

Notas para Examen Parcial- Sistemas Operativos

Carlos Zapata Arango

Noviembre-2024

Virtualización de la CPU

El sistema operativo virtualiza la CPU para simular múltiples CPUs, usando mecanismos de bajo nivel como trampas e interrupciones y políticas de planificación de procesos.



Figure 1: Virtualización de la memoria: Ilustración de cómo el sistema operativo gestiona múltiples procesos simulando múltiples CPUs.

Mecanismos

Los mecanismos son los métodos o protocolos de bajo nivel que implementan una funcionalidad necesaria. Algunos ejemplos son:

- **Trampas e Interrupciones:** Estos mecanismos permiten al SO recuperar el control de la CPU de un proceso en ejecución.
- **Conmutación de Contexto:** Se guarda el estado de un proceso (registros, contador de programa, etc.) y se restaura el estado de otro proceso, permitiendo la multitarea.

Políticas

SJF (Shortest Job First): Ejecuta primero el proceso con el tiempo de ejecución más corto. El tiempo promedio de espera se calcula como:

$$T = \frac{\sum_{i=1}^n (T_{\text{final},i} - T_{\text{llegada},i} - T_{\text{ejecución},i})}{n}$$

STCF (Shortest Time-to-Completion First): Similar a SJF pero expropiativo. La misma fórmula de SJF se aplica, considerando interrupciones.

RR (Round Robin): Asigna un quantum de tiempo Q a cada proceso. El tiempo promedio de espera se calcula como:

$$T = \frac{\sum_{i=1}^n (T_{\text{respuesta},i} - T_{\text{llegada},i} - T_{\text{ejecución},i})}{n}$$

donde $T_{\text{respuesta},i}$ es el tiempo total transcurrido desde la llegada hasta la finalización.

MLFQ (Multi-Level Feedback Queue): Utiliza múltiples colas con diferentes prioridades. El tiempo de espera promedio y el tiempo de respuesta dependen de la estrategia de reubicación entre colas.

Planificación Proporcional: Asigna tiempo de CPU en proporción a los "tickets" o pesos. La probabilidad de selección es:

$$P(i) = \frac{\text{Tickets del proceso } i}{\sum_{j=1}^n \text{Tickets del proceso } j}$$

CFS (Completely Fair Scheduler): Utiliza tiempo de ejecución virtual para lograr una distribución justa. El tiempo de ejecución virtual se calcula como:

$$V_{\text{runtime},i} = \frac{T_{\text{ejecución},i}}{\text{Peso}_i}$$

donde $T_{\text{ejecución},i}$ es el tiempo real que el proceso ha utilizado y Peso_i es el peso asignado al proceso.

El Proceso

Tiempos en la Planificación de Procesos

La planificación de procesos implica medir y analizar tiempos importantes para evaluar el rendimiento y eficiencia del sistema operativo. Dos de los tiempos más relevantes son:

- **Turnaround Time:** Es el tiempo total desde la llegada de un proceso al sistema hasta su finalización. Se calcula como:

$$T_{\text{turnaround}} = T_{\text{completion}} - T_{\text{arrival}}$$

donde $T_{\text{completion}}$ es el tiempo de finalización del proceso, y T_{arrival} es el tiempo de llegada.

- **Response Time:** Es el tiempo transcurrido desde que el trabajo llega al sistema hasta que es programado por primera vez. Su fórmula es:

$$T_{\text{response}} = T_{\text{firstrun}} - T_{\text{arrival}}$$

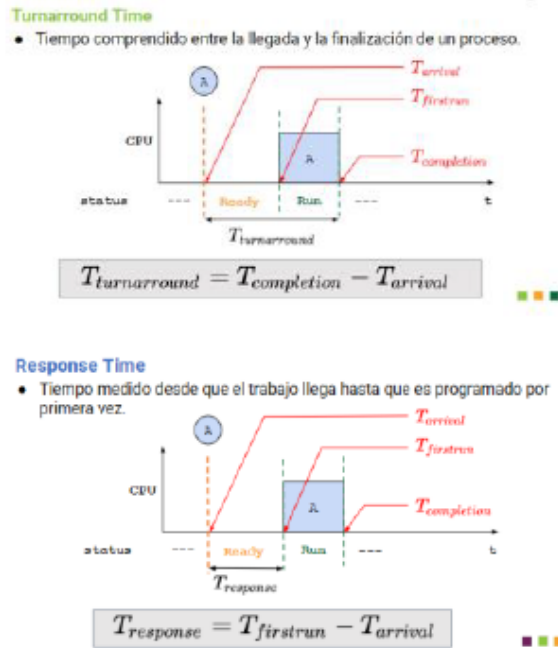


Figure 2: Visualización de los tiempos de Turnaround y Response Time.

donde $T_{firstrun}$ es el tiempo de inicio del proceso en el procesador.

Un proceso es una abstracción del SO para un programa en ejecución. Incluye el estado de la máquina del programa, como su memoria, registros, contador de programa y archivos abiertos. El SO proporciona una API de proceso para que los programas creen, destruyan y gestionen los procesos.

- **Estados del Proceso:** Los procesos pueden estar en diferentes estados, incluyendo en ejecución, listo para ejecutarse y bloqueado (esperando E/S).
- **Estructuras de Datos del Proceso:** El SO mantiene estructuras de datos, como la lista de procesos y el bloque de control del proceso (PCB), para rastrear el estado de los procesos.

