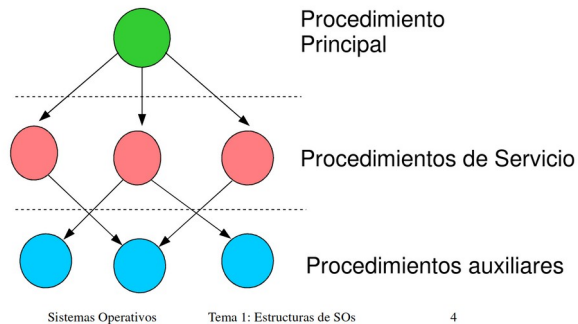


ESTRUCTURAS DE SO'S

1.1 Arquitectura: Sistema Monolítico

Modelo simple de estructura de un SO monolítico



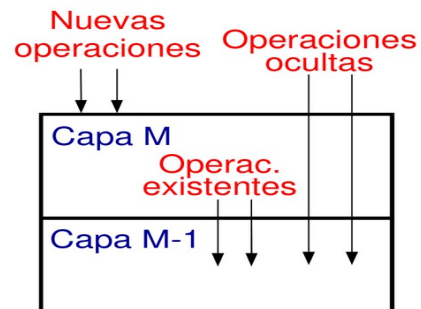
Los sistemas monolíticos no tienen una estructura bien definida. El sistema operativo está constituido por un conjunto de procedimientos de forma que cada uno puede llamar a los demás cuando lo necesite. Todos los procedimientos se ejecutan en modo supervisor. El SO se reduce a un único archivo ejecutable. No se aplica el principio de ocultación de información.

Problemas de los S. Monolíticos

Son difíciles de comprender y de modificar, al mismo tiempo que poco fiables, en el sentido de que un error en alguna parte puede provocar la caída del sistema. Además, son difíciles de mantener.

1.2 Arquitectura: Sistema por Capas

El sistema se implementa como un conjunto de capas en el cual, cada capa es una máquina más abstracta que la capa superior. Por motivos de modularidad, las capas se seleccionan para que cada una utilice funciones sólo de las capas inferiores.



Ejemplo: El Sistema THE

5: Programas de Usuario
4: Búfering para dispositivos de E/S
3: Manejador de consola del operador
2: Gestión de memoria
1: Planificación de la CPU
Nivel 0: Hardware

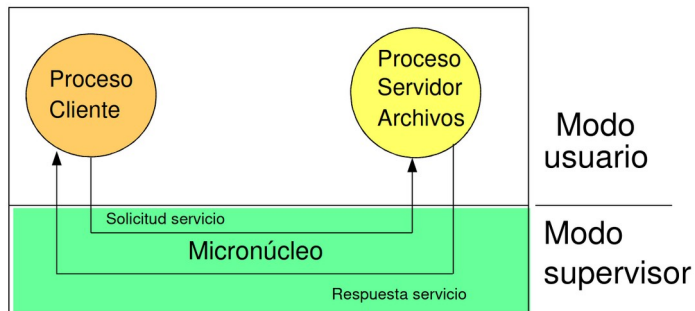
- El sistema estaba compuesto de una serie de procesos secuenciales
- Los procesos se sincronizan con declaraciones explícitas de sincronización
- Se puede probar y verificar de forma independiente cada proceso

Problemas de THE

- Los sistemas de capas deben ser jerárquicos pero los sist. reales son más complejos, p.ej.,
 - » El sistema de archivos podría ser un proceso en la capa de memoria virtual
 - » La capa de memoria virtual podría usar archivos como almacén de apoyo de E/S
- Sobrecarga de comunicaciones entre procesos de distintas capas
- A menudo, los sistemas están modelados con esta estructura pero no están así contruidos

1.3 Arquitectura: Microkernel o Micronúcleo

Modelo cliente-Servidor



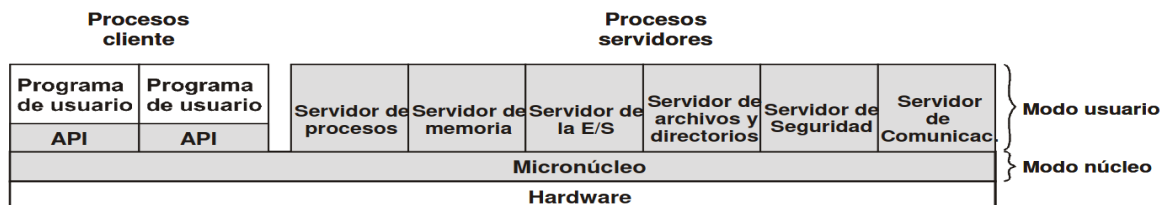
El objetivo que se persigue es reducir el SO a un núcleo mínimo: implementar la mayoría de las funciones del SO como procesos de usuario, lo que aporta mayor flexibilidad.

