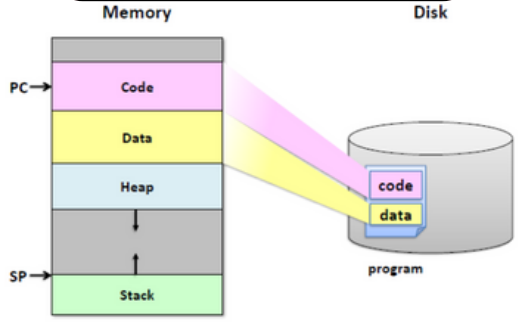
¿Cómo se ejecuta un programa?

1. Fetch
2. Decode
3. Execute
4. Again

Modos del procesador

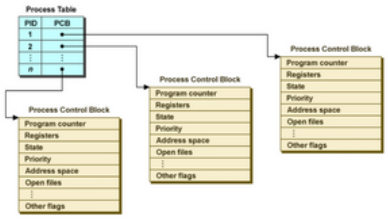
- **Modo Usuario:** No privilegiado, aplicaciones
- **Modo Kernel:** Privilegiado, acceso directo al hardware

Creación de un proceso



Estructuras de datos asociadas a un proceso

- Process table (PT)
- Process control Block (PCB)



¿Cuándo y cómo se cambia entre modos?

Cuando sucede un evento:

- **Llamada al sistema (Syscall):** Instrucciones *trap* (pasar a modo kernel) y *return from trap* (regresar a modo usuario).
- **Interrupción:** Mecanismo de hardware para que el SO retome control (timers, exceptions).

Danilo Antonio Tovar Arias
(danilo.tovar@udea.edu.co)
Juan Pablo Gomez López
(juan.gomez148@udea.edu.co)

Definición de un Sistema Operativo (SO)

Es un software que administra y controla de manera eficiente y justa la ejecución de programas.

↓

Realiza Virtualización y Administración de recursos

↓

Permite a las aplicaciones comunicarse con el SO mediante "System Calls"

Componentes de un proceso

1. Estado del procesador (Registros del procesador)
   - Program counter (PC).
   - Stack pointer (SP).
   - Registros de proposito general (GP).
2. Espacio de direcciones
   - Instrucciones.
   - Datos.
3. Información E/S

Ejecución directa limitada (EDL)

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.

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).

↓

Problema 1: Operaciones Restringidas
Solución: Usar transferencia de control restringida; En **modo usuario** las aplicaciones **no tienen** acceso total a los recursos, mientras que en **modo kernel** el SO **tiene** acceso total.

↓

Problema 2: Cambio entre procesos
Solución: Permitir al SO realizar sus procesos; versión cooperativa (yield) y versión no-cooperativa (timer interrupt and scheduler)

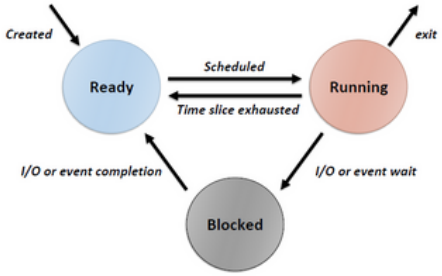
Virtualización de recursos físicos

Procesador	→	Proceso
Memoria	→	Espacio de direccionamiento
Disco	→	Archivo

Programa vs Proceso

Un programa es una entidad pasiva almacenada en disco.	Un proceso es un programa en ejecución, una entidad activa.

Estados de un proceso (Modelo de tres estados)



Multiprogramación

- **Multiprogramación:** Permite la ejecución de varios programas a la vez.
- **Time Sharing:** Técnica empleada por el sistema operativo para compartir la CPU entre varios procesos.

Planificación de procesos

¿Cuál proceso sigue según el scheduler? Se utiliza la carga de trabajo (*workload)* para caracterizar los procesos.

Métricas:

- **Turn-Around Time:** Es una métrica de desempeño, tiempo de finalización - tiempo de entrada.
- **Turn-Around Time Promedio:** Promedio del turn-around time entre diferentes procesos.
- **Tiempo de respuesta: Es** una métrica de equidad, tiempo de primera ejecución - tiempo de llegada.

