



Instituto Politécnico Nacional

Escuela Superior De Cómputo



Ingeniería En Sistemas Computacionales

Sistemas Operativos

Profesor: Salas Ramírez Israel

Equipo: Tacos linuxeros

Integrantes:

García Monzalvo Victoria Estefanía
Juárez Quintos Alejandro
Matías Medina Danna Lizbeth
Olguín Martínez José Arturo

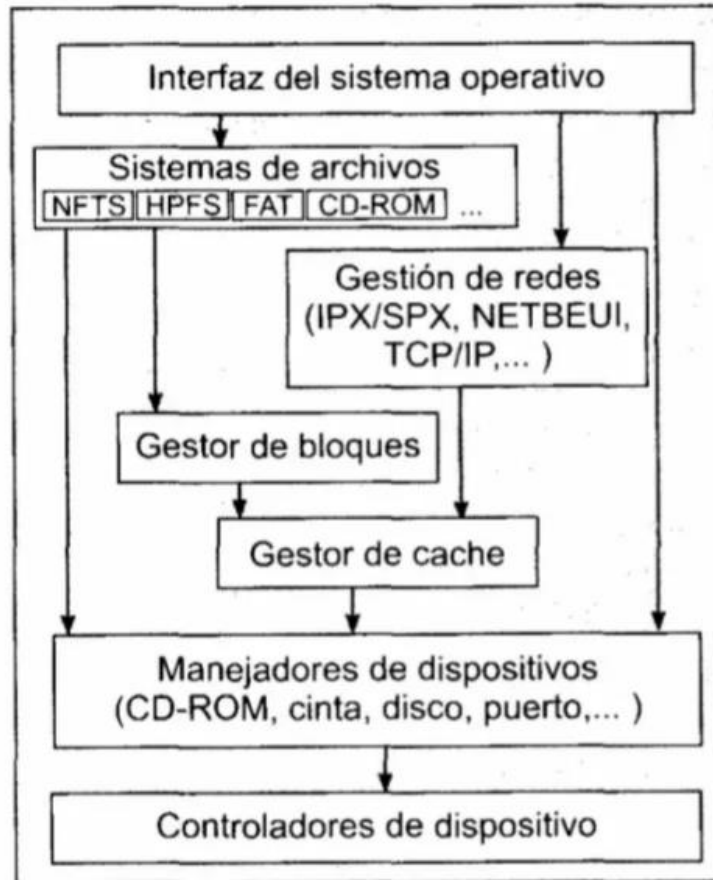
Tema 5.3

Capas de software

CAPAS DE SOFTWARE

Software modo usuario para E/S

Este software se organiza en una serie de capas. Estas capas se corresponden, en general, con los niveles de la arquitectura de E/S



Arquitectura del sistema de entrada/salida

Los procesos de usuario emiten peticiones de entrada/salida al sistema operativo. Cuando un proceso solicita una operación E/S, el sistema operativo prepara dicha operación y bloquea al proceso hasta que se recibe una interrupción del controlador del dispositivo indicando que la operación está completa.

Las peticiones se procesan de forma estructurada en las siguientes capas:

- Manejadores de interrupción
- Manejadores de dispositivos o drivers
- Software de E/S independiente de los dispositivos. Formado por parte de alto nivel de los manejadores, el gestor de caché, de bloques y el servidor de archivos.
- Interfaz del sistema operativo. Llamadas al sistema que usan las aplicaciones de usuario.

El uso de capas conlleva la realización de varias copias de datos, algunas de las cuales son inevitables. En algunos casos, la copia que se realiza en el núcleo del sistema operativo puede ser innecesaria, por lo que existen mecanismos para acceder directamente a los controladores desde la interfaz de E/S del sistema a los manejadores. Como norma general, esa copia siempre existe.

El sistema operativo estructura el software de gestión de E/S de esta forma para ofrecer a los usuarios una serie de servicios de E/S independientes de los dispositivos. Esta independencia implica que deben emplearse los mismos servicios y operaciones de E/S para leer datos de un disquete, un disco duro, de un CD-ROM o de un teclado.

Ejemplo

El servicio read de POSIX puede utilizarse para leer datos de cualquiera de los dispositivos. La estructuración en capas permite hacer frente a la gestión de errores que se pueden producir en el acceso a los periféricos en el nivel de tratamiento más adecuado.

