

I.E.S Alonso de Ercilla

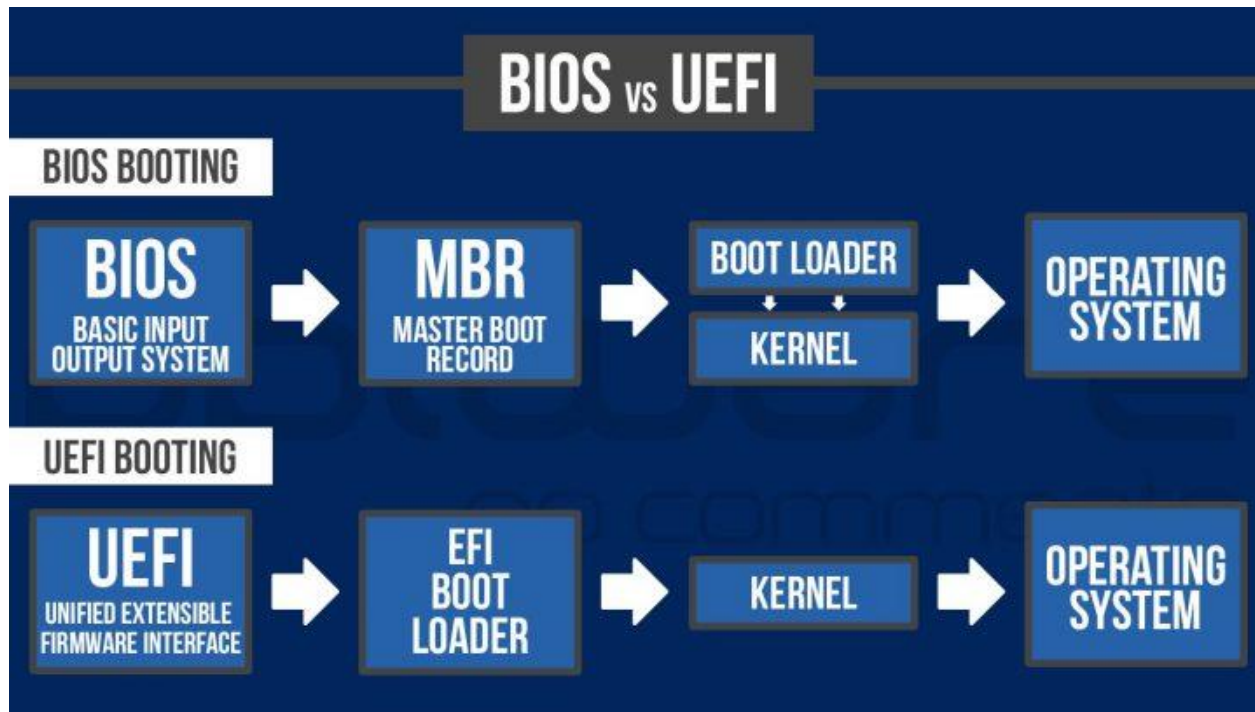
CFGS Desarrollo de Aplicaciones Multiplataforma

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, Hiren's boot(conjunto de programas: EaseUS Partition Master, EaseUS Partition Master, AOMEI Partition Assistant, DiskGenius, etc...).

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.

Tipo de particiones

Partición primaria: Divisiones primarias 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 está localizado en partición activa, y será el que se cargue al arrancar el ordenador. Para que un dispositivo de almacenamiento sea utilizable, al menos tiene que tener una partición primaria.

Partición extendida (MBR): sólo puede haber una partición de este tipo por disco, y sirve para contener particiones lógicas. Fue diseñada para solucionar el problema en MBR de sólo poder disponer de 4 particiones.

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

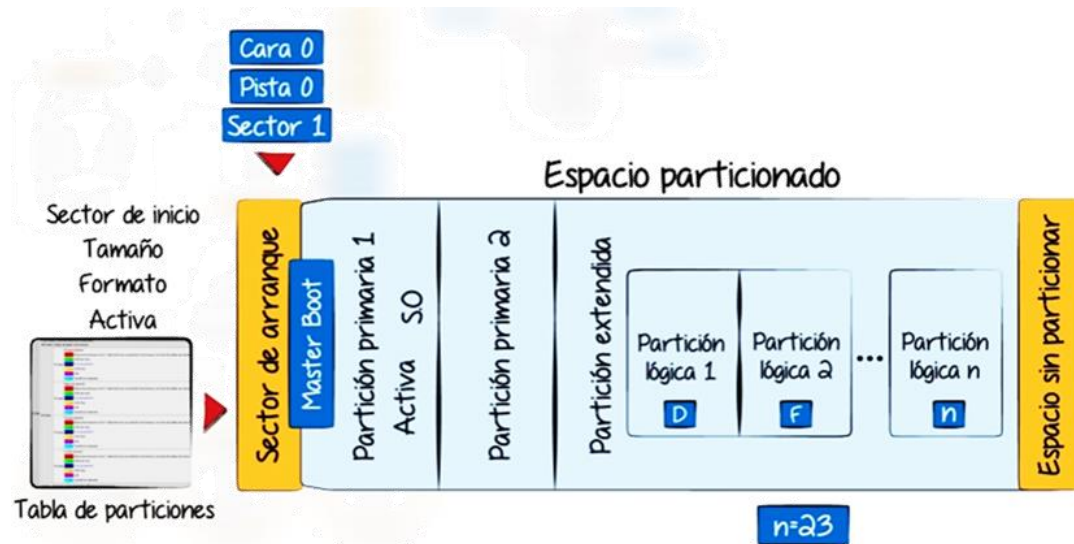
En el caso de Windows las particiones primarias y las particiones lógicas, tienen asociadas una letra para poder ser identificadas (C:\, D:\, E:\,...)

