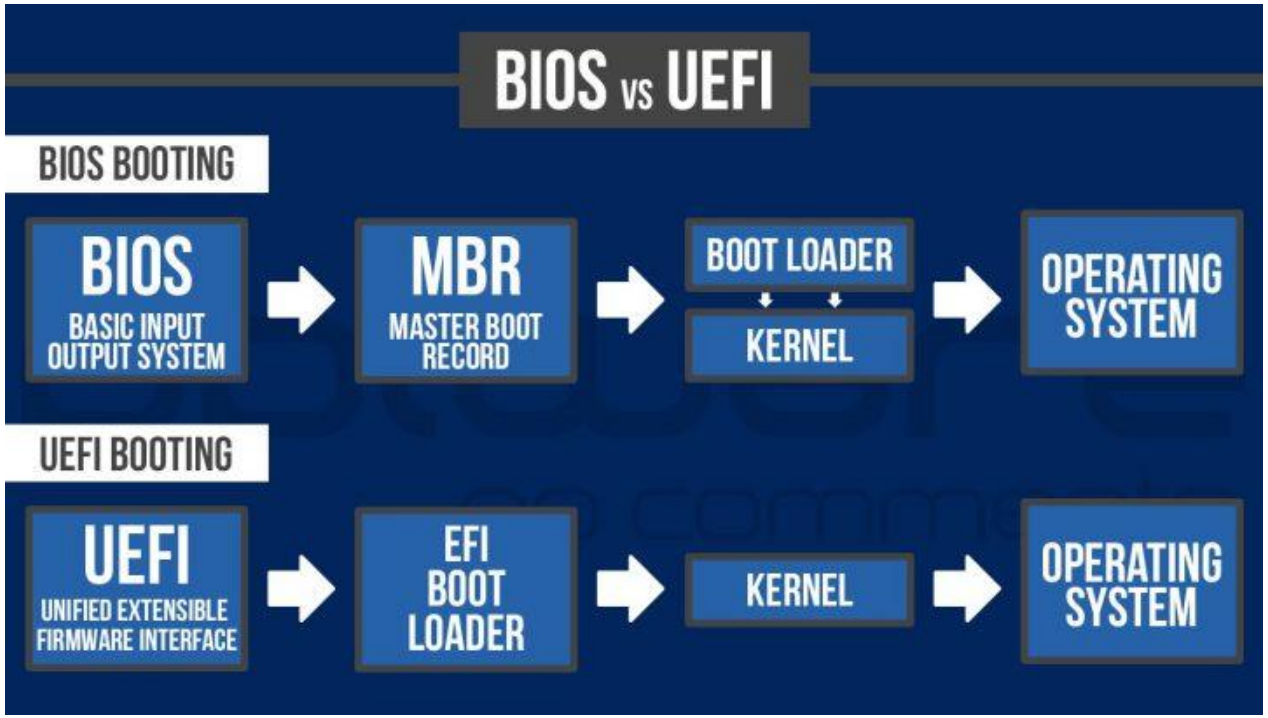
I.E.S Alonso de Ercilla
CFGS Desarrollo de Aplicaciones Multiplataforma

SISTEMAS INFORMÁTICOS

UT2
Introducción a los sistemas operativos

PARTICIONES Y SISTEMA DE ARCHIVOS



Esquema de particiones

En un equipo en reposo (apagado), los archivos están almacenados en la memoria secundaria, es decir, en los dispositivos de almacenamiento ya sean HDD o SSD, conectados con una interfaz SATA, Slot PCIe, M.2, etc... Antes de instalar un sistema operativo, es necesario conocer ciertas características.

Los dispositivos de almacenamiento disponen de **particiones** para organizar la información, datos e instalar sistemas operativos.

Podemos particionar el almacenamiento con herramientas propias dentro del sistema operativo instalado, o herramientas específicas externas, por ejemplo: GParted, WinPE Strelec (conjunto de programas: EaseUS Partition Master, AOMEI Partition Assistant, DiskGenius, etc…).

Antes de crear particiones es necesario definir una tabla de particiones. En la actualidad, los esquemas más utilizados son **MBR (Master Boot Record) y GPT (GUID Partition Table).**

Para almacenar información en una partición se necesita un **sistema de archivos.** Cada partición únicamente puede tener un sistema de archivos.

El proceso por el cual se asigna un **sistema de archivos** a una partición se conoce como **formato.**

MBR *(Master Boot Record)*

También conocido como registro de arranque maestro o sector de arranque, es la primera parte del dispositivo de almacenamiento que es leída por el sistema (BIOS/UEFI) al iniciarse el equipo.

Es el esquema de particionamiento de los sistemas con firmware BIOS. Es más antiguo, pero aún se sigue empleando por su compatibilidad con sistemas operativos. Este esquema indica como están organizadas las particiones, y que sistema de archivos (formato), tiene cada una.

Existe una serie de limitaciones:

- **Sólo funciona en unidades de hasta 2TB**
- **Sólo admite 4 particiones primarias, siendo necesario crear una partición extendida para crear particiones lógicas dentro de ella.**

GPT *(GUID Partition Table)*

Es el **esquema de particionado** utilizado por los sistemas con **firmware UEFI**, que sustituye al antiguo BIOS. Cada partición posee un identificador global único (GUID), lo que permite una gestión más flexible y segura del almacenamiento.

Este estándar supera las limitaciones del MBR y ofrece numerosas ventajas:

- Permite hasta **128 particiones** sin necesidad de particiones extendidas.
- Soporta discos de más de **2 TB** de capacidad.
- **Almacena copias de respaldo** de la tabla de particiones para evitar pérdida de datos por corrupción.
- Facilita **arranques más rápidos y fiables** cuando se utiliza junto con UEFI.

Estructura de la tabla GPT

Los dispositivos con formato GPT deben incluir una partición de sistema EFI (ESP, **EFI System Partition**). Esta partición ha de estar formateada en **FAT32** y normalmente suele estar ubicada al inicio del disco. La partición ESP almacena los cargadores de arranque de los sistemas operativos instalados, así como otros archivos necesarios para el proceso de arranque en sistemas UEFI.

Un dispositivo con esquema GPT dispone lo que se conoce como “Protective MBR”, que actúa como una especie de "cubierta protectora" para evitar que programas de gestión de particiones antiguos que no reconocen GPT puedan dañar los datos.

Tipo de particiones

Particiones: Divisiones del disco. Sólo puede haber 4 en el caso de MBR, y 128 en GPT, y sólo una de ellas puede estar activa. El sistema operativo que arranca el equipo debe estar ubicado en la partición activa.

Para que un dispositivo de almacenamiento sea utilizable, debe tener una tabla de particiones y al menos una partición formateada con un sistema de archivos.

Partición extendida (MBR): sólo puede haber una partición de este tipo por disco, y sirve para contener particiones lógicas. Fue diseñada para solventar la limitación de MBR, que solo permite un máximo de cuatro particiones primarias.

Partición lógica (MBR): ocupa una parte de la partición extendida o la totalidad de la misma.

