

UT3.1: Clasificación, evolución y funciones de un SO



Definición Sistema Operativo



El **Sistema Operativo (SO)** es el software básico de cualquier ordenador o computadora. Este software gestiona los recursos hardware del sistema informático y proporciona una base para la creación y ejecución del software de diferentes aplicaciones e interacción con el usuario.

Gracias al Sistema Operativo, el hardware se identifica, se reconoce y el Sistema Informático empieza a funcionar.

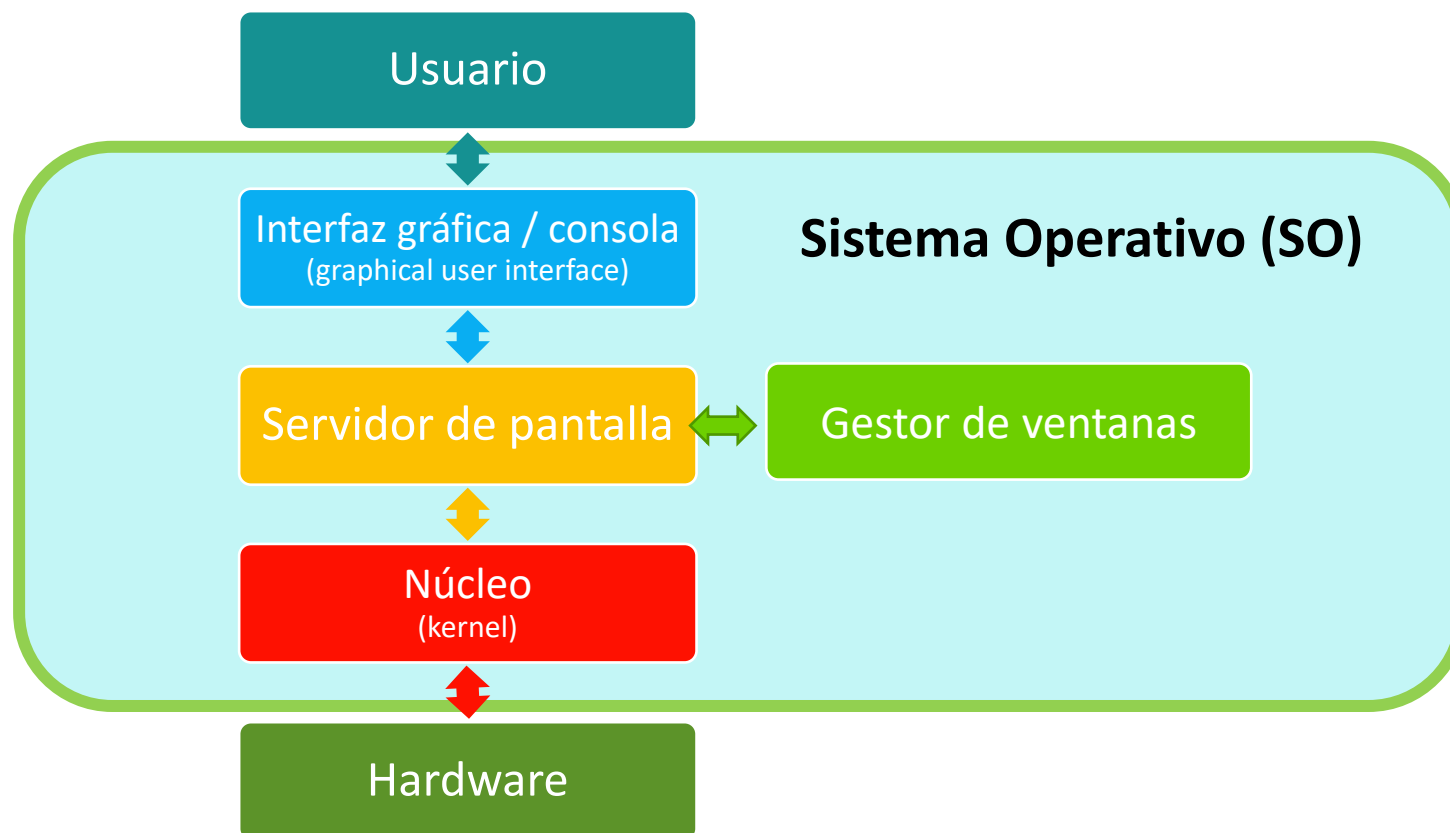
Gracias a los programas y aplicaciones del propio Sistema Operativo (SO), el usuario podrá realizar determinadas funciones específicas.



Definición Sistema Operativo



El **Sistema Operativo** ofrece al usuario la forma de comunicarse con el ordenador, bien mediante el teclado (*interfaz texto*), bien mediante otros dispositivos como el ratón, la pantalla táctil, etc. (*interfaz gráfica*). El SO es el medio que utilizan los usuarios para realizar sus funciones en cualquier sistema informático.



Clasificación de los Sistemas Operativos



Clasificación de los Sistemas Operativos



Se pueden clasificar los SO teniendo en cuenta la gestión que hacen del software y el hardware, y la forma en que el usuario los puede utilizar:

1. **Sistemas operativos monopuesto (SOMO):** Un solo usuario utiliza a la vez el Sistema Operativo y sus recursos. Se denominan también de escritorio.
2. **Sistemas operativos multiusuario (SOMU):** Permiten que varios usuarios puedan ejecutar a la vez distintos programas y hacer uso de los recursos de una máquina al mismo tiempo.
3. **Sistemas operativos en Red (SORED):** Son aquellos que permite la interconexión de ordenadores a través de una Red para poder acceder a servicios, recursos, hardware y software.
4. **Sistemas operativos distribuidos (SOD):** administran una colección de nodos distribuidos que trabajan en red, se comunican entre ellos y comparten recursos
5. **Sistemas operativos en tiempo real:** Son sistemas orientados a la respuesta en tiempo real, usados en sistemas críticos o cuando el tiempo de respuesta debe ser inmediato.

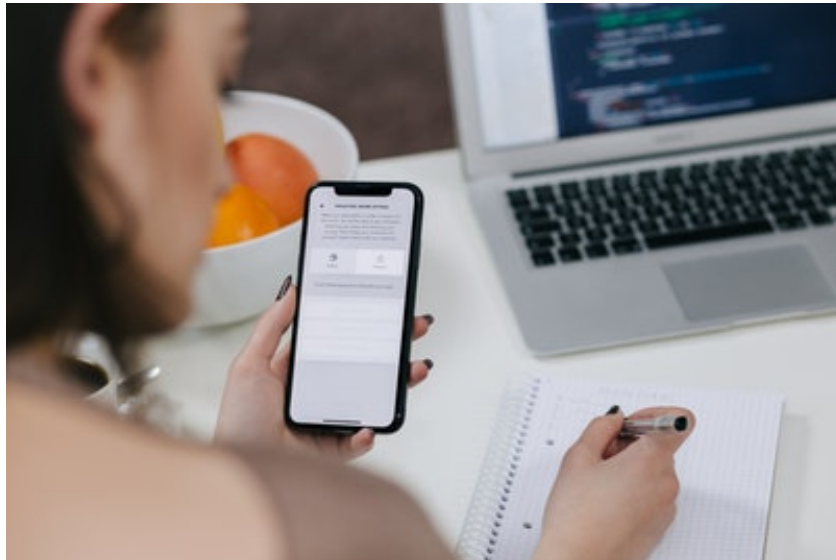
Clasificación de los Sistemas Operativos



Sistemas Operativos Monopuesto

Los Sistemas Operativos Monopuesto (SOM) o de escritorio, son aquellos que se instalan para un uso personal o particular o en estaciones de trabajo. Solo un usuario utiliza dicho equipo en un momento determinado y tiene todos los recursos (tanto hardware como software) disponibles para él mismo.

Los usuarios que inician sesión en estos equipos se denominarán usuarios locales y están dados de alta en el mismo equipo que utiliza dicho usuario.



Clasificación de los Sistemas Operativos



Sistemas Operativos Multiusuario

Un Sistema Operativo Multiusuario (SOMU) permite acceder de manera simultánea a más de un usuario a la vez en un mismo equipo. Por tanto, dicho SO será capaz de proveer de servicio y procesamiento a múltiples usuarios a la vez. En los SOMU existe un equipo principal. A dicho equipo podrían estar conectados varios terminales, en los que los usuarios realizan su trabajo.

Existe un solo conjunto de memoria RAM y uno o varios discos duros dentro del mismo equipo, etc.



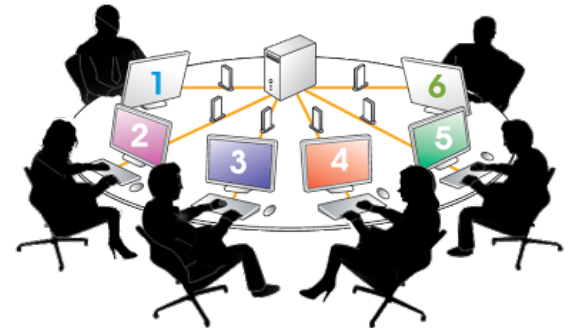
Clasificación de los Sistemas Operativos



Sistemas Operativos Multiusuario

Supongamos que hay cinco usuarios utilizando este sistema y que cada uno ejecuta un programa distinto. El SO deberá trabajar mucho, ya que tendrá que ubicar en memoria cinco programas diferentes, e ir ejecutando rotativamente, en principio, instrucciones de cada uno de los programas para que a todos los usuarios se les dé una respuesta en el menor tiempo posible. La velocidad de respuesta por tanto será inferior a la que daría un SOMO o un sólo usuario.

