

Estructuras de Sistemas Operativos

Definición de Sistema Operativos

Un sistema operativo es un programa que actúa como inter entre el usuario y el hardware de un computador y su propósito es proporcionar un entorno en el cual el usuario puede ejecutar programas. El objetivo principal de un sistema operativo es lograr que el sistema de computación se use de manera cómoda y el objetivo secundario es que el hardware del computador se emplee de manera eficiente

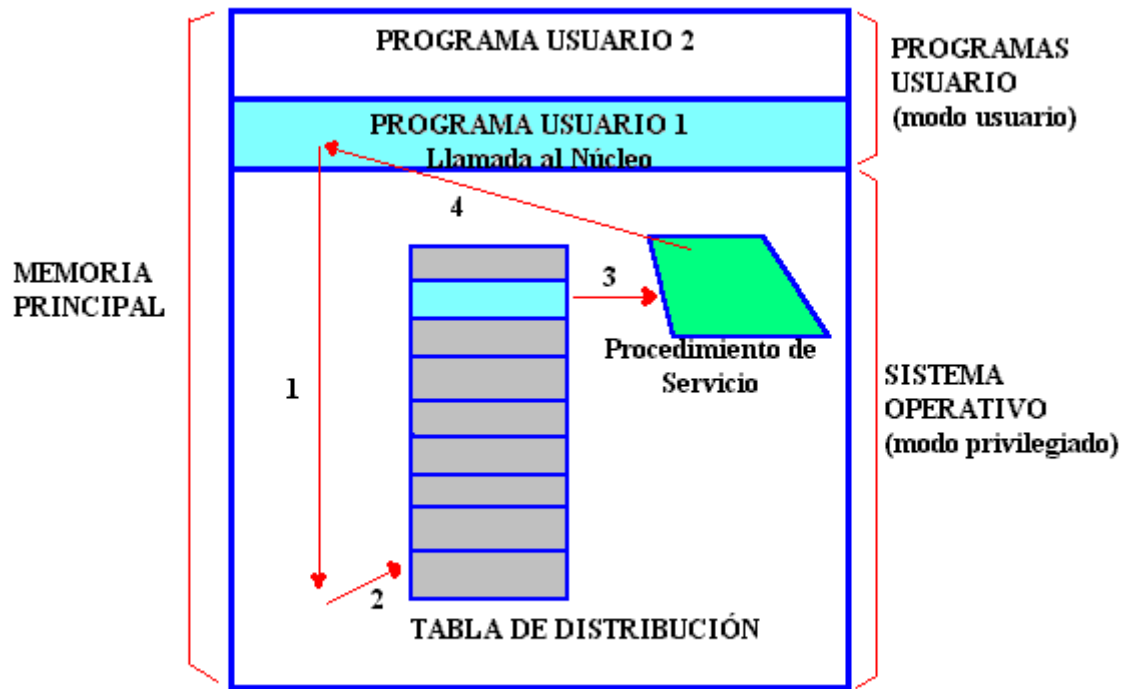
Sistemas monolíticos

Este tipo de organización es, con diferencia, la más común. El sistema operativo se escribe como una colección de procedimientos, cada uno de los cuales puede llamar a los demás cada vez que así lo requiera. Cuando se usa esta técnica, cada procedimiento del sistema tiene una interfaz bien definida en términos de *parámetros y resultados*, y cada uno de ellos es libre de llamar a cualquier otro, si éste último proporciona un cálculo útil para el primero. Para construir el programa objeto real del sistema operativo siguiendo este punto de vista, se compilan de forma individual los procedimientos, o los ficheros que contienen los procedimientos, y después se enlazan en un sólo fichero objeto con el enlazador. En términos de ocultación de la información, ésta es prácticamente nula: cada procedimiento es visible a los demás (en contraste con una estructura con módulos o paquetes, en la que la mayoría de la información es local a un módulo, y donde sólo los datos señalados de forma expresa pueden ser llamados desde el exterior del módulo).

Los servicios (mediante llamadas al sistema) que proporciona el sistema operativo se solicitan colocando los parámetros en lugares bien definidos, como los registros o la pila, para después ejecutar una instrucción especial de trampa, a veces referida como llamada al núcleo o llamada al supervisor. Esta instrucción cambia la máquina del modo usuario al modo núcleo (también conocido como modo supervisor), y transfiere el control al sistema operativo, lo que se muestra en el evento (1) de la próxima figura

El sistema operativo examina entonces los parámetros de la llamada para determinar cual de ellas se desea realizar, como se muestra en (2) de la figura.