En Windows, las particiones primarias y las particiones lógicas (caso MBR) tienen asociada una letra de unidad para poder ser identificadas (C:\, D:\, E:\,…)

Los dispositivos de almacenamiento en Linux son detectados dentro del sistema operativo por identificadores del tipo sda, sdb, sdc, etc, donde la letra correspondería a un dispositivo (a,b,c,…). Las particiones se identifican con números. Si por ejemplo el dispositivo a tiene 3 particiones: sda1, sda2, sda3

Formatos

Cada sistema operativo utiliza determinados sistemas de archivos (formatos), que definen cómo se organizan, acceden y almacenan los datos en una unidad de almacenamiento. No todos los sistemas operativos son compatibles con todos los formatos; por ejemplo, Windows no puede leer nativamente particiones con formato ext4.

FAT 32:

- Tamaño máximo de archivo: 4GB.
- Tamaño máximo de volumen (partición): 8TB.
- Se sigue usando en: Memorias USB, Tarjetas SD, discos duros externos siempre que los archivos no superen los 4GB.
- No recomendable en discos duros internos debido a problemas de compatibilidad con sistemas operativos.

NTFS (New Technology File System):

- Tamaño máximo de archivo: 256 TB.
- Tamaño máximo de volumen (partición): 256 TB.
- No es recomendable para sistemas de almacenamiento de poca capacidad, debido al espacio que necesita la estructura de este formato.
- Es compatible con el sistema operativo **Windows**.

Formatos

ext4 (fourth extended filesystem):

- Tamaño máximo de archivo: 1024 PB.
- Tamaño máximo de volumen (partición): 1EB.
- Menor uso del CPU.
- Mejoras en la velocidad de lectura y escritura.
- Utilizado en sistemas operativos de distribución Linux

exFAT (Extended File Allocation Table):

- Tamaño máximo de archivo: 16EB.
- Tamaño máximo de volumen (partición): 64ZB.
- Mejoras en el rendimiento de la asignación de espacio libre.
- Especialmente adaptado para memorias flash tipo SD y pendrive.

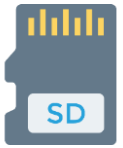
APFS (Apple File System):

- Posibilidad de realizar snapshots (instantáneas) y clonado rápido.
- Tamaño máximo de volumen (partición): 8EB.
- Especialmente diseñado para memorias de estado sólido (SSD) en todos los dispositivos Apple.

Formatos

A continuación, se muestra una tabla de compatibilidad entre formatos y sistemas operativos:

Sistema de Archivos	Windows	macOS	Linux	BSD	Unix
NTFS	Sí	No	Lectura	No	No
FAT32	Sí	Sí	Sí	Sí	Sí
exFAT	Sí	Sí	Sí	No	No
HFS+	No	Sí	Lectura	No	No
APFS	No	Sí	Lectura	No	No
EXT2/3/4	No	No	Sí	Sí	Sí
XFS	No	No	Sí	Sí	Sí
Btrfs	No	No	Sí	Sí	Sí
ZFS	No	No	Sí	Sí	Sí
UFS	No	No	Sí	Sí	Sí

UEFI *(Unified Extensible Firmware Interface)*

UEFI, también conocida como EFI, fue desarrollada por Intel en el año 2002 para solventar las dificultades técnicas de la interfaz BIOS.

UEFI es una interfaz que reemplaza a la antigua interfaz del Sistema Básico de Entrada y Salida (BIOS).

UEFI puede proporcionar menús gráficos adicionales, e incluso proporcionar acceso remoto para la solución de problemas y mantenimiento.

Es compatible con la mayoría de los sistemas operativos, incluyendo el soporte para tabla de particiones GUID (GPT), además de la conocida tabla de particiones MBR.

Gestor de arranque integrado, que permite seleccionar y cargar directamente los sistemas operativos instalados, sin depender de gestores externos.

UEFI *(GUID Partition Table)*

El sistema UEFI actúa como una capa intermedia entre el firmware de bajo nivel (anteriormente representado por el BIOS) y el sistema operativo, lo que le otorga una gran flexibilidad y mayores capacidades de gestión durante el arranque del equipo.

Por otro lado, el término BIOS se utiliza a menudo de forma genérica para referirse al sistema de arranque del firmware, pudiendo englobar tanto a los sistemas con estándar BIOS tradicional como a los basados en UEFI.

Modo de arranque UEFI

Los sistemas UEFI pueden establecer el modo de arranque. En general, se pueden configurar dos modos:

- **Heredado o Legacy BIOS:** establece compatibilidad hacia atrás con discos que utilizan esquemas MBR. En este modo no se aprovechan las ventajas del estándar UEFI. Durante la instalación del sistema operativo, las particiones se crearán con esquema **MBR**.
- **UEFI:** es el modo recomendado por la mayoría de sistemas operativos, dadas sus ventajas. Durante la instalación del sistema operativo en modo UEFI, se crean por defecto esquemas de particiones **GPT**.

Debe existir una **concordancia** entre el modo de arranque del equipo y el esquema de particionamiento del disco de arranque (el primero en el orden de inicio).

Los discos con sistemas operativos instalados bajo esquemas de particionamiento **GPT y MBR** deben iniciarse en modo **UEFI y Heredado**, respectivamente.

De lo contrario, el firmware UEFI o BIOS **no podrá arrancar el sistema operativo**, al estar el disco configurado con un tipo de particionado incompatible.

E.particionado GPT Windows

Cuando se realiza la instalación de Windows en un equipo con arranque UEFI, el esquema predeterminado de particiones es el siguiente:

EFI System Partition (ESP):
- **Tipo: Sistema EFI**
- **Tamaño: Aproximadamente 100 MB**
- **Función: Almacena los archivos de arranque del sistema operativo y otros archivos necesarios para el proceso de arranque en sistemas UEFI.**