De esta forma, todo se procesa en un ordenador principal. Primero se ejecutan unas instrucciones del primer programa lanzado por el primer usuario. Acto seguido, este programa se queda detenido para dar un poco de tiempo en la CPU al proceso lanzado por el segundo usuario. Este proceso se repetirá secuencialmente para dar servicio a todos los programas de todos los usuarios.

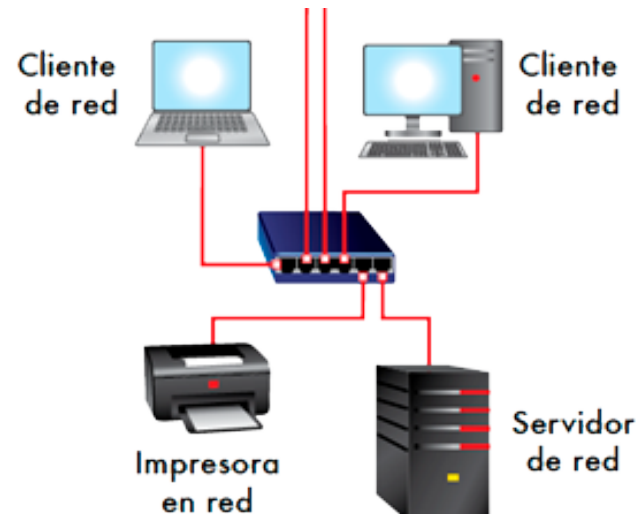


Clasificación de los Sistemas Operativos



Sistemas Operativos en Red

En los SO en red (*SORED*) cada usuario tendrá su propio equipo personal con un SO, normalmente monousuario. Cada usuario encenderá su ordenador y establecerá comunicación con el ordenador principal que dispone del SORED. Si partiéramos del mismo ejemplo anterior con cuatro usuarios ejecutando cada uno un programa diferente, tendríamos de cinco CPUs, cinco bloques de memoria independientes, cinco o más dispositivos de almacenamiento, etc.



Clasificación de los Sistemas Operativos



Sistemas Operativos Distribuidos

Un Sistema Operativo Distribuido (SOD) se define como una colección de equipos informáticos separados físicamente y conectados entre sí por una red de comunicaciones distribuida; cada máquina posee sus componentes de hardware y software de modo que el usuario percibe que existe un solo sistema (no necesita saber qué cosas están en qué máquinas). El usuario accede a los recursos remotos de la misma manera en que accede a recursos locales ya que no percibe que existan varios ordenadores, sino que solo es capaz de ver uno formado por todos los anteriores.

Una ventaja fundamental de los sistemas distribuidos es que permiten aumentar la potencia del sistema informático, de modo que 100 ordenadores trabajando en conjunto, permiten formar un único ordenador que sería 100 veces más potente.



Clasificación de los Sistemas Operativos



Sistemas Operativos en tiempo real

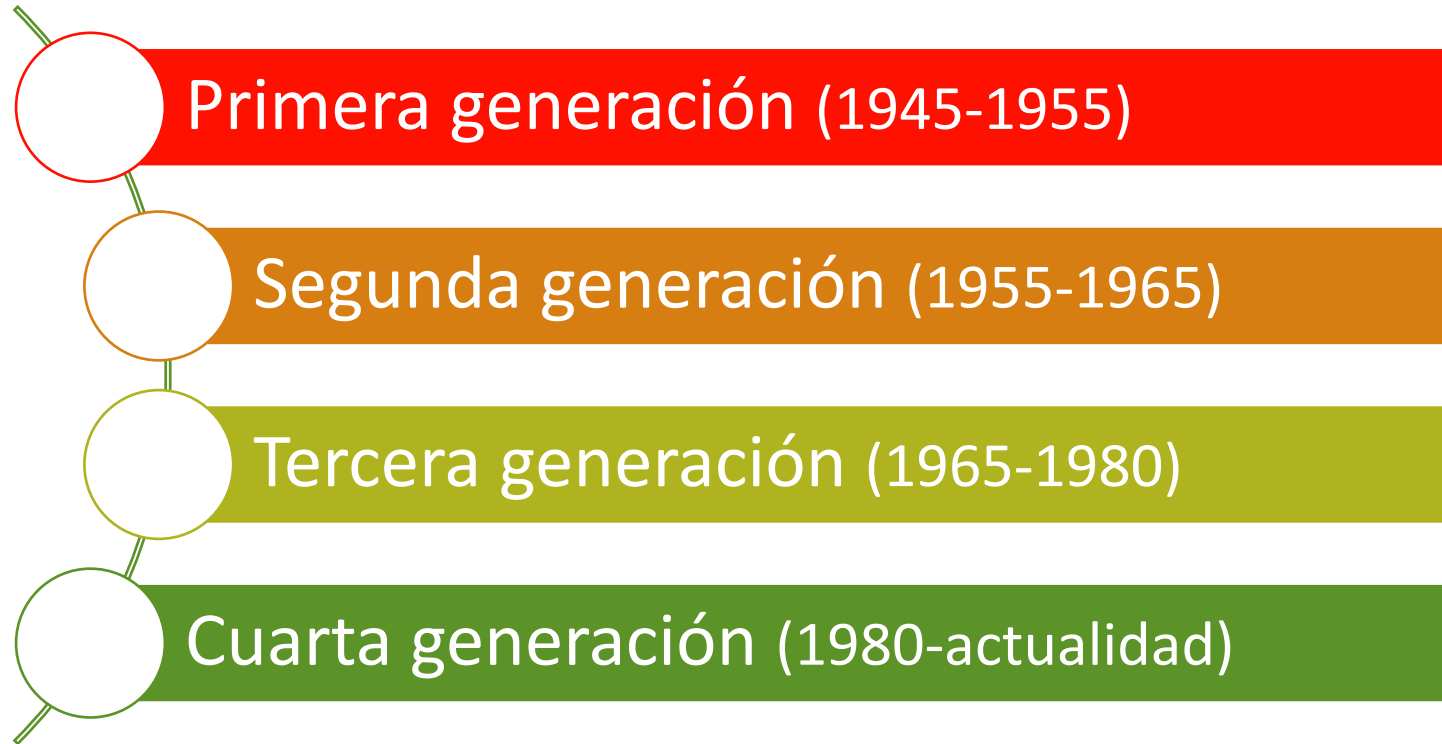
Son sistemas en los que los procesos requieren un tiempo de respuesta prácticamente inmediato. Se utilizan en sistemas críticos, en los que un tiempo de respuesta alto podría tener serias consecuencias, como sistemas médicos, sistemas aeroespaciales o en centrales energéticas.



Evolución histórica



En general, podemos hablar de cuatro **generaciones** de sistemas operativos, relacionadas siempre con la evolución del hardware:



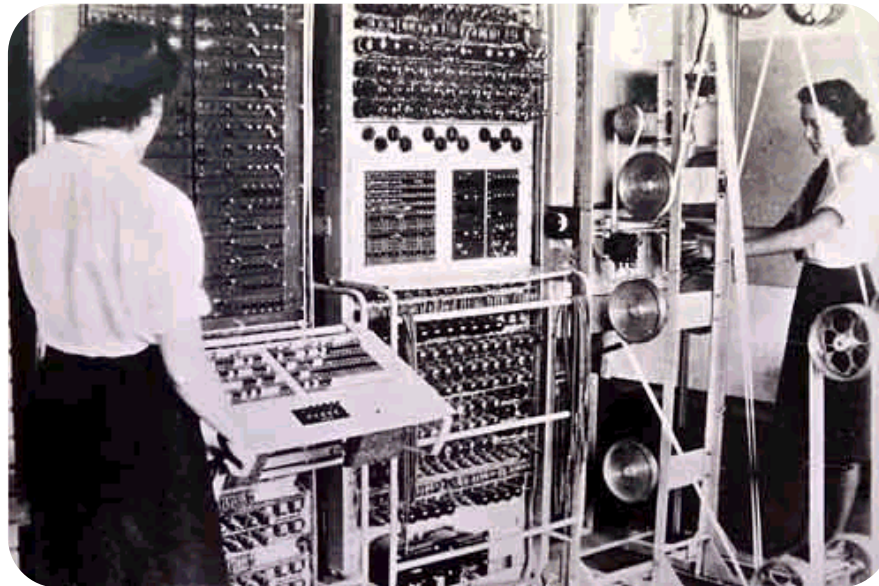
Los primeros Sistemas Operativos se denominaban **monolíticos**. La característica fundamental de estos SO es que su software básico era prácticamente imposible de modificar una vez creado e instalado en un sistema informático.

