

Sistemas Operativos

Índice

- [Máquina desnuda](#)
- [Concepto de Sistema Operativo](#)
- [Funciones del Sistema Operativo](#)
- [Componentes y estructura del Sistema Operativo](#)
- [Historia y evolución de los Sistemas Operativos](#)
- [Sistemas por lotes sencillos](#)
- [Sistemas por lotes multiprogramados](#)
- [Sistemas de tiempo compartido](#)
- [Sistemas de computador personal](#)
- [Sistemas paralelos](#)
- [Sistemas distribuidos](#)
- [Sistemas de tiempo real](#)

Máquina desnuda

- El término *máquina desnuda* se aplica a una computadora sin sistema operativo
 - Repite a alta velocidad la secuencia:
 - **lectura de instrucción máquina,**
 - **Incremento contador**
 - **Ejecución de la instrucción leída.**
 - Realiza la función si:
 - **Tiene en memoria principal un programa máquina específico para realizar dicha función**
 - **El registro contenga la dirección de comienzo del programa**

◆ La misión del S.O. Es completar (vestir) la máquina mediante una serie de

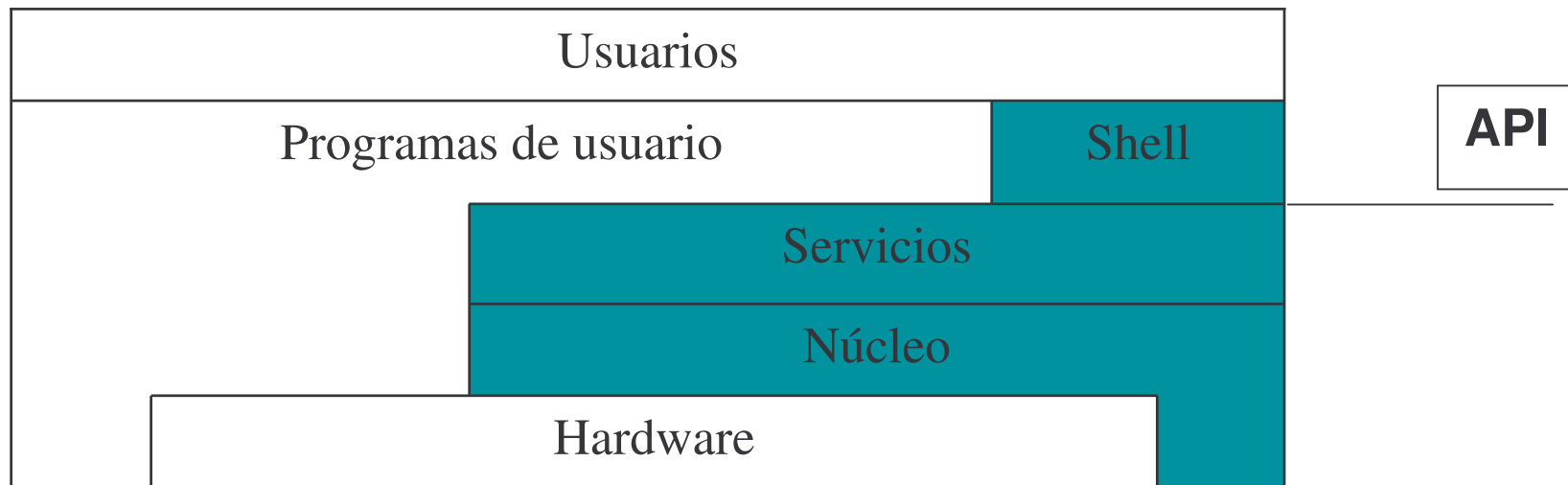
Concepto de Sistema Operativo

- *Un sistema operativo (S.O.) es un programa que tiene encomendadas una serie de funciones diferentes cuyo objetivo es simplificar el manejo y la utilización de la computadora, haciéndolo seguro y eficiente.*
- Históricamente se han ido agregando funciones al S.O. como interfaces gráficas, protocolos de comunicación, etc.

Funciones del Sistema Operativo

- Funciones clásicas:
 - Gestión de los recursos de la computadora
 - Ejecución de servicios para los programas
 - Ejecución de los mandatos de los usuarios

Componentes de un Sistema Operativo



- Núcleo o kernel, capa más cercana al hardware, encargada de gestionar los recursos hardware del sistema y suministra la funcionalidad básica del S.O.

- La capa de *servicios* o *llamadas al sistema* ofrece a los programas unos servicios en forma de interfaz de programación o API (application programming interface)
- Desde el punto de vista de los programas, esta capa extiende la funcionalidad de la computadora, se la denomina también máquina virtual extendida a los programas. Facilita la elaboración de estos.

- La capa de *intérprete de comandos* o *shell* suministra una interfaz a través de la cual el usuario puede dialogar en forma interactiva con la computadora.
- El shell recibe los mandatos del usuario, los interpreta y si puede los ejecuta.
- Como el shell se ejecuta en modo usuario a veces se lo considera que no forma parte del S.O., nosotros vamos a considerarlo como una parte de él.

