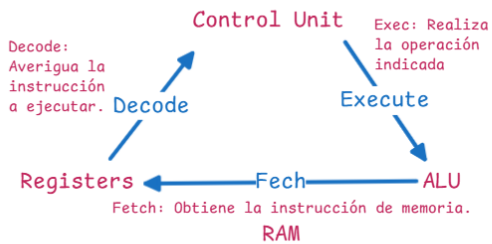


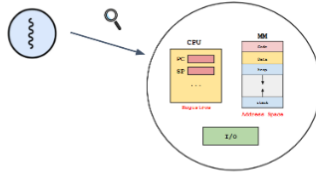
¿Cómo se ejecuta un programa?



Procesos

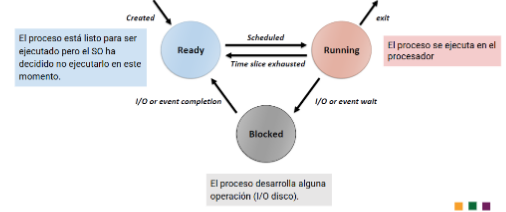
-Un proceso es un programa en ejecución.

Proceso = CPU + Memory + I/O Info



Estados de un proceso

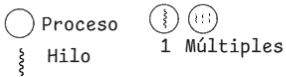
-Modelo de tres estados



Sistema Operativo

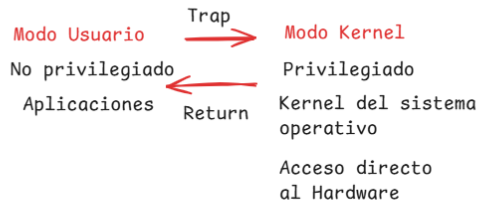
Controla y coordina el uso del hardware entre varias aplicaciones y usuarios
 -Administra justo y eficientemente los recursos (procesador, memoria, disco duro.)
 -Se encarga de controlar la ejecución de programas
 Previene errores.
 Evita el uso inadecuado del sistema de computo

Concurrencia



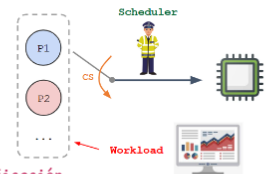
Operaciones del sistema operativo

-Los modos del procesador permiten que el sistema operativo se proteja a sí mismo y a otros componentes del sistema.



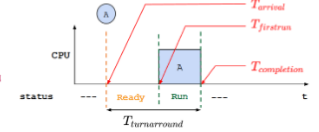
Planificación de Procesos

Scheduler (planificador): Decide como y cuando los procesos acceden a la CPU (o CPUs compartidas) de acuerdo a unas métricas.
Workload (carga de trabajo): Procesos en ejecución en un sistema.

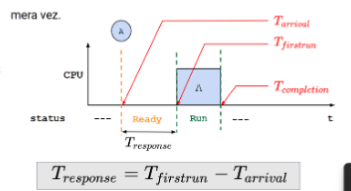


Métricas de planificación

Turnaround Time: Tiempo comprendido entre la llegada y la finalización de un proceso.



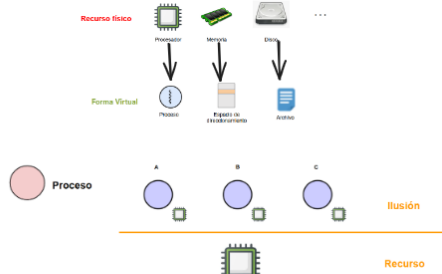
Response Time: Tiempo medido desde que el trabajo llega hasta que es programado por primera vez.



Proceso Arrival time First Run Time Completion Time
 Turnaround Time response Time

Virtualización de Recursos

-El sistema operativo toma un recurso físico y lo transforma en una forma virtual de sí mismo. (Realiza una abstracción)



Time sharing Es la que hace la ilusión de las CPUs infinitas



Persistencia: memoria volátil, hardware, software sistema de archivos (controla el disco).

Mecanismos: Ejecución Directa Limitada

Limited Direct Execution

Limited

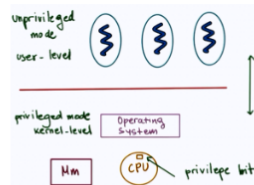
Protección:

- Los procesos deben estar aislados (protegidos) entre sí
- El kernel del sistema operativo debe estar aislado de los procesos.
- Los dispositivos de hardware deben estar aislados de los procesos

Direct Execution

Desempeño:

- Las aplicaciones deben ejecutarse directamente en el procesador (El SO no debe intervenir y verificar la validez de cada una de las instrucciones que la aplicación desea ejecutar)



¿Cómo hace el SO para retomar el control de la CPU y realizar cambios entre procesos?