Las particiones en Linux son identificadas dentro del sistema operativo por identificadores del tipo sda, sdb, etc, y son accesibles por su punto de montaje, que es el directorio en el cual la partición es accesible en el sistema de archivos (/ (raíz), /home, /boot).

Esquema de particiones

El sistema operativo manipula e interpreta cómo unidades independientes las particiones primarias y lógicas. Estas particiones por si solas no pueden almacenar datos. Para ello necesitan tener un sistema de archivos conocido como formato.

Es importante no confundir el formato o sistema de archivos, con las particiones.



Esquema en MBR

Formatos

Cada sistema operativo opera con unos determinados sistemas de archivos (formatos), es decir, como se organizan, acceden, y almacenan los archivos en una unidad de almacenamiento. No todos los formatos pueden ser leídos por los sistemas operativos. Windows no podría leer un formato ext4 por ejemplo.

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 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



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 CPU.

Es el esquema de particionamiento de los sistemas con estándar 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*)

Esquema de particionamiento de los sistemas con estándar UEFI. A cada partición se le asigna un identificador global único (GUID). Este estándar mejora el estándar BIOS tradicional y solventa multitud de limitaciones, al ser más flexible, potente y fácil de usar.

Puede contener hasta 128 particiones, eliminando las limitaciones de cuatro particiones primarias.

Soporta discos de tamaño muy superior al esquema MBR, que limitaba en 2TB su capacidad máxima.

Permiten trabajar en modos de 32 y 64 bits.

Inicio del sistema con mayor rapidez.

Es mucho más seguro que los sistemas BIOS:

- Carga el núcleo de los sistemas operativos comprobando su originalidad.
- GPT realiza copias de la estructura de la tabla de particiones.

Se puede conectar directamente con Internet.

Dispone de una interfaz mucho más amigable.

Consta de un gestor de arranque propio, es decir, no está vinculado a ningún sistema operativo.

Estructura de la tabla GPT

Los dispositivos con formato GPT deben tener una partición de sistema EFI (ESP). Esta partición ha de estar formateada en FAT32 y suele ser la primera. 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.

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).

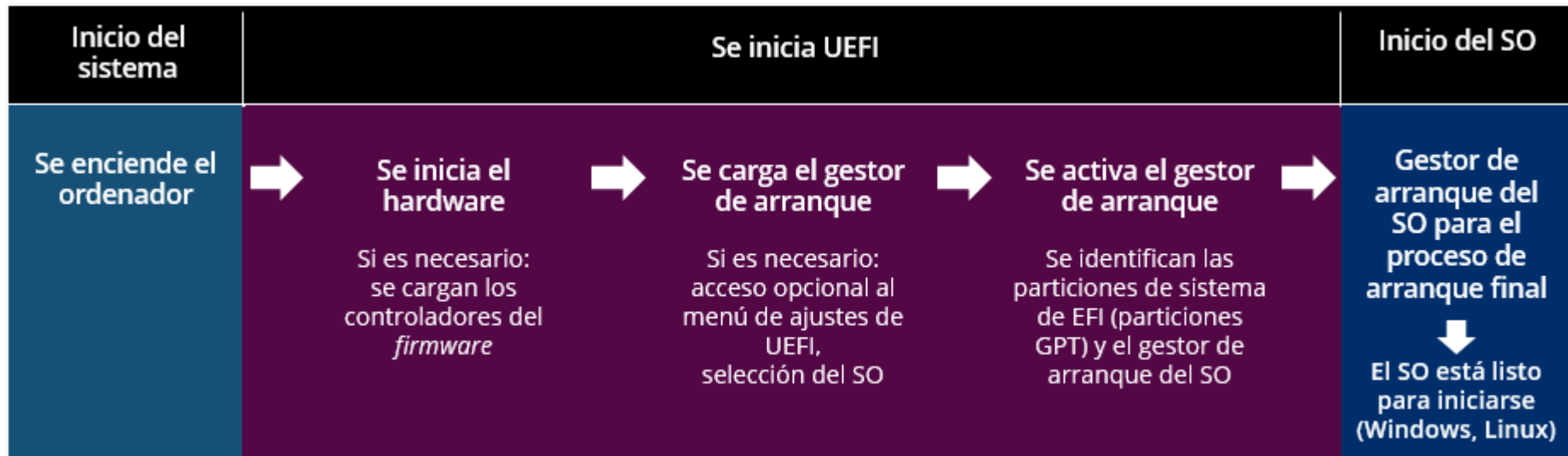
UEFI puede proporcionar menús gráficos adicionales, e incluso proporcionar acceso remoto para la solución de problemas y mantenimiento.