El Sistema Operativo como gestor de recursos

- En una computadora suelen haber varios programas ejecutándose simultáneamente. El S.O. debe a cada uno:
 - Asignar recursos
 - Garantizar su protección de datos como de ejecución
 - Y suministrar información del uso de los recursos

Asignar recursos

- El S.O. se encarga de asignar recursos a los programas en ejecución, para conseguirlo debe:
 - Llevar estructuras que le permitan saber que recursos están libres y cuales están asignados a cada programa
 - Los recursos se asignan según los requerimientos de cada programa y según la prioridad de los mismos.
 - Recuperar los recursos luego de utilizados porque el programa comunica o porque este terminó de ejecutarse.

- Los recursos pueden ser:
 - *Físicos*:
 - **procesador,**
 - **memoria principal y**
 - **los periféricos**
 - *Lógicos*
 - **Archivos**
 - **Puertos de comunicación**

Protección

- S.O. debe garantizar la protección entre los usuarios del sistema.
- Asegurar la confidencialidad de la información y los trabajos no interfieran entre sí.
- Para conseguir esto debe impedir que los recursos asignados a un programa sean accedidos por otro.

Contabilidad

- La contabilidad permite medir la cantidad de recursos que utiliza cada programa.
- Se puede medir también la carga de utilización de cada recurso.
- Cuando la contabilidad se emplea sólo para conocer la carga de los componentes del sistema se la denomina *monitorización*.

S.O. como máquina extendida

- El S.O. ofrece a los programas un conjunto de servicios o *llamadas al sistema* que los pueden solicitar cuando necesiten, estos complementan los servicios de hardware (asignación de recursos).
- Los recursos se clasifican en:
 - Ejecución de programas
 - Operaciones de E/S
 - Operaciones sobre archivos
 - Detección y tratamiento de errores.

Ejecución de programas

- El S.O. incluye servicios para:
 - Lanzar la ejecución de un programa
 - Parar o abortar la ejecución
 - Conocer y modificar las condiciones de ejecución
 - Comunicar y sincronizar programas entre sí.

Órdenes de E/S

- Los servicios de E/S ofrecen una gran comodidad y protección al proveer a los programas de operaciones de lectura, escritura y modificación del estado de los periféricos.
- La programación de las operaciones de E/S es muy compleja y dependiente del hardware específico de cada periférico.
- Los servicios del S.O. Ofrecen un alto nivel de abstracción de forma que el programador de aplicaciones no tenga que preocuparse de esos detalles.

Operaciones sobre archivos

- Los archivos ofrecen un nivel de abstracción mayor que el de las órdenes de E/S, permitiendo operaciones como:
 - Creación
 - Borrado,
 - Renombrado
 - Apertura
 - Escritura y
 - Lectura de archivos

Detección y tratamiento de errores

- Además de analizar todas las órdenes que recibe el S.O. se encarga de tratar todas las condiciones de error que detecte el hardware.
- Las condiciones de error que pueden aparecer son:
 - Errores en las operaciones de E/S
 - Errores de paridad en los accesos a memoria o en los buses
 - Errores de ejecución en los programas, como desbordamientos, violaciones de memoria, código de instrucción prohibidos

El S.O. Como interfaz de usuario

- Módulo del sistema que permite la interacción con el usuario es el Shell o intérprete de comandos.
- El shell se comporta como un bucle infinito que está repitiendo lo siguiente:
 - Espera una orden del usuario.
 - **En interfaz textual, está pendiente de lo que escribe el usuario en la línea de comandos.**
 - **En la interfaz gráfica está pendiente de los eventos del apuntador (ratón) además del teclado.**
 - Analiza la orden, en caso de ser correcta la ejecuta empleando los servicios del S.O.
 - Concluida la orden vuelve a la espera

Componentes y estructura del Sistema Operativo

- Componentes del Sistema Operativo



Componentes del S.O.

- Los servicios se agrupan según su funcionalidad en:
 - Gestión de procesos:
 - **encargada de la creación, planificación y destrucción de procesos.**
 - Gestión de memoria:
 - **Encargada de llevar el registro de las direcciones libres y de las ocupadas, así mismo de la asignación y liberación de memoria.**

- Gestión de E/S:
 - **Facilita el manejo de los dispositivos periféricos.**
- Gestión de archivos y directorios:
 - **Manejo de archivos y directorios, también de administrar el almacenamiento secundario.**
- Comunicación y sincronización entre procesos:
 - **Crea mecanismos para que los procesos puedan comunicarse y sincronizarse.**
- Seguridad y protección:
 - **Garantiza la identidad de los usuarios y los derechos de los mismos.**

Estructura de los S.O.

- Un S.O. es un programa grande y complejo que está compuesto, por una serie de componentes con funciones bien definidas. Cada S.O. Estructura estos componentes de distinta forma.
- En función de su estructura se clasifican en: monolíticos y estructurados.

S.O. monolíticos

- No tienen estructura clara y bien definida.
- Todos los componentes integrados en un único programa que ejecuta en un solo espacio de direcciones.
- Todas las funciones se ejecutan en modo núcleo únicamente.