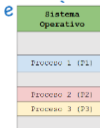
Hay dos formas:

Propuesta cooperativa: Se cede el control mediante llamados a sistema.

Propuesta no-cooperativa: El SO toma el control.

Multiprogramación

-Hace posible la ejecución de varios programas (que se encuentran en memoria) a la vez gracias al uso compartido de la CPU (por parte de e

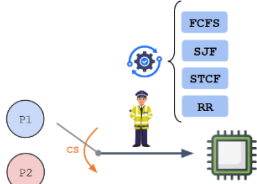


Time Sharing

Técnica empleada por el Sistema Operativo para compartir la CPU entre varios procesos.

Algoritmos de planificación

El planificador de procesos depende del algoritmo de planificación de procesos, el cual, basado en una política específica, asigna la CPU y mueve los diferentes trabajos a través del sistema.



First-Come, First-Served - Fifo (FCFS)

Atiende los procesos en el orden en que llegan, utilizando una cola FIFO. Es no apropiativo, lo que significa que cada proceso se ejecuta hasta que finaliza, sin interrupciones. Su implementación es simple y directa, pero puede generar tiempos de espera largos si un proceso requiere mucho tiempo de ejecución. Es más eficiente cuando los procesos tienen tiempos de procesamiento similares o equivalentes.

- ✓ Response Time
- ✗ Turnaround Time

Shortest Job First (SJF)

Organiza las tareas según su tiempo de ejecución estimado, dando prioridad a los procesos más cortos. Es un algoritmo no apropiativo, lo que significa que un proceso se ejecuta hasta que finaliza antes de que otro pueda iniciar.

- ✗ Response Time
- ✓ Turnaround Time

Posee problemas de inanición cuando hay muchos trabajos cortos y algún proceso largo.

Shortest Time-to-Completion First (STCF)

es la versión apropiativa del algoritmo SJF, donde el procesador siempre se asigna al proceso con el menor tiempo restante para completar su ejecución. Si llega un nuevo proceso con un tiempo de finalización menor que el del proceso actual, este es interrumpido y la CPU se reasigna al nuevo proceso.

- ✗ Response Time
- ✓ Turnaround Time

Round Robin (RR)

La planificación por porciones de tiempo (time slicing) ejecuta cada proceso durante un intervalo fijo llamado quantum, y luego pasa al siguiente en la cola de listos, repitiendo el ciclo hasta que todos los procesos terminen

- ✓ Response Time
- ✗ Turnaround Time

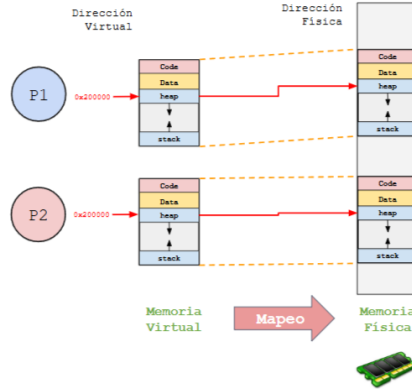
Puede adaptarse para incluir I/O

Virtualización de memoria

Cada proceso tiene su espacio de direcciones propio, si un proceso falla, no lo hace todo el sistema. Grande y continuo. Usa direcciones virtuales para referencias a memoria. Las direcciones virtuales son privadas a cada proceso.

Beneficios de la virtualización de memoria

- Facilidad de uso para el programador.
- Eficiencia de memoria en términos de tiempo y espacio.
- Protección (seguridad)
- Garantía de aislamiento entre procesos y SO.
- Protección ante accesos ilegales de otros procesos



MLFQ (Multi-Level Feedback Queue)

Conceptos importantes:

Alta prioridad: Procesos I/O bound.
Bajas prioridades: Procesos CPU-bound.

MLFQ: Reglas básicas

Regla 1: Si prioridad(A) > prioridad(B); A se ejecuta.

Regla 2: Si prioridad(A) = prioridad(B); RR para A y B.

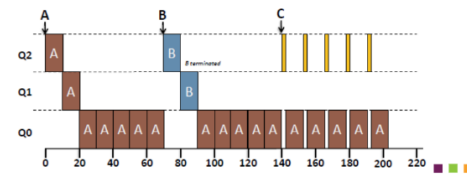
Regla 3: Cuando un trabajo llega al sistema es ubicado en la cola con la prioridad más alta.

Regla 4a: Si el trabajo usa completamente el quantum de tiempo, se reduce su prioridad.

Regla 4b: Si el trabajo entrega la CPU antes de finalizar su quantum de tiempo, mantiene el mismo nivel de prioridad.