Glosario: Stack (memoria automática), Stack (memoria dinámica), Cambio de contexto (Cambios de proceso en CPU), TrapTable (Almacen de funciones a nivel kernel), Algoritmo apropiativo (permite tomar el control de la CPU), Algoritmo no-apropiativo (no permite tomar el control de la CPU)

Algoritmos de planificación de procesos

- **FIFO (First In, First Out):** Primer proceso que llega, es el primero que termina.
- **SFJ (Shorstest Job First):** Se ejecuta primero el trabajo más corto; no apropiativo.
- **STCF (Shorstest Time-to-Completion First):** Se ejecuta el trabajo que demore menos en ser completado; apropiativo.
- **RR (Round Robin):** Ejecute un trabajo por una porción de tiempo (quantum), luego ejecute el siguiente trabajo en la cola.
- **MLFQ (Multi-Level Feedback Queue):** Utiliza multiples colas donde cada una tiene una prioridad, tal que se ejecuten los procesos con mayor prioridad (si hay empate de prioridad se aplica RR).
  - Asigna mayor prioridad a procesos que no hacen uso intensivo de CPU.
  - Asigna nuevos procesos al nivel de prioridad más alto.
  - Priority-Boost (Aging): Despues de un tiempo S, mover todos los procesos a la mayor prioridad.
  - Cuando un proceso cumple su tiempo de quantum, baja su prioridad.
  - **Mejoras al MLFQ:** Quantums diferentes a cada cola (menor prioridad → mayor quantum)

Problemas asociados a la asignación de memoria

Fragmentación Interna: Ocurre cuando al interior de los espacios de dirección de memoria física existen direcciones inutilizadas. Generalmente sucede con el método Base&Bound.

Fragmentación Externa: Ocurre cuando existen pequeños espacios libres de memoria que no son contiguos. Se intenta solucionar con compactación o algoritmos adicionales. Generalmente ocurre cuando se tienen particiones de tamaño variable.

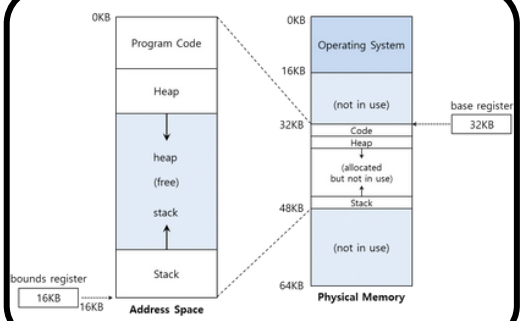
Danilo Antonio Tovar Arias
(danilo.tovar@udea.edu.co)
Juan Pablo Gomez López
(juan.gomez148@udea.edu.co)

Virtualización de memoria

El SO proporciona una ilusión sobre el espacio de memoria para cada proceso. Esto permite facilidad de uso, eficiencia y aislamiento entre procesos. A través del uso compartido de la memoria entre diferentes procesos. Sin embargo se generan problemas de protección (intentar modificar espacios de memoria que no pertenecen al proceso en ejecución).

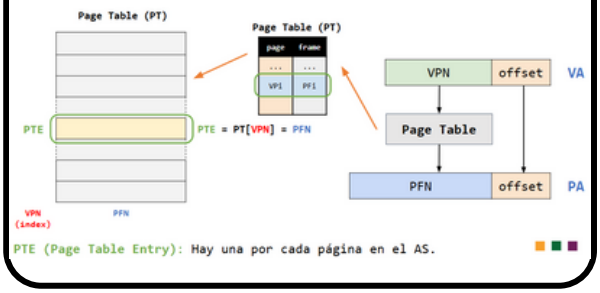
Para solucionar el problema se crean los espacios de direccionamiento (address space), la dirección virtual (virtual address) y la traducción de direcciones (Address Translation)

El SO es capaz de asignar el espacio de direcciones virtuales a cualquier espacio no ocupado de la memoria físca



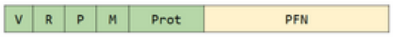
Kb = 2^10 bytes
Mb = 2^20 bytes
...

Tabla de direcciones



Estructura de una PTE (Page Table Entry)

- Bit de validez (V): Indica si la traducción es válida.
- Bits de protección (Prot): read - write - execute.
- Bit de presencia (P): Indica si la página está en la mem. física.
- Bit sucio (M): Indica si la página ha sido modificada.
- Bit de referencia (R): Indica si la página ha sido accedida.