Para solicitar un servicio, el proceso de usuario (cliente) envía un mensaje al proceso servidor, que realiza el servicio y devuelve al cliente

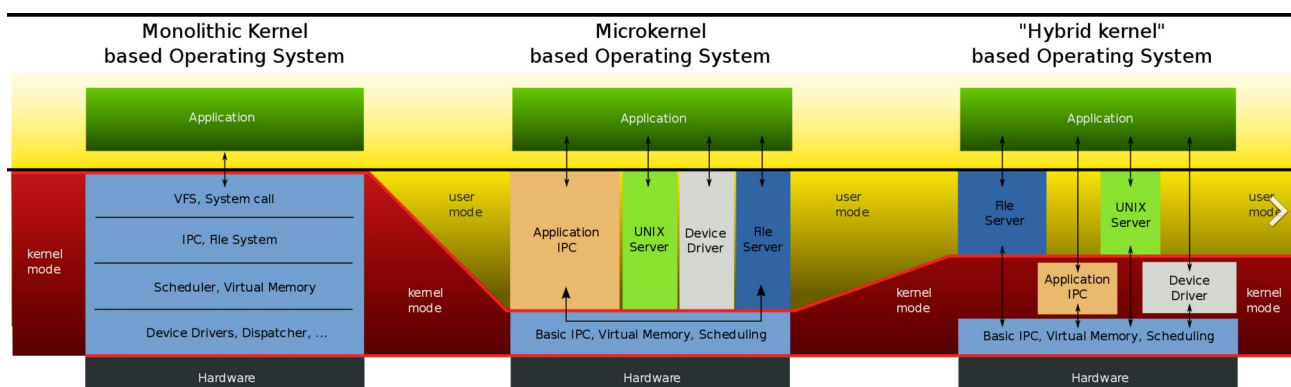
una respuesta, lo que conlleva una mayor sobrecarga por envío/recepción de mensajes.

Algunos sistemas Micronúcleo permiten servidores en modo sistema, lo cual es más eficiente, pero rompe la filosofía micronúcleo. Los servidores son programa independientes pero se ejecutan en el mismo espacio de direcciones del micronúcleo, y no usan IPCs para comunicarse.

