

FUNDAMENTOS DE LOS SISTEMAS OPERATIVOS

Guadalupe León López
IES CASTELAR



- INTRODUCCIÓN
- HISTORIA
- ARQUITECTURA DE LOS SSOO
- INTERFAZ DE USUARIO
- MODOS DE EXPLOTACIÓN
- FUNCIONES DE LOS SSOO



INTRODUCCIÓN

- ▶ La evolución de los SO no ha parado de crecer desde su primera aparición en la década de los años 50.
- ▶ Hoy en día, tener un SO como Windows en nuestro ordenador o Android/IOS en nuestro teléfono móvil, sea algo normal, pero necesario.
- ▶ Un **SO** lo podemos definir como un programa o conjunto de programas, que controla el hardware del dispositivo, y gestiona sus recursos, facilitando al usuario su uso, por medio de una serie de capas que lo abstraen de los elementos hardware.
- ▶ A lo largo de este tema, veremos los puntos más importantes en la historia de los SO desde sus inicios, hasta la actualidad. Trataremos los diferentes tipos de arquitecturas de los SO. Conoceremos la interfaz con el usuario que permite interactuar con el SO. Desarrollaremos los modos de explotación de un SO en función de diferentes formas. Veremos de manera resumida las distintas funciones de los SO y por último, trataremos los principales SO que podemos encontrarnos en la actualidad.

HISTORIA DE LOS SSOO

Siempre han estado conectados junto con la evolución del hardware.

- ▶ En los **inicios (1940-1955)** no se contaba con ningún SO en los ordenadores, su componente principal era la **válvula de vacío** y estaban destinados a aplicaciones científicas y militares, además la programación se realizaba en lenguaje máquina puro.
- ▶ En la **segunda generación** de ordenadores (**1955-1964**), nacían los primeros SO y los primeros lenguajes de programación como ensamblador, además de empezar a aparecer los **transistores**, dispositivos electrónicos de vital importancia.
- ▶ Continuando con la **tercera generación (1964-1972)**, se generalizó el uso de transistores y aparecen **los circuitos integrados** también conocidos como **chips o microchips**, permitiendo la creación de ordenadores más **pequeños** de propósito general, además aparecen los primeros sistemas de **multiprogramación**, que podían soportar dos o más procesos activos simultáneamente y el concepto de **memoria virtual** entre otras cosas.

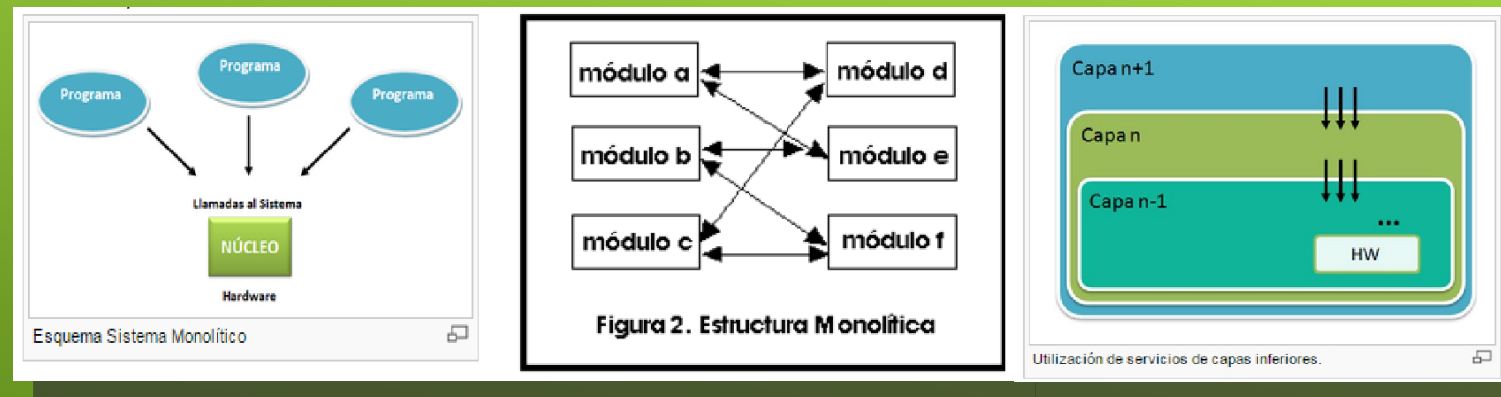
- ▶ Pasando a la **cuarta generación (1972-1981)** aparecen los microprocesadores (CPU), el chip central más complejo de un SO, además, cada vez se comercializan más los ordenadores personales y el desarrollo de programas. Pero destacamos esta época por el inicio del despegue del SO UNIX.
- ▶ Por último, desde la **quinta generación en adelante (1982 - X)** los SO han ido evolucionando enormemente, en parte, gracias a los entornos gráficos de usuario (GUI). Aparecen las primeras versiones de Windows, que no para de evolucionar, y, además, destacamos los SO distribuidos, el modelo cliente-servidor y las técnicas de protección de datos para garantizar la seguridad en la red.