Evolución histórica



Primera generación (1945-1955)

En esta generación los ordenadores utilizaban las **válvulas de vacío** y tableros enchufables. Eran máquinas de gran tamaño de diseño único, programadas en lenguaje de máquina puro y complejas interconexiones con sus válvulas. Llegaron de la mano de diseños de Howard Aiken y Jon Von Neumann. Sus operaciones se reducían a cálculos y operaciones matemáticas previamente programadas y no poseían sistema operativo como tal (programadas a mano)



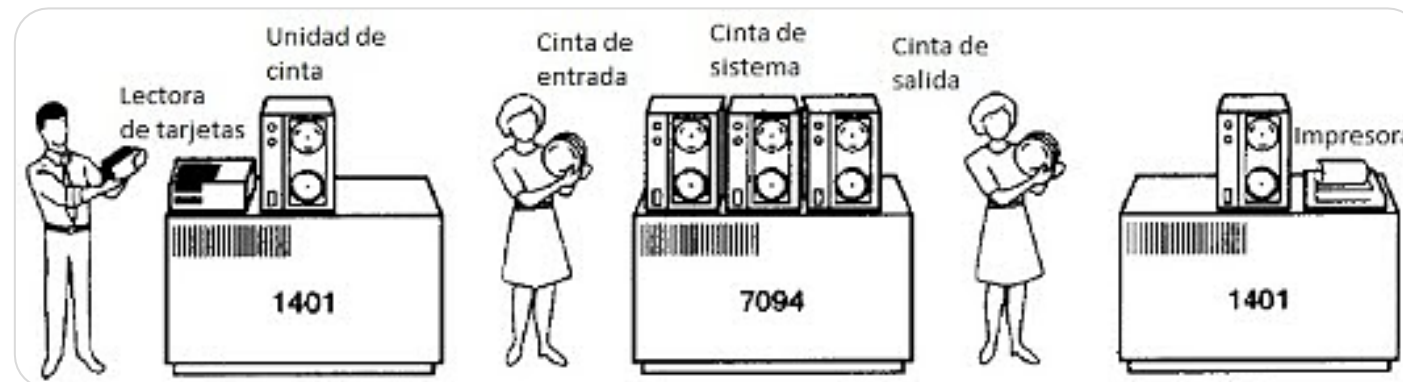
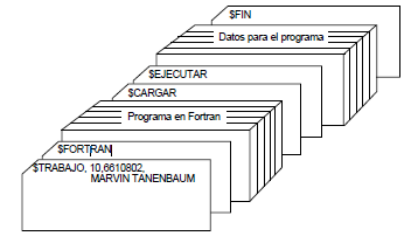
Evolución histórica



Segunda generación (1955-1965)

Se produce con la aparición de los **transistores** lo cual reduce el tamaño de los ordenadores aumentando su eficiencia. En esta generación aparece el **procesamiento por lotes** que constaba de tres fases:

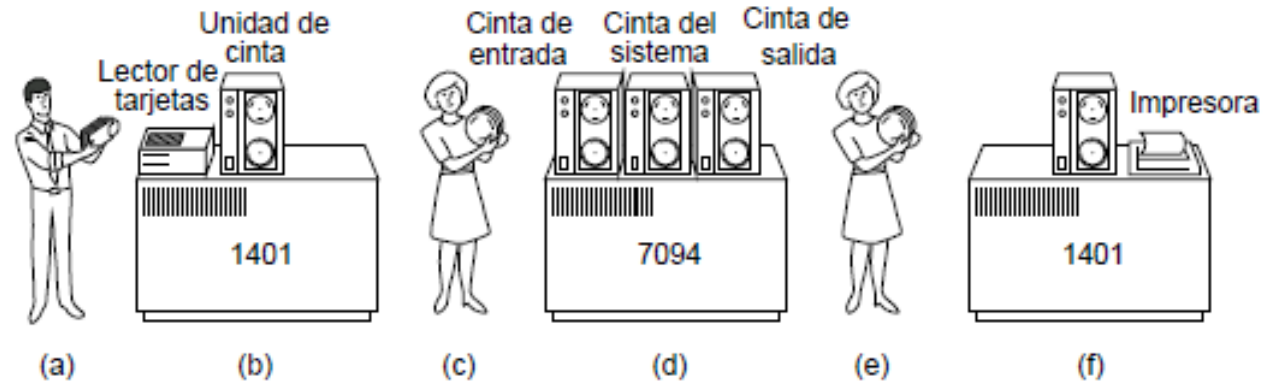
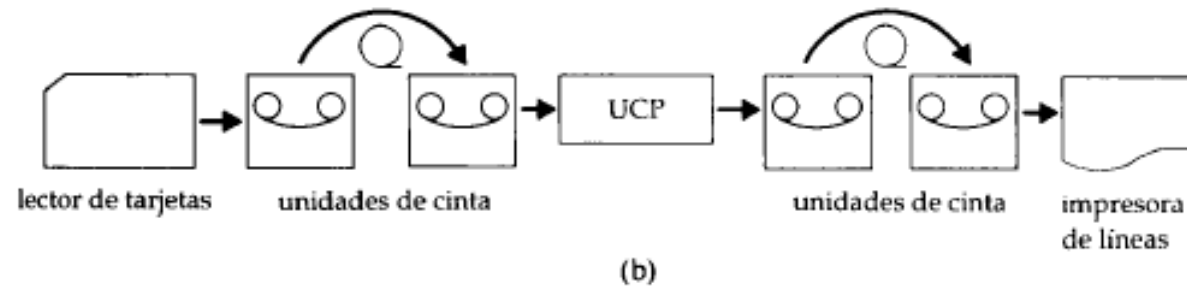
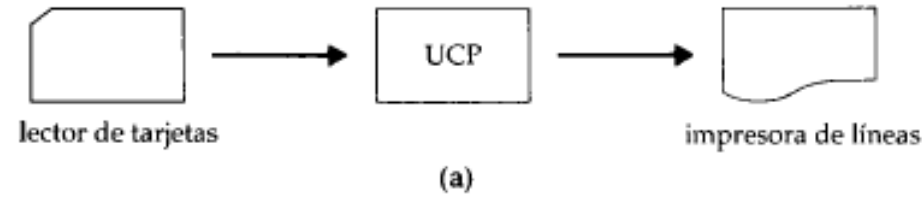
1. Se introducían datos en las tarjetas perforadas, cinta magnética o soportes magnéticos.
2. Se transportaba e introducía el soporte con datos en la computadora para que los procesara devolviendo resultados en otro soporte.
3. Se lleva ese soporte a otro distinto a la computadora para la generación de los resultados.



Evolución histórica



Segunda generación (1955-1965)



Evolución histórica



Segunda generación (1955-1965)

Estas aún grandes computadoras (solían ocupar una habitación) se usaban primordialmente para cálculos científicos y de ingeniería, como la resolución de ecuaciones diferenciales parciales. Generalmente se programaban en lenguaje FORTRAN o en lenguaje ensamblador.

Aparecen los primeros sistemas operativos típicos como **FMS** (Fortran Monitor System) e **IBSYS**, el sistema operativo de IBM para la 7094



Evolución histórica



Tercera generación (1965-1980)

La aparición de los **circuitos integrados** y de la **multiprogramación** supuso toda una revolución.

Multiprogramación

- Varios trabajos en memoria: cuando uno espera para E/S, otro se puede ejecutar
- Necesidad de planificación y protección

Tiempo compartido (multitarea)

- Variante del anterior
- Cambio rápido entre tareas: uso interactivo
- Más complejo



Evolución histórica



Tercera generación (1965-1980)

Se redujo el tamaño y consumo de energía de los ordenadores gracias a la sustitución de transistores por **circuitos integrados**, aún más baratos, rápidos y fiables.

En esta generación destaca el IBM 360 como una máquina capaz de realizar cualquier tipo de cálculo, con su sistema operativo denominado **OS/360**.



Evolución histórica



Tercera generación (1965-1980)

MIT, Bell laboratories y General Electric favorecen el desarrollo de la “computadora de servicio publico” así como el Sistema Operativo para ella **MULTICS** (*Multiplexed information and computing service*)

Se produce un gran crecimiento de las **minicomputadoras**.

Ken Thompson, empezó a escribir un nuevo Sistema Operativo conocido como **UNICS** (*Uniplexed information and computing service*) su ortografía cambió más tarde por **UNIX**.



Evolución histórica



Cuarta generación (1980-actualidad)

Se produce la aparición de las **computadoras personales**. Se utilizan complejas técnicas de miniaturización de componentes electrónicos.

Se produce la aparición de los sistemas operativos modernos que conocemos en la actualidad. Mucho más fáciles de utilizar y con la creación de interfaces gráficas para una mejor comunicación entre máquina y usuario.



Evolución histórica



Cuarta generación (1980-actualidad)

Los primeros ordenadores personales: Apple II y el IBM (PC):



Apple II (1978)



IBM PC (1981)

Evolución histórica



Cuarta generación (1980-actualidad)

La disponibilidad de una cada vez mayor potencia de cómputo, junto con una cada vez mayor potencia gráfica, y disponibilidad de discos de almacenamiento masivo en ordenadores personales, dio pie al crecimiento de la importante industria de software que conocemos hoy en día.

Los sistemas operativos que cobran gran popularidad inicialmente en este período (abuelos de los actuales) son **MS-DOS** y **UNIX**:

- **MS-DOS** de Microsoft fue el primer sistema operativo creado para ordenadores personales en la década de los 80 y su sucesor fue Windows (inicialmente se ejecutaba encima de MS-DOS). Se trata de un sistema operativo grabado en un disco, de modo que se encontraba permanentemente activo y residente en memoria
- **UNIX** dominaba las estaciones de trabajo y servidores en red, así como máquinas de alto rendimiento.

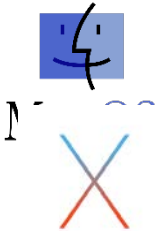
Evolución histórica



Cuarta generación (1980-actualidad)



- **MSDOS** fue creado como tal en 1981 después de que Microsoft comprara QDOS modificándolo y haciéndolo compatible con *IBM-PC*, que fue el primer ordenador personal.



- En 1984 Apple lanzaba su ordenador personal Macintosh, que incluía por primera vez su sistema operativo **MacOS**, cuyas características novedosas eran una GUI avanzada, multitarea y ratón. MacOS ha seguido evolucionando hasta la actualidad a través de las sucesivas versiones de **Mac OsX** basadas ya en UNIX y que van preinstaladas en los equipos de Apple.



- A mediados de los años 80 nace la primera versión de **Windows**, un mero gestor de ventanas que corre sobre MSDOS, pero no es hasta Windows 95 que se considera un SO separado completamente. La familia siguió creciendo con Windows 98, Me, XP, 7, 8, 10 y 11 así como las versiones de Windows del ámbito empresarial a partir de NT, 2000 y Windows Server.