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.

Otra de las características, es que dispone de un gestor de arranque propio, que permite la selección y carga directa de los sistemas operativos, eliminando la necesidad de recurrir a gestores de arranque.

UEFI (*GUID Partition Table*)

El sistema UEFI se encuentra a medio camino entre un sistema BIOS (firmware) y el sistema operativo, por lo que ofrece así muchas ventajas gracias a su flexibilidad. Por otro lado, la denominación BIOS se puede hacer extensible a sistemas con estándar BIOS o estándar UEFI cuando haga mención al sistema de arranque en sentido genérico.



Modo de arranque UEFI

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

1. **Heredado o Legacy BIOS:** establece la compatibilidad hacia atrás con discos con esquemas MBR. En este modo, no se hacen efectivas las ventajas del estándar UEFI. Durante la instalación del sistema operativo, la configuración de las particiones se hará con esquema MBR.
2. **UEFI:** es el modo que recomienda 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 consonancia entre el modo de arranque del equipo y el esquema de particionamiento del disco de arranque (el primero en el orden de arranque).

Los discos con sistemas operativos instalados en esquemas de particionamiento GPT y MBR deben ser iniciados en modo UEFI y Heredado, respectivamente. De lo contrario, el sistema UEFI o BIOS no reconocerá el sistema operativo al encontrarse en un disco con sistema de particionamiento no reconocido.

E.partitionado 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):

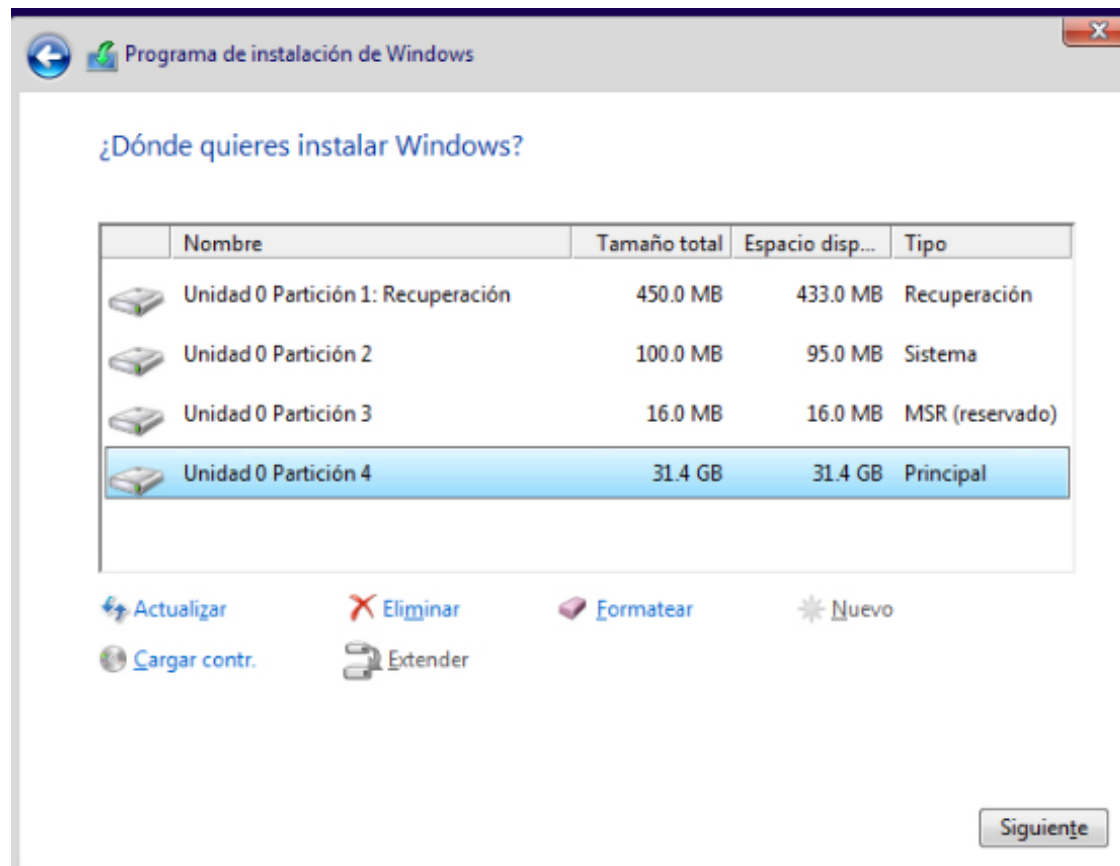
- Tipo: Partición de recuperación
- Tamaño: Variable (generalmente varios GB)
- Función: Almacena los archivos necesarios para la recuperación del sistema, incluyendo herramientas de solución de problemas y opciones de restauración.