Manejador de interrupción

En el contexto de la informática, una interrupción (del inglés Interrupt Request, también conocida como petición de interrupción) es una señal recibida por el procesador de un ordenador, indicando que debe "interrumpir" el curso de ejecución actual y pasar a ejecutar código específico para tratar esta situación.

Una interrupción es una suspensión temporal de la ejecución de un proceso, para pasar a ejecutar una subrutina de servicio de interrupción, la cual, por lo general, no forma parte del programa, sino que pertenece al sistema operativo o al BIOS). Una vez finalizada dicha subrutina, se reanuda la ejecución del programa.

Las interrupciones surgen de la necesidad que tienen los dispositivos periféricos de enviar información al procesador principal de un sistema informático.

La primera técnica que se empleó para esto fue el polling, que consistía en que el propio procesador se encargará de sondear los dispositivos periféricos cada cierto tiempo para averiguar si tenía pendiente alguna comunicación para él. Este método presentaba el inconveniente de ser muy ineficiente, ya que el procesador consumía constantemente tiempo y recursos en realizar estas instrucciones de sondeo.

Los manejadores de interrupción se encargan de tratar las interrupciones que generan los controladores de dispositivos una vez que estos están listos para la transferencia de datos o bien han leído o escrito los datos de memoria principal en caso de acceso directo a memoria.

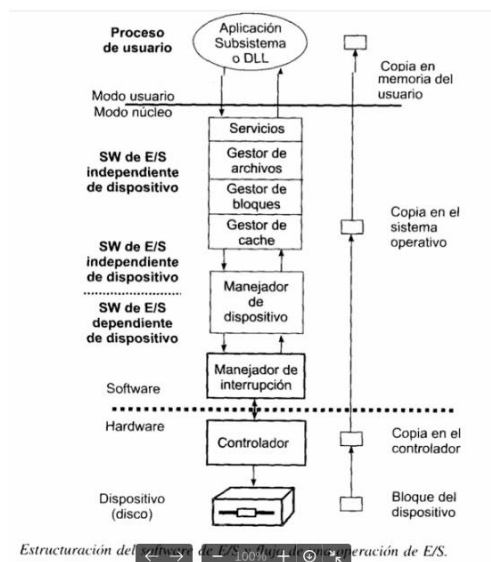
Cómo tratar la interrupción:

- Se utiliza el correspondiente manejador de interrupción cuyo efecto es el de salvar los registros, comunicar el evento al manejador del dispositivo y restaurar la ejecución de un proceso (que no tiene por qué ser interrumpido).
- El modelo de interrupciones está íntimamente ligado a la arquitectura del procesador. Casi todas las UCP actuales incluyen interrupciones vectorizadas y enmascarables, es decir, un rango de interrupciones entre 0 y 255.
- Cada interrupción se asigna a un dispositivo o un rango de ellos en caso de un controlador SCSI o una cadena de dispositivos tipo daisy chain, que usa el vector correspondiente para indicar eventos de E/S a la UCP.
- Cuando se programa una operación en un dispositivo como por ejemplo una búsqueda en un disco, este se puede dedicar a ejecutar código de otro proceso, optimizando de esta forma el uso del procesador. Al cabo de un cierto tiempo, cuando el disco ha efectuado la búsqueda y las cabezas del disco están sobre la posición deseada, genera una interrupción (poniendo un 1 en el vector correspondiente),

La rutina de tratamiento de la interrupción se encargará de leer o enviar el dato al controlador. Por razones de seguridad la tabla de interrupciones como la rutina de tratamiento de la interrupción se consideran parte del sistema operativo.

Tipos de interrupciones:

- Operaciones de hardware: se producen por una operación de entrada y salida producidas por dispositivos preferidos.
- Excepciones: se produce por una condición de error en un programa.
- Interrupciones de software: llamadas al sistema generadas por un programa mientras se está ejecutando.

