

1. Un sistema operativo (SO) es un software que actúa como intermediario entre el hardware de una computadora y las aplicaciones de software. Sus funciones principales incluyen la gestión de recursos, la interfaz de usuario, la gestión de archivos, procesos, memoria y dispositivos, así como la seguridad del sistema. Permite que la computadora funcione eficientemente al coordinar la interacción entre el hardware y el software.
2. Historia
 - a. Primera generación
 - i. EXEC I
 - b. Segunda generación
 - i. EXEC II
 - ii. EXEC 8
 - iii. CTSS: No fue un sistema operativo influyente en con sus aspectos técnicos, tuvo una gran influencia al mostrar que el tiempo compartido era viable, las nuevas aplicaciones de los ordenadores fueron planteadas por primera vez entonces, y gracias a su sucesor, Multics, del que todos los sistemas operativos modernos toman planteamientos teóricos.
 - c. Tercera generación
 - i. OS/360: Fue un sistema operativo que recogía 3 programas simples.
 1. PCP (Primary Control Program) controlaba procesos secuencialmente por lotes.
 2. MFT (Multiprogramming with a Fixed number of Tasks) Permitía la multitarea, pero con un número fijo de ellas.
 3. MVT (Multiprogramming with a Variable number of Tasks)
 - ii. CP/CMS
 1. CP-40: Virtualización
 2. CP-67: CP-40 para OS/360
 3. CP-370: CP-67 para OS/370 (no salió), fue utilizado para IBM VM/370
 - iii. TOPS-10 (Timesharing / Total Operating System-10)
 - iv. MULTICS
 - v. UNIX
 - d. Cuarta generación
 - i. TOPS-20
 - ii. BSD UNIX
 - iii. VMS
 - e. Quinta Generación

- i. MS-DOS
- ii. Windows
- iii. XENIX
- iv. Novell Netware (NT)
- v. OS 400
- vi. Solaris
- vii. MAC OS

Década de 1940 - Sistemas Operativos Primitivos:

En los primeros días de las computadoras electrónicas, los usuarios interactuaban directamente con el hardware y programaban en lenguaje de máquina.

Las tarjetas perforadas y los interruptores manuales se utilizaban para cargar programas y datos.

Década de 1950 - Batch Processing:

Con el aumento de la complejidad de los programas, se introdujo el procesamiento por lotes (batch processing).

El sistema operativo ejecutaba secuencialmente un conjunto de trabajos sin intervención manual.

Década de 1960 - Multiprogramación y Tiempo Compartido:

IBM desarrolló el sistema operativo OS/360 para la serie de computadoras System/360, introduciendo conceptos de multiprogramación y sistemas operativos de gran envergadura.

Los sistemas operativos para tiempo compartido, como CTSS (Compatible Time-Sharing System), empezaron a permitir que varios usuarios interactuaran con la computadora simultáneamente.

Década de 1970 - Sistemas Operativos Distribuidos:

Se desarrollaron sistemas operativos para entornos distribuidos y redes, como el sistema operativo UNIX en los laboratorios Bell.

IBM lanzó el sistema operativo VM/370, que permitía la virtualización y la ejecución de múltiples sistemas operativos en una misma máquina.

Década de 1980 - PC y Sistemas Operativos Personales:

La llegada de las computadoras personales (PC) popularizó sistemas operativos como MS-DOS (Microsoft Disk Operating System) y posteriormente Microsoft Windows.

Apple lanzó el sistema operativo Macintosh System Software para sus computadoras Macintosh.

Década de 1990 - Sistemas Operativos Móviles y Linux:

Microsoft lanzó Windows 95, que introdujo una interfaz gráfica de usuario más amigable.

Linux, un sistema operativo de código abierto basado en UNIX, comenzó a ganar popularidad.

Surgieron sistemas operativos específicos para dispositivos móviles, como Palm OS y Windows CE.

Década de 2000 - Dominio de Sistemas Operativos Móviles:

La introducción del iPhone en 2007 impulsó el crecimiento de iOS y Android como sistemas operativos móviles dominantes.

Linux se consolidó como una opción poderosa para servidores y sistemas embebidos.

Década de 2010 - Continuación de Móviles y Auge de la Virtualización:

Android y iOS se convirtieron en los principales sistemas operativos para dispositivos móviles.

La virtualización se volvió prominente con tecnologías como VMware y Hyper-V.

3.

Tipos de sistemas operativos

Los sistemas operativos pueden clasificarse en función de varios criterios, y dos de ellos son el tiempo de respuesta y el número de usuarios. Aquí hay una breve descripción de los tipos de sistemas operativos según estos criterios:

1. Tiempos de respuesta:

Según el Tiempo de Respuesta:

- Sistemas de Tiempo Real (RTOS - Real-Time Operating Systems):
 - Características:
 - Diseñados para aplicaciones que requieren respuestas inmediatas.
 - Cumplen con estrictos plazos de tiempo.
 - Se utilizan en sistemas embebidos, controladores industriales, sistemas de navegación, etc.
- Sistemas de Tiempo Compartido (Time-sharing Systems):
 - Características:
 - Permiten a varios usuarios compartir simultáneamente los recursos de la computadora.
 - Proporcionan una rápida alternancia entre diferentes tareas.
 - Ejemplos: Unix, Linux, Windows (en entornos multiusuario).
- Sistemas de Tiempo Multiprogramado (Multiprogramming Systems):
 - Características:
 - Múltiples programas se cargan en la memoria al mismo tiempo.
 - Selecciona y ejecuta uno de los programas en función de las prioridades y recursos disponibles.
 - Optimizan la utilización del CPU.

