

Tema 15

SISTEMAS OPERATIVOS. COMPONENTES. ESTRUCTURA. FUNCIONES. TIPOS

15.1 INTRODUCCIÓN

Un sistema operativo es un conjunto de programas que actúan como intermediarios entre los componentes hardware del ordenador y el usuario, facilitando así la utilización del equipo. Los sistemas operativos ofrecen al usuario una interfaz atractiva y sencilla de utilizar.

Las funciones de un sistema operativo son las siguientes:

- Gestión de procesos.
- Gestión de memoria.
- Gestión de entrada/salida.
- Gestión de sistemas de ficheros.
- Gestión de la red.

Hace unos años solo había sistemas operativos en los ordenadores, como por ejemplo Windows XP en su momento. Había también sistemas operativos especiales para servidores, un ejemplo era el sistema operativo Solaris. A nivel de usuario los más populares siempre han sido Windows, Linux y Mac OS y ahora todo el mundo lleva, en su día a día, un sistema operativo auestas, en los smartphones, los más utilizados son Android y iOS.

La evolución de los sistemas operativos ha ido siempre ligada a la evolución del hardware. Primero surgieron las calculadoras, después los ordenadores electro-mecánicos y poco a poco se fue haciendo necesario el uso de una interfaz entre el usuario y la máquina que facilitara el uso del equipo.

- Primera generación (década de los 40): los ordenadores no tenían sistema operativo. El programador introducía un programa que se ejecutaba y lo escribía en un dispositivo parecido a una impresora.
- Segunda generación (década de los 50): surgen los primeros sistemas operativos cuando surgen los transistores, utilizaban el procesamiento por lotes.
- Tercera generación (década de los 60): aparecen los sistemas integrados y los sistemas operativos mejoran y perfeccionan el trabajo por lotes. Surge la multiprogramación.
- Cuarta generación (década de los 70): el microprocesador revoluciona los ordenadores, en especial a los sistemas operativos. Nace UNIX y algunos lenguajes de programación como SQL.
- Quinta generación (década de los 80): nace el concepto de PC, ordenador personal, primero con MS-DOS y Linux. Se desarrollan los sistemas operativos distribuidos permitiendo que el usuario ejecute varios programas en tiempo simultáneo.
- Actualidad: los sistemas operativos son multiusuario y multiprocesador, están disponibles para ordenador y para tablets y smartphones. Las versiones más utilizadas en los ordenadores personales son Windows, Ubuntu y MacOS. La última gran revolución ha sido la de conectar el sistema operativo a la nube.

Este tema se centra en los sistemas operativos como tal, en cuáles son sus componentes, su estructura y sus funciones.

15.1.1 Contextualización¹

Al tratarse de un tema general de Sistemas Operativos, se contextualiza directamente en todos aquellos módulos en los que se estudian.

Así pues, este tema se contextualiza en los siguientes módulos:

- 3030. Operaciones auxiliares para la configuración y la explotación. Perteneciente al Título Profesional Básico en Informática y Comunicaciones (R.D. 127/2014).
- 3030. Operaciones auxiliares para la configuración y la explotación. Perteneciente al Título Profesional Básico en Informática de Oficina (R.D. 356/2014).

¹ El opositor debe recordar que esta contextualización debe ser completada con la legislación perteneciente a su Comunidad Autónoma.

- 0222. Sistemas Operativos Monopuesto del título de Técnico en Sistemas Microinformáticos y Redes (R.D. 1691/2007).
- 0224. Sistemas Operativos en red, también de Técnico en Sistemas Microinformáticos y Redes.
- 0369. Implantación de Sistemas Operativos del título de Técnico Superior en Administración de Sistemas Informáticos en Red (R.D. 1629/2009).
- 0374. Administración de Sistemas Operativos también del título de Administración de Sistemas Informáticos en Red.
- 0483. Sistemas Informáticos que pertenece al título de Técnico Superior en Desarrollo de Aplicaciones Web definido en el R.D. 450/2010 y al título de Técnico Superior en Desarrollo de Aplicaciones Multiplataforma definido en el R.D. 686/2010.

Además, se tratarán algunos aspectos de forma transversal en otros módulos, como por ejemplo en aquellos en los que se trabaja con máquinas virtuales, configuración de servicios en red, etc.