Arquitectura micronúcleo

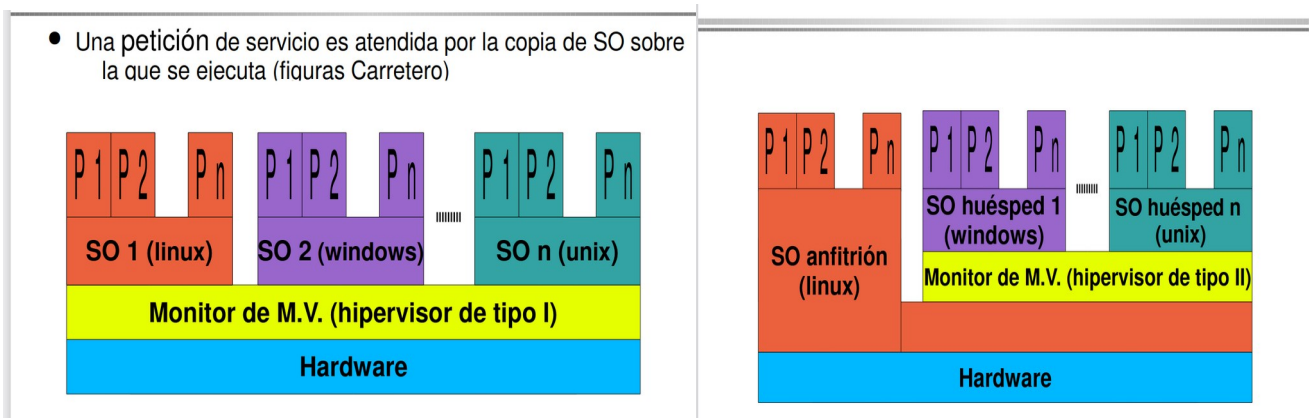


Monolítico VS Micronúcleo



1.4 Arquitectura: Máquinas Virtuales

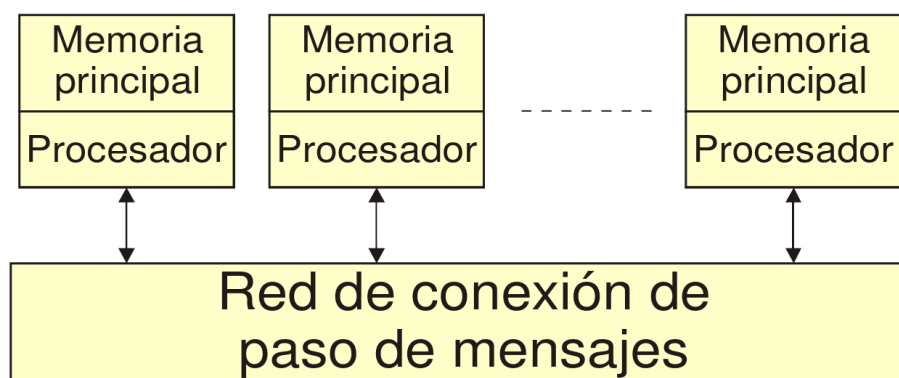
Software que implementa una máquina virtual (puede ser o no una máquina real). Cada copia es una réplica exacta del hardware. El sistema operativo crea una máquina virtual pero extendida (abstracción del hardware). La capacidad de procesamiento actual mitiga la ineficiencia de las Máquinas Virtuales. Los procesadores actuales incluyen soporte para la misma. Es una técnica que esa de nuevo en auge en la actualidad.



2. SO's de propósito específico: SO de Tiempo Real

Los SOTR se utilizan para aplicaciones especializadas como son por ejemplo los sistemas de control, etc. La idea básica es la siguiente: el SO debe garantizar la respuesta a sucesos físicos en intervalos de tiempo fijos. El problema está en que en planificar las actividades con el fin de satisfacer todos los requisitos críticos. Con el uso de aplicaciones de video sobre PC's, todos los SO's tendrán pronto requisitos tiempo-real.

Arquitectura Multicomputador



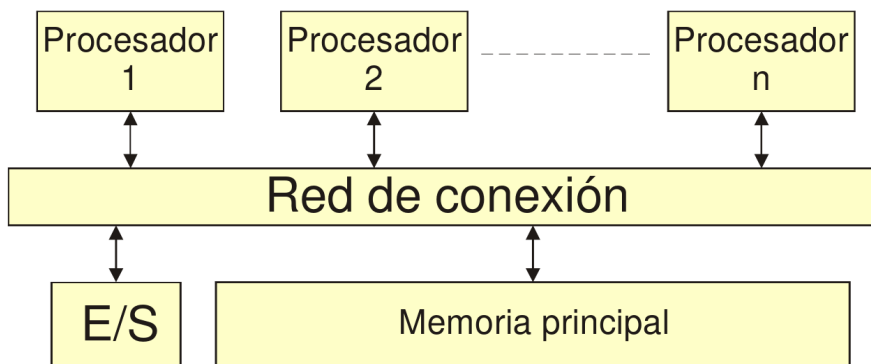
2.1 SO's de Red

Los usuarios son conscientes de la existencia de varias computadoras. Cada nodo ejecuta su propio sistema operativo local y tiene sus propios usuarios. Lo que les diferencia de los sistemas operativos de un solo procesador es la necesidad de software especial como: controlador de interfaz de la red, programas de conexión y acceso a archivos remoto.

2.2 SO's Distribuidos

Son sistemas débilmente acoplados; sistemas sin memoria común. La característica fundamental de estos sistemas es la transparencia. Permiten la compartición de recursos distribuidos, hardware o software. Permiten algún paralelismo, pero el aumento de velocidad no es el objetivo. Además, aumentan la fiabilidad del sistema.

Arquitectura multiprocesador



2.3 SO's Paralelos

Son sistemas multiprocesadores fuertemente acoplados (los procesadores comparten una memoria y el reloj).

Se distinguen dos tipos de multiprocesamiento:

- ◆ **Simétrico (SMP):** cada procesador ejecuta una copia idéntica del SO (buen rendimiento).
- ◆ **Asimétrico (ASMP):** un procesador *maestro* ejecuta el SO, los procesadores *esclavos* ejecutan procesos de usuario. Peor escalabilidad.

Soportan aplicaciones paralelas que desean obtener un aumento de velocidad de tareas computacionalmente complejas. Necesitan primitivas básicas para dividir una tarea en múltiples actividades paralelas. Proporciona una comunicación y sincronización eficiente entre esas actividades son tolerantes a fallos.