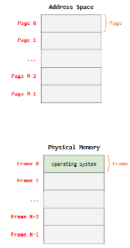
Regla 5: Después de un tiempo S, mueva todos los trabajos al mayor nivel de prioridad

Ejemplo resumen



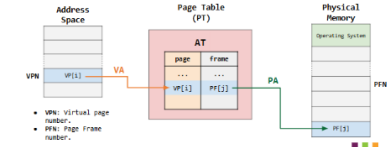
Paginación

Dividir el espacio de direcciones (address space) en unidades de tamaño fijo conocidas como páginas.



Page Frame (Marco)
Una marco (frame) es un bloque de memoria física de tamaño fijo el cual se encuentra asociado a una página (page) de la memoria virtual.

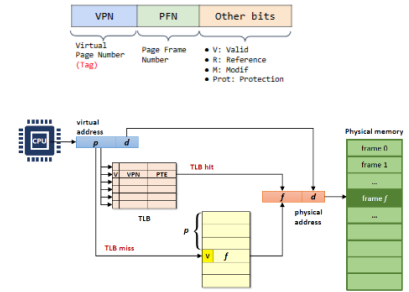
Page Tables
Permite la traducción de direcciones virtuales (VA) a direcciones físicas (PA) mapeando páginas (pages) en memoria virtual a marcos (frames) en memoria física.



TLB (Translation Lookaside Buffer)

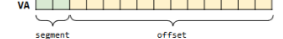
La TLB es una memoria cache on chip (En la CPU). Pequeña y rápida. Mantiene las direcciones más populares de la tabla de página (PT).
Tamaños típicos: 16 ~ 256 entradas. Usualmente es full asociativo (todas las entradas son comparadas en paralelo), pero puede haber asociatividad para reducir la latencia.

TLB Entry = VPN + PFN + Other bits



Traducción de direcciones- Pag

Las direcciones virtuales y físicas están compuestas de 2 partes:
VPN (virtual page number): Índice asociado a una página en la dirección virtual
PFN (Page frame number): Frame dentro del que se encuentra la dirección física.
offset: Desplazamiento dentro de la página seleccionada y son los bits menos significativos de la V.A.



Para obtener el número de páginas o marcos #pages, frames = size(address space)/size(page)
Para calcular los bits necesarios de la dirección virtual log, <posiciones de memoria> = m

Para calcular los bits necesarios del offset:

log, <bytes de las páginas> a n
Para calcular los bits necesarios del número de página (VPN):
V.A - offset = m - n

Traducción de direcciones

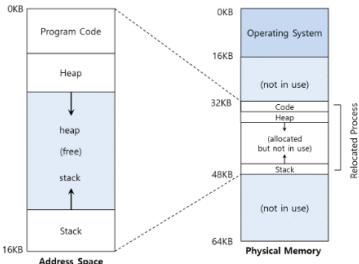
Se necesita un soporte en hardware (MMU)
MMU: Hardware encargado de realizar la traducción de direcciones lógicas (VA) a físicas (PA).

Fija un valor para el registro base

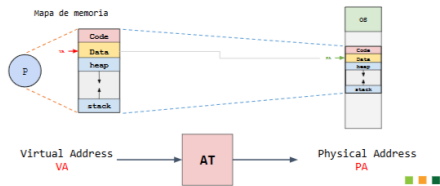
$$PA = VA + base$$

Las direcciones virtuales no deben ser mayores al valor en el registro bounds (limit) ni negativas.

$$0 \leq VA < bounds$$



Asociación entre direcciones virtuales y físicas



Políticas

Las políticas definen los criterios para seleccionar cuál página debe salir de memoria.

Política de reemplazo óptimo (OPT)

Se reemplaza la página que se accede más lejos en el futuro.
-No es práctica, solo sirve para comparar (caso más ideal).
-Se requiere que el sistema tenga conocimientos de los eventos futuros.

| | | | | | | | | | | | |
|---------------|---|---|---|---|---|---|---|---|---|---|---|
| Reference row | 0 | 1 | 2 | 0 | 1 | 3 | 0 | 3 | 1 | 2 | 1 |
| Page Frame | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| | | | 2 | 2 | 2 | 2 | 3 | 3 | 3 | 2 | 2 |
| | M | M | M | H | H | M | H | H | H | M | H |
| Evict | | | | | | | 2 | | | 3 | |

Política de reemplazo FIFO

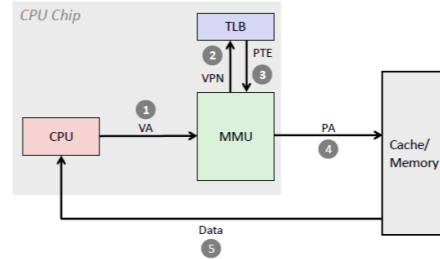
Reemplaza la página que ha entrado primero.
Las páginas entran a una cola cada que llegan a memoria.
Cuando se requiere un reemplazo, se selecciona la página al final de la cola (la primera que entró).