- Surgieron de S.O. sencillos y pequeños a los que se les ha ido añadiendo funciones. Esto los llevó a convertirse en programas grandes y complejos formados por muchas funciones todas al mismo nivel.
- Ejemplos son MS-DOS y UNIX
- Defecto:
 - La complejidad que representa modificarlo para añadirle nuevas funcionalidades y servicios pues se trata de un gran programa con muchas funciones que se llaman entre sí y sin ocultamiento.

S.O. Estructurados

- Habitualmente se utilizan estos dos tipos para que el sistema tenga cierta estructura:
 - Sistema por capas
 - Sistema cliente-servidor

Sistema por capas

- Organización de jerarquía por capas.
- Cada capa ofrece una interfaz clara y bien definida a la capa superior
- Cada capa utiliza los servicios que le brinda la capa inferior.
- Ventaja: modularidad y ocultamiento de la información. Facilita la depuración y verificación del sistema.
- Ejemplo: el S.O. OS/2 que es descendiente del MS-DOS.

Modelo cliente-servidor

- Se implementan la mayor parte de los servicios y funciones en modo usuario. A los procesos que ejecutan estos servicios se les llama *servidores*.
- Muy pocas funciones se ejecutan en modo núcleo, llamada *micronúcleo*.
- Cada servidor y también el micronúcleo se pueden desarrollar por separado.

- Por lo general las funciones del micronúcleo son:
 - Gestión de interrupciones, gestión básica de procesos y de memoria y servicios básicos de comunicación entre procesos.
- Para solicitar un servicio en este tipo de sistemas, el proceso usuario solicita el servicio al servidor correspondiente. Puede suceder, además que este servidor necesite solicitar a otro servidor una función que él no la tiene, en este caso se convierte en cliente del nuevo servicio.

- Ventajas:
 - Gran flexibilidad.
 - Cada proceso servidor se ocupa de una funcionalidad concreta, entonces son procesos pequeños y manejables. Fácil desarrollo y depuración.
- Desventajas:
 - Mayor sobrecarga en el tratamiento de los servicios que en los monolíticos, porque cada componente se ejecuta en un espacio de direcciones distinto, lo que hace que su activación sea requiera más tiempo.
- Ejemplos: Minix, Mach, Amoeba, Windows NT.



Historia y evolución de los Sistemas Operativos.

Historia de los Sistemas Operativos

- Introducción
 - Los sistemas operativos han evolucionado estando relacionados con la arquitectura de las computadoras.
 - La primera computadora digital real fue diseñada por el matemático Charles Babbage (1792-1871), diseño puramente mecánico y por lo tanto sin S/O.

Historia de los Sistemas Operativos

- La segunda generación(1955-1965): Transistores y sistemas de procesamiento por lotes.
- Características:
 - Computadoras confiables y operaban en cuartos c/aire acondicionado.
 - Clara separación entre: diseñadores, constructores, operadores, programadores y personal de mantenimiento.

Historia de los Sistemas Operativos

- La primera generación (1945-1955): Bulbos y conexiones.
 - Toda la programación se llevaba a cabo en lenguaje de máquina absoluto y se utilizaban conexiones para controlar las funciones básica de la máquina.
 - Los lenguajes de programación y los sistemas operativos eran desconocidos



Sistemas por lotes sencillos

Características generales

- Las primeras computadoras eran máquinas enormes (físicamente) que se controlaban desde una consola.
- Los dispositivos de entrada comunes eran lectores de tarjetas y unidades de cinta.
- Los dispositivos de salida comunes eran impresoras de líneas, unidades de cinta y perforadoras de tarjetas.
- Los usuarios de tales sistemas no interactuaban directamente con las computadoras; más bien, el usuario preparaba un trabajo - que consistía en el programa, los datos, y cierta información de control acerca de la naturaleza del trabajo (tarjetas de control) - y lo entregaba al operador de la computadora.

Características generales

- El trabajo casi siempre se presentaba en forma de tarjetas perforadas.
- En algún momento posterior (minutos, horas o días después) aparecía la salida, que consistía en el resultado del programa junto con un vaciado de la memoria y los registros en caso de haber un error de programa.
- El sistema operativo en estas primeras computadoras era sencillo. Su principal obligación era transferir el control automáticamente de un trabajo al siguiente.
- El sistema operativo siempre estaba residente en memoria.