15.2 SISTEMAS OPERATIVOS. FUNCIONES

15.2.1 Funciones

La función principal del sistema operativo, como ya se ha comentado, es la de ofrecer una interfaz al usuario para que éste interaccione con el ordenador de una forma amigable. Estos son algunos de sus objetivos principales:

- Comodidad: debe permitir al usuario que el uso del ordenador le resulte más cómodo a través del sistema operativo.
- Eficiencia: permite que los recursos del equipo se utilicen de la forma más eficiente posible.
- Adaptabilidad: debe adaptarse a la evolución del software y del hardware y evolucionar introduciendo, si es necesario, nuevas funciones del sistema.

Las funciones que debe realizar un sistema operativo dependen del punto de vista desde el que se mire. A los programadores debe ofrecerles una interfaz simple ocultando la complejidad del hardware, y por otra debe ofrecer al usuario una interfaz cómoda, como ya se ha comentado antes.

Así pues, las funciones generales de un sistema operativo, desde el punto de vista del equipo, son las siguientes:

- Control de procesos: se encarga de controlar los procesos que debe ejecutar la CPU.
- Gestión de memoria: asigna el espacio necesario de memoria a cada proceso, a veces necesitará hacer uso de la memoria virtual.
- Gestión de entrada y salida: gestiona la entrada y salida de datos a los periféricos conectados al sistema, a través de drivers.
- Gestión de archivos: se encarga de gestionar los archivos con operaciones de lectura y escritura, además debe garantizar la protección de estos archivos.
- Gestión de permisos: debe proporcionar seguridad al sistema, para ello gestiona permisos a los distintos programas y archivos que así lo requieran.

Desde el punto de vista del usuario, las funciones del sistema operativo son otras. Se debe proporcionar al usuario una interfaz amigable independiente de la complejidad del hardware del equipo. Existen una serie de programas que facilitan, a su vez, que se creen programas, se gestionen los periféricos y los archivos del sistema de una forma fácil e intuitiva.

Las funciones, desde el punto de vista de usuario, se resumen en:

- Creación de programas: ofrece aplicaciones preparadas para que el programador cree programas desde una interfaz atractiva. Suelen ser programas independientes del sistema operativo.
- Ejecución de programas: desde el sistema operativo se podrán ejecutar diversas aplicaciones, se cargan las instrucciones y datos en memoria principal.
- Acceso a los periféricos de E/S: desde el sistema operativo se ofrece al usuario un fácil acceso a los periféricos. El sistema operativo se encarga además de las instrucciones y señales de control.
- Acceso a los archivos: el sistema operativo facilita el acceso y uso de ficheros. Por ejemplo, el sistema de archivos del sistema operativo Windows, se encarga de cambiar el icono de un archivo según su tipo de extensión, facilitando así al usuario la interacción con el sistema.
- Control de errores: se deben detectar los errores del sistema y responder con una solución. Un claro ejemplo de control de errores podría ser en la gestión de E/S, si un usuario quiere imprimir y no queda papel en la impresora, el sistema operativo es el encargado de informar al usuario.

15.3 COMPONENTES

Los componentes más importantes de un sistema operativo son los que facilitan que se realicen sus tareas más importantes: gestor de procesos, gestor de memoria, gestor de entrada y salida y gestor de archivos y dispositivos.

15.3.1 Gestor de procesos

Los programas en ejecución se dividen en procesos. Un proceso es un programa en ejecución. Es habitual confundir el concepto de proceso con el de programa, pero es sencillo diferenciarlos. Mientras que un programa es una entidad pasiva que si no se ejecuta no hace nada, un proceso es una entidad activa que va pasando además por distintos estados. La ejecución de dicho programa es lo que se conoce como proceso.

El Doctor Tanenbaum, experto en sistemas operativos, hacía una analogía muy sencilla para entender la diferencia entre proceso y sistema operativo comparándolos con una receta alimenticia. Si se quiere seguir la receta de una tarta de cumpleaños, la receta escrita que hay que seguir paso a paso representaría el programa mientras que el cocinero es el procesador y los ingredientes las entradas del programa. El proceso consiste en que el cocinero vaya leyendo la receta y mediante los ingredientes hornear el pastel resultante.

Un proceso está formado por: instrucciones, memoria de trabajo, estado de ejecución y planificación.

Para que un proceso funcione éste debe de cargarse en memoria y tener disponibles ciertos recursos del sistema para llevar a cabo su objetivo: la CPU, el uso de la memoria principal, dispositivos de E/S, etc. Estos recursos se los asigna el sistema operativo cuando considera conveniente.

Cuando un proceso crea otro proceso nuevo puede ocurrir que padre e hijo se ejecuten al mismo tiempo o que el proceso padre espere a que finalice la ejecución del hijo.