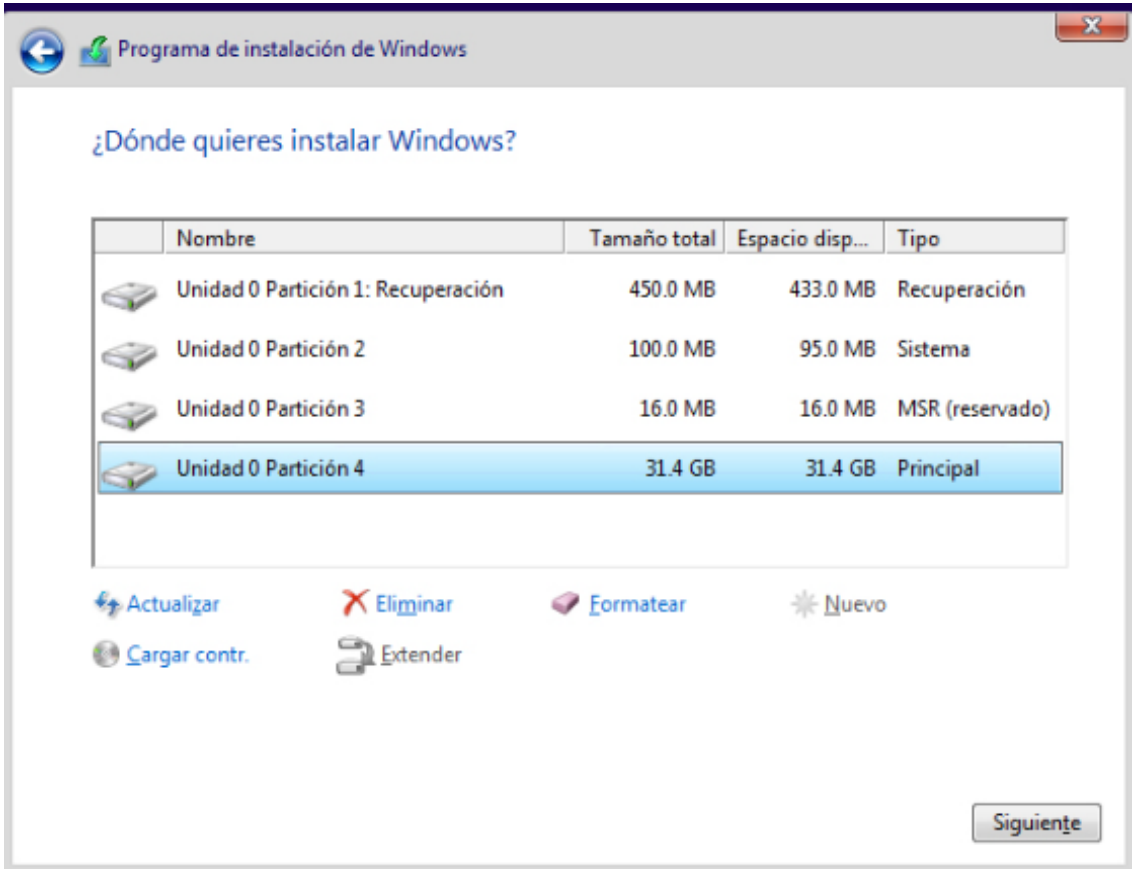
Microsoft Reserved Partition (MSR):
- **Tipo: Partición reservada de Microsoft**
- **Tamaño: 16 MB**
- **Función: Reservada por el sistema para un uso futuro. Generalmente está vacía y no contiene datos.**

Recovery Partition (Recuperación):
- **Tipo: Partición de recuperación**
- **Tamaño: Variable (generalmente varios GB)**
- **Función: Almacena los archivos necesarios para la recuperación del sistema, incluyendo herramientas de solución de problemas y opciones de restauración.**

Primary Partition (Partición Primaria):
- **Tipo: Partición primaria**
- **Tamaño: Resto del espacio disponible en el disco**
- **Función: Contiene el sistema operativo, programas y archivos del usuario.**

E.particionado GPT Windows

La siguiente imagen muestra el esquema de particionado generado por el propio instalador de Windows. Se puede ver las 4 particiones antes mencionadas. No todas son visibles desde el administrador de discos.



E.particionado GPT Ubuntu (Linux)

Cuando se realiza la instalación de ubuntu en un equipo con arranque UEFI, el esquema predeterminado de particiones es el siguiente:

EFI System Partition (ESP):
- Tipo: Sistema EFI
- Tamaño: Alrededor de 100 MB
- Función: Almacena los archivos de arranque del sistema operativo Ubuntu y otros archivos necesarios para el proceso de arranque en sistemas UEFI.

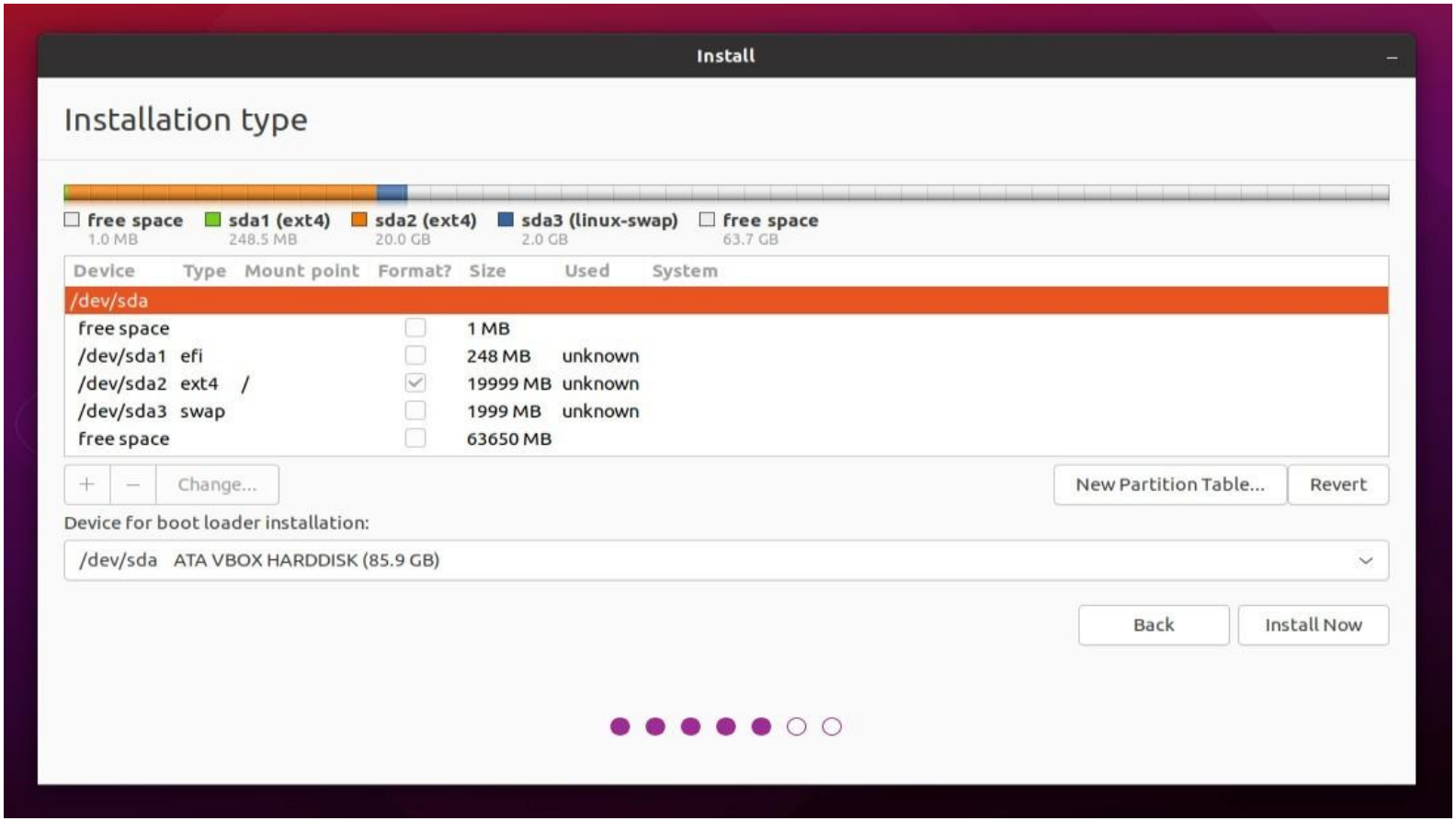
Root Partition (/):
- Tipo: Partición primaria
- Tamaño: Variable, dependiendo del tamaño del disco y la configuración del usuario
- Función: Contiene el sistema operativo Ubuntu, incluyendo el sistema de archivos y la mayoría de los programas.