Estructura de un trabajo en esos tiempos:

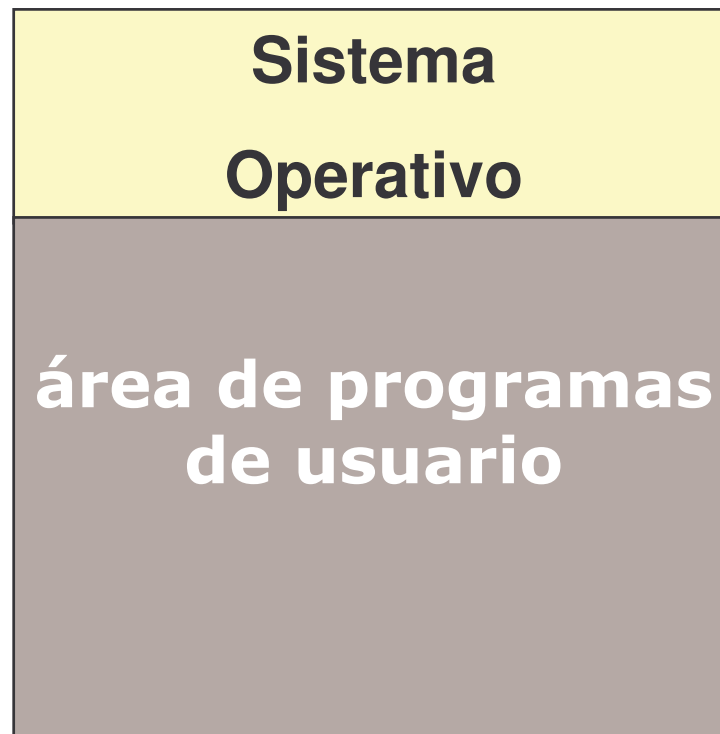
- El programador escribe el programa en hojas (FORTAN o ensamblador).
- Perfora las tarjetas
- lleva el paquete de tarjetas perforadas al cuarto de lectura.
- Un operador toma uno de los paquetes de tarjetas del cuarto de lectura y lo lee.
- Si es necesario utilizar el compilador Fortan, deberá ir al gabinete de archivos y ejecutarlo.
- Luego la computadora ejecuta el programa.
- El operador pasa a la impresora el grupo de tarjetas perforadas, separa la salida y lleva al cuarto de salida.
- **Lo que podemos observar que se pierde mucho tiempo de cómputo caminando por la sala.**

Características Generales

- Un sistema operativo por lotes normalmente lee un flujo de trabajos individuales (de un lector de tarjetas, por ejemplo), cada uno con sus propias tarjetas de control que predefinen lo que el trabajo hace.
Una vez terminado el trabajo, su salida generalmente se imprime.
La característica definitiva de este sistema es la falta de interacción entre el usuario y el trabajo mientras éste se ejecuta.
- El trabajo se prepara y se entrega, y cierto tiempo después aparece la salida.

- El retardo entre la presentación de un trabajo y su terminación (llamado tiempo de retorno) puede ser el resultado del volumen de cómputo requerido, o de retardos antes de que el sistema operativo comience a procesar el trabajo.
- En este entorno de ejecución, la CPU con frecuencia está ociosa. Esta inactividad ocurre porque los dispositivos de E/S mecánicos son más lentos que los dispositivos electrónicos.. Incluso una CPU lenta realiza sus operaciones en tiempo del orden de microsegundos, y ejecuta miles de instrucciones cada segundo. Un lector de tarjetas rápido lee 1200 tarjetas por minuto, 20 tarjetas por segundo.

