

Virtualización en Sistemas Operativos

Capítulos 2.1 – 2.4 Integrados con Memoria Virtual Avanzada

Basado en material de *02 Virtualización.pdf* y *M2S2-MemoriaVirtual_s6.pdf*

10 de marzo de 2025

Índice

1. Capítulo 2.1: Introducción	2
2. Capítulo 2.2: El Sistema Operativo como Manejador de un Ambiente Virtual	2
3. Capítulo 2.3: Memoria Virtual	2
3.1. 2.3.1 Espacio de Direcciones	2
3.2. 2.3.2 Diseño de la Memoria Virtual	3
4. Capítulo 2.4: Sistema de Archivos	4
4.1. 2.4.1 Introducción	4
4.2. 2.4.2 Presentación a los Usuarios	4
4.3. 2.4.3 Administración de Espacio Asignado	4
4.4. 2.4.4 Administración de Espacio Libre	4
5. Ejemplos	5
6. Preguntas	5
7. Conclusión	6

1. Capítulo 2.1: Introducción

La virtualización se basa en el concepto de “lo virtual”, es decir, en elementos que tienen una existencia aparente pero no real. En los sistemas operativos, esto se traduce en la creación de ambientes de ejecución en los cuales el hardware físico (procesador, memoria, disco) se presenta al usuario y a las aplicaciones de forma abstracta y exclusiva, aunque los recursos sean compartidos.

Históricamente, la virtualización permitió la creación de una máquina virtual sobre una máquina física; con el tiempo, la tecnología ha evolucionado para permitir la ejecución de múltiples máquinas virtuales sobre un único hardware, optimizando el uso de los recursos y ofreciendo mayor flexibilidad.

2. Capítulo 2.2: El Sistema Operativo como Manejador de un Ambiente Virtual

El sistema operativo actúa como el administrador del hardware y, mediante técnicas de virtualización, proporciona a cada usuario o proceso la ilusión de contar con recursos exclusivos. Entre las funciones principales se encuentran:

- **Virtualización del procesador:** El sistema operativo asigna turnos de ejecución a los procesos (utilizando estructuras como el PCB, o Process Control Block) y utiliza técnicas de hyperthreading para simular múltiples núcleos lógicos en un solo core físico.
- **Virtualización de la memoria RAM:** Se separa el espacio de direcciones utilizado por cada proceso del espacio físico, permitiendo que los procesos tengan su propio espacio virtual, incluso si en realidad comparten la memoria física.
- **Virtualización del disco:** El sistema operativo abstrae la complejidad física de los discos (platos, pistas, sectores) y presenta a los usuarios nombres y rutas de archivos, gestionando el acceso a través de llamadas al sistema.

Esta capa de abstracción permite a los usuarios y aplicaciones trabajar sin preocuparse por la complejidad del hardware subyacente.

3. Capítulo 2.3: Memoria Virtual

La memoria virtual permite a un sistema operativo ejecutar procesos sin que éstos tengan que estar completamente cargados en la memoria física. Se parte de la idea de que cada proceso dispone de un **espacio de direcciones virtual** que luego se traduce en direcciones físicas mediante mecanismos de hardware y software.

3.1. 2.3.1 Espacio de Direcciones

Cada proceso posee su propio espacio de direcciones virtual. Una de las primeras soluciones fue el uso de registros base y límite:

- El **registro base** indica la dirección física donde comienza el espacio asignado al proceso.
- El **registro límite** indica el tamaño del área asignada.

Con este mecanismo, cada dirección generada por el proceso se suma al valor del registro base y se verifica que no exceda el límite. Sin embargo, para sistemas multiproceso y de alta concurrencia se requiere un esquema más flexible: la memoria virtual.

3.2. 2.3.2 Diseño de la Memoria Virtual

La memoria virtual divide el espacio de direcciones de un proceso en unidades fijas denominadas **páginas**, y la memoria física se divide en **marcos de página** (del mismo tamaño que las páginas). La traducción de una dirección virtual a una dirección física se realiza de la siguiente forma:

1. Se divide la dirección virtual en dos partes: el **número de página virtual** y el **desplazamiento** dentro de la página.
2. Se consulta la **tabla de páginas** para determinar a qué marco de página corresponde la página virtual.
3. Si la página se encuentra en memoria física, se combina el número del marco con el desplazamiento para formar la dirección física.
4. Si la página no está presente (se produce un **fallo de página**), el sistema debe cargarla desde el área de **swap** en el disco y actualizar la tabla de páginas.

Traducción de Direcciones Virtuales y el Rol de la MMU

La **Unidad de Manejo de Memoria (MMU)** es el componente que intercepta las direcciones virtuales generadas por la CPU y se encarga de su traducción a direcciones reales antes de que se envíen al bus de direcciones de la memoria. En este proceso:

- La MMU divide la dirección virtual en **número de página y desplazamiento**.
- Con la información de la tabla de páginas, se obtiene el número de página física.
- La dirección real se forma concatenando el número de página física con el desplazamiento.

La TLB (Translation Lookaside Buffer)

Para acelerar la traducción, se utiliza una memoria caché especial llamada **TLB**. Esta almacena las entradas más recientes de la tabla de páginas. Si la traducción requerida se encuentra en la TLB, el tiempo de acceso se reduce drásticamente. En caso contrario, se debe acceder a la tabla de páginas en la memoria principal, lo que implica mayor latencia.

Fallas de Página y Algoritmos de Reemplazo

Cuando una dirección virtual solicita una página que no se encuentra en la memoria física, se produce un **fallo de página**. El sistema operativo:

1. Detecta la ausencia y detiene temporalmente el proceso.
2. Recupera la página desde el área de **swap** (almacenamiento en disco).
3. Si no hay espacio libre en memoria, se selecciona una página a desalojar utilizando algoritmos de reemplazo (por ejemplo, FIFO, LRU o mediante el uso de bits de referencia).

