

## Lecturas 10 de Octubre

- 

Sistemas Operativos de tiempo real

- 

Sistemas Operativos Distribuidos y de Red

- 

Sistemas Operativos Multiprocesador (o Paralelos)

### Sistemas Operativos de tiempo real: Stalling pp. 463-466

La computación de tiempo real puede definirse como aquella en la que la corrección del sistema depende no sólo del resultado lógico de la computación sino también del momento en el que se producen los resultados.

En un sistema de tiempo real se producen unas tareas que dan soluciones a problemas que vienen del mundo exterior y que se producen en un espacio de tiempo determinado, pudiéndose establecer así un margen de tiempo límite para realizar las tareas. De este modo se pueden dividir las tareas en tareas de **tiempo real duro** si tienen que realizarse en un tiempo determinado y un exceso de tiempo en dicha actividad puede suponer un error grave para el sistema, o tareas de **tiempo real blando** cuyo plazo de tiempo no es estricto y la tarea se puede seguir ejecutando después de haber cumplido el plazo sin desencadenar errores graves.

Estas tareas se distinguen a su vez dada la periodicidad de su ejecución, de este modo hay **tareas aperiódicas** si su ejecución es única y está restringida a un plazo de tiempo, o **tareas periódicas** si se repiten n veces en un espacio de tiempo o una vez cada período t.

Los sistemas operativos de tiempo real pueden ser caracterizados por tener requisitos únicos en ***cinco áreas generales:***

- Determinismo
- Reactividad
- Control del usuario
- Fiabilidad
- Operación de fallo suave

El **determinismo** nos dice que unas tareas concretas van a tardar un tiempo concreto en ser finalizadas, un sistema no puede ser completamente determinista, pues se ve limitado por sus recursos y por la prioridad de las tareas a ejecutar, se centra en el tiempo previo al reconocimiento previo de una instrucción.

La **reactividad**, relacionada con el determinismo, se enfoca en el tiempo que tarda en servir a esa interrupción, además incluye algunos aspectos como:

1. La cantidad de tiempo necesario para ejecutar la rutina de servicio de la interrupción actual, RSI (si es necesario cambiar de contexto requerirá un tiempo adicional).
2. El tiempo que requiere para procesarla, que se define a nivel hardware.
3. El efecto anidamiento. Si una interrupción puede ser interrumpida por otra.

El **control del usuario** es generalmente mucho mayor en un sistema operativo de tiempo real que en sistemas operativos ordinarios. El usuario tiene un control “de grano fino” sobre la prioridad de la tarea. En él, el usuario distingue las tareas y se encarga de gestionar algunas características de gestión de memoria.

La **fiabilidad** en sistemas de tiempo real es mucho más trascendental que en los sistemas que no son de tiempo real, pues en estos se puede solucionar el problema re-arrancando el sistema, en los de tiempo real la degradación de los servicios ofrecidos a un sistema que responde a tareas en tiempo real puede ser mucho más catastrófica.

La **operación de fallo suave** es una característica de los sistemas para responder ante un fallo y ser capaz de minimizar las pérdidas sufridas, manteniendo la consistencia de ficheros y datos cuando sea posible. Además un sistema se considera estable cuando a pesar de no ser capaz de responder a todas las peticiones, es capaz de responder a las tareas más críticas.

Para cumplir los requisitos precedentes, los sistemas operativos de tiempo real incluyen de forma representativa las siguientes características:

- *Cambio de proceso o hilo rápido.*
- *Pequeño tamaño (que está asociado con funcionalidades mínimas).*
- *Capacidad para responder rápidamente a interrupciones externas.*
- *Multitarea con herramientas para la comunicación entre procesos como semáforos, señales y eventos.*
- *Utilización de ficheros secuenciales especiales que pueden acumular datos a alta velocidad.*
- *Planificación expulsiva basada en prioridades.*
- *Minimización de los intervalos durante los cuales se deshabilitan las interrupciones.*
- *Primitivas para retardar tareas durante una cantidad dada de tiempo y para parar/retomar tareas.*
- *Alarmas y temporizaciones especiales.*

En la mayoría de los casos los sistemas no son capaces de lidiar con todas las tareas asignadas, por lo que una de las principales áreas de trabajo hoy en día son los planificadores de tiempo real.