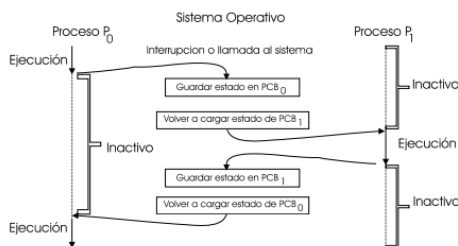


Figure 3: El ciclo de vida de un proceso: representación de los estados y transiciones comunes.

Estados de un Proceso

Un proceso, durante su ciclo de vida, puede encontrarse en uno de varios estados, gestionados por el sistema operativo:

- **En ejecución (Running):** El proceso está siendo ejecutado activamente en el procesador.
- **Listo (Ready):** El proceso está preparado para ejecutarse, pero el sistema operativo ha decidido posponer su ejecución en ese momento.
- **Bloqueado (Blocked):** El proceso está esperando que ocurra un evento, como la finalización de una operación de entrada/salida (E/S).

El sistema operativo gestiona las transiciones entre estos estados. Por ejemplo, un proceso bloqueado pasa a estar listo cuando se completa la operación de E/S asociada.

API de Procesos

El sistema operativo proporciona una API (Interfaz de Programación de Aplicaciones) que permite a los programas interactuar con los procesos. Algunas funciones comunes incluyen:

- **Crear (create):** Genera un nuevo proceso.
- **Eliminar (destroy):** Finaliza la ejecución de un proceso.
- **Esperar (wait):** Suspende la ejecución hasta que un proceso termine.
- **Operaciones de control:** Permite suspender o reanudar un proceso.
- **Estado (status):** Obtiene información sobre el estado de un proceso.

Ejemplos de llamadas al sistema

- **fork():** Crea un nuevo proceso que es una copia exacta del proceso actual (padre).
- **wait():** Espera a que un proceso hijo complete su ejecución.
- **exec():** Reemplaza el código del proceso actual con el de otro programa.

Estructuras de Datos del Sistema Operativo

El sistema operativo utiliza diversas estructuras de datos para la gestión de los procesos, entre las que se destacan:

- **Lista de Procesos (Process List):** Contiene información sobre todos los procesos activos en el sistema.
- **Bloque de Control de Proceso (PCB):** Almacena detalles específicos de cada proceso, como:

- Estado del proceso.
- Contador de programa.
- Registros de CPU.
- Límites de memoria.

Planificación Multiprocesador

La planificación en sistemas multiprocesador presenta desafíos adicionales, como la afinidad de la caché y el desequilibrio de la carga.

- **Afinidad de la Caché:** Es deseable mantener los procesos en el mismo CPU para aprovechar los datos almacenados en la caché.
- **Desequilibrio de Carga:** Los procesos deben distribuirse entre los CPUs para evitar que algunos CPUs estén inactivos mientras otros están sobrecargados.
- **SQMS (Single-Queue Multiprocessor Scheduling):** Utiliza una sola cola para todos los procesos, lo que facilita la implementación pero no escala bien.
- **MQMS (Multiple-Queue Multiprocessor Scheduling):** Utiliza múltiples colas para los procesos, lo que mejora la escalabilidad y la afinidad de la caché, pero puede llevar al desequilibrio de la carga.
- **Planificadores Multiprocesador de Linux:** Linux ha utilizado varios planificadores, como O(1), CFS y BF Scheduler, cada uno con sus propias fortalezas y debilidades.

Process	Arrival time	Execution time
P1	0	6
P2	1	4
P3	5	3
P4	6	1

P1	●
P2	●
P3	●
P4	●

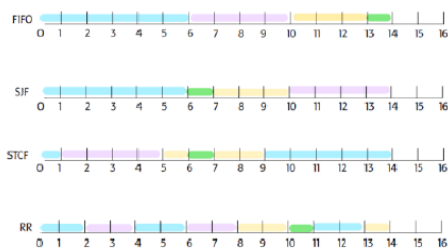


Figure 4: Ejemplo.

Fórmulas Importantes

- **Tiempo de Ejecución (TT):** $TT = T_{\text{completion}} - T_{\text{arrival}}$
- **Tiempo de Respuesta (RT):** $RT = T_{\text{firstrun}} - T_{\text{arrival}}$
- **Cálculo de Quantum Efectivo en CFS:**

$$\text{time slice}_k = \left(\frac{\text{weight}_k}{\sum_{i=0}^{n-1} \text{weight}_i} \right) \cdot \text{sched latency}$$

- **Cálculo de Tiempo de Ejecución Virtual en CFS:**

$$\text{vruntime}_i = \text{vruntime}_i + \left(\frac{\text{weight}_0}{\text{weight}_i} \right) \cdot \text{runtime}_i$$

Consejos para el Examen

- **Entender los conceptos básicos:** Asegúrese de comprender los conceptos clave como la virtualización de la CPU, los estados del proceso, los diferentes algoritmos de planificación y las métricas de planificación.
- **Conocer las fórmulas:** Memorice las fórmulas importantes para calcular el tiempo de ejecución, el tiempo de respuesta y otros parámetros relevantes.
- **Practicar con ejemplos:** Resuelva ejercicios de planificación de procesos, como el que resolvimos juntos, para familiarizarse con los diferentes algoritmos y cómo calcular las métricas.
- **Leer las referencias proporcionadas:** Los documentos contienen información detallada sobre los temas discutidos. Consulte las referencias para obtener una comprensión más profunda.