- ▶ En la actualidad las tecnologías incorporadas a los SO tienen como finalidad **aumentar la productividad y facilitar la tarea a los usuarios**. Algunas de estas tecnologías son
- ▶ El **multiprocesamiento**, procesadores con múltiples núcleos que al principio estaban destinados a entornos profesionales y grandes sistemas, y que ahora podemos tener en cualquiera de nuestros dispositivos domésticos como el móvil o el ordenador.
- ▶ La **virtualización**, ya que cada vez son más los sistemas que se ejecutan en entornos virtualizados y eso hace que se mejoren los procesadores y SO's para virtualizar con mayor eficiencia.
- ▶ La **inteligencia artificial**, multitud de herramientas que han evolucionado para ayudar a los usuarios tomando la función de asistentes como: Siri de Apple, Alexa de Amazon o Cortana de Microsoft. Interaccionan con los usuarios a través del uso de la voz.
- ▶ La integración en la **nube**, debido al gran uso de internet, cada vez ofrecen mejores herramientas para mejorar la conectividad y la experiencia de usuario, navegadores web, mecanismos para almacenar nuestra información y compartirla entre distintos dispositivos de forma transparente para el usuario o incluso ejecutar todo el SO a través de la red.
- ▶ Por último, destacar la **seguridad**, debido al auge de las redes de dispositivos cada vez mayor, es necesario tener técnicas para prevenir accesos no autorizados a nuestros datos, por lo que las técnicas para garantizar la seguridad, no han parado de actualizarse.

ARQUITECTURA DE LOS SISTEMAS OPERATIVOS

Han ido evolucionando con el paso de los años, los primeros SO contaban con una **estructura monolítica**, prácticamente obsoleta en la actualidad, dónde el SO estaba construido por un sólo programa compuesto de un conjunto de procedimientos que podían llamarse entre sí sin ningún tipo de limitación. Sistemas en los que la interdependencia entre sus elementos es total, no pueden trabajar unos sin los otros.



A medida que fueron incluyéndose nuevas funciones, fue aumentando el tamaño del SO y, por lo tanto, fue necesario crear una estructura más organizada, estableciéndose una **jerarquía de capas** o niveles virtuales contruidos sobre el hardware.

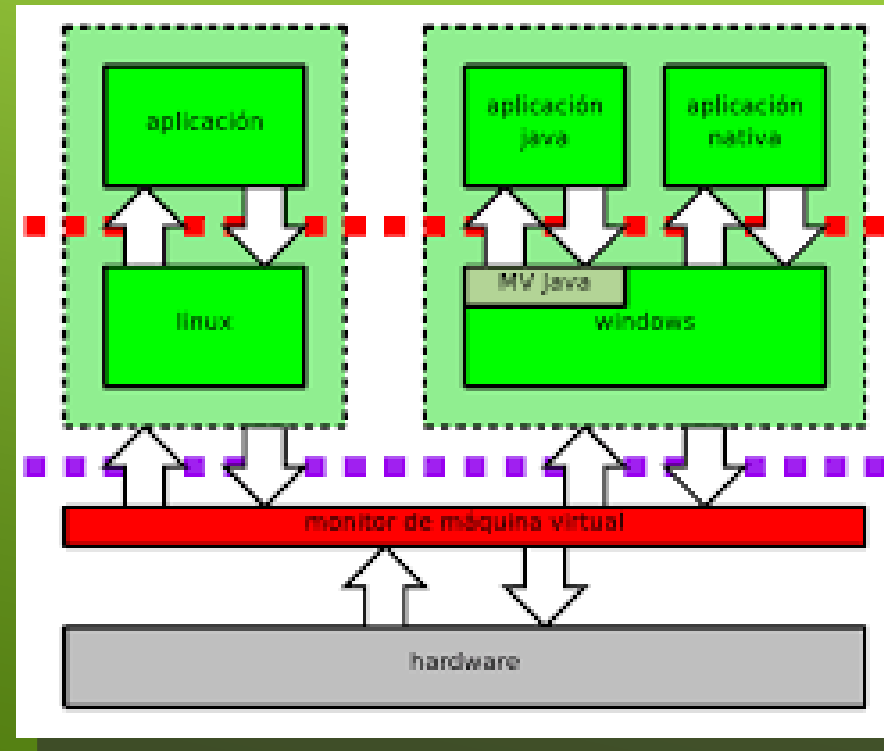
La capa 0 está formada por el hardware del sistema informático, cuya función es aislar al resto de capas de las particularidades del hardware sobre el que va a correr el SO, en algunos SO se denomina capa de abstracción del hardware.