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.partitionado 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.partitionado 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 o lógica
- 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):

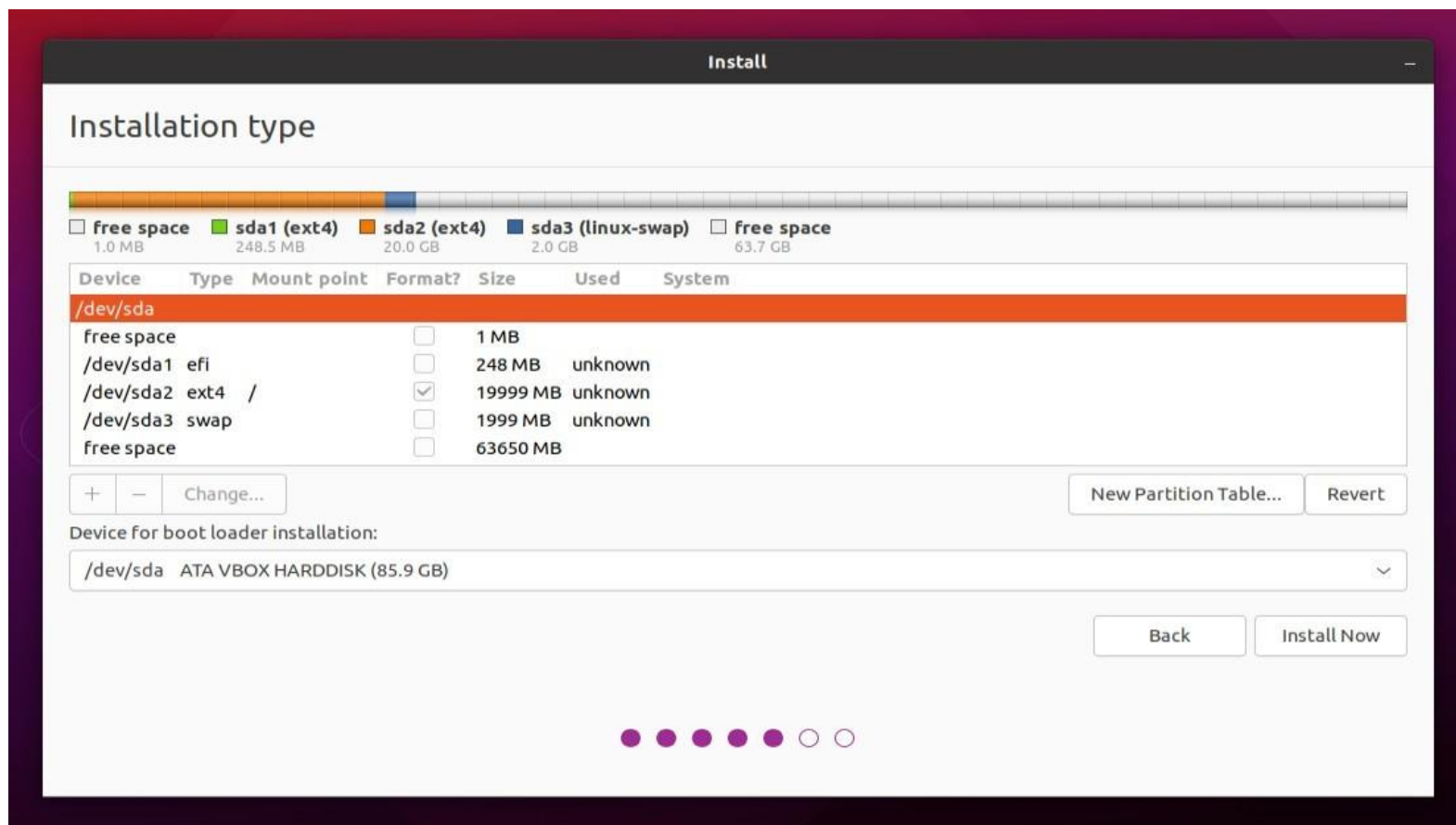
- Tipo: Partición primaria
- Tamaño: Variable, dependiendo de la cantidad de RAM en el sistema (a menudo recomendado tener al menos el tamaño de la RAM física)
- Función: Espacio de intercambio utilizado como extensión de la memoria RAM física en caso de que esta se agote.