El estado de un proceso determina su actividad en un momento preciso y para pasar de un estado a otro se realizan una serie de transiciones. Los estados en los que puede estar un proceso son: activo (en ejecución), terminado, suspendido o preparado. Las transiciones más importantes son:

- Preparado-ejecución: el planificador determina que se va a ejecutar el proceso.
- Ejecución-preparado: el proceso que se está ejecutando se interrumpe (por ejemplo, si ya ha agotado el tiempo que se le ha asignado) y se queda en estado preparado de nuevo.

- Ejecución-suspendido: el proceso abandona de forma voluntaria la ejecución y se queda esperando algún evento.
- Suspendido-preparado: el proceso suspendido ya puede volver a ejecutarse.
- Ejecución-terminado: el proceso ha finalizado su ejecución.

Cuando se va a ejecutar un proceso, el sistema operativo necesita almacenar toda la información disponible con respecto a ese proceso. Esta información se almacena en un registro especial que se llama **bloque de control de proceso (PCB)**. Una vez el proceso termina su ejecución, su PCB es borrado y puede utilizarse para almacenar la información de otros procesos.

Todos los procesos conllevan una serie de prioridades de ejecución. Las prioridades determinan cuando un proceso debe tener preferencia sobre otro.

15.3.2 Gestor de memoria

La gestión de memoria de un sistema operativo consiste en controlar qué partes de la memoria se están utilizando para controlar cuáles son los espacios libres. Cuando un proceso lo necesite se le asignará una zona libre de memoria, una vez el proceso ya se ha ejecutado, la memoria queda liberada.

Las funciones de un gestor de memoria son, entre otras:

- Administrar la memoria utilizada por los procesos.
- Garantizar un buen aprovechamiento de la memoria.
- Proteger la memoria frente a la llegada de nuevos procesos.
- Usar el disco como memoria virtual.
- Determinar las zonas de memoria ocupadas por procesos.
- Determinar las zonas de memoria libres.
- Memoria compartida.

Los requisitos que se deben cumplir para una buena gestión de la memoria son:

- Reubicación: El usuario del equipo se desentiende de la ubicación de los programas durante su ejecución. El gestor de memoria (una combinación entre el hardware y el software del sistema) será quien se encargue de almacenar las direcciones de memoria a las que acceder.
- Protección: las zonas asignadas a cada proceso deben estar protegidas frente a accesos de terceros. La protección se realiza en tiempo de ejecución.

- **Compartición:** aunque es importante proteger la memoria asignada a un proceso de accesos de terceros, a veces es necesario que se comparta la misma zona de memoria, estos accesos deben controlarse.
- **Organización lógica:** los datos se almacenan en un ordenador por medio de archivos y carpetas, este es un tipo de organización lógica, el gestor de memoria conocerá la ubicación física real de la información.
- **Organización física:** la localización de datos en memoria y los huecos libres sin asignar son organizados por el gestor de memoria.

La gestión de memoria también se encarga de controlar el intercambio entre la memoria principal del sistema y la memoria de sistemas de almacenamiento secundario cuando en la memoria principal no pueda almacenarse ya más información.

En ámbitos generales, existen dos tipos de asignación de memoria:

- **Asignación de memoria contigua:** los procesos se almacenan en posiciones consecutivas de memoria.
- **Asignación de memoria no contigua:** los procesos se almacenan en posiciones de memoria no consecutivas.

15.3.3 Gestor de E/S

Cuando se hace referencia a la E/S de información entre un ordenador y un periférico siempre se hace desde el punto de vista del ordenador. Las operaciones que se pueden realizar pueden ser de dos tipos:

- **Operaciones de entrada:** el periférico es el emisor y el ordenador es el receptor. El ordenador puede ser receptor bien desde la memoria o desde el procesador.
- **Operaciones de salida:** el emisor es el ordenador y el receptor es el periférico. El ordenador puede ser emisor bien desde la memoria o desde el procesador.

Todas las operaciones que se realizan de entrada y salida en un ordenador son controladas por el procesador, él es quien controla toda la operación. El procesador determina si la operación es de lectura o escritura y qué datos se van a compartir, también será quien determine la operación como finalizada.