Swap Partition (opcional):
- Tipo: Partición primaria
- Tamaño: Variable, dependiendo de la cantidad de RAM en el sistema (a menudo recomendado tener al menos el tamaño de la RAM física)
- Función: Espacio de intercambio utilizado como extensión de la memoria RAM física en caso de que esta se agote.

Home Partition (opcional):
- Tipo: Partición primaria
- Tamaño: Variable, dependiendo del espacio disponible y las preferencias del usuario
- Función: Almacena los archivos y datos personales de los usuarios, permitiendo separar los datos del sistema para facilitar las actualizaciones o reinstalaciones futuras.

E.particionado GPT Ubuntu (Linux)

La siguiente imagen muestra el esquema de particionado generado por el propio instalador de Ubuntu. Se puede ver las particiones antes mencionadas.



E.particionado Heredado (MBR)

Cuando la instalación se realiza en un equipo con arranque heredado (Legacy BIOS), el esquema de particiones recomendado por Microsoft es el siguiente:

a) **Partición del sistema (System): ha de estar configurada como la partición activa del disco de arranque.**
b) **Partición de Windows (Windows): partición que aloja el sistema operativo.**
c) **Partición de recuperación (Recovery): almacena las herramientas del entorno de recuperación de Windows**

SISTEMAS OPERATIVOS

Introducción

Cuando trabajamos con un ordenador con sistema operativo, el usuario no necesita preocuparse de las direcciones de memoria RAM usadas, de la gestión de las interrupciones, de la interfaz gráfica.

Los sistemas operativos actuales están formados por un conjunto de programas que tratan de facilitar el empleo del dispositivo al usuario lo máximo posible

Las funciones básicas de un sistema operativo son:

1. Actuar de interfaz entre el usuario y el hardware de manera transparente para el primero. Debe ofrecer soporte a los usuarios para que sus acciones se transmitan con facilidad. Los usuarios no tienen por qué ser especialistas de software o hardware para usarlo (depende del SO).
2. Gestionar los recursos software y hardware del equipo. El uso eficiente de los recursos es primordial puesto que son limitados. Por ejemplo, la eficiencia buscada en un equipo de sobremesa en nuestro hogar es diferente a la eficiencia de un sistema que gestione un conjunto de alarmas en tiempo real.

Funciones y características

El sistema operativo es un software con características particulares, ya que debe administrar todos los recursos del sistema y coordinar su uso entre los usuarios y el resto de software. Por tanto, las características fundamentales que debe cumplir cualquier sistema operativo genérico son las siguientes:

- **Adaptabilidad:** debe ajustarse a dos situaciones que evolucionan en paralelo, nuevo software y nuevo hardware. El sistema operativo debe ser capaz de reacondicionarse (normalmente mediante actualizaciones).
- **Facilidad de uso:** Normalmente, una mayor comodidad implica mayor gasto de recursos (como por ejemplo un sistema gráfico de ventanas). Por ello, existen sistemas operativos que ganan en eficiencia a costa de restringir su manejabilidad.
- **Eficiencia:** los recursos (procesadores y núcleos, RAM, acceso a discos, red o cola de impresión) son limitados. El sistema operativo debe atender todas las peticiones de usuarios, programas y el propio sistema operativo para facilitar el acceso a los recursos.

Funciones y características

El sistema operativo debe administrar de forma eficiente los recursos, atendiendo a los objetivos del propio sistema. Los recursos más solicitados son los siguientes:

- **Memoria RAM:** La parte del sistema operativo que siempre reside en memoria se denomina núcleo o kernel. Es un subconjunto de software esencial del propio sistema operativo que, por su importancia en la gestión del sistema, no puede ser descargado de la memoria principal.
- **Procesador:** Aunque un sistema disponga de varios núcleos y, por tanto, pueda ejecutar múltiples procesos de forma simultánea (multitarea), el sistema operativo debe planificar la CPU para decidir qué proceso se ejecuta en cada momento.
- **Adaptadores de red:** Diversas aplicaciones pueden utilizar la red de manera simultánea, por lo que el sistema operativo debe gestionar las conexiones entre aplicaciones, procesos y usuarios para evitar conflictos y optimizar el rendimiento.
- **Medios de almacenamiento:** El acceso a la memoria secundaria (discos duros o SSD) puede representar un cuello de botella, por lo que el sistema operativo gestiona la lectura, escritura y caché de datos para mejorar la eficiencia.
- **Colas de impresión:** Pueden existir varias peticiones de impresión dirigidas a una misma impresora, por lo que el sistema operativo debe organizar la cola de trabajos y asignar prioridades para garantizar un funcionamiento ordenado.

Gestión de procesos

El procesador, como recurso fundamental del sistema, ha de repartir su tiempo entre los diferentes procesos que deseen ejecutarse. El SO debe organizar el paso de estos procesos por el procesador (o procesadores) y sus núcleos, de tal manera que los tiempos de ejecución de las diferentes tareas sigan los objetivos del SO. Por tanto, el SO debe gestionar:

- La asignación de procesos a varios procesadores (si dispone de varios).
- Uso de multiprogramación sobre procesadores individuales y sus núcleos.
- La ejecución de una aplicación o proceso en cuanto a su sincronización con otros procesos o hilos.

Estos objetivos son definidos por políticas de planificación con orientaciones diferentes:

- Planificación orientada a los usuarios (orientada a las entradas y salidas): intenta agilizar las acciones de procesos como accesos a discos, señales de pantallas táctiles o accesos a Internet. Prima el tiempo de respuesta a los usuarios.
- Planificación orientada al sistema (orientada a procesos de cálculo).

Gestión de memoria

Por gestión de memoria se entiende la planificación y gestión global de la memoria principal con extensión a la memoria secundaria. Hoy en día los sistemas disponen de memoria RAM suficiente para albergar el sistema operativo y mucho más software. Pero también se debe planificar cómo actuar en caso de necesitar mayor espacio de memoria empleando el almacenamiento secundario.

El sistema operativo amplía virtualmente la memoria RAM, tomando prestado del disco duro espacio como si fuese una extensión de la primera.

A este concepto se denomina memoria virtual, paginación (Windows), memoria de intercambio/SWAP (distribuciones Linux).

Toda la transferencia de información entre memorias requiere una planificación vital para ahorrar tiempo y no lastrar la eficiencia del sistema.