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.partitionado 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.partitionado Heredado (MBR)

Y cuando la instalación se realiza en un equipo con arranque 'Heredado', el esquema ideal que plantea 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 se tiene que preocupar de las direcciones de memoria RAM usadas, de la gestión de las interrupciones, de la interfaz gráfica.

Los sistemas operativos actuales están compuestos por un conjunto de software que trata 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 entre los usuarios y el resto de software. Por tanto, las características fundamentales que debe soportar cualquier sistema operativo genérico son:

- **Adaptabilidad:** se debe acomodar 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 al objetivo de dicho sistema operativo. Los más solicitados son:

- **Memoria RAM.** La parte del sistema operativo que siempre reside en memoria RAM se denomina núcleo o kernel. Es un subconjunto software del propio sistema operativo que por su importancia en la gestión del sistema no puede abandonar la memoria principal.
- **Procesador.** Aunque disponga de varios núcleos y, por tanto, pueda ejecutar varios procesos a la vez (multitarea), existe multitud de software que desea ejecutarse.
- **Adaptadores de red.** Múltiples aplicaciones hacen uso de la red simultáneamente, debiendo administrar las conexiones de red entre aplicaciones, procesos y usuarios.
- **Medios de almacenamiento.** El acceso a memoria secundaria puede representar un cuello de botella importante.
- **Colas de impresión.** Pueden existir más de una petición de impresión a una misma impresora, por lo que se debe gestionar la cola de trabajos de impresión adecuadamente.

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

Por tanto, y de manera general, los procesos que “no son propios” de dicho sistema operativo son penalizados. Es decir, si en un sistema operativo de tendencia interactiva se lanza un conjunto de tareas interactivas y batch, las segundas serán penalizadas, en cuanto a sus tiempos de ejecución.

Existen SO más orientados a uno u otro tipo de proceso, puesto que la eficiencia de estos se planifica desde el diseño de los mismos:

- **Sistemas operativos en tiempo real:** donde se deben cumplir escrupulosamente los plazos de ejecución de los procesos y, además, deben tener un comportamiento predecible. Ejemplos: en aviónica, instrumentación médica, sistemas de alertas en una central nuclear, etc.
- **Sistemas operativos interactivos o de tiempo compartido:** orientados a la participación continua del usuario, los cuales hacen uso de los programas antes comentados, tales como un procesador de textos o un editor de imágenes. Son sistemas de propósito general en los que, a diferencia de los sistemas de tiempo real, no priman los tiempos de respuesta en la ejecución de procesos (Windows, Apple/Mac OS, Linux, ..).

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 despleables 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 un 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

Podríamos definir el concepto de Sistema Operativo como un programa, o un conjunto de programas que colaboran entre ellos para administrar los elementos físicos de un sistema informático, optimizando su uso y ofreciendo determinados servicios a los programas de aplicación.

Un sistema operativo se encargará de aspectos como:

- El uso, compartido y ordenado, de los recursos entre diferentes usuarios.
- La protección de recursos, para evitar que un usuario acceda a recursos para los que no está autorizado.

Arquitectura de los SO

Para que esta protección sea posible, el sistema informático debe ser capaz de ejecutar instrucciones en dos niveles diferentes.

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

EJEMPLOS:

- ***Navegador web:*** Cuando estás utilizando un navegador web para buscar información en internet, estás interactuando en modo usuario. Puedes hacer clic en enlaces, leer contenido y enviar formularios. Sin embargo, el navegador no puede acceder directamente al hardware de tu computadora ni realizar cambios en la configuración del sistema. Está limitado a las operaciones permitidas para un programa de usuario.
- ***Procesador de texto:*** Cuando escribes un documento en un procesador de texto como Microsoft Word o Google Docs, estás trabajando en modo usuario. Puedes escribir, formatear texto y guardar archivos en tu disco duro. El procesador de texto no puede manipular el hardware de la computadora directamente ni realizar operaciones que afecten a otros programas en ejecución.

Arquitectura de los SO

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.

EJEMPLOS:

- ***Actualizaciones del sistema operativo:*** Cuando instalas actualizaciones del sistema operativo en tu computadora, el proceso de actualización opera en modo núcleo. El sistema operativo tiene que realizar cambios significativos en la configuración del sistema y actualizar archivos del sistema, lo que requiere privilegios elevados. Estas operaciones solo pueden ser realizadas en modo núcleo para garantizar la integridad y seguridad del sistema.
- ***Controladores de dispositivos:*** Los controladores de dispositivos son programas que permiten que el sistema operativo interactúe con hardware específico, como impresoras o tarjetas de video. Estos controladores operan en modo núcleo, ya que necesitan acceder directamente al hardware y controlar sus funciones. Por ejemplo, un controlador de impresora en modo núcleo facilita la comunicación entre el sistema operativo y la impresora, permitiendo que los documentos se impriman correctamente.

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.