La capa 1 controla la gestión y planificación de la CPU de los distintos procesos, **la capa 2** realiza la administración de memoria, asignando espacio de memoria principal a los procesos y gestionando su ubicación mediante mecanismos de memoria virtual, **la capa 3** es la encargada de la gestión de los dispositivos de entrada y salida y de los archivos.

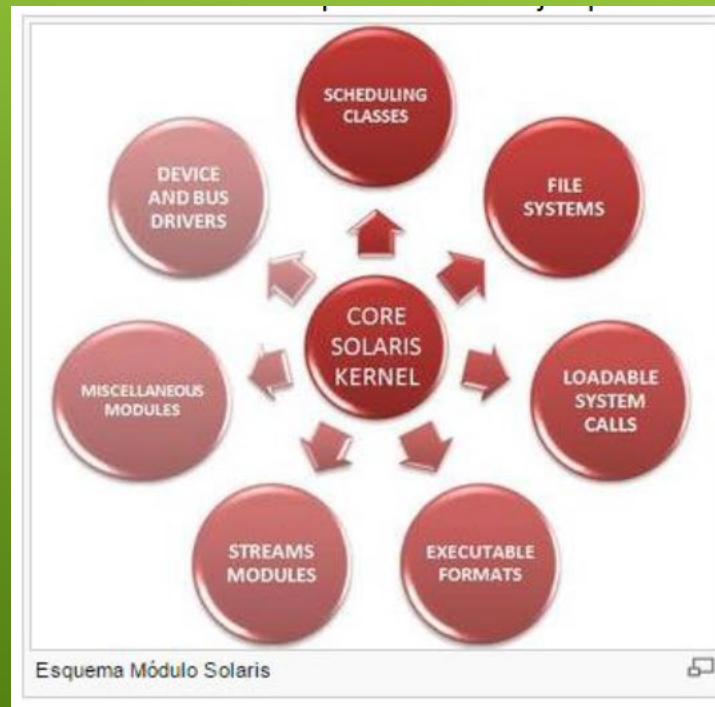
La capa 4 soporta la interfaz de llamadas al sistema, a partir de las cuales los procesos de usuario pueden requerir y recibir los servicios disponibles, y finalmente, **la capa 5**, es dónde se ejecutan los programas de usuario. Cuanto más cerca del nivel cero se encuentra una capa, mayor **privilegio** tendrán los procesos que se ejecuten en ella.



- A continuación, tenemos la estructura de **máquina virtual**, compuesta de un módulo principal llamado *monitor virtual* o **hypervisor**, cuya función primordial es llevar a cabo la multiprogramación, presentando tantas máquinas virtuales como se soliciten, y ser el encargado de gestionar los elementos hardware del sistema para que los usen las distintas máquinas virtuales creadas. Cada máquina virtual es *independiente* y es capaz de ejecutar un SO distinto.