| | | | | | | | | | | | |
|---------------|---|---|---|---|---|---|---|---|---|---|---|
| Reference row | 0 | 1 | 2 | 0 | 1 | 3 | 0 | 3 | 1 | 2 | 1 |
| Page Frame | 0 | 0 | 0 | 0 | 0 | 3 | 3 | 3 | 3 | 2 | 2 |
| | | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| | | | 2 | 2 | 2 | 2 | 2 | 2 | 1 | 1 | 1 |
| | M | M | M | H | H | M | H | M | M | H | H |
| Evict | | | | | | | 0 | 1 | | 2 | 3 |

Address Translation usando TLB

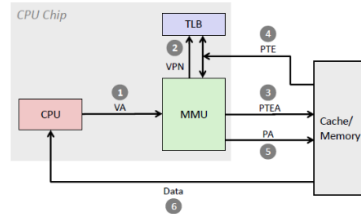
Caso Hit

Se elimina la necesidad de acceder a memoria física:
El procesador envía la VA a la MMU.
La MMU extrae la VPN de la VA.
Se verifica que la VPN se encuentre en la TBL, como en este caso esto es verdadero (TLB hit), se obtiene la PTE.
Se extrae la PFN de la PTE y se procede a formar la dirección física (PA) asociada a la dirección virtual.
Con la dirección física obtenida, se procede a cargar el dato que allí se encuentra a la CPU.



Caso Miss

En este caso, se da un acceso adicional a memoria (más exactamente a la PT). Afortunadamente los TLB misses son raros:
El procesador envía la VA a la MMU.
La MMU extrae la VPN de la VA.
Se verifica que la VPN se encuentre en la TBL, como en este caso no se encuentra (TLB miss), se tiene que acceder a la PT.
Una vez al obtiene el PFN (al acceder a la PT) asociado al VPN deseado, se procede a la actualización de la TLB.
Se realiza nuevamente la búsqueda original en la TLB ya actualizada produciéndose un TLB Hit lo cual hace posible formar la dirección física deseada.
Con la dirección física obtenida, se procede a cargar el dato que allí se encuentra a la CPU.



Bit present

El bit de presencia (P) permite conocer dónde se encuentra la página

| Valor | Significado |
|-------|---|
| 1 | La página requerida está presente en la memoria física. |
| 0 | La página requerida no está presente en la memoria física, pero sí en el disco. |

Nota: Pueden ocurrir fallos de pagina cuando un proceso se encuentra en el espacio swap y tiene que ser llevada a la memoria principal.

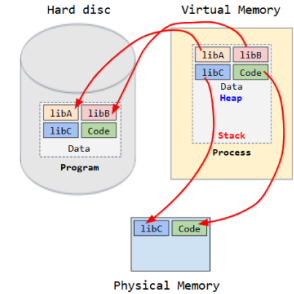
Más allá de la memoria física: mecanismos

¿Cómo ir más allá de la memoria física?

¿Cómo puede el sistema operativo usar un dispositivo más grande y lento para proporcionar de forma transparente la ilusión de un gran espacio de direcciones virtuales?

Swapping

El swapping consiste en mover procesos completos (o partes de ellos) desde la memoria principal (RAM) al disco duro (generalmente a un archivo o partición llamada swap) y viceversa. Esto se hace para liberar espacio en RAM y permitir que otros procesos se ejecuten.



Políticas

Política de reemplazo aleatoria

- Selecciona de manera aleatoria la página a reemplazar
- No tiene en cuenta qué bloque de datos se saca de memoria.
- Desempeño es aleatorio (Depende de la suerte cuando se elige la página a reemplazar).

| | | | | | | | | | | | |
|---------------|---|---|---|---|---|---|---|---|---|---|---|
| Reference row | 0 | 1 | 2 | 0 | 1 | 3 | 0 | 3 | 1 | 2 | 1 |
| Page Frame | 0 | 0 | 0 | 0 | 0 | 3 | 3 | 3 | 1 | 1 | 1 |
| | | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| | | | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| | M | M | M | H | H | M | M | H | M | H | H |
| Evict | | | | | | | 0 | 1 | | 3 | |

Política LRU (Least Recently Used)

Reemplaza la página menos recientemente usada.
Usa el pasado para predecir el futuro.
Cuando hay localidad la política LRU se aproxima al OPT.

| | | | | | | | | | | | |
|---------------|---|---|---|---|---|---|---|---|---|---|---|
| Reference row | 0 | 1 | 2 | 0 | 1 | 3 | 0 | 3 | 1 | 2 | 1 |
| Page Frame | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 2 |
| | | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| | | | 2 | 2 | 2 | 2 | 3 | 3 | 3 | 3 | 3 |
| | M | M | M | H | H | M | H | H | H | M | H |
| Evict | | | | | | | 2 | | | 0 | |