El ordenador debe proporcionar mecanismos de control y mecanismos de transferencia de datos, estos mecanismos se llevan a cabo entre la unidad de control del procesador, los programas de E/S y el módulo de E/S:

- Los mecanismos de control indican el principio y fin de una operación de E/S, cantidad de datos que se transmite, detección de errores, etc.
- Los mecanismos de transferencia de datos indican la conversión de códigos, la manera de dirigir el tráfico, la sincronización, etc. En la transferencia de datos hay que tener en cuenta si es una transferencia individual o si la transferencia es por bloques:
 - Un ejemplo de transferencia individual podría ser un clic del ratón o la presión de una tecla.
 - Un ejemplo de transferencia por bloques podría ser actualizar el contenido de la pantalla o abrir un fichero, en estos casos se transfiere mayor cantidad de información.

En un ordenador lo lógico y normal es que haya varios periféricos conectados, así pues, si se desencadenan operaciones de E/S de forma simultánea, el sistema de E/S debe ser capaz de gestionar todas estas operaciones sin que se pierda información en el proceso.

15.3.4 Gestor de archivos y dispositivos

Un **archivo** (también llamado fichero) es un conjunto de información no volátil que se almacena en un dispositivo.

Los sistemas operativos utilizados en la actualidad deben ser capaces de almacenar información en cualquier dispositivo informático que se esté utilizando. El almacenamiento en un ordenador puede ser volátil o no volátil. El almacenamiento de datos en memoria principal es un ejemplo de almacenamiento volátil, si se apaga el ordenador la información se pierde. El almacenamiento en una memoria secundaria, por ejemplo, en un disco duro externo, es un ejemplo de memoria no volátil puesto que proporciona un almacenamiento a largo plazo.

En los primeros ordenadores de la historia había un único directorio en el que se almacenaban todos los archivos. En la actualidad, se utiliza una estructura jerárquica en forma de árbol donde cada nodo es un archivo o directorio.

El sistema de archivos debe llevar un registro de los bloques de disco utilizados y los bloques de disco libres, este registro tiene en cuenta la dirección de disco donde se encuentra cada bloque lógico del archivo. Un archivo se almacena en bloques de tamaño fijo que no necesariamente contiguos.

La gestión de dispositivos es una de las operaciones más importantes que realiza un sistema operativo. Es importante que se gestione el acceso a cada uno de los dispositivos, se debe controlar la entrada y salida de datos.

Los dispositivos no son homogéneos y esto hace que se complique su gestión ya que no puede haber una metodología general para acceder a todos ellos.

Los dispositivos no están en contacto directo con el ordenador, todos los dispositivos de entrada/salida deben de tener un controlador encargado de interpretar las órdenes que llegan al procesador, teniendo en cuenta si se está emitiendo o recibiendo información, dependiendo de si se trata de periféricos de entrada o de salida. El controlador (comúnmente conocido como **driver**) genera las señales de control precisas para activar los elementos electromecánicos que correspondan. Los sistemas operativos deben contener una rutina especial conocida como gestor del periférico para controlar las transferencias de información.

15.4 ESTRUCTURA

Un sistema operativo robusto y con un alto rendimiento debe contar con una estructuración interna que facilite su mantenimiento.

15.4.1 Estructura monolítica

En un sistema monolítico su centro es un grupo de estructuras fijas, no tienen una estructura definida, existen módulos grandes en el núcleo que interactúan entre sí, las partes del kernel se compilan por capas.

Un ejemplo de sistema monolítico es Unix, Linux y FreeBSD. Es fácil relacionar Linux con los sistemas monolíticos por su famoso y potente kernel.

Estos sistemas se caracterizan porque cualquier cambio realizado requiere de nuevo la compilación del núcleo y el reinicio del sistema.

La mayor desventaja es que si falla el núcleo falla todo el sistema.

15.4.2 Estructura por capas

En estos sistemas se organiza el sistema operativo en una jerarquía de capas, cada una construida sobre la inferior. Su diseño es muy costoso y es menos eficiente que el sistema monolítico.

El sistema consta de seis capas:

- Capa 5: Interfaz de usuario.
- Capa 4: Programas de usuario.
- Capa 3: Dispositivos de E/S.
- Capa 2: Administración de procesos. Consola.
- Capa 1: Administración de memoria y discos.
- Capa 0: Hardware.

El primer sistema por capas construido fue el THE, un sistema operativo que trabajaba por lotes. Más adelante surgió MULTICS que estaba organizado en capas en forma de anillos concéntricos.

15.4.3 Estructura por módulos

En estos sistemas, el núcleo se compone de módulos separados entre sí. Si uno de los módulos falla, el sistema no se ve afectado.

El sistema por módulos se parece al de capas, pero este es mucho más flexible.

Ejemplos de sistemas operativos por módulos: algunas distribuciones de Linux, Mac OS X y Solaris.