Evolución histórica



Cuarta generación (1980-actualidad)



- Respecto a **Linux** nace en 1991 creado por Linus Towards e inspirado en Unix, usando una licencia **GNU** que permitía distribuir y modificar su kernel. Gracias a su comunidad ha evolucionado enormemente gracias a las distribuciones en las que se utiliza a entornos gráfico con gestores de ventanas como KDE, GNOME así como distintas distribuciones que se han ido haciendo famosas (Ubuntu, Debian, Suse, Fedora, Mint...)



iOS

- Entre 2003-2007 surgen los SO para smartphones, a partir del primer iPhone con su SO **iOS** y le sigue Google con su SO **Android**, basado en Linux, que en la actualidad es el más utilizado en el mundo. Dichos SO se han diversificado a otros dispositivos como tabletas, *wereables* e IOT.

En esta generación aparecen, por tanto, las distintas variantes de SO conocidas en la actualidad: los **multiusuario**, en **red**, **distribuidos**, **móviles** y en **tiempo real** para otros dispositivos.

Evolución histórica



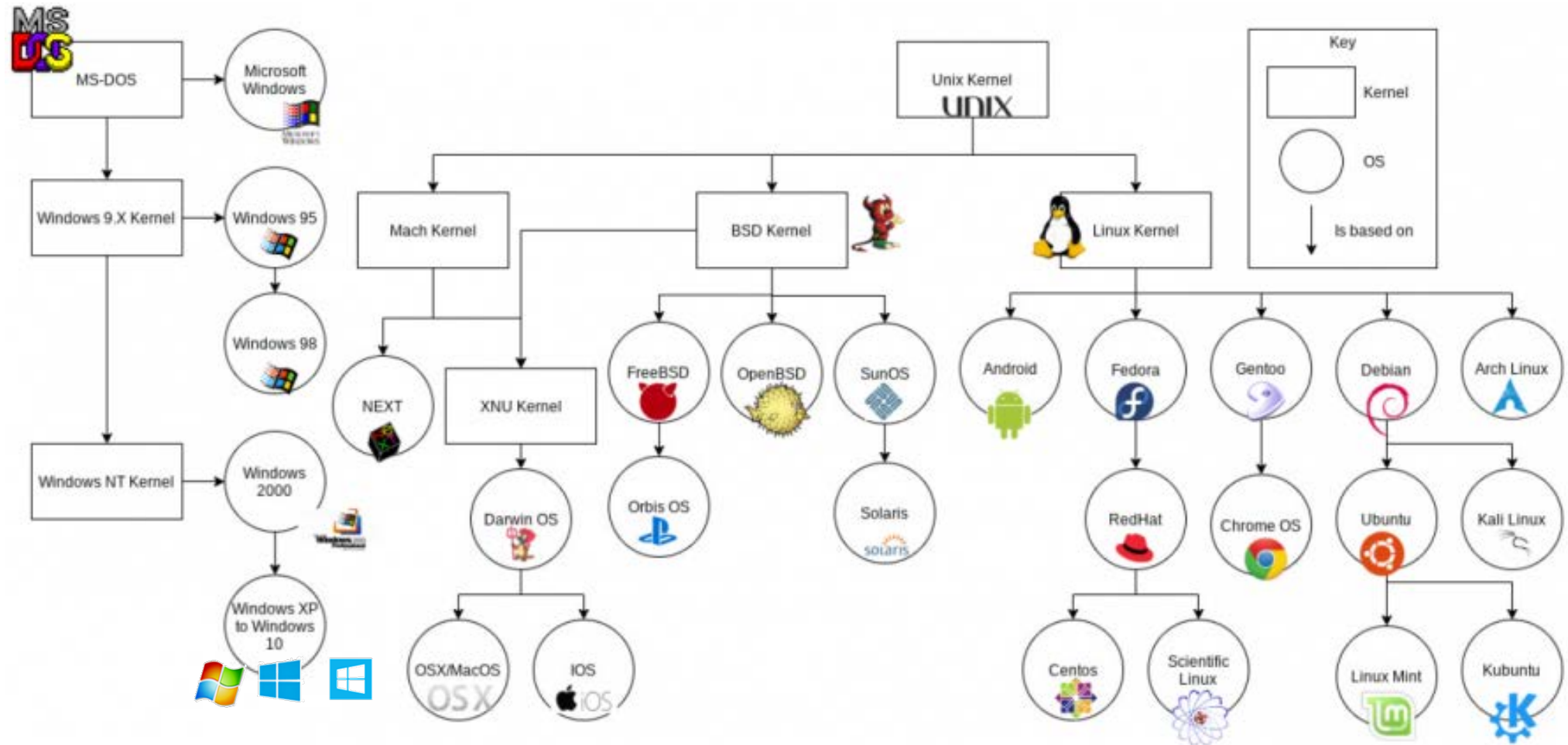
Cuarta generación (1980-actualidad)



Evolución histórica



Cuarta generación (1980-actualidad)



Funciones y arquitectura



El fin principal del Sistema Operativo es coordinar la utilización que se hace del hardware, dependiendo de los programas o aplicaciones que se estén utilizando.

Las partes de un SO se estructuran de la siguiente forma **jerárquica**:



La comunicación entre los distintos niveles se realiza mediante las llamadas **interfaces**.

Un mismo sistema operativo puede trabajar en múltiples plataformas hardware, por lo que debe poder adaptarse a cada una de ellas.

Funciones y arquitectura



Si dejamos al margen los sistemas operativos monolíticos, actualmente todos los sistemas operativos se organizan en **capas**.

En la actualidad, la mayoría de los sistemas operativos son sistemas operativos que admiten uno o varios usuarios y que pueden trabajar en multitarea.

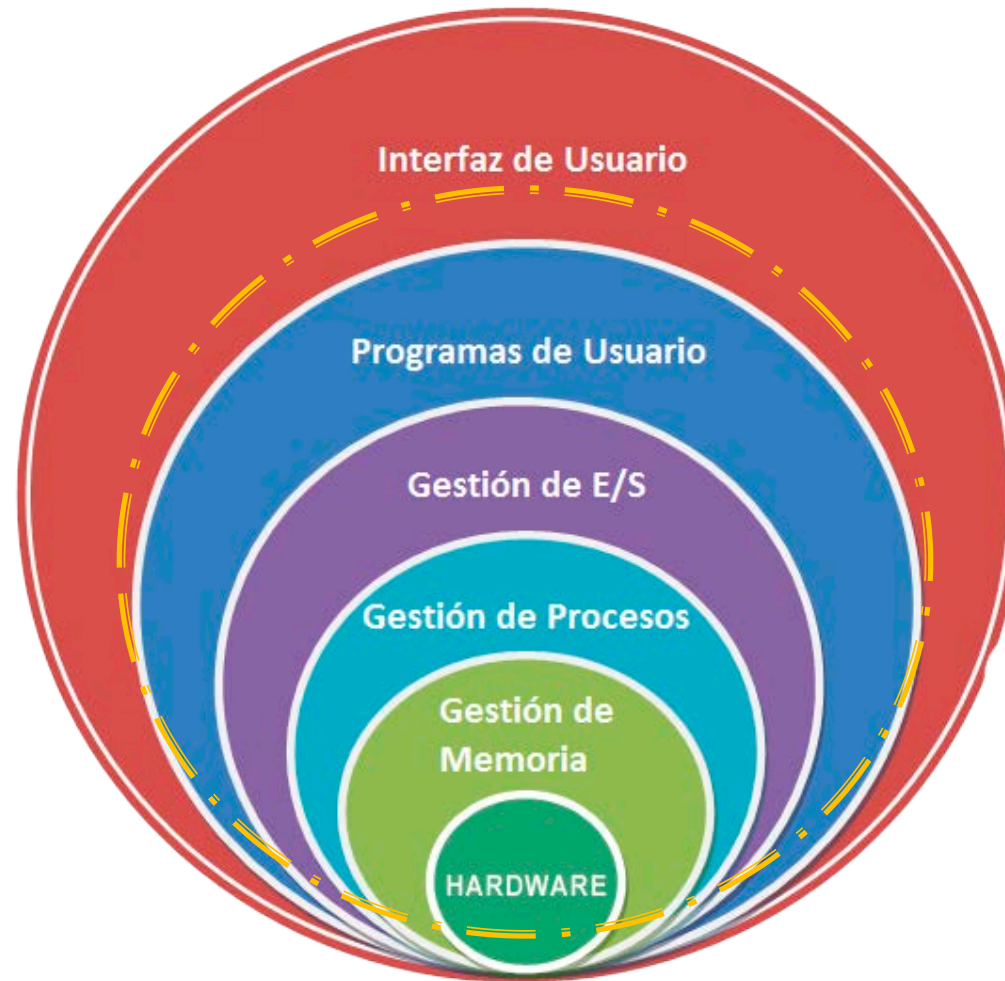
Estos sistemas operativos pueden funcionar en varias plataformas de hardware gracias a la abstracción de hardware que se hace (como Linux, Windows, Android)



Funciones y arquitectura



La **jerarquía o arquitectura** de los sistemas operativos (SO) que veremos en unidades posteriores puede verse de la siguiente forma ampliada:



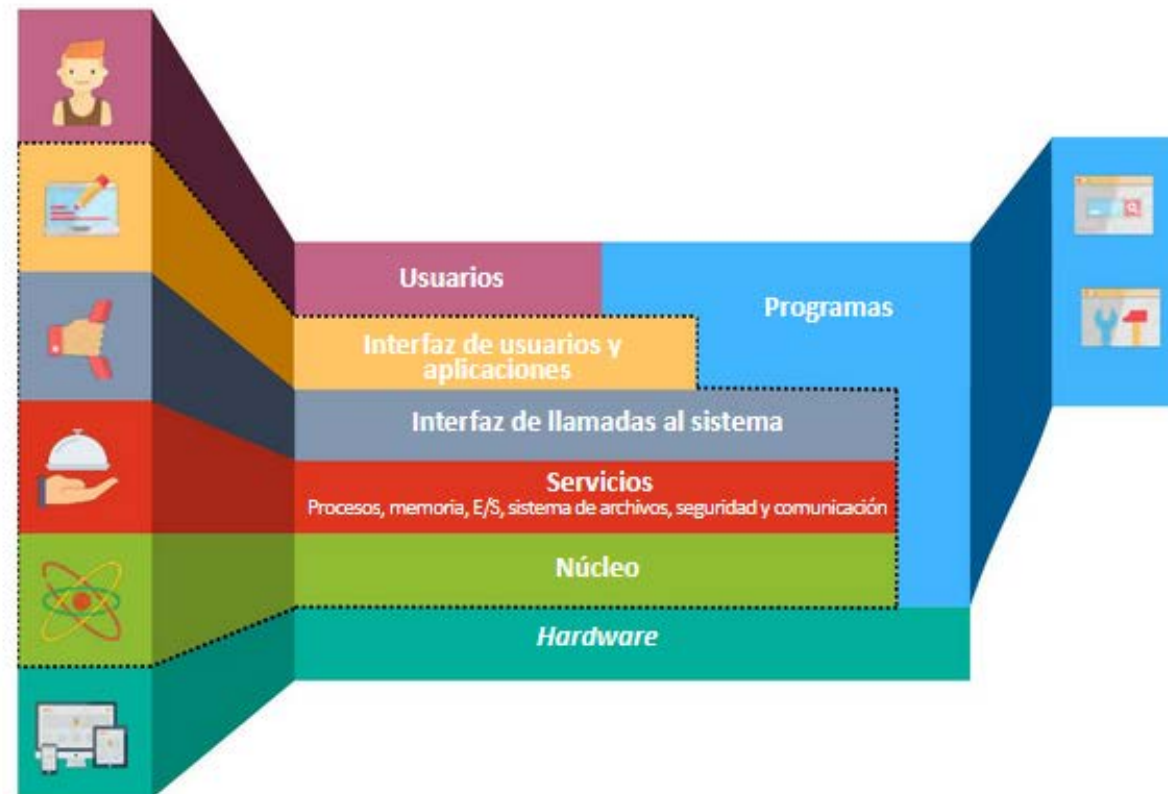
La comunicación entre niveles se lleva a cabo a través de **interfaces**.

Funciones y arquitectura



El sistema Operativo está formado por:

- El núcleo o kernel
- Los servicios
- La interfaz de usuarios y aplicaciones



Funciones y arquitectura



Las **funciones** principales que realiza todo sistema operativo son las siguientes:

Control y ejecución de programas

Administración de hardware, red y periféricos

Gestionar de ficheros, usuarios y permisos

Control de concurrencia y errores

Administración de memoria (procesos)

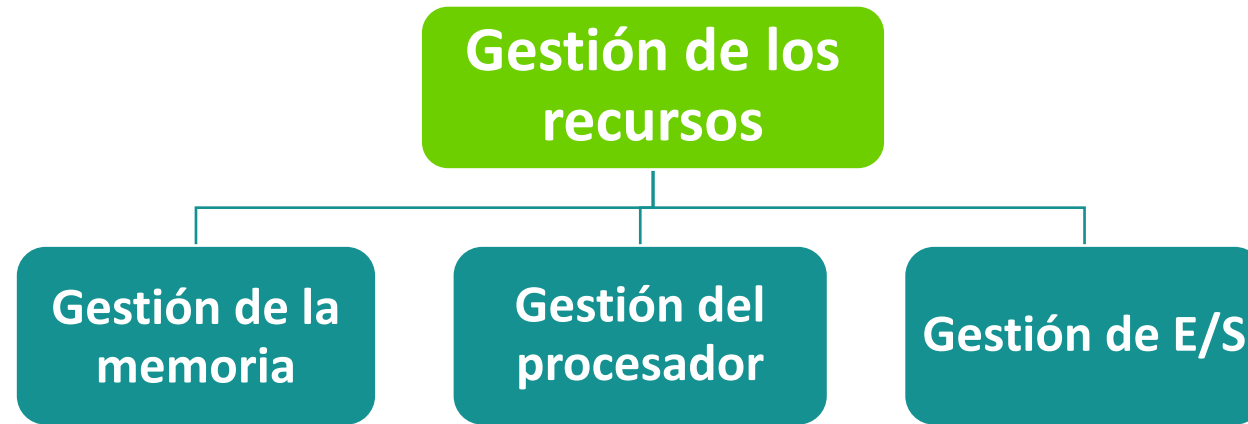
Control de seguridad

Funciones y arquitectura



- **Control y ejecución de los programas.** Para ello, acepta los trabajos, administra la manera en que se realizan, les asigna los recursos y los conserva hasta cerrarse.
- **Administración de hardware, red y periféricos (E/S).** Coordinando y manipulando los dispositivos conectados al ordenador.
- **Gestión de permisos de archivos y usuarios.** Adjudica los permisos de acceso a los usuarios y a los archivos de un ordenador.
- **Control de concurrencia y errores.** Establece prioridades cuando diferentes procesos solicitan un mismo recurso. Gestiona errores de hardware y de datos.
- **Administración de memoria.** Asigna memoria a los procesos y gestiona su uso.
- **Control de seguridad.** Proporciona seguridad tanto para los usuarios como para el software y la información almacenada en los sistemas.

Gestión de los recursos

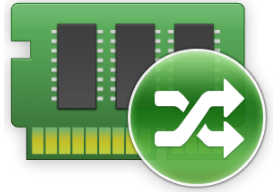


Gestión de los recursos



La memoria

La parte del sistema operativo que gestiona la memoria se denomina **administrador de memoria**.



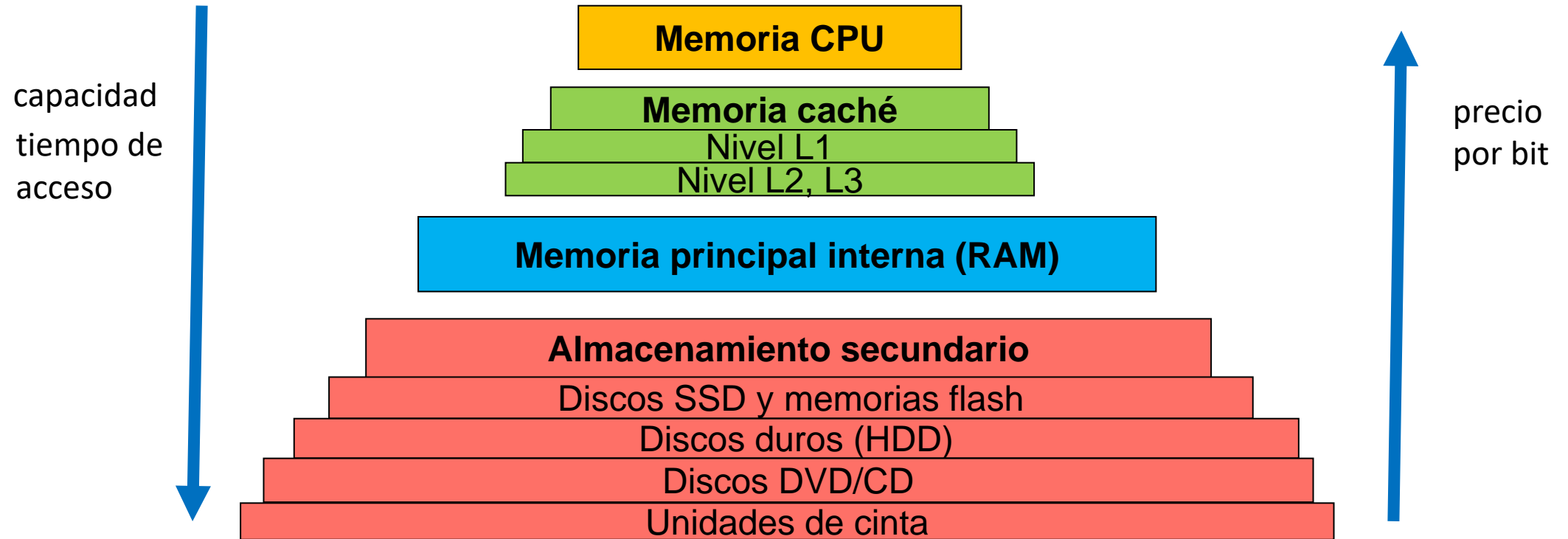
El **administrador de memoria** es una parte crítica del sistema operativo que se encarga de reservar espacio en la memoria para los nuevos procesos (programas o servicios) y liberarla cuando estos ya no se sigan utilizando así como el intercambio de datos entre memoria y disco duro.

La gestión de la memoria es importante cuando trabajamos en sistemas operativos multitarea (cualquier SO moderno) ya que trabajan con varios **procesos** a la vez.

Gestión de los recursos



La memoria



Gestión de los recursos



La memoria

El **administrador de memoria** proporcionará por tanto protección y **uso compartido**, es decir, debe proporcionar un espacio de memoria para cada proceso que lo necesite y controlar que ningún otro proceso trabaje en zonas de memoria que no le correspondan produciendo colisiones o inconsistencias en los datos.

