

Unidad 1: Introducción a los sistemas operativos (parte 2)

Clasificación de los sistemas operativos

Por usuarios:

1. Monousuario: Solo soportan un usuario a la vez en ejecución por limitaciones definidas (ej: MS-DOS, AmigaOS, etc)
2. Multiusuario: Soporta múltiples usuarios en ejecución que comparten los mismos recursos. En este punto se necesita un sistema operativo que sea efectivo a la hora de realizar la distribución de dichos recursos (Ubuntu, familia Windows NT)

Por tarea:

1. Monotarea: Los sistemas monotarea son aquellos que sólo permiten una tarea a la vez por usuario. Puede darse el caso de un sistema multiusuario y monotarea, en el cual se admiten varios usuarios al mismo tiempo pero cada uno de ellos puede estar haciendo solo una tarea a la vez.
2. Multitarea: Un sistema operativo multitarea es aquél que le permite al usuario estar realizando varias labores al mismo tiempo. Por ejemplo, puede estar editando el código fuente de un programa durante su depuración mientras compila otro programa, a la vez que está recibiendo correo electrónico en un proceso en background. Es común encontrar en ellos interfaces gráficas orientadas al uso de menús y el ratón, lo cual permite un rápido intercambio entre las tareas para el usuario, mejorando su productividad.
 - a. **Tipos de multitarea:**
 - i. **Nula:** El sistema operativo es monotarea. El ejemplo típico es MS-DOS. Aún así puede lograrse a veces algo parecido a una multitarea implementando en espacio de usuario, o usando

trucos como los TSR de MS-DOS.

- ii. **Cooperativa:** Los procesos de usuario son quienes ceden la CPU al sistema operativo a intervalos regulares. Muy problemática, puesto que si el proceso de usuario se cuelga y no cede la CPU al sistema operativo, todo el sistema quedará entonces colgado. Da lugar también a latencias muy irregulares, y la imposibilidad de tener en cuenta este esquema en sistemas operativos de tiempo real. Un ejemplo sería Windows, hasta la versión 3.11.
- iii. **Preceptiva:** El sistema operativo es el encargado de administrar el/los procesador/es, repartiendo el tiempo de uso de este entre los procesos que estén esperando para utilizarlo. Cada proceso utiliza el procesador durante cortos periodos de tiempo, pero el resultado final es prácticamente igual que si estuviesen ejecutándose al mismo tiempo. Ejemplos de sistemas de este tipo serían Unix y clones (FreeBSD, Linux, etc.), VMS y derivados, AmigaOS, etc.
- iv. **Real:** Sólo se da en sistemas multiprocesador. Es aquella en la que varios procesos se ejecutan realmente al mismo tiempo, en distintos microprocesadores. Suele ser también preceptiva. Ejemplos de sistemas operativos con esa capacidad: Linux y OS X

Por número de procesadores:

UNIPROCESO

Un sistema operativo uniproceto es aquél que es capaz de manejar solamente un procesador de la computadora, de manera que si la computadora tuviese más de uno le sería inútil. El ejemplo más típico de este tipo de sistemas es el DOS y MacOS.

MULTIPROCESO

Un sistema operativo multiproceto se refiere al número de procesadores del sistema, que es más de uno y éste es capaz de usarlos todos para distribuir su carga de trabajo. Generalmente estos sistemas trabajan de dos formas: simétrica o

asimétricamente. Cuando se trabaja de manera asimétrica, el sistema operativo selecciona a uno de los procesadores el cual jugará el papel de procesador maestro y servirá como pivote para distribuir la carga a los demás procesadores, que reciben el nombre de esclavos. Cuando se trabaja de manera simétrica, los procesos o partes de ellos (threads) son enviados indistintamente a cualquiera de los procesadores disponibles, teniendo, teóricamente, una mejor distribución y equilibrio en la carga de trabajo bajo este esquema. Se dice que un thread es la parte activa en memoria y corriendo de un proceso, lo cual puede consistir de un área de memoria, un conjunto de registros con valores específicos, la pila y otros valores de contexto. Un aspecto importante a considerar en estos sistemas es la forma de crear aplicaciones para aprovechar los varios procesadores. Existen aplicaciones que fueron hechas para correr en sistemas monoproceso que no toman ninguna ventaja a menos que el sistema operativo o el compilador detecte secciones de código paralelizables, los cuales son ejecutados al mismo tiempo en procesadores diferentes. Por otro lado, el programador puede modificar sus algoritmos y aprovechar por sí mismo esta facilidad, pero esta última opción las más de las veces es costosa en horas hombre y muy tediosa, obligando al programador a ocupar tanto o más tiempo a la paralelización que a elaborar el algoritmo inicial.

Por tipo de Hardware:

1. Supercomputadoras: son aquellas computadoras diseñadas para realizar tareas específicas. Su costo es muy elevado y es utilizado por grandes empresas para investigación (astronómica, petrolera, etc) o comercio (minado de criptomonedas). Están compuestas por varias CPU y GPU trabajando en paralelo y su potencia se mide en FLOPS (Operaciones de coma flotante por segundo) y MIPS (Millones de instrucciones por segundo)
2. Mainframes: son computadoras de gran capacidad usadas por gobiernos y corporaciones para procesar gran cantidad de datos (censos, estadísticas, transacciones bancarias, tus datos de navegación, etc). Tienen una gran capacidad de conexión de dispositivos de E/S que son usualmente terminales tontas que proveen sus datos al mainframe. Tienen que tener una gran capacidad de manejo de procesos concurrentes para poder operar con

todas las entradas y salidas simultáneamente. Estos sistemas deben ser muy confiables y con una alta disponibilidad para mantener la utilidad de los datos recolectados. Además, deben ser sistemas mantenibles para poder ser reparados en caso de ser necesarios, en el menor tiempo posible.