A continuación, el sistema operativo analiza una tabla que contiene en la entrada k un apuntador al procedimiento que implementa la k -ésima llamada al sistema. Esta operación, que se muestra en (3) de la figura, identifica el procedimiento de servicio, al cual se llama. Por último, la llamada al sistema termina y el control vuelve al programa del usuario.



Sistemas Monolíticos - 1.

Esta organización sugiere una estructura básica del sistema operativo:

- ✓ Un programa principal que llama al procedimiento del servicio solicitado.
- ✓ Un conjunto de procedimientos de servicio que lleva a cabo las llamadas al sistema.
- ✓ Un conjunto de procedimientos de utilidades que ayudan a los procedimientos de servicio.

En este modelo, para cada llamada al sistema existe un procedimiento de servicio que se encarga de ella. Los procedimientos de utilidad hacen cosas necesarias para varios procedimientos de servicio, como por ejemplo, buscar los datos del programa del usuario. Esta división de los procedimientos en tres capas se muestra en la figura 2

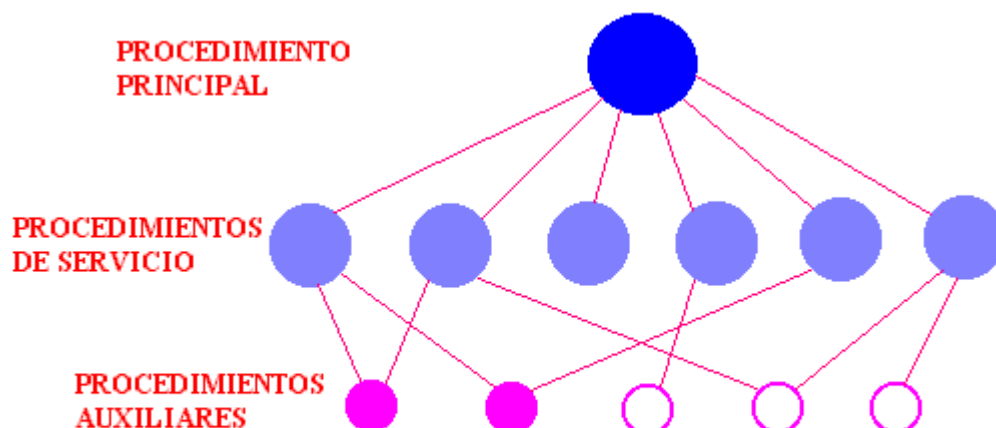


Figura 2

Modelo cliente-servidor

Una tendencia de los sistemas operativos modernos es la de trasladar el código a capas superiores, y eliminar la mayor parte posible del sistema operativo para mantener un núcleo mínimo. El punto de vista usual es el implantar la mayoría de las funciones del sistema operativo como procesos de usuario. Para solicitar un servicio, como la lectura de un bloque de cierto fichero, un proceso de usuario (denominado en este caso **proceso cliente**) envía la solicitud a un **proceso servidor**, que realiza el trabajo y devuelve la respuesta.

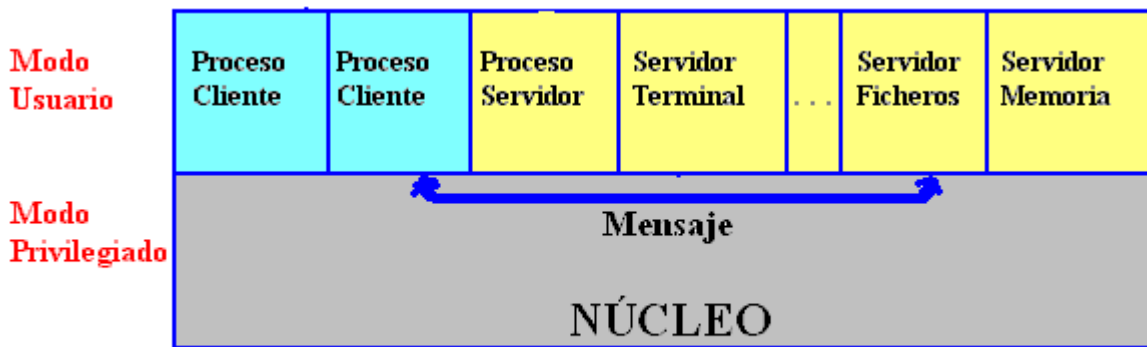


Fig. 3. Modelo Cliente-Servidor.

