ls file[1-3].txt # Lista archivos como file1.txt, file2.txt, file3.txt
```

5. **Negación (! [ ])**: Representa cualquier carácter que no esté dentro de los corchetes.

```
ls file[!1-3].txt # Lista archivos que no sean file1.txt, file2.txt, file3.txt
```

6. **Rangos de caracteres**: Puedes especificar rangos de caracteres dentro de los corchetes.

```
ls file[a-c].txt # Lista archivos como filea.txt, fileb.txt, filec.txt
```

7. **Combinación**: combina diversas opciones

```
ls file[a-c,A-C,1-3]?.txt # Lista ficheros como filea1.txt, fileBc.txt, filecc.txt
```

## Comandos Avanzados

Comando	Descripción	Ejemplo
<b>grep</b>	busca patrones en archivos.	grep root /etc/passwd
<b>sort</b>	ordena las líneas de un archivo.	sort -f /etc/passwd
<b>wc</b>	cuenta líneas, palabras o caracteres.	wc /etc/passwd
<b>cut</b>	extrae secciones específicas.	cut -d':' -f1 /etc/passwd
<b>uniq</b>	elimina o muestra líneas duplicadas.	sort /etc/passwd   uniq
<b>find</b>	busca archivos y directorios.	find /home -name "*.txt"
<b>xargs</b>	construye y ejecuta comandos.	find . -name "*.txt"   xargs rm
<b>head</b>	muestra las primeras líneas.	head -n 10 /etc/passwd
<b>tail</b>	muestra las últimas líneas.	tail -f /var/log/syslog
<b>tee</b>	divide la salida y la guarda en un archivo.	ls   tee lista.txt

MINISTERIO  
DE EDUCACIÓN  
Y FORMACIÓN PROFESIONALPlan de Recuperación,  
Transformación  
y Resiliencia

Comando	Descripción	Ejemplo
tr	traduce o elimina caracteres.	echo "linux"   tr 'a-z' 'A-Z'
sed	edita texto en línea.	sed 's/root/admin/g' /etc/passwd
tar	comprime o descomprime archivos.	tar -czvf archivo.tar.gz carpeta/

**TAR**tar

## Comandos de Gestión

Comando	Descripción	Ejemplo
chmod	cambia los permisos de archivos.	chmod 755 script.sh
chown	cambia el propietario de un archivo.	chown user:group archivo.txt
scp	copia archivos entre máquinas.	scp archivo.txt usuario@192.168.1.100:/home/usuario/
rsync	sincroniza archivos y directorios.	rsync -av carpeta/ usuario@192.168.1.100:/destino/
df	muestra el espacio en disco.	df -h
du	calcula el tamaño de archivos.	du -sh /home/usuario/
ps	lista los procesos activos.	ps aux   grep apache
top	muestra procesos en tiempo real.	top
kill	termina procesos usando su PID.	kill 1234
fsck	comprueba la integridad de los discos.	fsck /dev/sda1
mount	monta particiones y volúmenes.	mount /dev/sda1 /mnt

Comando	Descripción	Ejemplo
umount	desmonta particiones y volúmenes.	umount /mnt
fdisk	administra particiones del disco.	fdisk /dev/sda
ln	crea enlaces físicos y simbólicos.	ln -s archivo_original enlace

### Prueba

1. ¿Qué hace el comando uname -a?

2. Prueba los siguientes comandos:

- cat /etc/passwd | wc -l
- ls -l | sort -k 5n

[Cheatshet - Terminal](#)

### Actividad

- Estructura de directorios en Linux
- Speed command dating
- Linux Advanced Commands