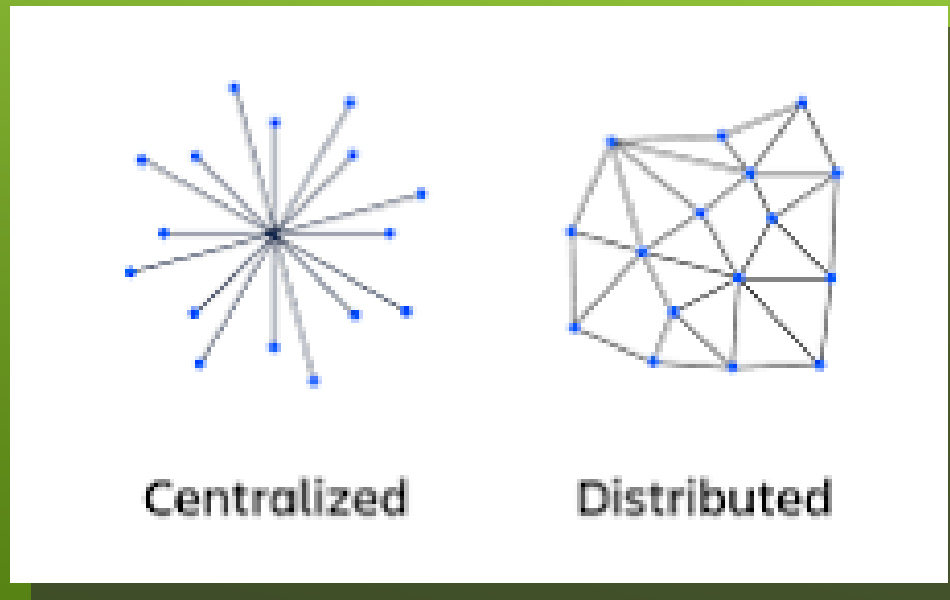
- La siguiente es la estructura **orientada al objeto**, basada en utilizar la metodología de diseño orientado a objetos, de esta forma, en lugar de ver el sistema operativo como una colección de funciones que se pueden ejecutar, lo que se tiene es una colección de objetos, los programas, los ficheros, los dispositivos periféricos, todos serían un tipo de objeto.



- Por último, la estructura más difundida en la actualidad es la de **cliente servidor**, que consiste en un proceso principal, llamado **núcleo**, que se encarga de las relaciones y comunicaciones entre procesos clientes y servidores. El objetivo de esta estructura es implantar la mayoría de las funciones del SO como si de procesos de usuario se tratase, de tal forma que, para solicitar un servicio del sistema, como la asignación de memoria, *un proceso de usuario denominado proceso cliente, envía la solicitud, a un proceso servidor que lleva a cabo el trabajo y devuelve la respuesta, la función del núcleo será controlar la comunicación entre los clientes y los servidores.* Esta estructura se adapta muy bien a **sistemas distribuidos**.



- **Sistemas distribuidos**, ya que no importa de donde venga la respuesta, si de la red o del propio sistema, siempre que se obtenga el resultado solicitado. Éstos son un conjunto de componentes y máquinas independientes ubicados en diferentes sistemas de una red, que se comunican con el fin de operar como una sola unidad.



INTERFAZ DE USUARIO

Desde el punto de vista del usuario, una parte fundamental de todo SO, es el conjunto de mecanismos que están disponibles para que se puedan llevar a cabo tanto el servicio de los procesos, como el acceso directo del usuario. Los programas de usuario solicitan al SO servicios mediante las **llamadas al sistema** a través de una **Interfaz de Programación de Aplicaciones**, conocida como **API**.



Por otra parte, los SO actuales incorporan **interpretes para comandos**, conocidos como **Shell**, que sirven como ejemplo para realizar llamadas al sistema. También son la principal interfaz entre un usuario y el SO.

Dentro de los interfaces del SO distinguimos los de **tipo texto**, dónde las ordenes se realizan en forma de líneas de texto que el usuario escribe sobre el ***terminal***, respetando ciertas reglas sintácticas y semánticas. E interfaces de **tipo gráfico**, más simples, intuitivos y muy usados por disponer de ventanas, menús y dibujos aclaratorios para facilitar el uso al usuario.



MODOS DE EXPLOTACIÓN DEL SO

Es posible clasificar las distintas formas en las que un usuario explota un sistema, es decir, la forma en la que emplea los recursos de un sistema.

- En primer lugar, es en función del número de usuarios que utilizan el sistema simultáneamente, éste puede ser **monousuario**, cuando sólo puede trabajar un usuario con el ordenador y **multiusuarios**, cuando varios usuarios pueden ejecutar procesos distintos sobre la misma CPU y compartir el uso de los recursos del sistema simultáneamente.
- En segundo lugar, en función del número de procesos que se pueden ejecutar simultáneamente, podemos diferenciar **monoprogramación** o **monotarea**, que puede ejecutar sólo un proceso, y hasta que este no acabe no puede iniciarse el siguiente. Y **multiprogramación** o **multitarea**, capacitado para poder ejecutar varios procesos a la vez. Añadir que hoy en día no quedan prácticamente SO monotarea, todos los que se comercializan son multitarea.

- En tercer lugar, es en función del número de procesadores que se pueden usar simultáneamente en un mismo ordenador, un sistema será **monoproceso** cuando hay un único procesador y **multiproceso** cuando hay varios procesadores.