En este modelo, que se muestra en la figura 3, lo único que hace el núcleo es controlar la comunicación entre los clientes y los servidores. Al separar el sistema operativo en partes, cada una de ellas controla una faceta del sistema, como el servicio a ficheros, servicio a procesos, servicio a terminales o servicio a la memoria; cada parte es pequeña y controlable. Además, puesto que todos los servidores se ejecutan como procesos en modo usuario, y no en modo núcleo, no tienen acceso directo al *hardware*. En consecuencia, si hay un error en el servidor de ficheros éste puede fallar, pero esto no afectará en general a toda la máquina.

Otra de las ventajas del modelo cliente-servidor es su capacidad de adaptación para su uso en sistemas distribuidos (véase la figura 4). Si un cliente se comunica con un servidor mediante mensajes, el cliente no necesita saber si el mensaje se gestiona de forma local, en su máquina, o si se envía por medio de una red a un servidor en una máquina remota. En lo que respecta al cliente, lo mismo ocurre en ambos casos: se envió una solicitud y se recibió una respuesta.

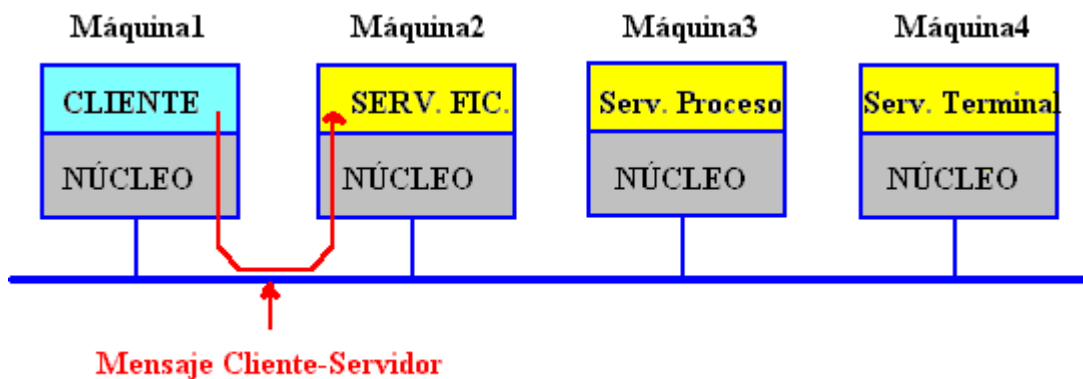


Fig. 4 Modelo Cliente-Servidor.

PRESTACIONES DE UN SISTEMA OPERATIVO

La misión de un sistema operativo es la de ayudar a los usuarios en el manejo de la computadora; para ello deberá proporcionar servicios que se puedan considerar desde dos puntos de vista distintos:

✓ Punto de vista del Programador

- **Ejecución de programas:** Facilidades para cargar un programa en memoria y ejecutarlo.
- **Operaciones de Entrada /Salida:** Facilidades para que un programa pueda tratar un archivo, enviar o recibir datos a un dispositivo, etc
- **Gestión de archivos:** Facilidades de uso y organización del sistema de archivos.

Puntos de Vista del Sistema

- **Asignación de Recursos:** mecanismos de resolución de conflictos de asignación de recursos cuando varios procesos o usuarios están compitiendo por ellos.
- **Contabilidad:** control de tiempos de utilización de recursos por los usuarios para su facturación o simplemente para la obtención de estadísticas de utilización.
- **Protección:** Defensa contra acciones no deseadas.

Servicios de Usuarios

El sistema operativo ofrece a los usuarios sus servicios de dos formas diferentes: Las llamadas al sistema operativo desde un proceso y la ejecución de programas del propio sistema.

Llamada al sistema operativo

constituyen el interface entre un programa en ejecución y el sistema operativo. Estas llamadas se pueden agrupar de la siguiente forma:

- Gestión de procesos.
- Gestión de operaciones de entrada/ salida.
- Gestión del sistema de archivos.
- Protección.