Organización de la memoria para un sistema por lotes sencillos

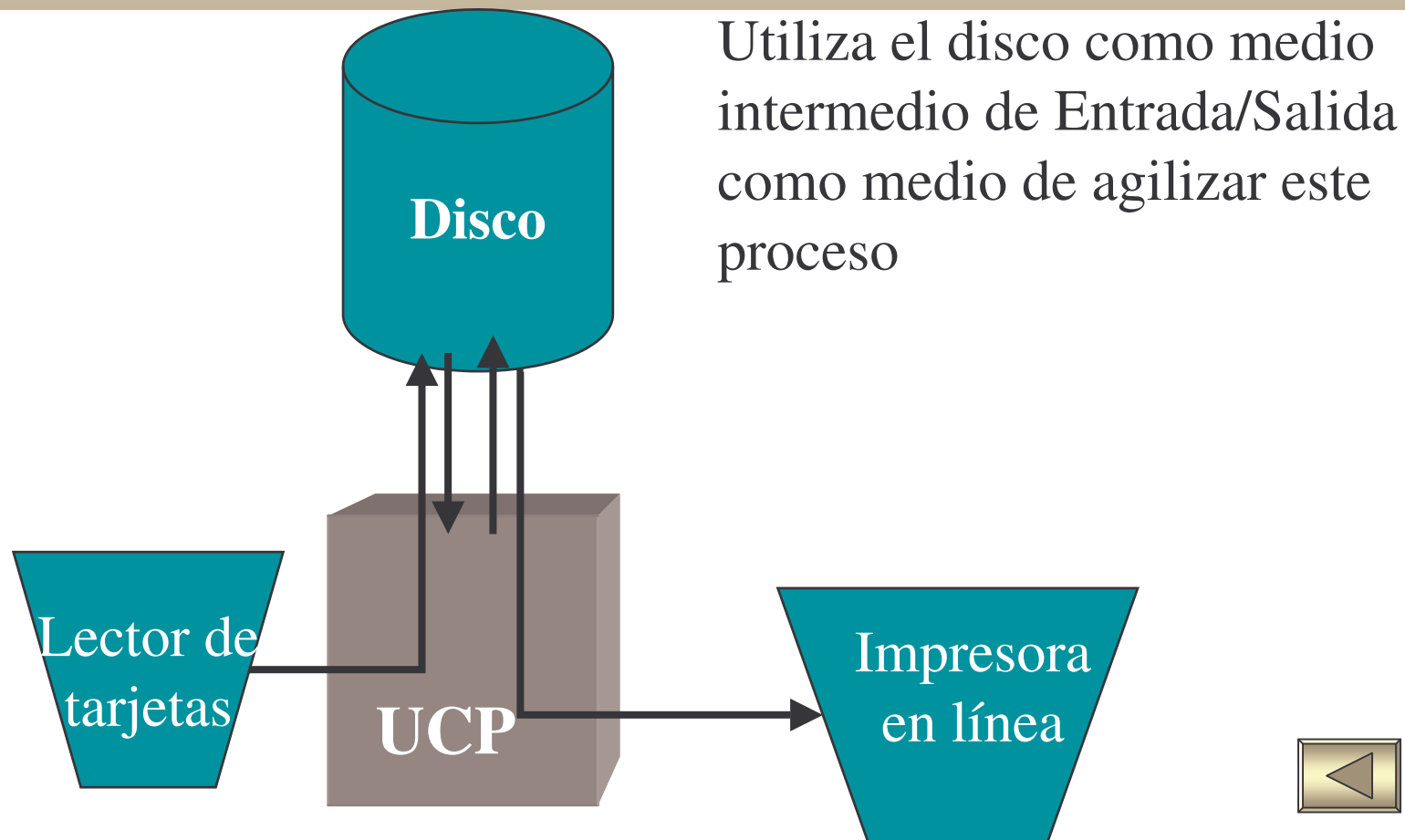


- El esquema nos muestra que existe un solo trabajo en la memoria, cuando lo ejecuta la UCP espera a que se imprima para leer el nuevo trabajo.

- Con la tecnología de discos, en vez de leer los trabajos desde tarjetas directamente del lector a la memoria, para después procesarla. Se leen las tarjetas y se graban los trabajos en el disco y luego se leen del disco a la memoria, con lo que se consigue bajar los retardos de procesamiento.
- Cuando el trabajo solicita la impresión de una línea, es línea se copia en un buffer del sistema y se escribe en el disco. Una vez que el trabajo termina, la salida se imprime realmente.
- **Esta forma de procesamiento, se llama *spooling*.**
(*simultaneous peripheral operation on-line*)

- En esencia, el disco se utiliza como *buffer* de gran tamaño, para leer por adelantado hasta donde sea posible de los dispositivos de entrada, y para guardar los archivos de salida hasta que los dispositivos de salida puedan aceptarlos.
- El *spooling* también sirve para procesar datos en sitios remotos.

Spooling



Sistemas por lotes multiprogramados

Características generales

- El *spooling* da origen a una importante estructura de datos: la *reserva de trabajos*.
- Como resultado del *spooling*, puede haber varios trabajos ya leídos esperando en el disco, listo para ejecutarse.
- Al tener reserva de trabajos en disco, el sistema operativo puede escoger cuál trabajo ejecutará a continuación, a fin de mejorar el aprovechamiento de la CPU.
- Si los trabajos llegan directamente en tarjetas o incluso en cinta magnética, no es posible ejecutarlos en un orden distinto. Los trabajos se deben ejecutar secuencialmente, bajo un régimen de servicio por orden de llegada.

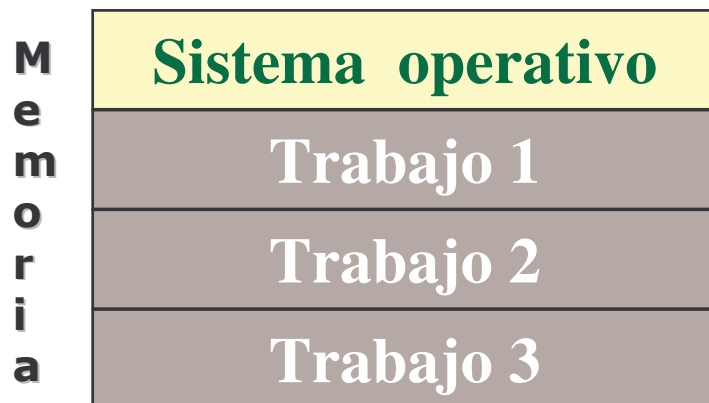
Características generales

- En cambio, si varios trabajos están en un dispositivo de acceso directo, como un disco, es posible la *planificación de trabajos*.
- El aspecto más importante de la planificación de trabajos es la capacidad de *multiprogramar*.
- La operación fuera de línea y el *spooling* para traslapar la E/S tienen sus limitaciones. En general, un solo usuario no puede mantener ni la CPU ni los dispositivos de E/S ocupados todo el tiempo.
- La multiprogramación aumenta el aprovechamiento de la XPU organizando los trabajos de tal forma que la CPU siempre tenga uno para ejecutar.