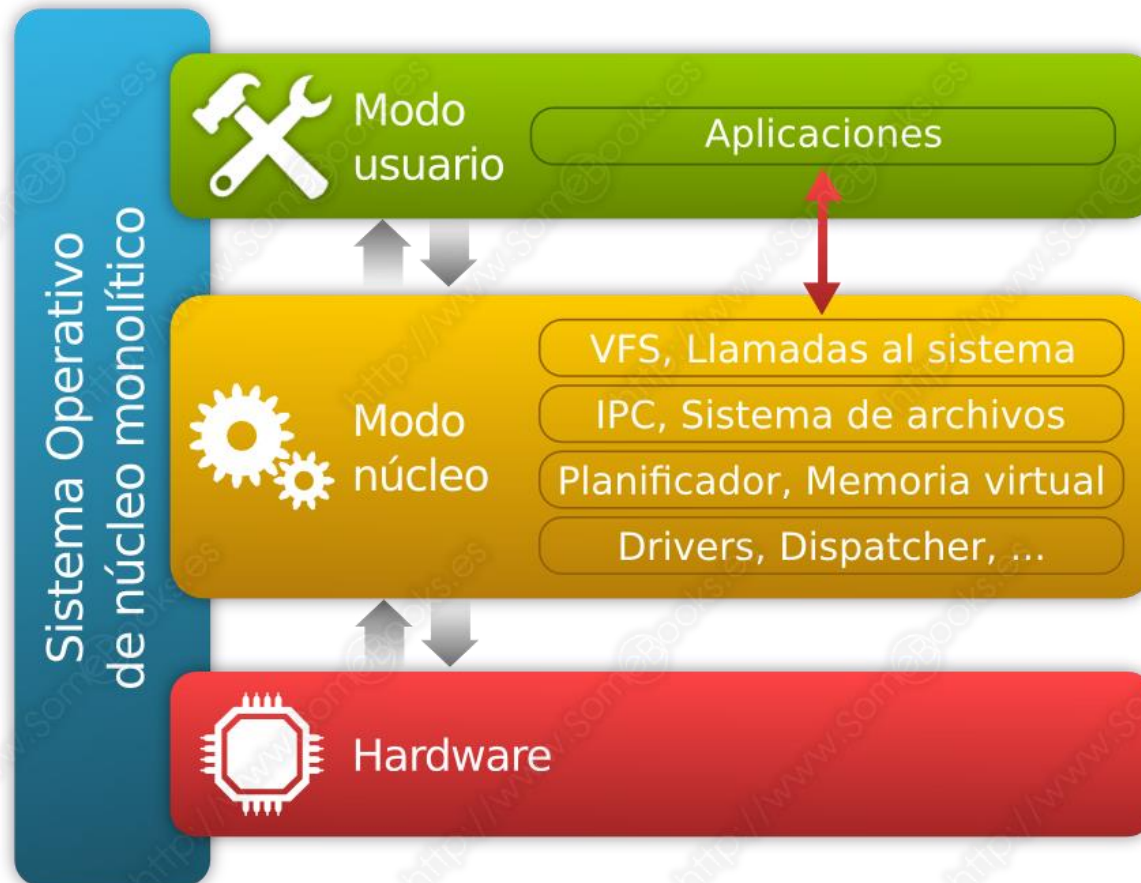
SO Monolíticos

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.

Los programas de usuario, por otro lado, se ejecutan en un espacio de memoria separado y en modo usuario. Estos programas de usuario tienen restricciones de acceso y no pueden realizar operaciones directas en el hardware o interferir con el funcionamiento del sistema operativo. Si un programa de usuario necesita realizar una operación que requiere privilegios elevados, debe hacer una llamada al sistema para que el kernel del sistema operativo ejecute esa operación en su nombre.

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. Las aplicaciones deben realizar llamadas al sistema para ejecutar tareas que requieran permisos elevados.