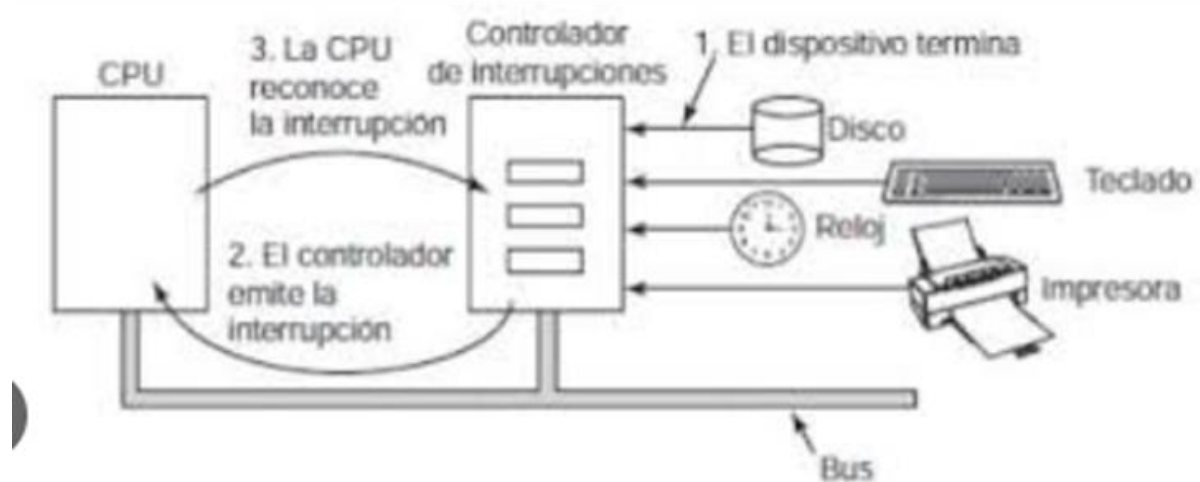


El manejador de interrupciones es un componente crítico del sistema operativo que permite la gestión eficiente de las interrupciones generadas por el hardware. Las interrupciones son señales enviadas por dispositivos hardware al procesador para indicar que requieren atención, como la llegada de datos desde un teclado o la finalización de una operación de disco. El manejador de interrupciones se encarga de:

Recibir y clasificar las interrupciones: Determina qué dispositivo ha generado la interrupción y prioriza su atención.

Ejecutar el código correspondiente: Una vez identificada la interrupción, el manejador ejecuta un conjunto específico de instrucciones para atender la solicitud del dispositivo.

Restaurar el contexto: Después de procesar la interrupción, el manejador debe restaurar el estado anterior del sistema para continuar con las tareas previamente en ejecución.



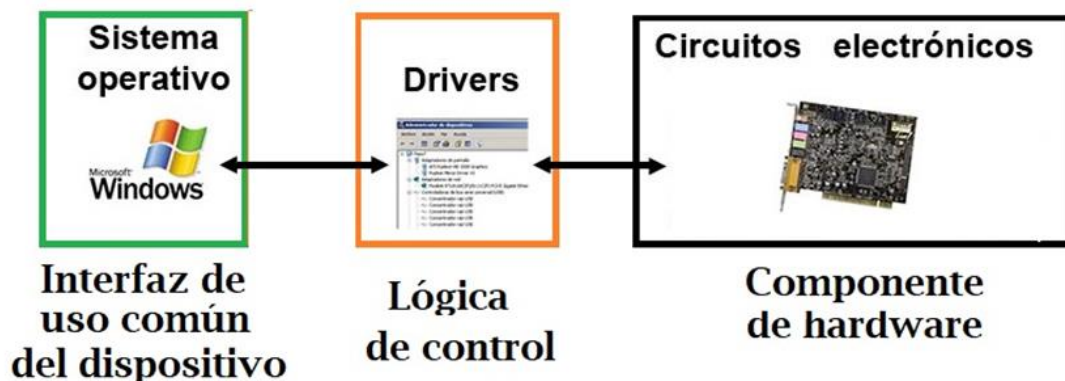
Controladores de dispositivos

Los controladores de dispositivos son programas que permiten al sistema operativo interactuar con el hardware. Actúan como intermediarios entre el sistema operativo y los dispositivos periféricos, facilitando la comunicación y asegurando que las instrucciones del software se traduzcan adecuadamente en acciones del hardware. Sus funciones principales incluyen:

Abstracción del hardware: Proporcionan una interfaz estandarizada que oculta los detalles técnicos del hardware, permitiendo que aplicaciones y sistemas operativos interactúen sin necesidad de conocer las especificaciones exactas del dispositivo.

Gestión de operaciones: Cuando se envían comandos desde el sistema operativo, los controladores se encargan de traducir estos comandos en instrucciones que el hardware puede entender y ejecutar.

Actualización y mantenimiento: Es fundamental mantener actualizados los controladores para garantizar la compatibilidad con nuevas versiones del sistema operativo y para mejorar el rendimiento y la seguridad del hardware.



Cada dispositivo E/S o cada clase de dispositivos, tiene un manejador asociado en el sistema operativo.