Gestión de entradas salidas y almacenamiento

- **Gestión E/S:** Acciones como tocar una pantalla táctil, imprimir un documento, acceder a un fichero del disco duro o navegar por Internet, requieren que el sistema operativo necesite administrar dichos recursos, ofreciendo soluciones rápidas y de la forma menos costosa posible.
- Cada dispositivo de E/S tiene una forma peculiar de interaccionar con el sistema operativo, y este ha de gestionarlo estableciendo un diálogo claro y fluido.
- **Gestión de almacenamiento secundario:** Los discos duros son dispositivos de E/S por sí mismos, por lo que la gestión de los archivos y directorios en ellos es fundamental. La estructura organizativa de los archivos y su gestión viene determinada por los sistemas de archivos (FAT32, ntfs, ext4, APFS,…)
- **Gestión de la seguridad.** Se deben evitar actuaciones originadas por errores software, errores hardware o por actuaciones maliciosas de usuarios, ya sean intencionadas o no, dando lugar a inconsistencias en el sistema. Por ello, el sistema debe garantizar: **El servicio y la disponibilidad de sus recursos, y la confidencialidad, protección e integridad del sistema y los datos.**

Gestión de errores e interfaz de usuario

La gestión de errores es un elemento fundamental en todo sistema operativo. El control de la totalidad de las acciones que puedan derivarse del software de terceros, el hardware y el propio sistema operativo es prácticamente imposible. Por ello, el sistema operativo debe gestionar todo tipo de errores de la manera más liviana posible, informando al usuario y salvaguardando de forma prioritaria la seguridad del sistema y los datos.

Gestión de la interfaz de usuario. Todas las acciones encomendadas al sistema operativo tratadas hasta ahora no tendrían sentido sin una interfaz que permita una clara manejabilidad del sistema. Por tanto, los sistemas operativos con interfaz gráfica o textual (por línea de comandos) deben ofrecer un soporte que permita una fluida comunicación, así como realizar todas las acciones necesarias para la gestión, administración o explotación del mismo

Tipos de sistemas operativos

Los objetivos de los sistemas operativos marcan la eficiencia en el uso al que se destine el sistema. Se pueden diferenciar tipologías de sistemas operativos con objetivos antagónicos entre sí, aunque en la práctica podamos encontrar versiones intermedias muy variadas.

Existen distintos puntos de vista para catalogar los sistemas operativos:

a) Atendiendo al número de procesos que se pueden ejecutar concurrentemente:

- **Monotarea o monoprogramado**: un proceso únicamente puede ser ejecutado por un usuario. Esto quiere decir que un usuario solo puede estar ejecutando un programa, además del propio sistema operativo.

- **Multitarea o multiprogramado**: un usuario puede ejecutar varios procesos simultáneamente. De esta manera, pueden existir varios programas en memoria susceptibles de ser ejecutados.

Tipos de sistemas operativos

b) **Atendiendo al número de usuarios** que pueden ser atendidos por el sistema operativo simultáneamente:

- **Monousuario:** solo pueden atender a un usuario. El usuario goza de todos los recursos, a menos que el sistema operativo los acapare.
- **Multiusuario:** pueden atender a más de un usuario concurrentemente. Por tanto, los recursos del sistema deben ser gestionados para todos ellos.

De esto podemos deducir que los sistemas operativos multiusuario son también multitarea.

Actualmente la mayoría de sistemas operativos son multitarea y multiusuario.

Tipos de sistemas operativos

c) **Atendiendo al tipo de procesamiento:** el sistema operativo ha de estar preparado para ejecutar procesos con diferentes finalidades y requisitos. Los sistemas operativos intentan optimizar sus recursos, independientemente de los procesos que atiendan. Sin embargo, los procesos, según su forma de ejecutarse, pueden ser:

- **De tiempo real:** requieren unos plazos en su ejecución o tiempos de respuesta
- **Interactivos:** requieren de la participación del usuario.
- **Por lotes, batch o no interactivos:** se suministra un conjunto de tareas al sistema operativo con características similares, y este se encarga de ejecutarlas en serie y sin la intervención del usuario. En caso de producirse un error en una tarea del lote, el resto de tareas no se podrá ejecutar. Ejemplos: realización de facturas agrupadas, tareas de cómputo en investigación, envío de mensajes con informes o resúmenes en cadenas de producción, etc

Tipos de sistemas operativos

d) Atendiendo al sistema de interfaz empleado:

- **Textuales:** emplean un repertorio de comandos que se introducen en el sistema de forma escrita a través de un terminal de órdenes. Aunque, se necesitan mayores conocimientos de sintaxis y manejo del sistema operativo, las acciones pueden llegar a ser muy potentes desde un punto de vista de explotación del sistema operativo.

- **Gráficos:** usan un conjunto de ventanas, botones y desplegables gráficos donde se representan los diferentes volúmenes, unidades y sistemas de ficheros de forma muy intuitiva. Además, los programas lanzados presentan una vista gráfica. El manejo se realiza con un dispositivo de entrada/salida, como un ratón, y destaca por su fácil utilización. Este sistema emplea muchos más recursos que el textual a nivel de procesador, memoria e incluso, en algunos casos, se necesita de manera casi obligada un adaptador gráfico. Por tanto, en sistemas operativos donde se busca ahorrar todo tipo de recursos en favor de atender a peticiones de usuarios y procesos, la interfaz gráfica se desprecia.

Tipos de sistemas operativos

e) Atendiendo a la forma de ofrecer los servicios:

- **Sistemas operativos cliente o de escritorio.** Se encargan de realizar el procesamiento de la información, la gestión de los procesos, de la memoria, dispositivos de E/S de un solo equipo. Este SO es el que usamos normalmente en casa, en una oficina, en clase…
- **Sistemas operativos en red.** Se encargan de gestionar la red, los usuarios, y los recursos de una red de ordenadores en general, de forma centralizada mediante un servidor o varios como réplicas o extensiones del primero. Es en el servidor donde se instala este sistema operativo. El resto de equipos de la red (con sistemas operativos cliente) se conectan al servidor. Su principal objetivo es el intercambio de información.
- **Sistemas operativos distribuidos.** A diferencia de los anteriores, actúan varios ordenadores de manera transparente al usuario, de forma que da la sensación que este interactúa solo con uno de ellos. Por tanto, permiten emplear los recursos de varios ordenadores en paralelo.

Tipos de sistemas operativos (SO Distribuidos)

Los sistemas operativos distribuidos presentan muchas ventajas, aunque destacan por su:

- Escalabilidad: es relativamente sencillo ampliar la potencia de cálculo y los recursos del sistema, puesto que se pueden añadir más ordenadores.
- Confiabilidad: en caso de que una ordenador falle, el resto puede hacerse cargo de las tareas que se van a realizar.