2. Número de usuarios:

- Sistemas de Un Solo Usuario:
 - Características:
 - Diseñados para ser utilizados por una sola persona a la vez.
 - El sistema operativo gestiona las tareas del usuario sin la interferencia de otros usuarios.
- Sistemas Multiusuario:
 - Características:
 - Permiten que varios usuarios accedan y utilicen los recursos del sistema simultáneamente.
 - Proporcionan servicios de red para la comunicación entre usuarios y recursos compartidos.
- Sistemas de Tiempo Compartido (Time-sharing Systems):
 - Características:
 - Permite que varios usuarios compartan el tiempo del procesador de manera interactiva.

- Cada usuario tiene la sensación de tener la computadora para sí mismo durante su sesión.

3. Numero de procesos

Monoprogramacion:

La monoprogramación, también conocida como monotarea o ejecución única, es un modelo de ejecución de sistemas informáticos en el cual un solo programa o tarea se ejecuta en un sistema en un momento dado. En un entorno de monoprogramación, el sistema operativo asigna todos los recursos del sistema (como la CPU, la memoria y los dispositivos de entrada/salida) a un solo programa en ejecución hasta que esa tarea se complete o alcance un punto de espera, momento en el cual el sistema operativo puede cargar y ejecutar otro programa.

Multiprogramacion:

La multiprogramación es un modelo de ejecución de sistemas informáticos en el cual varios programas o tareas pueden ejecutarse simultáneamente en un sistema. En un entorno de multiprogramación, el sistema operativo asigna y comparte eficientemente los recursos del sistema, como la CPU, la memoria y los dispositivos de entrada/salida, entre varios programas en ejecución.

5.

Según su numero de procesadores

Monoproceso:

La expresión "sistema operativo monoproceso" se refiere a un sistema informático en el que solo se permite la ejecución de un proceso (o programa) a la vez. En otras palabras, en un sistema operativo monoproceso, solo se puede ejecutar una tarea o programa en un momento dado.

Multiproceso:

Un "sistema operativo multiproceso" es un tipo de sistema operativo diseñado para gestionar y ejecutar múltiples procesos simultáneamente en una computadora. Este enfoque es una extensión de la multiprogramación, donde varios programas pueden estar en memoria y listos para ejecutarse. La multiproceso lleva esto un paso más allá al permitir que múltiples procesos se ejecuten al mismo tiempo, compartiendo recursos del sistema como la CPU, la memoria y los dispositivos de entrada/salida.

Algunas características clave de un sistema operativo multiproceso incluyen:

- **Multitarea:** La capacidad de ejecutar varias tareas o procesos al mismo tiempo. Cada proceso puede tener su propio espacio de memoria y recursos asignados.
- **Concurrencia:** La habilidad de gestionar la ejecución concurrente de varios procesos, permitiendo que avancen de manera aparentemente simultánea.
- **Planificación de procesos:** El sistema operativo debe tomar decisiones sobre qué proceso ejecutar en cada momento y cómo asignar recursos para optimizar el rendimiento.
- **Gestión de memoria:** La administración eficiente de la memoria para permitir la ejecución de varios procesos sin interferencias indebidas.
- **Sincronización y comunicación:** Mecanismos para que los procesos se comuniquen y se sincronicen entre sí cuando sea necesario

Hay 2 tipos:

Simétrico:

Un "sistema operativo multiproceso simétrico" o "SMP" (por sus siglas en inglés, Symmetric Multiprocessing) es un tipo de sistema operativo diseñado para ejecutarse en hardware con múltiples procesadores idénticos que comparten la misma memoria principal y están conectados a través de un bus común. En un sistema SMP, todos los procesadores tienen acceso igualitario a la memoria y a otros recursos del sistema.

Las características principales de un sistema operativo multiproceso simétrico son:

- **Equidad en el Acceso:** Todos los procesadores tienen un acceso equitativo a la memoria y a otros recursos del sistema. No hay un procesador principal designado, y cada procesador puede ejecutar cualquier tarea o proceso.
- **Escalabilidad Lineal:** El rendimiento del sistema se puede mejorar añadiendo más procesadores, ya que el trabajo puede distribuirse entre ellos. Idealmente, la adición de un procesador debería resultar en un aumento proporcional en la capacidad de procesamiento.
- **Sincronización y Coordinación:** Se requiere una gestión eficiente de la sincronización y la coordinación entre los procesadores para evitar conflictos y garantizar la coherencia de la memoria compartida.
- **Alta Disponibilidad:** Si un procesador falla, los otros procesadores pueden continuar ejecutando tareas sin interrupciones significativas, lo que contribuye a la alta disponibilidad del sistema.