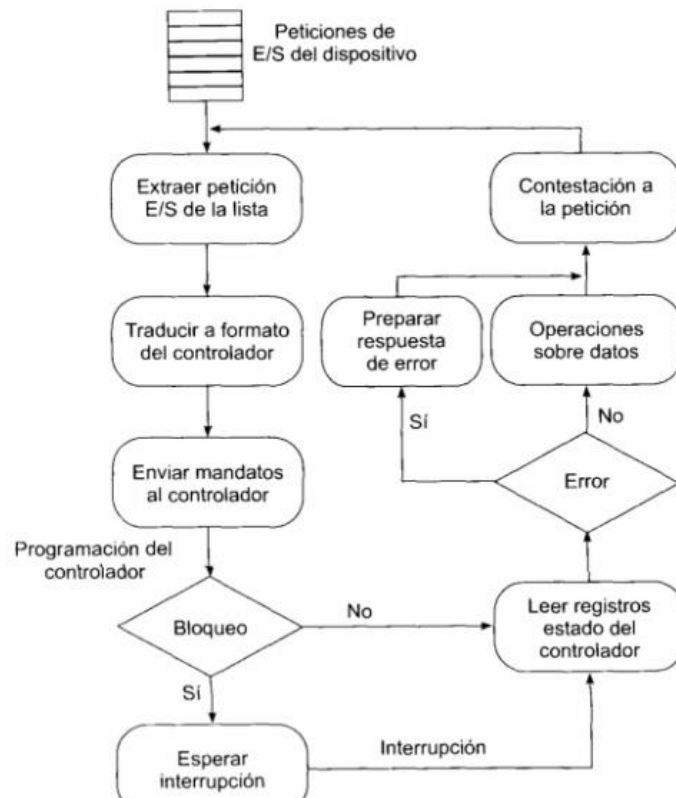
Este manejador incluye:

- Código independiente del dispositivo para proporcionar al nivel superior del sistema operativo en una interfaz de alto nivel
- Código dependiente del dispositivo necesario para programar el controlador del dispositivo a través de sus registros y mandatos.

Función de un controlador:

Aceptar peticiones en formato abstracto, de la parte del código E/S independiente del dispositivo, traducir dichas peticiones a términos que entienda el controlador, enviar al mismo tiempo las órdenes adecuadas en la secuencia correcta y esperar a que se cumplan

Diagrama de flujo con operaciones de un manejador:

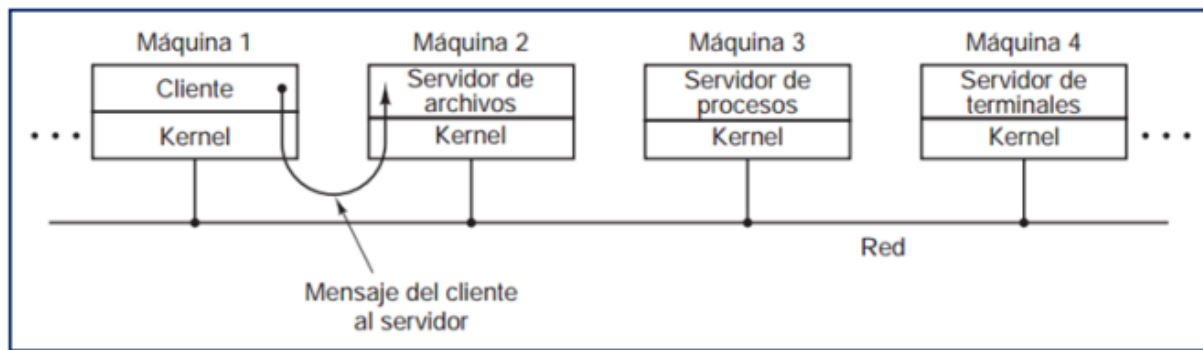


Todos los manejadores tienen una lista de peticiones pendiente por dispositivo donde se encolan las peticiones que llegan de niveles superiores.

El manejador explora la lista de peticiones pendientes, extrae una petición pendiente y ordena su ejecución.

La política de extracción de peticiones de la lista es dependiente del manejador y puede ser FIFO, con pr etc.

los servidores, cada uno de los cuales proporciona cierto servicio, y los clientes, que utilizan estos servicios. Este modelo se conoce como cliente-servidor. A menudo la capa inferior es un microkernel, pero eso no es requerido. La esencia es la presencia de procesos cliente y procesos servidor. La comunicación entre clientes y servidores se lleva a cabo comúnmente mediante el paso de mensajes. Para obtener un servicio, un cliente construye un mensaje indicando lo que desea y lo envía al servicio apropiado. Después el servicio hace el trabajo y envía de vuelta la respuesta. Si el cliente y el servidor se ejecutan en el mismo equipo se pueden hacer ciertas optimizaciones, pero en concepto estamos hablando sobre el paso de mensajes.