- Y en cuarto y último lugar, en función de los requerimientos temporales, un sistema puede trabajar en **tiempo real**, SO reservados para el ámbito militar. SO **interactivos**, que es una categoría equivalente a un tiempo real menos estricto en la que la velocidad de respuesta corresponde a las expectativas de los usuarios y, por último, un sistema puede trabajar en **procesos por lotes**, dónde un proceso empieza a ejecutarse cuando el último ha terminado. Se suele emplear en procesos que no requieren intervenciones ni resultados inmediatos.



FUNCIONES DE LOS SSOO

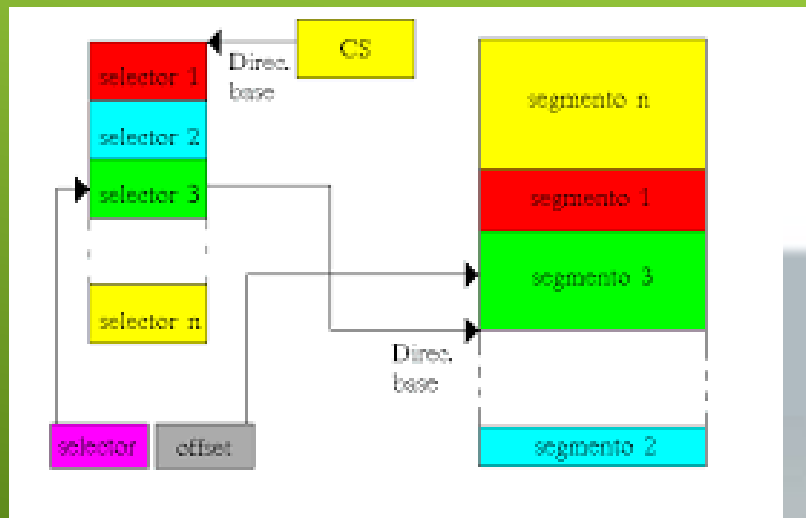
Hay un conjunto de servicios que son suministrados por la mayoría de SO, particularmente los que se destinan a entornos **multiusuario**. Estos son:

- ❑ Gestión de procesos.
- ❑ Gestión de memoria.
- ❑ Gestión de E/S.
- ❑ Sistema de archivos.



- En primer lugar, vamos a tratar la **gestión de los procesos**, definiendo proceso como una abstracción de un programa en ejecución. La gestión se realiza cuando hay varios procesos preparados para ejecutar y el SO debe decidir a quién darle el turno haciendo pasar al proceso al estado de ejecución. El planificador, es el módulo encargado de resolver este problema mediante el uso de alguno de los siguientes algoritmos de planificación que existen. Empezando con el **algoritmo FIFO**, siglas en inglés que significan que el primero que llega es el primero en salir, es decir, en ser atendido, cada proceso que entra es ejecutado hasta finalizar o realizar una llamada de entrada salida. El segundo algoritmo es el de **Round Robin**, una mejora del anterior, donde antes de cederle la CPU a un proceso, se pone en marcha un reloj para que se produzca una interrupción dentro de un periodo de tiempo establecido y así alternar los procesos. El siguiente es el algoritmo **SJF**, siglas que significan que el trabajo más corto, se ejecuta primero. Y, por último, está el algoritmo de **planificación por prioridad**, dónde resumidamente, a cada proceso se le asigna una prioridad y en función de ella, antes se ejecutan en la CPU.

- En segundo lugar, pasamos a como gestiona **el SO la memoria**, la parte que realiza esto se llama administrador de la memoria y es el encargado de controlar qué partes de la memoria están en uso y cuáles no, asignar memoria a procesos cuando la necesiten y retirársela cuando terminen entre otras funciones.



- ▶ En tercer lugar, está la forma de **gestión y control de los dispositivos de entrada y salida**, una función muy importante en los SO, se deben manejar errores, controlar los dispositivos y capturar interrupciones entre otras cosas. Además, un componente importante que se debe conocer, es el controlador de dispositivo, también conocido como driver, que suele especializarse en un tipo de dispositivo, para realizar la comunicación entre el SO y el adaptador.
- ▶ Por último, tenemos la forma en la que se gestiona **el sistema de archivos**, una parte del SO la cual se encarga de administrar el almacenamiento secundario y responsable de componer los medios para el almacenamiento de la información y para que ésta sea compartida por los usuarios de forma controlada. Las funciones más importantes que debe realizar son la creación, eliminación, consultar, abrir y cerrar archivos.