- **Balance de Carga Dinámico:** El sistema operativo debe ser capaz de distribuir dinámicamente las tareas entre los procesadores para mantener un equilibrio de carga y aprovechar eficientemente la capacidad de procesamiento.

Asimétrico:

A diferencia de un sistema operativo multiproceso simétrico (SMP), un "sistema operativo multiproceso asimétrico" implica que los procesadores en el sistema pueden tener roles y responsabilidades diferentes, y no todos tienen el mismo acceso o capacidad de ejecución de tareas. En un sistema asimétrico, los procesadores pueden ser especializados para realizar tipos específicos de tareas o funciones.

Algunas características clave de un sistema operativo multiproceso asimétrico son:

- **Jerarquía de Procesadores:** Los procesadores pueden clasificarse en función de sus capacidades o roles. Por ejemplo, puede haber un procesador principal (maestro) que gestiona tareas del sistema operativo y coordina la ejecución de otras tareas en procesadores secundarios (esclavos).
- **Diferencias en Recursos:** Los procesadores pueden tener diferentes velocidades de reloj, capacidades de caché o configuraciones de memoria. Estas diferencias permiten que algunos procesadores sean más adecuados para ciertos tipos de tareas que otros.
- **Gestión de Tareas Específicas:** En lugar de tener acceso equitativo a todas las tareas, ciertos procesadores pueden ser asignados específicamente para realizar tareas especializadas, como procesamiento gráfico, operaciones de red, o cálculos intensivos.
- **Escalabilidad Asimétrica:** A medida que se agregan más procesadores, el aumento de rendimiento puede no ser proporcional o uniforme. Algunos procesadores pueden contribuir más a tareas específicas que otros.
- **Flexibilidad en la Configuración:** La configuración asimétrica permite adaptarse a las necesidades específicas de las aplicaciones o tareas, asignando recursos según los requisitos particulares de cada tarea.

3.5 Trabajo en red

Los sistemas operativos que trabajan en red son aquellos diseñados para facilitar la comunicación y la interacción entre diferentes dispositivos en una red de computadoras. Estos sistemas operativos permiten compartir recursos,

como archivos e impresoras, y gestionar la comunicación entre los diversos nodos de la red.

Ejemplos de sistemas operativos de red y sus definiciones:

- Windows Server:

Definición: Windows Server es una versión del sistema operativo Microsoft Windows diseñada para entornos empresariales y de servidor. Ofrece funciones de red avanzadas y servicios como Active Directory para la gestión de usuarios y recursos en una red.

- Linux (varias distribuciones):

Definición: Linux es un sistema operativo de código abierto que tiene varias distribuciones diseñadas para trabajar en entornos de red. Algunas distribuciones populares para servidores incluyen Ubuntu Server, CentOS, y Debian. Linux es conocido por su estabilidad y rendimiento en entornos de red.

- Unix:

Definición: Unix es un sistema operativo multiusuario y multitarea que ha sido fundamental en el desarrollo de sistemas operativos modernos. Aunque no es tan común en entornos de escritorio, sigue siendo utilizado ampliamente en servidores y sistemas integrados.

- macOS Server:

Definición: macOS Server es la versión del sistema operativo macOS diseñada para ser utilizada como servidor. Ofrece funciones de red avanzadas y servicios como el Servidor de Archivos, el Servidor de Correo y otros servicios de red para entornos empresariales.

4.

Estructura de un sistema operativo

Un sistema operativo es un software que actúa como intermediario entre el hardware del ordenador y las aplicaciones de un usuario. Su función principal es gestionar los recursos del sistema y proporcionar servicios a las aplicaciones. La estructura de un sistema operativo generalmente se basa en:

1. **Kernel.** Es el núcleo del sistema operativo y se encarga de gestionar los recursos del hardware como la CPU o la memoria. Y proporciona servicios esenciales a aplicaciones.

2. **Gestor de procesos.** Se encarga de gestionar el tiempo de la CPU y de la comunicación entre procesos.
3. **Gestor de memoria.** Administra la memoria del sistema asignando espacio a los programas en ejecución y liberando espacio cuando no sea necesario, puede incluir técnicas de optimización como la paginación o la segmentación.
4. **Sistemas de archivos.** Proporciona una interfaz para el almacenamiento y la recuperación de datos en dispositivos de almacenamiento y organiza los archivos en directorios.
5. **Gestor de dispositivos.** Controla los dispositivos hardware y proporciona una interfaz para que los programas y el kernel puedan interactuar con los dispositivos.
6. **Interfaz de usuario.** Esta interfaz permite a los usuarios interactuar con el sistema operativo.
7. **Servicios del sistema.** Ofrece servicios adicionales a las aplicaciones y a los usuarios, como la gestión de usuarios o la seguridad.
8. **Controladores de dispositivos.** Son programas específicos que permiten la comunicación entre el sistema operativo y dispositivos hardware específicos.