## **Sistemas Operativos Distribuidos y de Red: Stallings pp. 596-599**

Hoy en día es común usar un *sistema distribuido de procesador de datos (DDP)* para dividir y optimizar el trabajo. Un sistema DDP implica dividir las funciones de computación y organizar de forma distribuida las bases de datos, el control de dispositivos y el control de interacciones. Así se permita establecer una relación entre los ordenadores personales capaces de ejecutar “pequeños programas y tareas” y los servidores que son grandes bases de datos mucho más sofisticados.

Existen distintas estructuras distribuidas:

1. **Arquitectura de comunicaciones:** Es un software que da soporte a un grupo de computadores independientes, en red. Proporciona soporte para aplicaciones distribuidas, tales como correo electrónico, transferencia de ficheros y acceso a terminales remotos. Cada cual puede tener su propio S.O. y sólo se pueden comunicar directamente por deseo expreso.
2. **Sistema operativo de red:** Normalmente lo componen un único usuario con una o varias máquinas de servidores, que proporcionan acceso a servicios y aplicaciones, el usuario conoce la existencia de múltiples computadores y debe trabajar con ellos de forma explícita. Normalmente se utiliza una arquitectura de comunicaciones común para dar soporte a estas aplicaciones de red.
3. **Sistema operativo distribuido:** Es un sistema operativo común compartido por una red de computadores. A los usuarios les proporciona acceso transparente a los recursos de diversas máquinas. Un sistema operativo distribuido puede depender de una arquitectura de comunicaciones para las funciones básicas de comunicación, pero normalmente se incorporan un conjunto de funciones de comunicación más sencillas para proporcionar mayor eficiencia.

Para hacer un intercambio de información entre dos computadores, ya sea algo tan sencillo como un fichero, es necesario establecer un enlace, tanto directo como en red de comunicaciones, pero además se necesitan tareas como:

1. Que el emisor active el enlace directo o informar a la propia red de comunicaciones
2. Debe comprobar que el receptor puede recibir estos datos
3. Que existe un programa que sea capaz de recepcionar el fichero
4. Que se encargue de realizar una traducción si las representaciones de datos son incompatibles entre ambos sistemas

En relación a la comunicación de computadores y redes de computadores, hay dos conceptos de suma importancia:

- Protocolos.
- Arquitectura de comunicaciones o arquitectura de protocolos.

Un **protocolo** se utiliza para comunicar entidades de diferentes sistemas. Lo que se comunica, cómo se comunica y cuándo se comunica, debe hacerse de acuerdo a unas convenciones entre las entidades involucradas. se pueden definir como un conjunto de reglas que gobiernan el intercambio de datos entre dos entidades. Los elementos principales de un protocolo son los siguientes:

- **Sintaxis:** Incluye cosas tales como formatos de datos y niveles de señales.
- **Semántica:** Incluye información de control para realizar coordinación y gestión de errores.
- **Temporización:** Incluye ajuste de velocidades y secuenciamiento.

#### Arquitectura de protocolos:

Dada la complejidad de los sistemas, en la red hay que establecer una arquitectura de protocolos, que será la encargada de realizar las comunicaciones entre máquinas dividiendo las tareas en subtareas, en módulos, que contienen claves, mandatos, registros... otros que ese encarguen de comprobar que las comunicaciones están bien establecidas... En general, una estructura que sea capaz de responder de manera eficiente ante las diferencias de los sistemas que se están comunicando.

### Sistemas Operativos Multiprocesador (o Paralelos): Stalling pp. 172-175

Históricamente el computador se ha considerado como una unidad de procesamiento secuencial, donde toda instrucción se sucedía una tras otra, pero este concepto ha ido cambiando, pues hoy en día, por motivos de optimización, se utilizan nuevas técnicas de paralelismo (procesamiento de funciones en varios dispositivos a la vez). En particular el *Multiprocesamiento Simétrico (SMP)* y *los clusters*.