Características generales

- **El concepto es el siguiente:** El sistema operativo mantiene varios trabajos en la memoria a la vez. Este conjunto de trabajos es un subconjunto de los que se mantienen en la reserva de trabajos (puesto que el número de trabajos que se pueden mantener simultáneamente es la memoria generalmente es mucho menor que el número de trabajos que pueden estar en la reserva).
- El sistema operativo escoge uno de los trabajos que están en la memoria y comienza a ejecutarlo. Tarde o temprano, el trabajo tendrá que esperar mientras se lleva a cabo alguna tarea, como, el montaje de una cinta o la terminación de una operación de E/S.

Organización de la memoria en un sistema multiprogramado



- Se particiona la memoria para colocar en cada una un trabajo, que será utilizado por la UCP, cada vez que el trabajo activo utilice los recursos de E/S

Características generales

- En un sistema sin multiprogramación, la CPU estaría ociosa.
- En un sistema multiprogramado, el sistema operativo simplemente selecciona otro trabajo y lo ejecuta. Cuando *ese* trabajo necesita esperar, la CPU se conmuta a otro trabajo, y así sucesivamente.
- En algún momento el primer trabajo terminará su espera y recuperará la CPU. En tanto haya algún trabajo que ejecutar, la CPU nunca estará ociosa.

Ejemplo vida real

- Esta idea es común a otras situaciones de la vida real. Un abogado no tiene un solo cliente a la vez. Más bien, varios clientes pueden estar siendo atendidos al mismo tiempo.
- Mientras un caso está esperando ir a juicio o la preparación de documentos, el abogado puede trabajar en otro caso. Si tiene suficientes clientes, el abogado nunca tendrá que estar ocioso. (Los abogados ociosos tienden a convertirse en políticos por lo que mantener a los abogados ocupados tiene cierto valor social)

Características generales

- La multiprogramación es el primer caso en que el sistema operativo debe tomar decisiones por los usuarios. Por ello, los sistemas operativos multiprogramados son relativamente complejos.
- Todos los trabajos que ingresan en el sistema se mantienen en un área auxiliar de trabajos. Esta área auxiliar consiste en todos los procesos que residen en el almacenamiento masivo y que esperan la asignación de espacio en la memoria principal.

Características generales

- Si varios trabajos están listos para colocarse en la memoria, y no hay espacio suficiente para todos, el sistema deberá escoger algunos de entre ellos.
- La toma de esta decisión es la planificación de trabajos.
- Cuando el sistema operativo escoge un trabajo del área auxiliar, lo carga en la memoria para ejecutarlo.
- Tener varios programas en la memoria al mismo tiempo requiere alguna forma de gestión de memoria.
- Además, si varios trabajos están listos para ejecutarse al mismo tiempo, el sistema debe escoger entre ellos, esta toma de decisión se llama: planificación de la CPU.

Características generales

- Por último, la ejecución concurrente de múltiples trabajos requiere limitar su capacidad para afectarse mutuamente en todas las fases del sistema operativo, incluidos la planificación de procesos, el almacenamiento en disco y la gestión de memoria.



Sistemas de tiempo compartido

Características generales

- Los sistemas por lotes multiprogramados proporcionan un entorno en el que los distintos recursos del sistema (por ej, CPU, memoria, dispositivos periféricos) se aprovechan de manera efectiva.
- Los sistemas por lotes impiden que usuarios interactúen con los trabajos mientras estos se ejecutan. En muchos casos la ejecución de un paso depende del resultado del paso anterior y no siempre es posible predecir todas las posibilidades necesarias.
- Los programadores no pueden modificar un programa durante su ejecución a fin de estudiar su comportamiento, sólo lo pueden depurar estáticamente.

- El tiempo compartido, o *multitarea*, es una consecuencia lógica de la multiprogramación.
- Se ejecutan múltiples trabajos mientras la UCP se conmuta entre ellos, pero la conmutación es tan frecuente que los usuarios pueden interactuar con cada programa durante su ejecución.
- Un sistema *interactivo*, o *manual*, permite la comunicación en línea entre el usuario y el sistema.

Un usuario da instrucciones al sistema Operativo, o una aplicación y obtiene una respuesta inmediata. Cuando el Sistema Operativo termina de ejecutar una orden, busca el siguiente “enunciado de control” del teclado del usuario, y así sucesivamente. Para el usuario es fácil experimentar, y puede ver los resultados de inmediato.

- Para que los usuarios puedan acceder con comodidad tanto a los datos como al código, deben contar con un sistema de archivos en línea.
- Un archivo es una colección de información relacionada definida por su creador.
- Por lo regular los archivos representan programas y datos.
- Los archivos de datos suelen ser: numéricos, alfabéticos o alfanuméricos.
- Los archivos pueden ser de forma libre, como los de texto o tener formato rígido.

- En general, un archivos es una secuencia de bit, bytes, líneas o registros cuyo significado ha sido definido por su creador y el usuario.
- El sistema operativo crea el concepto abstracto de archivo para administrar los dispositivos de almacenamiento masivo.
- Los archivos normalmente se organizan en grupos lógicos, o *directorios* para facilitar su localización y acceso. Como varios usuarios pueden acceder a los archivos, es necesario definir quienes pueden hacerlo y de qué forma.