Un error o colisión de zona de memoria provocará un fallo en cascada del SO provocando un cuelgue del sistema.

Por ejemplo, imagina un sistema de control de temperatura de una habitación con dos procesos. Uno de ellos se encarga de leer la temperatura de un sensor; el otro, a partir de dicha temperatura y del valor deseado, controla un generador de frío/calor. En función de estas condiciones, el primer proceso leerá la temperatura y la depositará en una posición de la memoria a la que el segundo proceso deberá tener también acceso.

Reservado
Sistema Operativo

Reservado Proceso 1

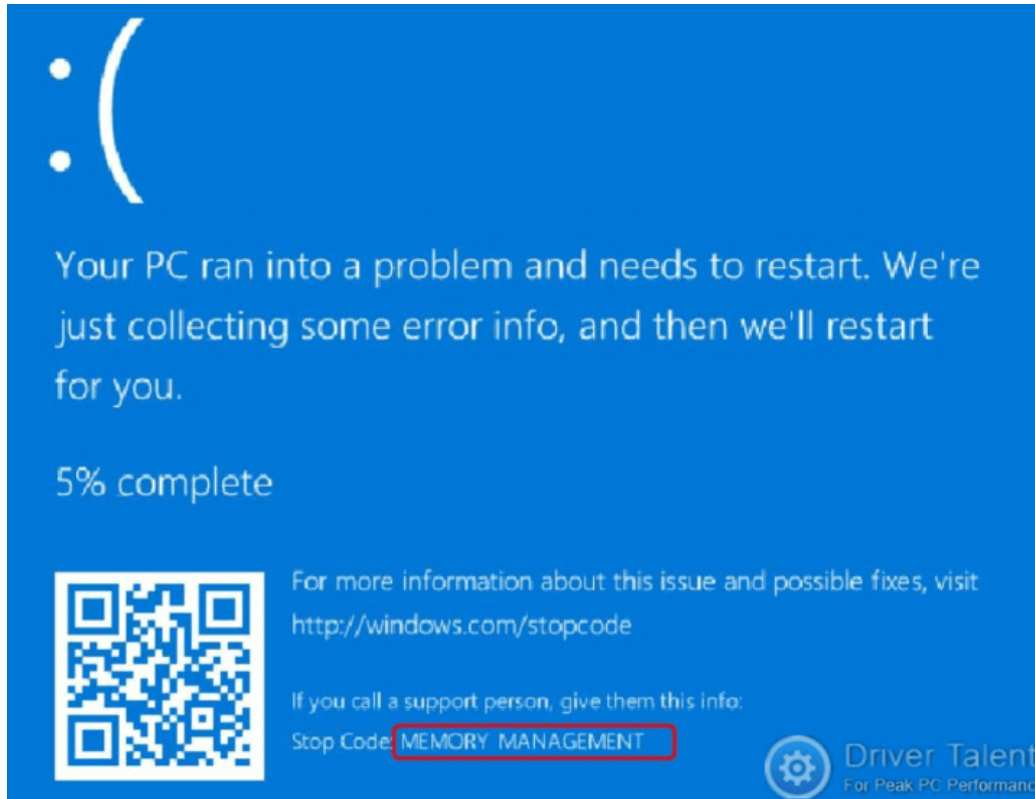
Reservado Proceso 2

Reservado Proceso 3

Gestión de los recursos



La memoria



BSOD (Blue Screen of The Dead) en Windows

Gestión de los recursos



El procesador

Un **proceso** tal y como vimos anteriormente consiste en un programa en ejecución gestionado en memoria por el administrador de memoria y que se va a mandar a ejecutar al **procesador** o CPU.

Cualquier programa que se esté ejecutando en un ordenador se convierte en un proceso, ya que desde ese momento el programa, denominado **proceso**, se puede **ejecutar**, se pueda **detener** o se puede **bloquear**, entre otras muchas cosas.

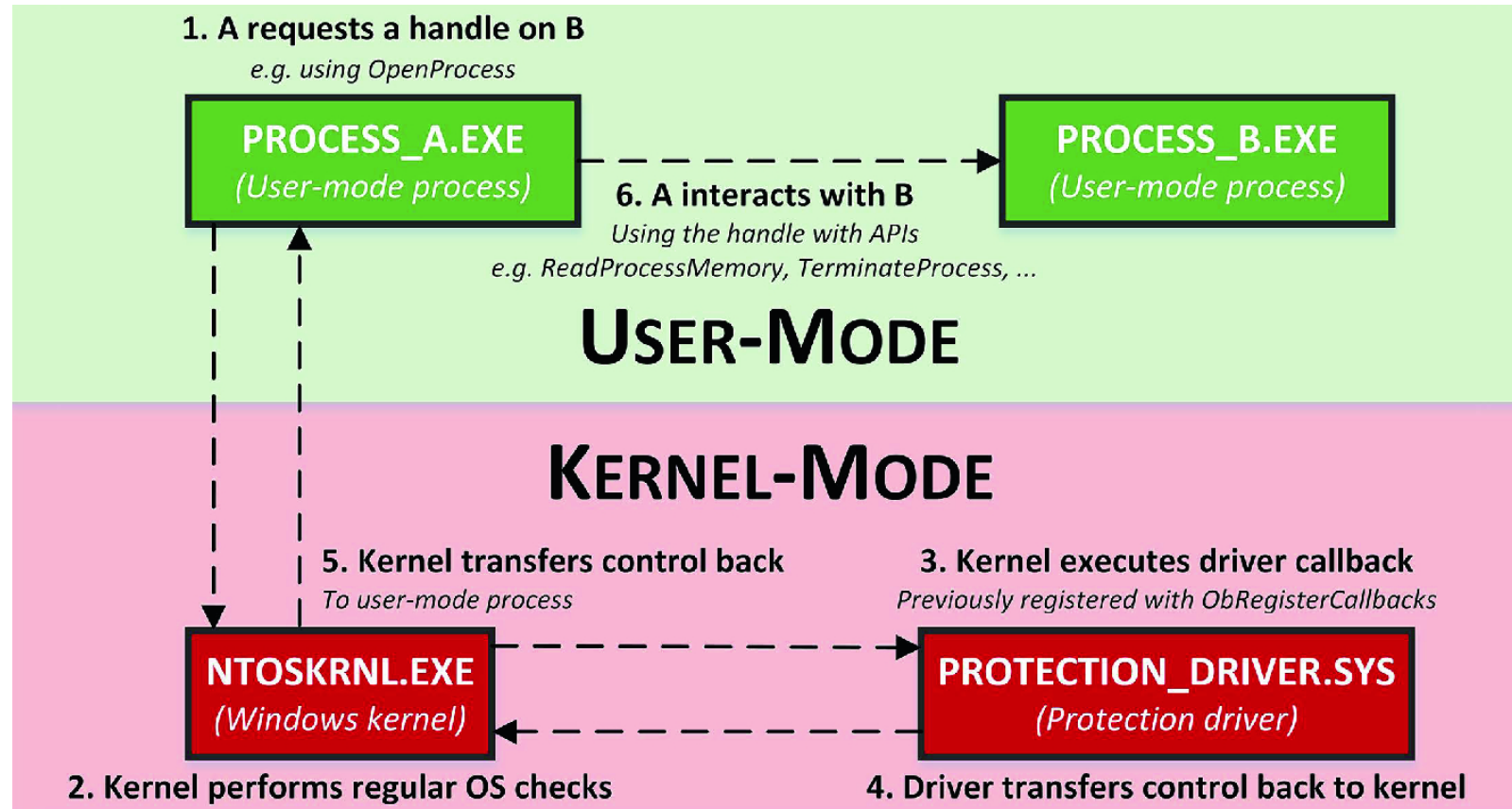
Existen dos tipos de procesos:

- **Procesos de usuario:** creados por cualquier usuario del sistema y los programas de aplicación instalados. Se ejecutan en modo usuario. Se tiene acceso limitado a los recursos del sistema.
- **Procesos del Sistema Operativo:** creados por el SO o sus servicios y que se ejecutan en modo privilegiado o **modo kernel**. Los procesos se ejecutan teniendo acceso directo a toda la memoria y recursos hardware de la máquina.

Gestión de los recursos



El procesador



Gestión de los recursos



El procesador



Gestión de los recursos



El procesador

Durante la ejecución de un proceso, este **compite** con el resto de los procesos que se están ejecutando de forma concurrente en el sistema por el uso de los recursos hardware (ya sea la CPU o la gráfica)

El reparto de los recursos del sistema entre los distintos procesos y su ejecución **concurrente** se conoce como **multiprogramación** y es gestionada en el **núcleo o kernel** del sistema operativo.

Un programa no es un proceso, sino que se convierte en tal en el momento en que se pone en **ejecución**. La aplicación WordPad, por ejemplo, cuando no se está ejecutando no deja de ser un archivo sin más. Y cuando se ejecuta, el archivo *WORDPAD.EXE* sigue estando almacenado donde estaba originalmente. Lo que ocurre es que al ejecutar el programa, las instrucciones necesarias pasan a la memoria principal. En este momento, cuando al programa está en ejecución, cuando se convierte en proceso, pero no antes.



Cuantas más instrucciones sea capaz de procesar un procesador, mayor será la velocidad en el sistema, debido a que los procesos esperarán menos tiempo a que dicha CPU los atienda y asigne recursos.