Debido a la complejidad en el diseño e implementación (principalmente por el concepto de transparencia) de los sistemas operativos distribuidos, estos no se han popularizado y desarrollado como tales. Sin embargo, muchas de sus ideas se han aplicado a los sistemas operativos de escritorio y en red, y sobre todos aplicamos este concepto a los **SISTEMAS DISTRIBUIDOS:**

https://www.youtube.com/watch?v=ozeEneFH6qs

Arquitectura de los SO

Un sistema informático debe ser capaz de ejecutar instrucciones en dos niveles de funcionamiento (no confundir con los usuarios de Windows, Linux, etc.):

- **En modo usuario:** es el modo menos privilegiado de funcionamiento del sistema. En este modo no se permite el acceso directo al hardware. Las instrucciones que se ejecutan en este modo sólo pueden acceder a su propio espacio de direcciones de memoria y utilizan el **API** del sistema para requerir **los servicios** del sistema operativo. Este es el modo de ejecución que utilizan todos los programas de aplicación que tengamos instalados
- **En modo núcleo (también llamado modo kernel) o modo supervisor:** En él, las instrucciones se ejecutan en un modo privilegiado, teniendo acceso directo a toda la memoria (incluidos los espacios de direcciones de todos los procesos que estén ejecutándose). También podrán acceder a todo el hardware disponible.

Arquitectura de los SO

Ahora que ya sabemos que el sistema operativo se divide en distintos elementos, podemos plantearnos el modo en el que dichos elementos se organizan dentro del sistema operativo para llevar a cabo su cometido.

También será importante para el diseño del sistema establecer qué componentes del mismo se ejecutan en modo núcleo y cuáles en modo usuario.

En este sentido, los planteamientos que se aplican en los sistemas operativos más conocidos son los siguientes:

- **Monolítico.**
- **Micronúcleo.**
- **Núcleo híbrido.**

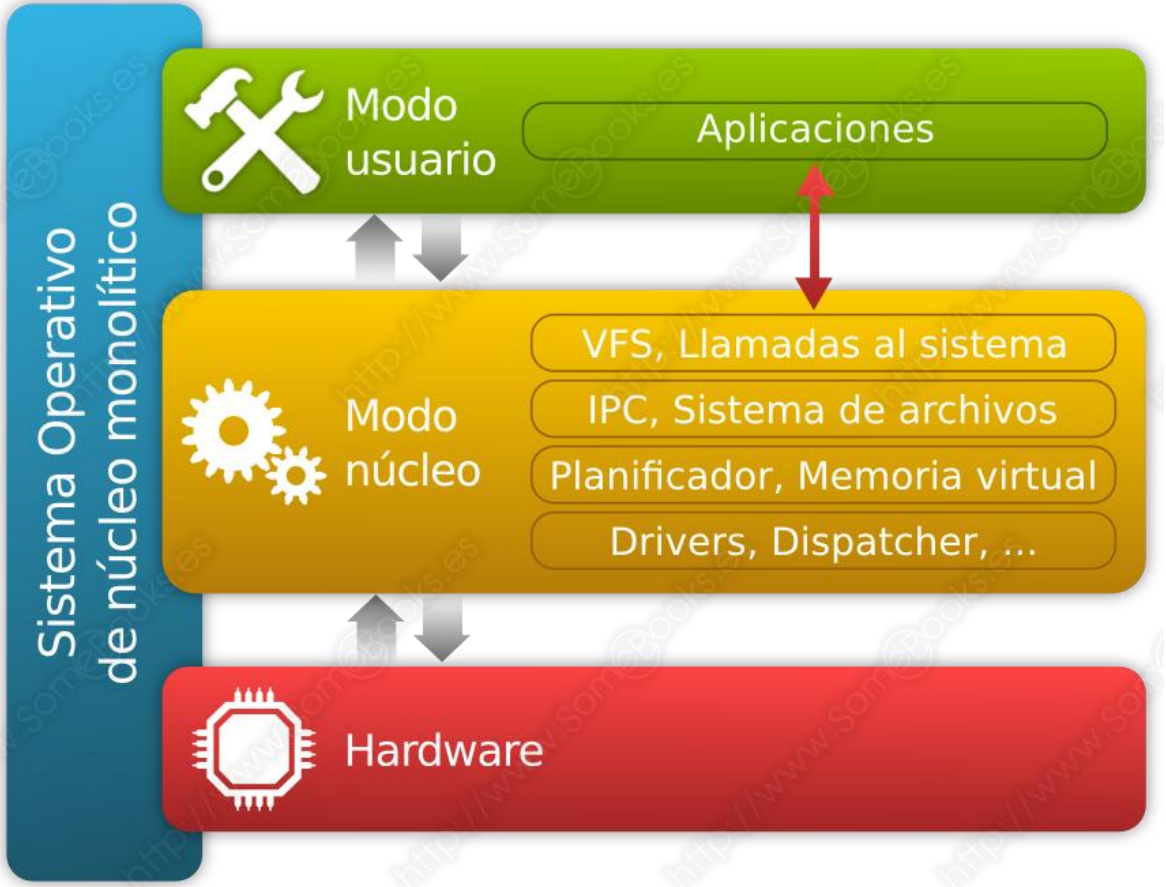
SO Monolíticos

En este tipo de sistemas, el núcleo concentra la mayor parte de la funcionalidad del sistema operativo (sistema de archivos, gestión de memoria, etc), de modo que todos sus componentes principales se ejecutarán en modo núcleo.

En una estructura monolítica pura tendríamos un núcleo complejo y de gran tamaño que debería ser recompilado por completo ante cualquier modificación. Sin embargo, cuando se utiliza la carga dinámica de módulos, éstos pueden compilarse por separado y cargarse durante la ejecución del sistema, ofreciendo mayor velocidad.

Como ejemplos de sistemas con estructura monolítica podemos nombrar Solaris, FreeBSD, OSX (versiones anteriores a la 9), GNU/Linux y las versiones de escritorio de Windows anteriores a XP.