- Los sistemas por lote son apropiados para ejecutar trabajos grandes que casi no necesitan interacción.
- Los trabajos interactivos suelen consistir en muchas acciones cortas. El usuario introduce la orden y espera los resultados, por lo tanto, el tiempo de *respuesta* debe ser corto, del orden de segundos. Se emplea un sistema interactivo cuando se requiere un tiempo de respuesta corto.

- Los sistemas de tiempo compartido se crearon para brindar el uso interactivo de un sistema de computadora a un costo razonable.
- Un sistema operativo de tiempo compartido utiliza:
 - planificación de UCP
 - multiprogramación.

para ofrecer a cada usuario una pequeña porción del tiempo de la UCP



Sistemas de computadora personal

Características

- Microcomputadores más pequeños y económicos que los sistemas macrocomputadores(mainframes)
- UCP carecían de protección de los programas de usuarios.
- No eran ni multiusuario ni multitarea
- Optimización de la comodidad del usuario en contraposición con la optimización de la UPC y los periféricos.

Características 2

- Los S.O. adoptaron funciones de los S.O. creados para macrocomputadores, también adelantos técnicos de los mismos.
- Costos bajos del hardware posibilitó que el aprovechamiento de la UCP dejara de ser la preocupación central. También la protección de archivos no tomó tanta importancia.
- Los costos bajos del hardware posibilita implementar conceptos de S.O. avanzados en un número mayor de sistemas.



Sistemas paralelos

Sistemas Distribuidos

Características generales

- Una tendencia reciente en los sistemas de computación es distribuir el cómputo entre varios procesadores.
- En contraste con los sistemas fuertemente acoplados (Sistemas Paralelos) los procesadores no comparten la memoria ni el reloj. Cada procesador tiene su propia memoria local
- Los procesadores se comunican entre sí a través de diversas líneas de comunicación, como buses de alta velocidad o líneas telefónicas.
- Los sistemas con estas características decimos que son débilmente acoplados o distribuidos

Características generales

- Los procesadores de un sistemas distribuido pueden tener diferentes tamaños y funciones; pueden incluir microprocesadores pequeños, estaciones de trabajo, minicomputadoras y sistemas de computadora de propósito general grandes.
- Tales procesadores reciben nombres distintos: sitios, nodos, computadoras, etc.. dependiendo del contexto en el que se mencionen

¿Por qué construirlos?

- Recursos compartidos:
 - Si varios sitios distintos (con diferentes capacidades) se conectan entre sí, un usuario de un sitio podría aprovechar los recursos disponibles en otro.
 - El uso de recursos compartidos en un sistema distribuido ofrece mecanismos para compartir archivos en sitios remotos, procesar información de una base de datos distribuida, imprimir archivos en sitios remotos, usar equipos especializados remotos, etc....

¿Por qué construirlos?-2

- Computación más rápida.
 - Si un cálculo dado se puede dividir en varios subcálculos susceptibles de ejecución concurrente, un sistema distribuido podría permitirnos distribuir el cálculo entre los distintos sitios, y ejecutarlo de forma concurrente.
 - Además, si un sitio en particular actualmente está sobrecargado de trabajos, algunos de ellos podrían transferirse a otros sitios cuya carga sea más ligera. Esta transferencia de trabajos se llama: carga compartida.

¿Por qué construirlos?-3

- Confiabilidad.
 - Si un sitio de un Sistema Operativo Distribuido falla, los sitios restantes podrían seguir funcionando. Si el sistema se compone de varias instalaciones automáticas grandes (computadoras de propósito general grandes), el fallo de uno no deberá afectar a los demás.
 - Por otro lado, si el sistema consiste en varias máquinas, cada una con una función crucial del sistema (como E/S de caracteres en terminales, o el sistema de archivos), un sólo fallo podría detener todo el sistema.
 - En general, si existe suficiente redundancia en el sistema (tanto en hardware como de datos), éste puede continuar con su operación, incluso si algunos de sus sitios han fallado.

¿Por qué construirlos?-4

- Comunicación.
 - Hay muchos casos en que los programas necesitan intercambiar datos con otros programas del mismo sistema. Los sistemas de ventanas son un ejemplo, ya que a menudo comparten datos o transfieren datos entre presentaciones.
 - Si muchos sitios están conectados a través de una red de comunicaciones, los procesos de diferentes sitios tienen la oportunidad de intercambiar información. Los usuarios podrían comunicarse entre sí por correo electrónico, en el mismo sitio o en uno distinto.



Sistemas de tiempo real

Características generales

- Otra forma de Sistema Operativo de propósito general es el: Sistema Operativo de tiempo real.
- Se usa un sistema de tiempo real cuando los requisitos de tiempo de la operación de un procesador o del flujo de datos son estrictos; por ello, a menudo se utilizan como dispositivos de control en aplicaciones dedicadas.
- Los sensores envían datos a la computadora, la cual debe analizar esos datos y posiblemente ajustar controles a fin de modificar las entradas de los sensores.