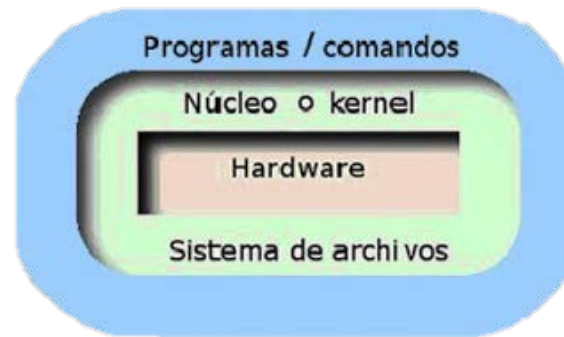
Gestión de los recursos



Núcleo o kernel del SO

La parte crítica de un sistema operativo encargada de gestionar los procesos entre sí se denomina **núcleo** o **kernel**.

El núcleo o kernel se encarga del manejo de las interrupciones, la asignación del trabajo al procesador y el proporcionar una vía de comunicación entre los distintos procesos.

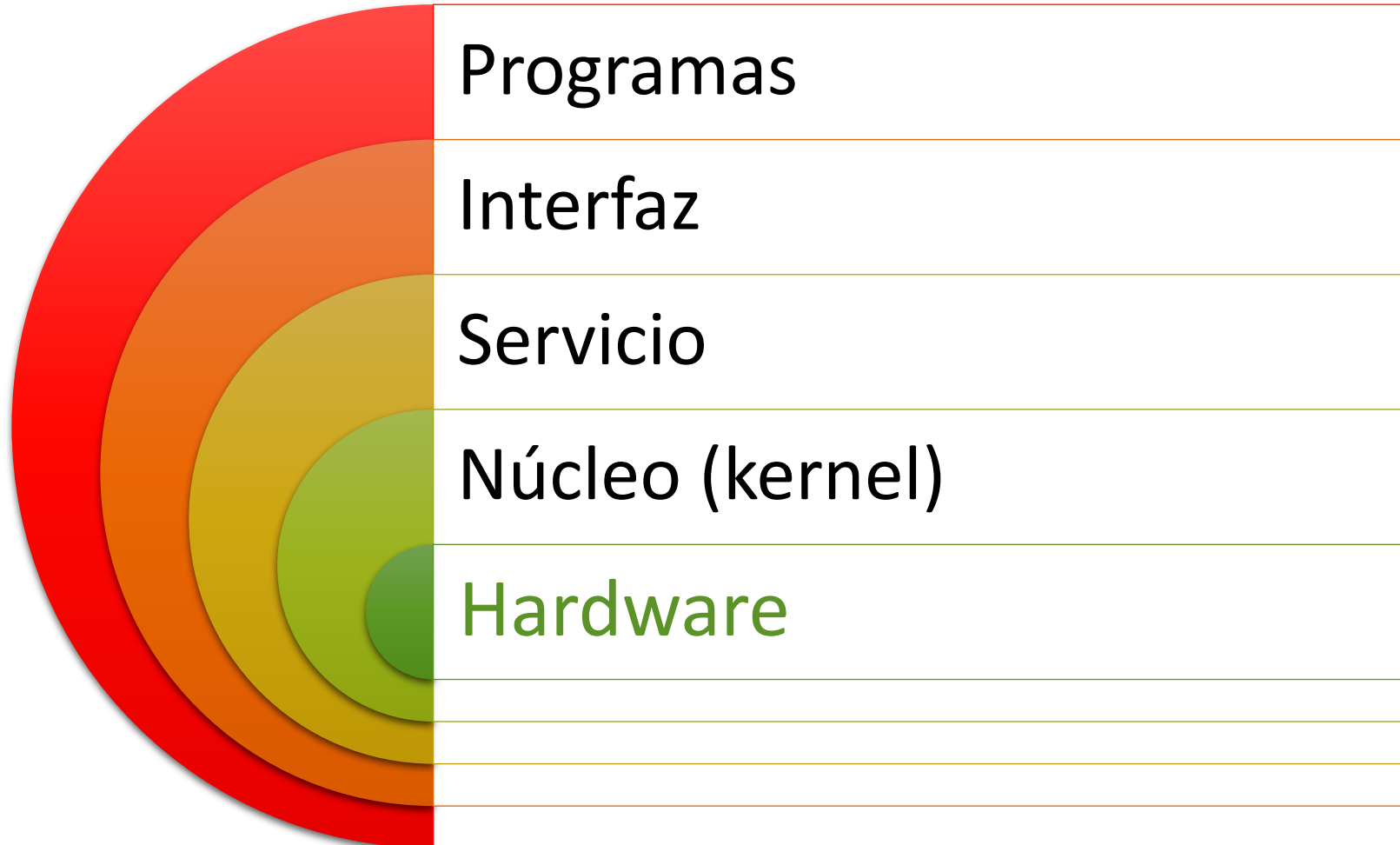


El núcleo tiene un submódulo denominado **planificador** el cual se encarga de asignar tiempo del procesador a los procesos, de acuerdo a una cierta política de planificación que varia de un sistema operativo a otro. El **administración de memoria** visto anteriormente también se controla desde el núcleo del SO.

Gestión de los recursos



Núcleo o kernel del SO

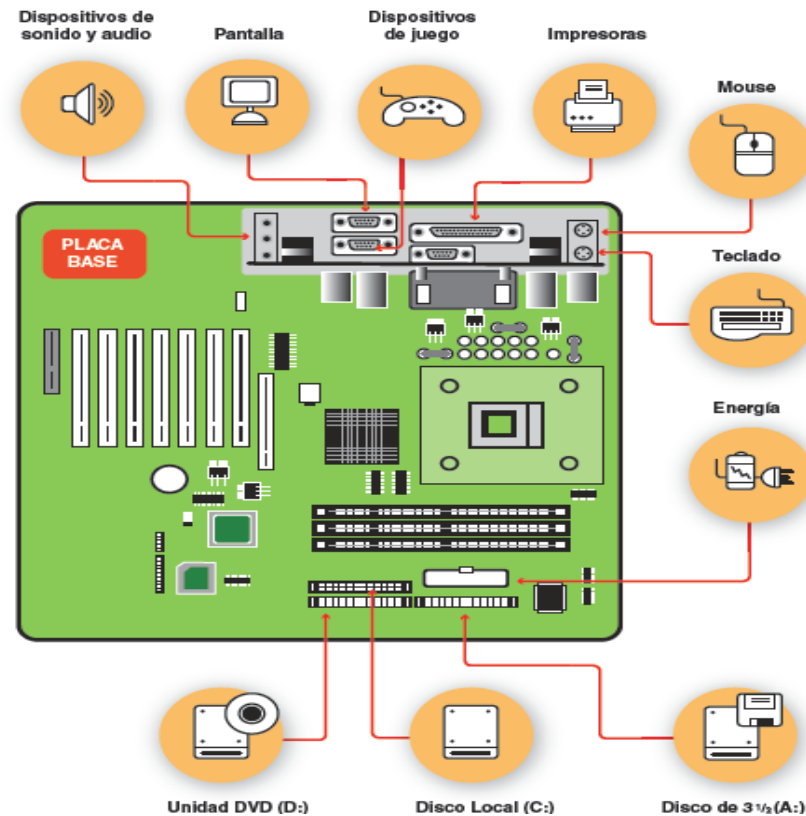


Gestión de los recursos



Gestión de E/S

La otra tarea fundamental de cualquier sistema operativo es la gestión de sus unidades periférica (ya sea de *entrada*, *salida*, *mixta* o de *comunicación*) que se conecten al ordenador o dispositivo.



La comunicación de los diferentes periféricos de E/S se lleva a cabo a nivel de hardware a través del **chipset** de la placa base, el cual comunica dichos componentes con la CPU y la memoria a través de buses internos.

Gestión de los recursos



Gestión de E/S

El sistema operativo se encarga de atender todas las particularidades de los dispositivos de E/S como su velocidad de operación, una técnica es la salida es el uso de **spoolers**, también llamadas colas.

Los datos de salida se almacenan de forma temporal en una cola situada en un dispositivo de almacenamiento masivo (**spool**) hasta que el dispositivo periférico requerido se encuentre libre, para evitar que el programa quede retenido porque el periférico no esta listo.