Programas del Sistema

En los sistemas operativos actuales, son como las llamadas a un subprograma desde el punto de vista del programa que llama. Tras la llamada al sistema operativo, existe un conjunto de programas del sistema o de utilidad cuya misión es resolver problemas comunes y frecuentes de los usuarios, ofreciéndolos de forma cómoda y sencilla

Estos programas los podemos agrupar de la siguiente forma:

- **Tratamiento de Archivos:** Crean, copian borran, renombran, imprimen, visualizan, vuelcan, ordenan, etc un archivo, facilitando la gestion de los mismos y los directorios.
- **Informacion:** Dan cualquier tipo de informacion relativa al estado del sistema, de la memoria, de los discos, de usuarios, fechas, hora, etc.
- **Editores:** Son programas que facilitan la edicion de archivos de texto o de programas fuente
- **Ejecucion:** Son programas para la construccion, depuracion y carga de programas ejecutables (Linkers, debugger. Etc)
- **Programas de utilidad:** Son programas para la gestion de Base de Datos, compiladores, comunicaciones, etc.
- **Interprete de comandos:** Es el mas importante de todos los programs del sistema operativo, ya que es el que crea el entorno de trabajo de los usuarios.

Servicios del Sistema

El interprete de comandos y los programas del sistema son los que fijan el entorno y la forma de ver el sistema operativo por los usuarios. En cambio, el programa tiene una vision totalmente diferente; para el todo son recursos fisicos y dispositivos que deben ser convertidos en entidades logicas para ofrecerselas a los usuarios.

Llamadas al Sistema Operativo

Estas se aguepan por el tipo de llamada no por la accion que realizan. Pueden ser:

- **Terminación normal:** Se realiza la devolucion del control al usuario cuyo proceso ha terminado, a traves del interprete de comandos.
- **Terminación anormal:** Cuando aparece un error en la ejecucion de un programa, este se da por terminado, devolviendo el control al interprete de comandos, que indicara tal situacion de error al usuario.
- **Peticiones de estado:** Se procesa la peticion solicitada y se devuelve el control al programa que lo solicito.
- **Peticiones de recursos:** Los programas solicitan recursos durante su ejecucion que seran atendidos de inmediato o se entrara en un estado de espera hasta que pueda ser atendidos.
- **Peticiones de entrada / salida:** De igual forma, los programas las solicitaran y sera atendidas de inmediato o tras un pequeño periodo de espera.

Interrupciones de los dispositivos de entrada / salida.

Una vez que un programa en ejecución realiza una petición de entrada/salida, se pueden tomar dos tipos de acción:

- El proceso queda en espera hasta que se termina la operación de entrada / salida: en este caso el dispositivo externo, cuando termine la operación, producirá una interrupción que dará control al sistema operativo, el cual activará el proceso que estaba en espera.
- El proceso seguirá realizando otras operaciones: En el caso el dispositivo externo también produce una interrupción en el sistema operativo, el cual no activará el proceso puesto que no estaba en espera, pero indicará que la operación solicitada ha terminado.

Gestión de excepciones

cuando un programa en ejecución comete un error, se producirá una interrupción; por ejemplo, una división por 0, intento de violación de un archivo protegido, intento de ejecución de una instrucción no permitida o privilegiada, etc. El tratamiento de estos errores se conoce como manejo de excepciones.

Protecciones

Los programas de aplicación de los usuarios no están exentos de errores, así como los sistemas tampoco están libres de usuarios como malas intenciones. Por ello, el sistema operativo debe incluir ciertas funciones de protección como objeto de evitar problemas entre procesos y entre estos y el propio sistema operativo.

● Protección de la entrada/salida

Todos los dispositivos externos cuentan, por parte del sistema operativo, con rutinas para el control de las operaciones de entrada/salida. Estas rutinas se denominan controladores o drivers de dispositivos y entre otras funciones protegen los accesos incorrectos, devolviendo el control al núcleo del sistema operativo, indicándole la situación errónea que se ha producido.

● Protección de la memoria

en general, cada proceso tiene una zona de memoria asignada para el tratamiento de sus datos denominada espacio de direccionamiento y no puede acceder a zonas asignadas al sistema operativo o a otros procesos. Para evitarlo existen unos registros frontera que indican el límite de memoria asignado a cada proceso.

Si un proceso intentase acceder a direcciones que están fuera de la zona de memoria de su espacio de direccionamiento, se producirá una interrupción que dará control al sistema operativo dando cuenta del error que se ha producido.

- **Proteccion del procesador**

Otro tipo de problemas que pueden presentarse es la presencia de bucles infinitos o accesos al procesador que no lo liberan nunca. En estos la unica solucion es la de apagar y volver a encender (*reset*) para volver a comenzar desde el principio.

Para evitarlo, hardware incluye un temporizador que marca periodos de tiempo de manera que al terminar un periodo de tiempo se produzcan una interrupcion y tome el control el sistema operativo.