4. Se actualizan las tablas de páginas y se reanuda la ejecución del proceso.

4. Capítulo 2.4: Sistema de Archivos

El sistema de archivos es el componente que permite el almacenamiento y recuperación persistente de la información. Su función es presentar a los usuarios un ambiente amigable, donde puedan interactuar con archivos y directorios sin conocer la estructura física del disco.

4.1. 2.4.1 Introducción

Cada disco se organiza en sectores, y el sector 0 contiene el **Master Boot Record (MBR)** y la tabla de particiones. Cada partición aloja su propio bloque de arranque, el sistema operativo y el sistema de archivos. Esto garantiza que el disco pueda ser particionado y que cada partición funcione de manera independiente.

4.2. 2.4.2 Presentación a los Usuarios

Los usuarios interactúan con el sistema de archivos mediante conceptos intuitivos:

- **Archivos:** Unidades básicas que almacenan datos.
- **Directorios:** Estructuras jerárquicas que organizan los archivos en forma de árbol.

Además, se soporta el acceso remoto a través de protocolos como NFS y SMB, lo que permite el montaje de sistemas de archivos distribuidos.

4.3. 2.4.3 Administración de Espacio Asignado

El sistema de archivos no sólo almacena el contenido de los archivos, sino también metadatos (por ejemplo, permisos, fechas, y la lista de bloques asignados). Para ello, se utilizan estructuras de datos como:

- **Descriptores de archivo:** Estructuras que contienen información (metadatos) y los apuntadores a los bloques en disco.
- **Representaciones de Bloques:** Se puede usar una lista enlazada o nodos-i para administrar los bloques asignados a cada archivo.

4.4. 2.4.4 Administración de Espacio Libre

El sistema operativo debe llevar un control sobre los bloques libres en el disco. Para ello se usan principalmente dos métodos:

- **Lista enlazada de bloques libres:** Donde se registran los identificadores de los bloques disponibles.
- **Mapa de bits:** Donde cada bit representa el estado (ocupado o libre) de un bloque.

La elección entre uno u otro método depende del tamaño del disco y de la eficiencia en las operaciones de asignación y liberación.

5. Ejemplos

Ejemplo 1: Traducción de Dirección Virtual a Real

Suponga que tenemos un sistema con páginas de 4096 direcciones (2^{12}) y que la dirección virtual se expresa en 32 bits.

1. Divida la dirección virtual en **número de página virtual** (los 20 bits más significativos) y **desplazamiento** (los 12 bits menos significativos).
2. Utilice la tabla de páginas para encontrar a qué marco de página corresponde la página virtual.
3. Combine el número del marco (por ejemplo, 7) con el desplazamiento para formar la dirección real.

Ejemplo 2: Uso de la TLB

Considere que la TLB tiene 32 entradas. Si una dirección virtual ya se encuentra en la TLB, la MMU traduce la dirección en 1 ciclo de reloj; si no, la traducción desde la tabla de páginas puede tomar 100 ciclos. Calcule el tiempo promedio de traducción si la tasa de aciertos en la TLB es del 99 %.

Ejemplo 3: Manejo de Fallo de Página

Cuando un proceso genera una dirección virtual para una página que no está en la memoria física, se produce un fallo de página. Describa el proceso:

1. La MMU detecta el fallo.
2. Se busca la página en el área de swap y se carga en un marco libre de memoria.
3. Si la memoria está llena, se selecciona una página para desalojar mediante un algoritmo (por ejemplo, LRU).
4. Se actualizan las tablas de páginas y se reanuda la ejecución del proceso.

6. Preguntas

- P1. Traducción de Direcciones:** ¿Cómo se divide una dirección virtual en número de página y desplazamiento? Explique el rol de la MMU y la TLB en este proceso.
- P2. Fallas de Página:** ¿Qué es un fallo de página y cuáles son los pasos que sigue el sistema operativo para manejarlo? Mencione al menos dos algoritmos de reemplazo de páginas.
- P3. Memoria Virtual vs. Física:** ¿Por qué es ventajoso usar memoria virtual en lugar de trabajar directamente con direcciones físicas? Explique con ejemplos.
- P4. Sistema de Archivos:** Describa la función de los descriptores de archivo y explique cómo se administra el espacio asignado y el espacio libre en un sistema de archivos.

- P5. Integración de Recursos:** ¿Cómo virtualiza el sistema operativo el procesador, la memoria y el disco? Relacione esta virtualización con la experiencia del usuario final.
- P6. Ejercicios Prácticos:** Plantee un algoritmo que simule la lógica de traducción de una dirección virtual a una dirección real, incluyendo la consulta a la TLB y el manejo de fallo de página.
- P7. Impacto en el Rendimiento:** ¿Cómo afecta la tasa de aciertos en la TLB al tiempo promedio de acceso a la memoria? Justifique su respuesta con un ejemplo numérico.

7. Conclusión

En este documento se han integrado y detallado los conceptos fundamentales de la virtualización en sistemas operativos, abarcando desde la definición de ambientes virtuales hasta la administración de la memoria virtual y el sistema de archivos. Se han incluido detalles metódicos sobre la traducción de direcciones, el funcionamiento de la TLB, el manejo de fallos de página y los algoritmos de reemplazo, complementando la visión general con ejemplos prácticos y preguntas de análisis. Esta integración permite comprender de forma holística cómo los sistemas operativos abstraen y gestionan los recursos físicos, ofreciendo un marco teórico y práctico esencial para estudiantes y profesionales.