Interfaces de Usuario

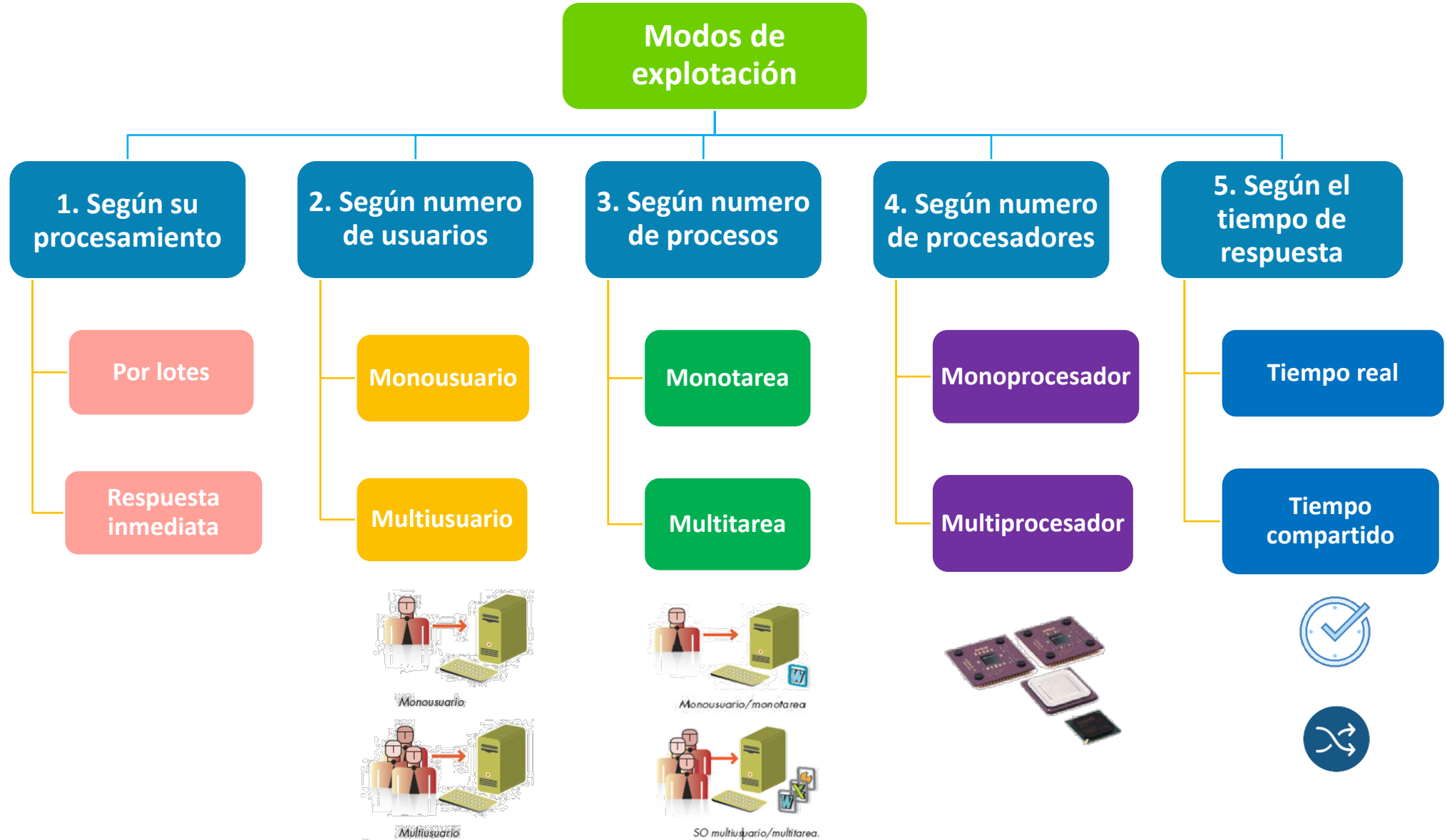


Las **interfaces de usuario** o shell son el medio de comunicación entre el usuario, el hardware y el software a través del SO. Las interfaces se clasifican en:

- **Interfaz tipo texto.** Si el SO es de tipo texto, todas las órdenes que el usuario introduzca y las respuestas que el SO dé se introducirán o visualizarán mediante cadenas de caracteres.
- **Interfaz tipo gráfico.** Hoy en día, la mayoría de los sistemas operativos utiliza medios de comunicación entre máquina y ordenador de tipo gráfico.
- **Interfaz mixta.** Hoy en día, la mayoría de los sistemas operativos utilizan ambas interfaces para comunicarse con el usuario.



Modos de explotación



Modos de explotación



Explotación según su procesamiento

Procesamiento por Lotes: (obsoleto)

Es el sistema que se utilizaba en la segunda generación de ordenadores con SO como FMS e IBSYS, y en el que había 4 pasos diferenciados: introducir los datos, transportarlos a la computadora y procesar los datos en la computadora y traerlos de vuelta.

Procesamiento de respuesta inmediata:

Se trata del sistema de procesamiento utilizado en la actualidad. Se introducen los datos en el momento y la respuesta también se obtiene de forma inmediata desde la CPU, es decir, hay que esperar poco entre lo que se tarda en introducir los datos y la respuesta, dependiendo eso de lo que tarde el proceso o programa en sí.

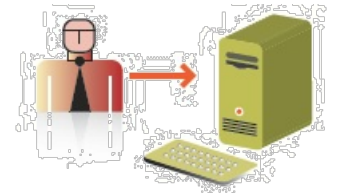
Modos de explotación



Explotación según número de usuarios

Monousuario:

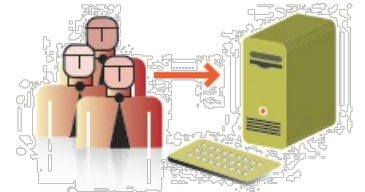
Un solo usuario utiliza a la vez el Sistema Operativo y sus recursos.



Monousuario

Multiusuario:

Permiten que varios usuarios puedan ejecutar a la vez distintos programas y hacer uso de los recursos de una máquina al mismo tiempo.



Multiusuario

Modos de explotación



Explotación según número de usuarios

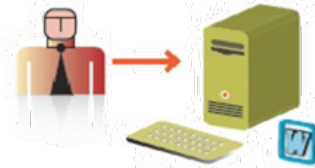
Monoprogramación o monotarea:

Solo se puede ejecutar un programa o proceso a la vez. Todos los recursos del sistema están dedicados a ese programa o recurso.

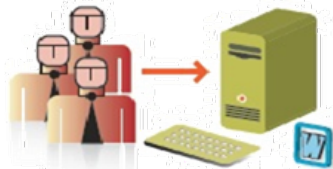
Un SO monotarea no significa que no pueda ser multiusuario, siempre y cuando esperen su turno de ejecución.

Multiprogramación o multitarea:

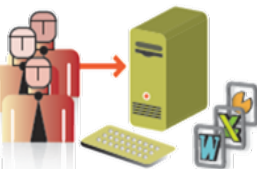
En este tipo de SO se pueden ejecutar varios programas o procesos concurrentemente. Si el SO se instala en una máquina con un solo procesador, la CPU compartirá su tiempo entre los diferentes procesos que requieran ejecutarse a la vez en ese momento.



Monousuario/monotarea



Multiusuario/monotarea



SO multiusuario/multitarea.

Modos de explotación



Explotación según nº de procesadores

Monoprocesador:

El ordenador consta de un único procesador y todos los procesos y tareas tendrán que pasar por él. Un ordenador monoprocesador podrá tener un SO monotarea, multitarea, monousuario o multiusuario.

Multiprocesador:

Todo aquel ordenador que tenga dos o más procesadores (dual core, quad core, octo..).

- a) **Multiproceso simétrico:** El SO utilizará la potencia de los procesadores de igual forma.
- b) **Multiproceso asimétrico:** El SO repartirá las tareas que esté realizando cada procesador de la forma más efectiva que considere.

Modos de explotación



Explotación según tiempo de respuesta

Tiempo real:

La respuesta es prácticamente inmediata tras lanzar un proceso.

Tiempo compartido:

Cada proceso utilizará ciclos de la CPU de forma fragmentada hasta finalizar.



Conceptos. Servicios y procesos

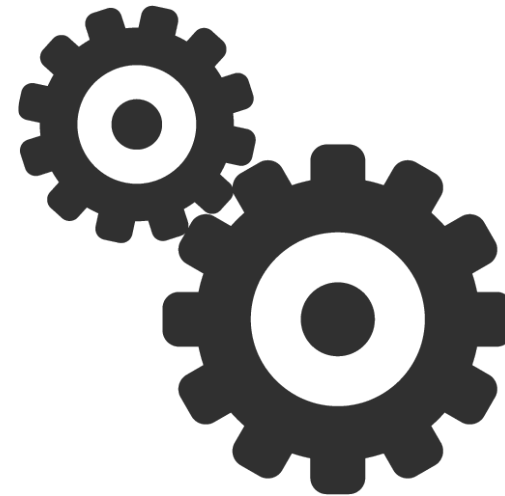


Para llevar a cabo las funciones descritas anteriormente un Sistema Operativo hace uso de un tipo de aplicaciones especiales denominadas **servicios** (o demonios) funcionando en segundo plano.

Un **servicio** (o demonio) es un tipo de aplicación cargado por el sistema operativo que normalmente se ejecutan en **segundo plano** y que permiten llevar a cabo funciones básicas de éste.

Muchos de los servicios (también llamados demonios o *daemons* en Linux) son esenciales para el funcionamiento de muchas de las aplicaciones y del propio Sistema Operativo.

Si estos programas, es decir, los servicios, no estuvieran ejecutándose, muchas aplicaciones no funcionarían o algo tan frecuente como el acceso a Internet no sería posible.



Conceptos. Servicios y procesos



Los **procesos** son otro importante concepto manejado por el sistema operativo, que veremos en la próximas unidades en profundidad:

Un **proceso** son el conjunto de instrucciones de un programa que se encuentra en ejecución en ese momento (van a procesarse en la CPU)

Los procesos son gestionados por el sistema operativo y están formados por:

- Las **instrucciones** de un programa destinadas a ser ejecutadas por el microprocesador.
- Su **estado de ejecución** en un momento dado, esto es, los valores de los registros de la unidad central de procesamiento para dicho programa.
- Su **memoria de trabajo**, es decir, la memoria que ha reservado y su contenido.

Conceptos. Servicios y procesos



▪ Diferencias entre servicios y procesos:

Un **proceso** es la instancia en memoria de un programa ejecutable (un archivo ejecutable **.exe** o binario) que se ejecuta. Una aplicación puede tener varios procesos que se ejecutan simultáneamente. Por ejemplo, algunos navegadores modernos como Google Chrome o Firefox, ejecutan varios procesos a la vez (cada uno de los cuales representa una pestaña).

Un **servicio** es también un proceso, pero que se ejecuta en segundo plano y no interactúa con nosotros de forma directa como un programa o aplicación. En sistemas Windows, los servicios casi siempre se ejecutan como una instancia del proceso **svchost**, también identificado como host de servicio (En Linux *systemd*)

- ✓ Abre el *administrador de tareas* de Windows 10.
¿Cuántos Host de servicio identificas abiertos a la vez?