Modelo Cliente-Servidor

Una generalización obvia de esta idea es hacer que los clientes y los servidores se ejecuten en distintas computadoras, conectadas mediante una red de área local o amplia, como se describe en la gráfica. Como los clientes se comunican con los servidores mediante el envío de mensajes, no necesitan saber si los mensajes se manejan en forma local en sus propios equipos o si se envían a través de una red a servidores en un equipo remoto. En cuanto a lo que al cliente concierne, lo mismo ocurre en ambos casos: se envían las peticiones y se regresan las respuestas. Por ende, el modelo cliente-servidor es una abstracción que se puede utilizar para un solo equipo o para una red de equipos. Cada vez hay más sistemas que involucran a los usuarios en sus PCs domésticas como clientes y equipos más grandes que operan en algún otro lado como servidores. De hecho, la mayor parte de la Web opera de esta forma. Una PC envía una petición de una página Web al servidor y la página Web se envía de vuelta. Éste es un uso común del modelo cliente-servidor en una red.

E/S programada o por interrupciones

Un controlador de dispositivo o unidad de E/S se encarga de controlar uno o más dispositivos del mismo tipo y de intercambiar información entre ellos y la memoria principal o unidad central de proceso de la computadora.

El controlador debe encargarse además de sincronizar la velocidad del procesador con la del periférico y de detectar los posibles errores que se produzcan en el acceso a los periféricos. En caso de un controlador de disco: Debe encargarse de convertir un flujo de bits procedente del disco a un bloque de bytes detectando y corrigiendo, si es posible, los errores que se produzcan en esta transferencia.

Cuando se obtiene el bloque y se haya comprobado que no tiene ningún error deberá encargarse de transferirlo a memoria principal.

Software en Modo Usuario para E/S

El software en modo usuario para entrada/salida (E/S) se refiere a las aplicaciones y procesos que operan en un nivel más alto que el núcleo del sistema operativo. Este software no tiene acceso directo a los recursos del sistema, lo que significa que cualquier operación de E/S debe ser gestionada a través del núcleo. Las características clave incluyen:

Interacción indirecta con el hardware: Las aplicaciones en modo usuario deben utilizar llamadas al sistema para realizar operaciones de E/S, lo que implica una mayor latencia en comparación con el acceso directo al hardware.

Seguridad y estabilidad: Al operar en modo usuario, se minimiza el riesgo de que un fallo en una aplicación afecte al núcleo del sistema operativo, mejorando así la estabilidad general del sistema.

Facilidad de desarrollo: Los desarrolladores pueden crear aplicaciones sin preocuparse por los detalles complejos del manejo directo del hardware, ya que estas interacciones son gestionadas por los controladores y el núcleo.

En resumen, estas capas de software trabajan conjuntamente para gestionar la comunicación entre el hardware y las aplicaciones, asegurando un funcionamiento eficiente y seguro del sistema operativo.

Bibliografía

- Carretero Pérez, J. (2001). Sistemas operativos: Una visión aplicada (Edición en español). McGraw-Hill.
- <https://gateway.pinata.cloud/ipfs/bafykbzacebkea4ot7nmt5j2nri5oa247d6onhvc3auaffij7laceynvbfule?filename=Jesus> Carretero Perez - Sistemas Operativos - Una Vision Aplicada (Spanish Edition)-MC Graw Hill (2001).pdf
- interrupciones – Sistemas Operativos. (2015, 2 marzo). Sistemas Operativos. <https://lcsistemasoperativos.wordpress.com/tag/interrupciones/>
- capas – Sistemas Operativos. (2015, 2 enero). Sistemas Operativos. <https://lcsistemasoperativos.wordpress.com/tag/capas/>
- Ballejos, L. (2024, 12 febrero). ¿Qué es un controlador de dispositivo? | NinjaOne. NinjaOne. <https://www.ninjaone.com/es/it-hub/endpoint-management/controlador-de-dispositivo-que-es/>
- Mhopkins-Msft. (2024, 19 junio). Paquetes de controladores - Windows drivers. Microsoft Learn. <https://learn.microsoft.com/es-es/windows-hardware/drivers/install/driver-packages>