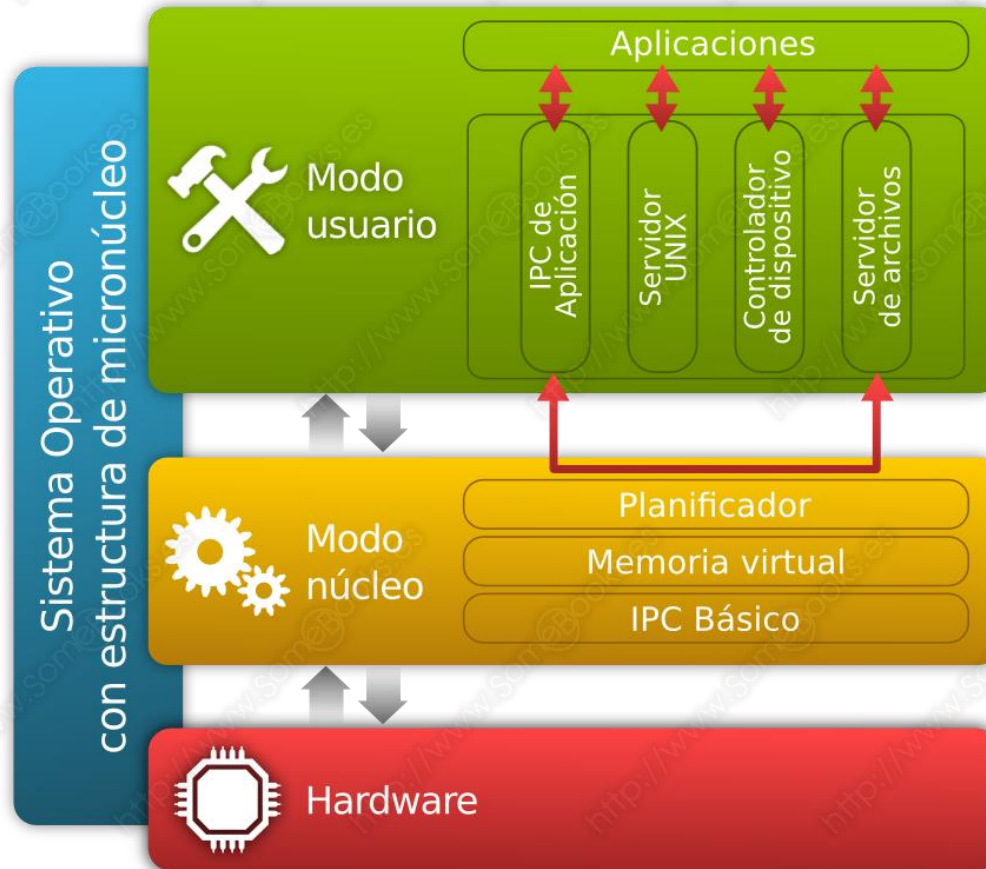
SO MicroKernel

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



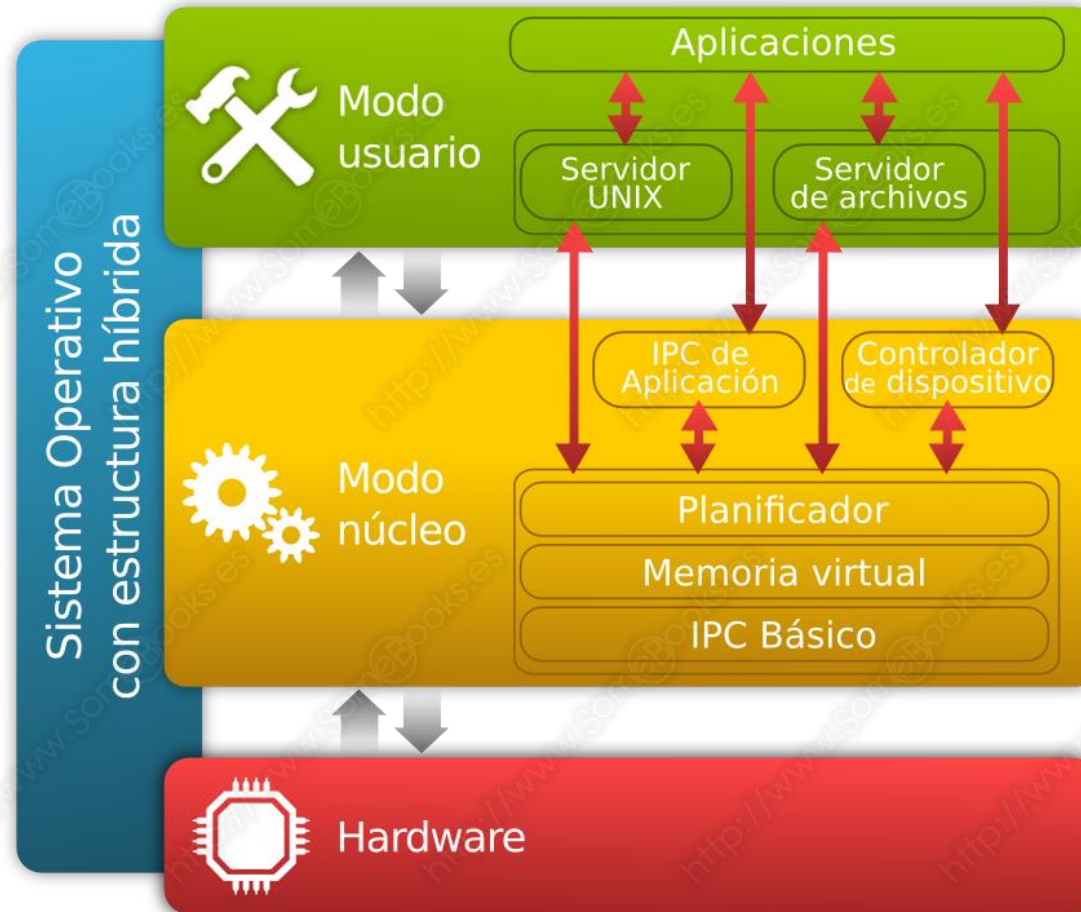
Los servicios o aplicaciones si se quieren comunicar con el Hardware, deben enviar peticiones al Kernel a modo de “mensajes” IPC (mensajes de interproceso). Será el Kernel el que gestione esa comunicación