Reasignación dinámica

Base & Bound: Para obtener la dirección física se realiza: ***dir. física = dir. virtual + reg. base*** ; con la restricción que
0 <= dir. virtual < reg. limite.

Segmentación: Dividir el espacio de direccionamiento en sus partes, tal que cada segmento puede ser ubicado en una sección diferente. Se requiere un par de registros base&bound para cada segmento.
dir. física = offset + reg. base

- Offset es la distancia entre la instrucción y su segmento lógico.
- Si se referencia una dirección invalida, se genera un fallo de segmentación (*out of bounds*).
- Se pueden compartir segmentos, y se utilizan bits de protección para definir los permisos (W - R - X)

Paginación: Se divide el espacio de direccionamiento en particiones de tamaño fijo llamadas paginas (*pages*), la memoria física en marcos de pagina (*frames*) del mismo tamaño y se crea una tabla de pagina (*page table*) para realizar la traducción.

Traducción de direcciones en paginación

Direcciones virtuales (VA): Están compuestas por un VPN (virtual page number) y un offset (desplazamiento al interior de la pagina seleccionada).

Direcciones físicas (PA): Están compuestas por un PFN (page frame number) y un offset (desplazamiento al interior del marco seleccionada).

Vir. Dir. length = $\log_2$(AS_size)
Phy. Dir. length = $\log_2$(Mem_size)
Offset = $\log_2$(page_size)
VPN/FPN = X. Dir. Length - Offset

Las tablas de pagina pueden ser muy grandes, y adicionalmente se crea una tabla por cada proceso, por lo que pueden ocupar mucho espacio. Así mismo aumentan los tiempos computacionales debido a la necesidad de acceder dos veces a memoria (una por la tabla de pagina y otra por el valor a utilizar en la instrucción)

Glosario: Starvation (un proceso no recibe tiempo de CPU), time slicing (equivalente al quantum), Address Space (espacio con la información necesaria para la ejecución de un proceso), Virtual Address (dirección lógica de memoria), MMU (memory management unit), Free List (lista de dirs. libres de memoria)

TLB (Translation Lookaside Buffer)

Corresponde a una memoria caché en HW (en la MMU) para acelerar el proceso de traducción de direcciones, a través del almacenamiento de valores populares (aprovechando la localidad espacial y temporal).

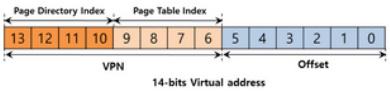
TAEM (tiempo de acceso efectivo a memoria) : HitRate * (Tiempo de acceso a TLB + Tiempo de acceso a memoria principal) + MissRate * (Tiempo de acceso a TLB + 2 * Tiempo de acceso a memoria principal)

Existe un problema en el cambio de contexto debido a que es necesario saber cuales entradas de la TLB corresponden al proceso en ejecución. Entonces existen dos soluciones, la primera es invalidar todas la entradas de la TLB al realizar el cambio de contexto; y la segunda es utilizar un identificador único para cada proceso denominado ASID.

A través de ASID se puede realizar un uso compartido de la misma dirección de memoria física.

Tablas de página multinivel

Se deja de utilizar una estructura lineal para las tablas de página, y se empieza a utilizar una estructura en árbol. Tal que se realiza una paginación de las tablas de página, y se gestionan a través de un directorio de páginas. Entonces se divide la tabla de página en pedazos del mismo tamaño de la página; y si una página entera de PTE es inválida, no se le asigna memoria a esa página.

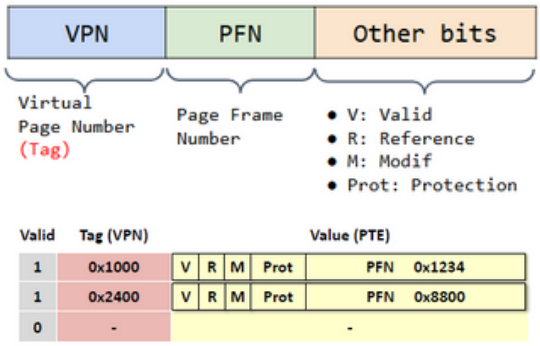


Casos de swapping

- **Page Fault (Fallo de página):** La página a la que se quiere acceder no se encuentra en memoria; se procede a ingresarla a memoria.
- **Memory full:** Se realiza *swap out* de una página, para hacer *swap in* de la página necesaria. En un caso real, se hace uso de *background thread* llamado *swap Daemon* encargado de mantener una pequeña porción de memoria libre.