#### ARQUITECTURA SMP

Es útil ver donde encaja la arquitectura SMP dentro de las categorías de procesamiento paralelo. La forma más común de categorizar estos sistemas es la clasificación de sistemas de procesamiento paralelo introducida por Flynn. El cual propone las siguientes categorías de sistemas de computadores:

- **Única instrucción, único flujo de datos:** (*Single instruction single data (SISD) stream*). Un solo procesador ejecuta una única instrucción que opera sobre datos almacenados en una sola memoria.

- **Única instrucción, múltiples flujos de datos :** (*Single instruction multiple data (SIMD) stream*). Una única instrucción de máquina controla la ejecución simultánea de un número de elementos de proceso. Cada elemento de proceso tiene una memoria de datos asociada, de forma que cada instrucción se ejecuta en un conjunto de datos diferente a través de los diferentes procesadores. Los procesadores vectoriales y matriciales entran dentro de esta categoría.
- **Múltiples instrucciones, único flujo de datos :** (*Multiple instruction single data (MISD) stream*). Se transmite una secuencia de datos a un conjunto de procesadores, cada uno de los cuales ejecuta una secuencia de instrucciones diferente. Esta estructura nunca se ha implementado.
- **Múltiples instrucciones, múltiples flujos de datos:** (*Multiple instruction multiple data (MIMD) stream*). Un conjunto de procesadores ejecuta simultáneamente diferentes secuencias de instrucciones en diferentes conjuntos de datos.

Un sistema con esta última organización MIMD recibe el nombre de **cluster**, donde si cada procesador accede a una misma memoria se le conoce como **multiprocesador de memoria compartida**. Y en estos se usa una estructura de maestro/esclavo para controlar y organizar, pero esto tiene algunas desventajas, por ejemplo un fallo en el maestro puede derrumbar todo el sistema, o las limitaciones en este pueden afectar a los que se encuentran subyugados.

## ORGANIZACIÓN SPM

La organización de procesamiento multisinétrico permite que se planifiquen todas las tareas y se dividan en hilos previos a su procesamiento, cada procesador contiene sus propias unidades, pero se encuentran unidas a una memoria compartida

## DISEÑOS DE SISTEMAS OPERATIVOS MULTIPROCESADOR

El diseño de un S.O. permite encargarse de todas las peticiones sin que el usuario tenga que responder a estas, por lo tanto este diseño debe de tener unas características especiales como:

- **Procesos o hilos simultáneos concurrentes.** Las rutinas del núcleo necesitan ser reentrantes para permitir que varios procesadores ejecuten el mismo código del núcleo simultáneamente. Debido a que múltiples procesadores pueden ejecutar la misma o diferentes partes del código del núcleo, las tablas y la gestión de las estructuras del núcleo deben ser gestionadas apropiadamente para impedir interbloqueos u operaciones inválidas.
- **Planificación.** La planificación se puede realizar por cualquier procesador, por lo que se deben evitar los conflictos. Si se utiliza multihilo a nivel de

núcleo, existe la posibilidad de planificar múltiples hilos del mismo proceso simultáneamente en múltiples procesadores.

- **Sincronización.** Con múltiples procesos activos, que pueden acceder a espacios de direcciones compartidas o recursos compartidos de E/S, se debe tener cuidado en proporcionar una sincronización eficaz. La sincronización es un servicio que fuerza la exclusión mutua y el orden de los eventos.
- **Gestión de memoria.** La gestión de memoria en un multiprocesador debe tratar con todos los aspectos encontrados en las máquinas uniprocador. Además, el sistema operativo necesita explotar el paralelismo hardware existente, como las memorias multipuerto, para lograr el mejor rendimiento. Los mecanismos de paginación de los diferentes procesadores deben estar coordinados para asegurar la consistencia cuando varios procesadores comparten una página o segmento y para decidir sobre el reemplazo de una página.
- **Fiabilidad y tolerancia a fallos.** El sistema operativo no se debe degradar en caso de fallo de un procesador. El planificador y otras partes del sistema operativo deben darse cuenta de la pérdida de un procesador y reestructurar las tablas de gestión apropiadamente.