15.4.4 Estructura con micronúcleo

Un sistema con micronúcleo consiste en tener un núcleo que ofrezca servicios mínimos de comunicación entre procesos. Tienen un diseño simple y funcional, escalable y portable.

Ejemplos de sistemas con micronúcleo: AIX, Minix, RadiOS y Symbian.

15.4.5 Estructura cliente-servidor

Son sistemas que se crean con la finalidad de desarrollar las funciones del sistema operativo como procesos de usuario. El núcleo solo controla la comunicación que se realiza mediante mensajes entre clientes y servidores.

Sirve para toda clase de aplicaciones, cumple con las mismas aplicaciones que los sistemas operativos tradicionales.

15.4.6 Estructura de máquina virtual

Son sistemas operativos que presentan una interfaz para cada proceso, muestran una máquina que simula una máquina real.

Su objetivo es el de integrar distintos sistemas operativos simulando tener varias máquinas.

15.5 TIPOS

15.5.1 Sistemas Operativos según el número de usuarios

Según el número de usuarios, un sistema operativo puede ser monousuario o multiusuario.

- Monousuario: soportan un único usuario a la vez. Los ordenadores personales siempre se han caracterizado por ser monousuario.
- Multiusuario: soportan varios usuarios al mismo tiempo. Puede que mediante varias terminales conectadas haya distintos usuarios o por medio de sesiones remotas.

15.5.2 Sistemas Operativos según el número de tareas

Según las tareas que un ordenador puede realizar al mismo tiempo, los sistemas operativos se pueden clasificar en monotarea y multitarea.

- Monotarea: solo permiten una tarea a la vez por usuario. Ejemplo: Windows Me.
- Multitarea: permite al usuario realizar varias tareas al mismo tiempo. Ejemplo: Mac OS, Windows 10.

15.5.3 Sistemas Operativos según el número de procesadores

Dependiendo del número de procesadores del sistema, existen sistemas operativos monoproceso y multiproceso.

- Monoproceso: es capaz de usar un único procesador, aunque hubiese más procesadores en el sistema no sería capaz de darles uso. Ejemplo: DOS y MacOS.
- Multiproceso: es un sistema que puede manejar más de un procesador. Trabajan de forma simétrica o asimétrica, dependiendo de a qué procesador se van enviando los procesos.

15.5.4 Según la forma de ofrecer los servicios

Según la forma de ofrecer los servicios se pueden clasificar los sistemas operativos en sistemas de red, distribuidos y sistemas en tiempo real:

- Sistemas de red: son sistemas operativos capaces de interactuar con otros sistemas operativos mediante la red. Pueden transferir archivos, ejecutar comandos remotos e incluso instalar aplicaciones.

- **Sistemas distribuidos:** abarcan a los sistemas de red. Son un conjunto de equipos independientes que trabajan como un único equipo.
- **Sistemas a tiempo real:** son capaces de realizar determinadas tareas en un tiempo concreto establecido. Suelen ser máquinas industriales.

15.6 CONCLUSIÓN

En este tema se muestra la importancia de un sistema operativo en un dispositivo. El tener una interfaz clara de cara al usuario es fundamental para hacer un buen uso del equipo, ya sea desde nivel de usuario como a nivel de programador.

Después de ver cuáles son las funciones y las características más importantes de los sistemas operativos, es necesario destacar la importante gestión que realiza el sistema operativo en todo el equipo: gestión de entrada y salida, gestión de archivos, gestión de procesos y gestión de memoria.

Los sistemas operativos pueden tener distintas estructuras internas como se ha comentado a lo largo del tema, junto con algunos ejemplos de los sistemas operativos conocidos.

Con el paso de los años, los sistemas operativos se van adaptando a los cambios. Por ejemplo, lo normal es trabajar mediante conexiones en red, por ejemplo, los smartphones trabajan constantemente en red, haciendo que el sistema operativo que lleva el teléfono inteligente instalado gestione los accesos al otro dispositivo. Es también bastante común en estos últimos años, realizar el mantenimiento de equipos por control remoto, accediendo desde el sistema operativo de un ordenador a otro sistema operativo, que puede ser un sistema diferente.

La evolución ha sido tan evidente que es fundamental contar con un sistema operativo robusto para gestionar los componentes del sistema de la forma más eficiente posible.

Como dato curioso, los sistemas operativos más utilizados del mundo son Windows 10, Linux y Mac OS. Cerca de un 90 % de usuarios utilizan Windows a nivel mundial. En cuanto a los teléfonos inteligentes, el sistema operativo que va en cabeza es Android, siguiéndole de cerca iOS de Apple.