SO Monolíticos



La imagen muestra como la mayoría de la funcionalidad queda en el Kernel de SO

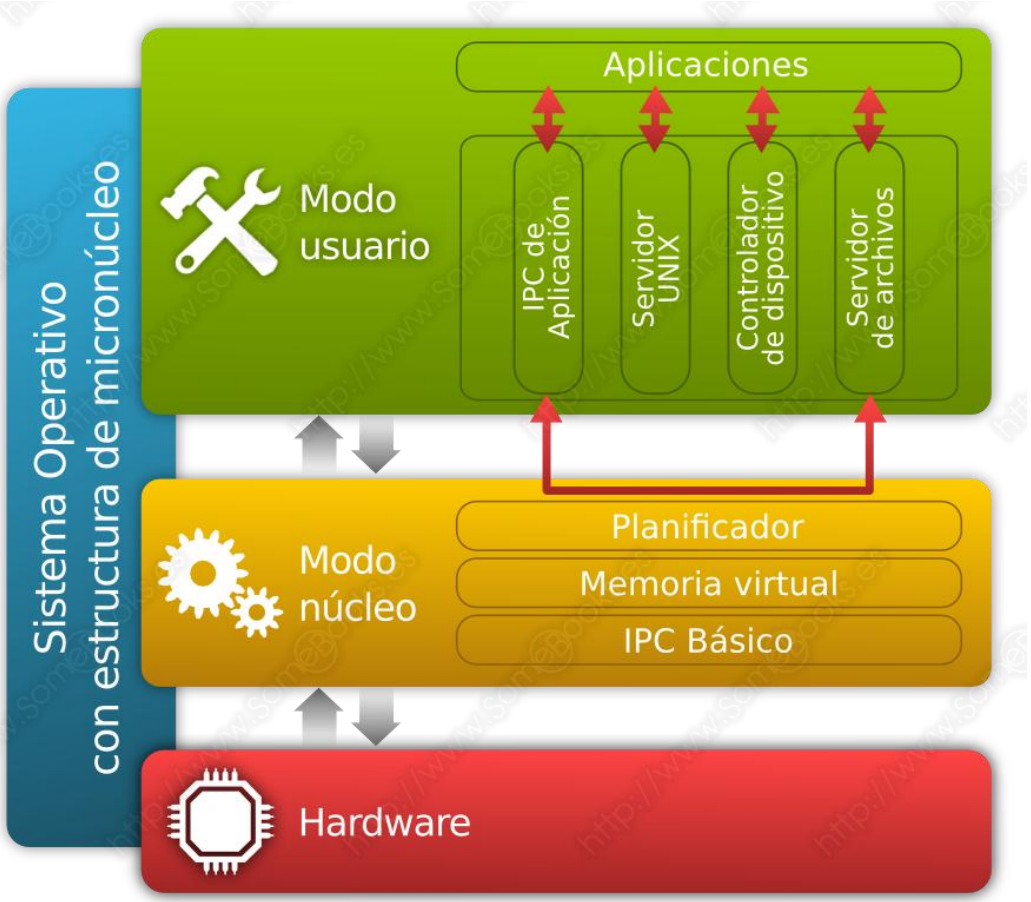
SO MicroKernel

En este tipo de sistemas, el núcleo sólo contiene la implementación de servicios básicos como el soporte de acceso a memoria de bajo nivel, la administración de tareas y la comunicación entre procesos (también conocida como IPC, del inglés, Inter-Process Communication).

En este tipo de arquitectura, el micronúcleo es el único componente que se ejecuta en modo privilegiado. El resto de las funciones del sistema, como los controladores de dispositivos (drivers), el sistema de archivos, la gestión de E/S, etc, se ejecutan en modo usuario. De esta forma, es más difícil que un error de programación en uno de los módulos afecten al funcionamiento del resto, haciendo que el sistema sea más fiable.

Como ejemplos de sistemas con estructura de micronúcleo podemos nombrar AIX, AmigaOS, Minix, Symbian (aunque en algunos textos aparece como monolítico con carga dinámica de módulos) y NeXTStep (aunque a veces lo encontramos entre los sistemas con núcleo híbrido).

SO MicroKernel



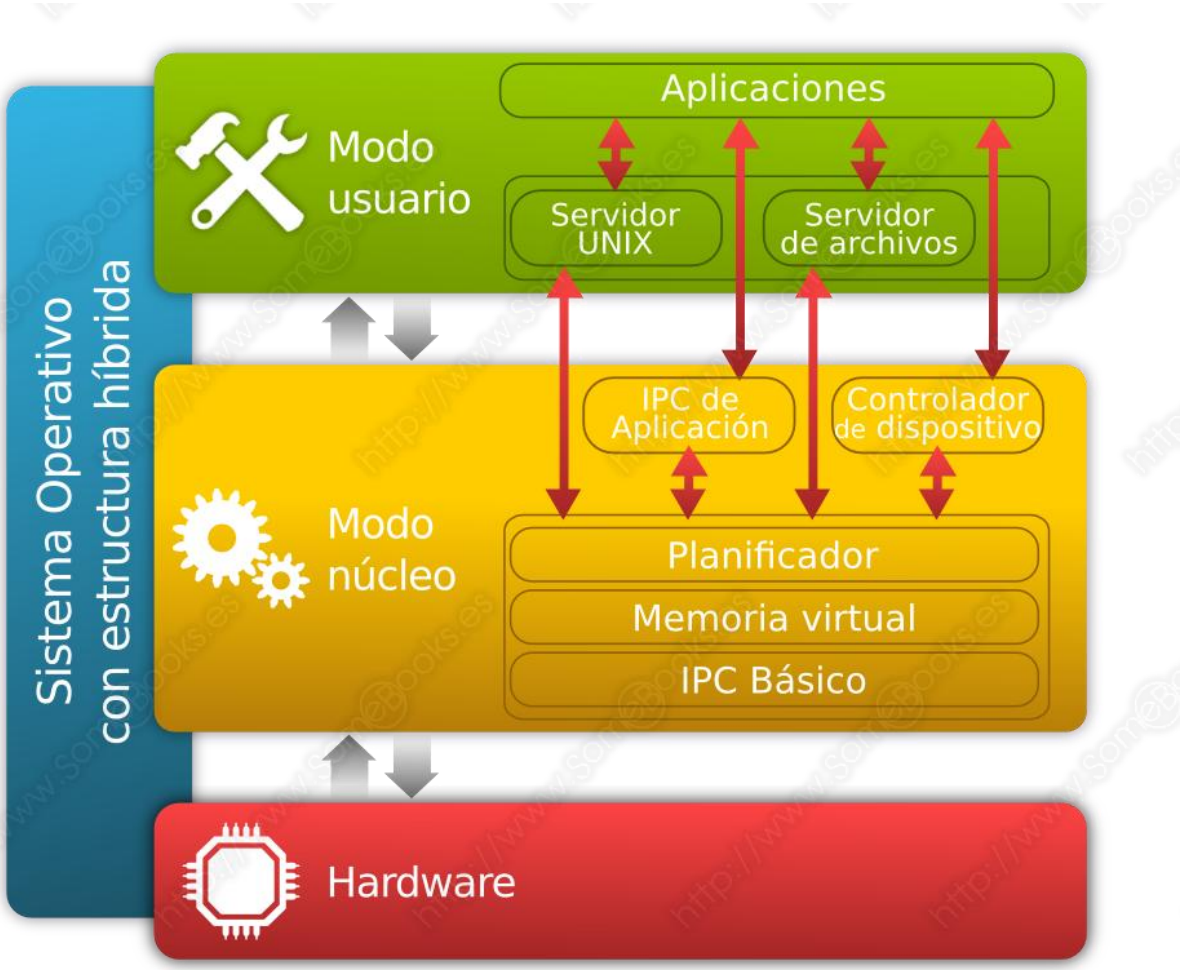
Los servicios o aplicaciones si se quieren comunicar con el Hardware, deben enviar peticiones al Kernel a modo de "mensajes". Será el Kernel el que gestione esa comunicación

SO Kernel Híbrido

Este tipo de arquitectura consiste básicamente en un esquema de micronúcleo que incluye algo de código complementario para hacerlo más rápido, aunque buena parte de las funciones del sistema operativo siguen ejecutándose en modo usuario.

Se trata de una solución de compromiso que han adoptado muchos de los sistemas operativos modernos, como las versiones de escritorio de Microsoft Windows, a partir de XP, y todas las versiones de Servidor.