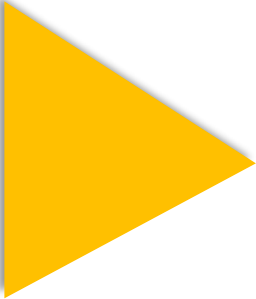
SO Kernel Híbrido

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=QhZi9r0PStI>

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, y sobre todo, por su medio de interacción táctil, y la facilidad de desarrollar aplicaciones para estas plataformas.

Lo que realmente quiere decir esto, es que muchos de los sistemas operativos incorporan el kernel de Linux.

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 o LINUX, denominan a sus diferentes sistemas operativos principales como DISTRIBUCIONES (se basan directamente en UNIX/LINUX), o se nombran como BASADO EN (cuando se basa en una distribución concreta) , o DE TIPO LINUX/UNIX (si su Kernel se basa en ese tipo)

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 (Open Source)

Fue desarrollado por Bell en 1969, y cuyo último lanzamiento fue realizado en 1979 en su versión V7. Desarrolló su propio kernel (kernel UNIX) monolítico. Estos sistemas operativos son multiplataforma, multiusuario y multitarea.

Fueron diseñados principalmente para servidores.

Es de código abierto, y su modo de interactuar con el usuario es por línea de comandos. En la práctica, el término UNIX se utiliza en sistemas operativos GNU/Linux, macOS X, FreeBSD, OpenBSD,...

Todos estos sistemas operativos incorporan el kernel de UNIX, pero no todos son UNIX como tal. Para ello deben alcanzar una certificación. Por ejemplo, FreeBSD es un UNIX como tal, pero Linux no es considerado UNIX, sino de tipo UNIX, ya que no dispone de la certificación.

GNU/Linux (Open Source)

Solemos utilizar el termino Linux para hacer referencia a un sistema operativo, pero el realidad Linux es el nombre del kernel de estos sistemas operativos. Originalmente pretendía reemplazar el núcleo de UNIX.

Fue iniciado en 1991 como proyecto personal del estudiante finlandés Linus Torvals, con el objetivo de crear un sistema operativo libre.

Estos SO son de tipo UNIX (no son UNIX, sólo parte de su kernel). Aún compartiendo parte del kernel de UNIX, Linux posee su propio kernel (kernel de Linux). Tendríamos el ejemplo de Android, que posee el núcleo Linux.

Las versiones de Linux se denominan Distribuciones. También existen diferentes e innumerables sistemas operativos basados en Linux, es decir, que derivan de distribuciones Linux, siendo las más conocidas:

Debian – Ubuntu (deriva de Debian) – Linux Lite; Kubuntu (Derivan de Ubuntu)

RedHat / Fedora

CentOS

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

La mayoría de estas distribuciones incorporan o pueden incorporar GUI (instalando el paquete adecuado). Las distribuciones con GUI, que intentan ser una alternativa competitiva a Windows para versiones de escritorio son: Ubuntu y Linux Lite.

Microsoft Windows

La primera versión fue lanzada en 1985 (Windows 1.0). En 1990 fue cuando se popularizó en todos los hogares, gracias a su interfaz gráfica de usuario en su versión Windows 3.0.

Originalmente, su Kernel estaba basado en MSDOS, para después derivar en el Kernel que se denominaría NT (a partir de Windows NT). Desde entonces, la empresa ha ido ofreciendo diferentes versiones de sistemas operativos, enfocados principalmente a hogares, empresas/oficina, servidores, manteniendo el Kernel de NT como base para su desarrollo.

Actualmente este SO se encuentra en su versión de Windows 11 para hogar y empresa, siendo sus ediciones más usadas:

Hogar: Home, Education, Pro

Empresa: Pro, Enterprise, Workstations

En servidores este SO es actualmente Windows Server 2022, disponiendo de ediciones tales como: Datacenter (virtualizado), Standard, Essentials.

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

macOS

Inicialmente conocido por macOS X, y desarrollado en 1984 por Apple Computer para su máquina Macintosh, es un sistema operativo con Kernel propio XNU (Híbrido) empleado en máquinas y dispositivos del fabricante Apple. Su primer Kernel implementaba UNIX, tal es el caso, que es considerado un sistema operativo UNIX.

La mayoría de los dispositivos que incorporan macOS son equipos de escritorio y portátiles, pero también dispone de versiones enfocadas a servidores, como es el caso de macOS Server.

Estos sistemas operativos vienen preinstalados en los equipos Apple. A pesar de que parte del sistema está disponible como código abierto, no está considerado SO de código abierto.