Características generales

- Los sistemas que controlan experimentos científicos, los que producen imágenes médicas, los de control industrial, los de control de combustible a inyección en los autos, controladores de aparatos domésticos, sistemas de armamentos y algunos sistemas de exhibición son sistemas de tiempo real.

Características generales

- Un sistema operativo de tiempo real tiene restricciones de tiempo fijas bien definidas. El procesamiento debe efectuarse dentro de los intervalos definidos, o el sistema fallará.
- Por ejemplo, no es conveniente ordenar a un brazo robot que se detenga después de haber chocado con el automóvil que está construyendo.
- Se considera que un sistema de tiempo real está funcionando correctamente sólo si produce el resultado correcto dentro de los intervalos de tiempo estipulados.

Características generales

- Podemos contrastar este requisito con un sistema de tiempo compartido, en el que es deseable (pero no obligatorio) responder rápidamente, o con un sistema por lotes, en el que tal vez no haya restricciones de tiempo.

Clasificación

- Hay dos tipos de sistemas de tiempo real.
 - Un sistema de tiempo real duro.
 - Un sistema de tiempo real blando.

Sistema de tiempo real duro

- Un sistema de tiempo real duro garantiza que las tareas críticas se terminarán a tiempo.
- Este objetivo requiere que todos los retardos del sistema estén limitados, desde la obtención de datos almacenados hasta el tiempo que el sistema operativo tarda en atender cada solicitud que se le presenta.
- Tales restricciones de tiempo determinan los recursos que están disponibles en este tipo de sistemas.
- El almacenamiento secundario de cualquier índole suele estar limitado o ausente, y los datos se almacenan de preferencia en memoria de corto plazo o en memoria sólo de lectura (ROM).

Sistema de tiempo real duro

- La ROM se encuentra en dispositivos de almacenamiento no volátil que conservan su contenido aún en caso de fallar el suministro de electricidad; casi todos los demás tipos de memoria son volátiles.
- También está ausente la mayor parte de las funciones avanzadas de los sistemas operativos, ya que tienden a separar al usuario aún más del hardware, y tal separación causa incertidumbre acerca del tiempo que una operación tarda.

Sistema de tiempo real duro

- Por ejemplo, los sistemas de tiempo real casi nunca tienen memoria virtual.
- Por ello, los sistemas de tiempo real duros son incompatibles con el funcionamiento de los sistemas de tiempo compartido y no pueden combinarse con ellos. Puesto que ninguno de los sistemas operativos de propósito general existentes apoya la funcionalidad de tiempo real dura, no se estudiará en este curso.

. Sistema de tiempo real blando

- Un tipo menos restrictivo de sistema de tiempo real es el de tiempo real blando, en el que una tarea de tiempo real crítica goza de prioridad respecto a otras tareas, y conserva esa prioridad hasta que se lleva a cabo.
- Al igual que en los sistemas de tiempo real duros, es preciso limitar los retardos del núcleo: no es posible mantener a una tarea de tiempo real esperando indefinidamente a que el núcleo la ejecute.
- El tiempo real blando es una meta alcanzable que puede combinarse con otros tipos de sistemas.

Sistema de tiempo real blando

- Los sistemas operativos de tiempo real blandos, tienen una utilidad más limitada que los duros.
- En vista de que no apoyan el cumplimiento estricto de plazos, es riesgoso utilizarlos en control industrial y robótica, aunque hay varias áreas en las que pueden ser útiles, como multimedia, realidad virtual y proyectos científicos avanzados como la exploración submarina y planetaria.
- Estos sistemas requieren características avanzadas de los sistemas operativos que no pueden incluirse en los sistemas de tiempo real duros.

Sistema de tiempo real blando

- La proliferación del uso de funciones de tiempo real blando ha hecho que se incluyan en la mayor parte de los sistemas operativos actuales, incluidas versiones importantes de UNIX y Windows NT



Ejercicios

- Mencione los tres propósitos principales de un sistema operativo.
- Un método extremo de spooling, llamado activación de cinta, consiste en leer todo el contenido de una cinta magnética y colocarlo en disco antes de usarlo. Analice la ventaja principal de tal esquema.
- En un entorno de multiprogramación y tiempo compartido, varios usuarios comparten el sistema simultáneamente. Esta situación puede dar pie a varios problemas de seguridad.
 - Mencione dos de esos problemas.
 - ¿Es posible alcanzar el mismo grado de seguridad en una máquina de tiempo compartido que en una máquina dedicada? Explique su respuesta.

- Mencione la principal ventaja de la multiprogramación.
- Cite las principales diferencias entre los sistemas operativos para macrocomputadoras y para computadoras personales.
- Defina las propiedades esenciales de los siguientes tipos de sistemas operativos:
 - Por lotes.
 - Interactivos
 - De tiempo compartido
 - De tiempo real
 - Distribuidos.
- ¿En qué circunstancias le convendría más a un usuario utilizar un sistema de tiempo compartido en vez de una computadora personal a una estación de trabajo monousuario?
- ¿Por qué son deseables los sistemas distribuidos?