SO Kernel Híbrido



VERSIONES, DISTRIBUCIONES Y EDICIONES, DE LOS PRINCIPALES SISTEMAS OPERATIVOS

https://www.youtube.com/watch?v=fPQCfy6FNE8

Distribuciones de SO más utilizados

Podemos encontrar versiones y distribuciones de sistemas operativos, que han sido diseñados y desarrollados para correr en entornos distintos. Desde un reloj, un control de temperatura de una casa, router, móvil, e-book, tablet, servidores, NAS, equipos de sobremesa/portátiles, televisores, consolas, cámaras, proyectores, frigoríficos, GPS, Chromecast, coches, impresoras, mesas, bicicletas…

Versiones de los SO más utilizados

La mayoría de aparatos electrónicos que utilizan microprocesadores para funcionar llevan incorporado un sistema operativo.

Entre todos estos ejemplos, la mayoría son SO basados en Android, por su interfaz gráfica, sobre todo, por su medio de interacción táctil, y la facilidad de desarrollar aplicaciones para estas plataformas.

En realidad, esto significa que muchos de los sistemas operativos incorporan el kernel de Linux como núcleo principal

Versiones de los SO más utilizados

Dentro de todos los sistemas operativos disponibles, **UNIX, LINUX y WINDOWS** son los más importantes.
Prácticamente cualquier sistema operativo que veas es o pertenece a alguna distribución de estos tres.

UNIX  LINUX Windows

Es importante destacar que tanto **UNIX como Linux** denominan a sus diferentes sistemas operativos principales como **distribuciones** (cuando se basan directamente en UNIX o Linux), **basados en** (si derivan de una distribución concreta) o de tipo **UNIX/Linux** (si su kernel se inspira en este modelo).

Windows, en cambio, denomina **VERSIONES** a sus SO, las cuales salen al mercado cada cierto tiempo. Estas versiones ofrecen al usuario **EDICIONES**, la cuales disponen de características técnicas que las diferencian dependiendo el ámbito de aplicación, pero siendo la misma versión de SO

UNIX®

Fue desarrollado por los laboratorios **Bell (AT&T Bell Labs)** en 1969. Su última versión ampliamente distribuida del sistema original fue **UNIX V7**, lanzada en 1979.

UNIX utiliza un **núcleo monolítico** y se caracteriza por ser **multiplataforma, multiusuario y multitarea**, diseñado principalmente para **entornos de servidor**. A diferencia de muchos de sus descendientes, **UNIX no es de código abierto**, aunque ha servido como base para numerosos sistemas de tipo UNIX que sí lo son.

La interacción con el usuario se realiza principalmente mediante **línea de comandos**, si bien existen entornos gráficos desarrollados posteriormente.

UNIX®

En la práctica, el término “UNIX” se emplea a menudo de forma genérica para referirse a sistemas operativos **de tipo UNIX**, como **GNU/Linux, macOS, FreeBSD u OpenBSD**. Sin embargo, no todos ellos son UNIX en sentido estricto: para ser considerados como tales deben poseer una **certificación oficial UNIX** (The Open Group UNIX®).

Por ejemplo, **FreeBSD y macOS** están certificados como sistemas **UNIX**, mientras que **Linux** se considera un sistema **tipo UNIX**, al no disponer de dicha certificación.

Algunas distribuciones, como **Debian GNU/kFreeBSD**, combinan el entorno GNU con el núcleo de FreeBSD, representando una fusión entre ambos mundos.

GNU/Linux (Open Source)

https://upload.wikimedia.org/wikipedia/commons/1/1b/Linux_Distribution_Timeline.svg

A menudo utilizamos el término **Linux** para referirnos a un sistema operativo completo, pero en realidad **Linux** es el nombre del **núcleo (kernel)** de estos sistemas operativos. El conjunto completo, que incluye herramientas y utilidades del proyecto GNU, se denomina correctamente **GNU/Linux**.

El proyecto fue iniciado en **1991** por el estudiante finlandés **Linus Torvalds**, con el objetivo de desarrollar un sistema operativo **libre y abierto** inspirado en UNIX.

Aunque Linux está basado en los principios de diseño de UNIX, **no comparte código** con este, por lo que se clasifica como un **sistema de tipo UNIX (Unix-like)**.

GNU/Linux (Open Source)

Las diferentes versiones de GNU/Linux se denominan **distribuciones**. Cada distribución combina el núcleo Linux con distintas herramientas, gestores de paquetes, interfaces y configuraciones. Entre las más conocidas se encuentran:

- **Debian**, junto con sus derivadas **Ubuntu**, **Linux Lite** o **Kubuntu** (estas últimas derivadas de Ubuntu).
- **Red Hat Enterprise Linux (RHEL)** y su sucesora comunitaria **Fedora**.
- **CentOS**, basada en el código de Red Hat.

La mayoría de las distribuciones pueden incorporar un **entorno gráfico (GUI)** instalando el paquete correspondiente.

Entre las distribuciones más populares orientadas a **usuarios de escritorio** se encuentran **Ubuntu** y **Linux Lite**, que buscan ofrecer una experiencia de uso más intuitiva y cercana a sistemas como Windows.

Microsoft Windows

https://upload.wikimedia.org/wikipedia/commons/0/0e/Windows_Family_Tree.svg

La primera versión de Windows fue lanzada en **1985** bajo el nombre de **Windows 1.0**. Sin embargo, no fue hasta **1990**, con **Windows 3.0**, cuando el sistema operativo comenzó a popularizarse en los hogares y oficinas, gracias a su **interfaz gráfica de usuario (GUI)** y a la creciente adopción de los ordenadores personales.

En sus inicios, Windows no era un sistema operativo independiente, sino una **capa gráfica que se ejecutaba sobre MS-DOS**, gestionando las ventanas y la interacción visual con el usuario.

Posteriormente, Microsoft desarrolló una nueva línea de sistemas operativos con un **núcleo (kernel) completamente independiente**, denominado **Windows NT (New Technology)**.

Microsoft Windows

Todas las versiones modernas de Windows —tanto domésticas como empresariales y de servidor— se basan en el kernel NT, más estable, seguro y preparado para entornos multitarea y multiusuario.

En la actualidad, las versiones de escritorio más utilizadas son las de Windows 11, que se ofrece en varias ediciones según su ámbito de uso.

En el ámbito profesional, Microsoft mantiene su línea de sistemas para servidores bajo la denominación Windows Server, cuya versión más reciente es Windows Server 2022, disponible en ediciones como:

- *Datacenter* (para entornos virtualizados o de alta densidad)
- *Standard* (para instalaciones físicas convencionales)
- *Essentials* (para pequeñas empresas)

macOS

Inicialmente conocido como **Mac OS X**, lanzado en **2001** por **Apple Computer** para sus equipos **Macintosh**, es un sistema operativo con **kernel propio XNU (híbrido)**, empleado en los dispositivos del fabricante Apple. Su primer kernel implementaba **UNIX**, motivo por el cual está **certificado como sistema operativo UNIX**.

La mayoría de los dispositivos que incorporan macOS son **equipos de escritorio y portátiles**, aunque también existen **versiones para servidores**, como **macOS Server**.

Estos sistemas operativos vienen **preinstalados en los equipos Apple** y son **en su mayoría de tipo propietario**, aunque integran componentes de código abierto procedentes del proyecto **Darwin**.