3. Servidores: son computadoras basadas en la arquitectura cliente-servidor. Su costo es mayor al de una computadora común pero pueden ser virtualizados contratando un servicio de datacenter que mantenga los servidores. Se dedican a brindar un servicio a los potenciales clientes que lo vayan a consumir. Deben tener una alta disponibilidad para que los clientes no tengan dificultades en consumir el servicio. Además, deben tener hardware redundante para poder mantenerse en funcionamiento en caso de que falle alguno de sus componentes. En base al punto anterior podemos decir que los componentes de hardware deben ser hot swap para poder intercambiarlos fácilmente sin necesidad de apagar el servidor.
4. Tiempo real: son sistemas basados en la sincronización de procesos industriales. Los procesos ejecutados deben ser predecibles, controlados y carecer de la mayor parte de errores para maximizar la ganancia que se puede obtener de ese equipo industrial.
5. Computadoras personales: son sistemas para uso doméstico que se centran en una interfaz amigable para los usuarios y tienen (en la mayoría de los casos) un precio de hardware asequible. Sus usos son variados, desde ofimática hasta multimedia y gaming. Una de las mayores capacidades en las que se centran es que deben tener un buen tiempo de respuesta para ser útiles a los usuarios
6. Smartphone: son sistemas compactos que tienen que controlar múltiples periféricos de entrada y salida que son opcionales en las computadoras personales (cámara, micrófono, linterna, etc). Su costo es accesible para las personas comunes (excepto Iphone) y también tienen múltiples usos. Utilizan una arquitectura ARM lo que

implica cambios sustanciales en la implementación del kernel del sistema operativo

7. Embebidos: son sistemas baratos y dedicados a un dispositivo doméstico. Tienen funcionalidades capadas para la realización de tareas determinadas. Existen muchas dificultades para actualizar estos sistemas. Están presentes en múltiples dispositivos con o sin conexión a internet (lavadora, heladera, aire acondicionado, aspiradora robot, etc)
8. De tarjeta: son sistemas utilizados en tarjetas inteligentes (débito y crédito con chip, tarjeta STM, etc). Almacenan cantidades limitadas de información y dependen de otro dispositivo externo para poder funcionar.

Licencias

Una Licencia de Software es la autorización o permiso concedida por el autor para utilizar su obra de una forma convenida, habiendo marcado unos límites y derechos respecto a su uso.

La Licencia puede restringir el territorio de aplicación del programa, su plazo de duración o cualquier otra cláusula que el autor decida incluir.

Las licencias sobre obras intelectuales originales son una forma de protección proporcionada por las leyes vigentes que incluyen tanto los trabajos publicados como los pendientes de publicación, y otorgan al autor el derecho exclusivo para autorizar a otros a utilizar, modificar y/o redistribuir su obra original.

El autor del software puede autorizar o limitar el uso, modificación y/o redistribución de su obra inscribiéndose a un determinado tipo de licencia.

Tipos de licencia de software

El software propietario (también llamado software no libre, software privativo, software privado, software con propietario o software de

propiedad) se refiere a cualquier programa informático en el que los usuarios tienen limitadas las posibilidades de usarlo, modificarlo o distribuirlo (con o sin modificaciones), o cuyo código fuente no está disponible o el acceso a éste se encuentra restringido.

En el software no libre de una persona física o jurídica (compañía, corporación, fundación, etc.) posee los derechos de autor sobre un software negando o no otorgando, al mismo tiempo, los derechos de usar el programa con cualquier propósito; de estudiar cómo funciona el programa y adaptarlo a las propias necesidades; de distribuir copias; o de mejorar el programa y hacer públicas las mejoras (para esto el acceso al código fuente es un requisito previo).

De esta manera, un software sigue siendo no libre aún si el código fuente es hecho público, cuando se mantiene la reserva de derechos sobre el uso, modificación o distribución (por ejemplo, la versión comercial de SSH de Microsoft).

Freeware es un software de computadora que se distribuye sin cargo. A veces se incluye el código fuente, pero no es lo usual. El freeware suele incluir una licencia de uso, que permite su redistribución pero con algunas restricciones, como no modificar la aplicación en sí, ni venderla.

Shareware es una modalidad de distribución de software (juegos o programas) para que el mismo pueda ser evaluado de forma gratuita, pero generalmente por un tiempo especificado, aunque también las limitaciones pueden estar en algunas de las formas de uso o las capacidades finales. Para

adquirir una licencia de software que permite el uso del software de manera completa se requiere de un pago

El software libre es aquel que, una vez obtenido, puede ser usado, copiado, estudiado, modificado y redistribuido libremente. El software libre suele estar disponible gratuitamente en Internet, o a precio

de la distribución a través de otros medios; sin embargo no es obligatorio que sea así y, aunque conserve su carácter de libre, puede ser vendido comercialmente. El software libre, garantiza los derechos de modificación y redistribución de dichas versiones modificadas del programa. No debe confundirse "software libre" con software de dominio público. Éste último es aquél por el que no es necesario solicitar ninguna licencia y cuyos derechos de explotación son para toda la humanidad, porque pertenece a todos por igual.