Danilo Antonio Tovar Arias
(danilo.tovar@udea.edu.co)
Juan Pablo Gomez López
(juan.gomez148@udea.edu.co)

Estructura de una TLB



Valid	Tag (VPN)	Value (PTE)
1	0x1000	V R M Prot PFN 0x1234
1	0x2400	V R M Prot PFN 0x8800
0	-	-

Políticas de reemplazo en TLB

- **Last Recently Used (LRU):** Reemplaza la entrada de la TLB que no haya sido usada recientemente (la que lleva más tiempo sin ser accedida).
- **Random:** Realiza un reemplazo seleccionando una entrada al azar de la TLB.

Formato de una dirección virtual con páginas multinivel

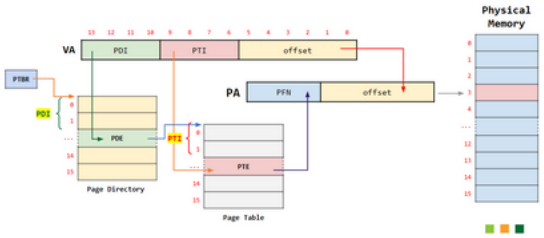
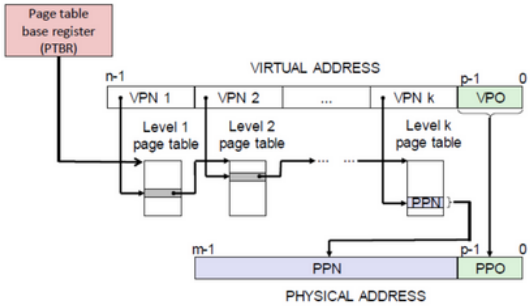


Tabla de página de k niveles



Políticas y métricas de reemplazo en swap

Métricas: **Average Memory Access Time → AMAT = P_HIT * T_M + P_MISS * T_D**, Simplificado: **T_M + P_MISS * T_D**

Políticas:

- **De reemplazo optimo (OPT):** Imposible de implementar, fácil de describir. Caso ideal.
- **FIFO:** Se saca la primera que entró, no puede determinar la importancia de las páginas.
- **Aleatoria:** Selección aleatoria, desempeño aleatorio.
- **LRU:** Selección del valor que lleva más tiempo sin ser utilizado, basado en datos pasados.
- **LFU:** Selección según la frecuencia de uso.

Problemas adicionales

Si se quiere reducir el tamaño de las tablas de pagina se puede aumentar el tamaño de cada pagina, sin embargo esto genera fragmentación interna.

Para mejorar el problema de tamaños grande de tabla de pagina se puede utilizar una aproximación hibrida entre paginación y segmentación, esto permite tener tamaños de pagina variable, a través de la definición de base & bounds para cada segmento de una tabla de pagina. Sin embargo, se genera fragmentación externa.

La traducción para estos casos se realiza así:
PTEAddr = base[seg] + (VPN * size(PTE))

Swapping Mechanisms

Existe una jerarquía de memoria en los sistemas modernos, tal que el componente más rápido y más pequeño son los registros de CPU, luego los Caché, la memoria principal y finalmente los discos que son los componentes más grandes pero más lentos.

Para proporcionar un tamaño más grande del que existe en la memoria principal se utilizan mecanismos como:

- **Memory Overlay:** Gestión manual de la memoria para cargar solo las secciones necesarias a memoria cuando sean requeridas.
- **Espacio Swap:** Sección del disco duro empleada para almacenar páginas de memoria. Tal que cada página del espacio de direcciones tiene tres posibilidades de mapeo:
  - Memoria física (V = 1, P = 1)
  - Memoria secundaria (V = 1, P = 0)
  - No se mapea - free (V = 0)
- Entre las funciones del espacio swap existe el swap out (*mem to disk*) y swap in (*disk to mem*); y al espacio de memoria (pagina) en espacio swap se conoce como bloque (*block*), mientras que en memoria principal se conoce como *frame*.

Glosario: Bit de presencia (P, corresponde al indicativo en una PTE que define si la pagina requerida se encuentra [1] o no [0] en memoria), Traza de memoria (cola/array de la cantidad de valores que puede "recordar" la memoria)