Administración de procesos: Procesos e hilos

Gunnar Wolf

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

Índice

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

¿Cuándo es proceso? ¿Cuando es programa?

Programa Una lista de instrucciones a seguir, una *entidad pasiva*

Proceso *Entidad activa* que:

- *Emplea* al programa
- Típicamente opera sobre un *conjunto de datos*
- Tiene *información de estado* que indica, entre otras cosas, en qué punto va la ejecución.

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

¿Cuando es proceso? ¿Cuándo es tarea?

Tarea Equivalente a un proceso en un *sistema por lotes*; requiere típicamente menos metainformación.

La distinción proceso-tarea no es del todo clara u objetiva.

Hay textos que emplean uno u otro término indistintamente

Nosotros emplearemos siempre el término *proceso*.

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

La ilusión de la concurrencia

- Un sistema actual nos da la *ilusión* de ejecución simultánea de muchos procesos
- La realidad: Casi todos están suspendidos, esperando que los active el planificador
- En un momento dado sólo puede estarse ejecutando un número de procesos igual o menor al número de CPUs que tenga el sistema.
- Esa ilusión tiene grandes costos... Especialmente pensando con suficiente velocidad

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

Índice

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

Estados de un proceso

Un proceso, a lo largo de su vida, alterna entre diferentes *estados* de ejecución. Estos son:

Nuevo Se solicitó al sistema la creación de un proceso, y sus recursos y estructuras están siendo creadas

Listo Está listo para ser asignado para su ejecución

En ejecución El proceso está siendo ejecutado

Bloqueado En espera de algún evento para poder continuar ejecutándose

Terminado El proceso terminó de ejecutarse; sus estructuras están a la espera de ser *limpiadas* por el sistema

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

Estados de un proceso

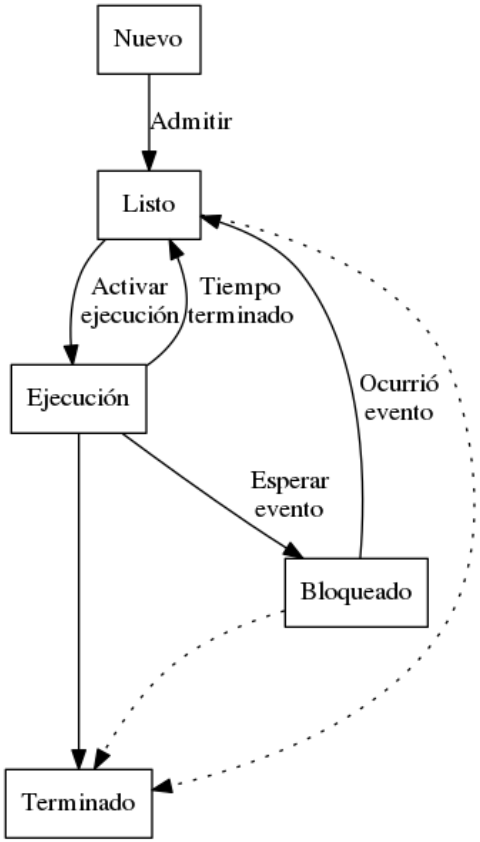

Figura: Diagrama de transición entre los estados de un proceso

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

El bloque de control del proceso (PCB) (1)

¿Qué información debe mantener el sistema acerca de cada proceso?
O, ¿Cuánto cuesta un cambio de contexto?

Estado del proceso El estado actual del proceso

Contador de programa Cuál es la siguiente instrucción a ser ejecutada

Registros del CPU Información específica del estado del CPU

Información de planificación (scheduling) La prioridad del proceso, la *cola* en que está agendado, y demás información que puede ayudar al sistema operativo a agendar al proceso

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

El bloque de control del proceso (PCB) (2)

Información de administración de memoria Las tablas de mapeo de memoria (páginas o segmentos, dependiendo del sistema operativo).

Estado de E/S Listado de dispositivos y archivos asignados que el proceso tiene *abiertos* en un momento dado.

Información de contabilidad Información de la utilización de recursos que ha tenido este proceso

- Tiempo *de usuario* y *de sistema*
- Uso acumulado de memoria y dispositivos
- etc.

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

Índice

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

El problema con los hilos

Aunque los hilos parecen ser un pequeño paso partiendo del cómputo secuencial, de hecho, son un paso inmenso. Descartan las propiedades más esenciales y atractivas del cómputo secuencial: Facilidad de comprensión, predictabilidad y determinismo. Los hilos, como un modelo de computación, son salvajemente no-determinísticos, y el trabajo del programador se convierte en podar ese no-determinismo.
— El problema con los hilos, Edward A. Lee, UC Berkeley, 2006

El peso de los procesos

- La cantidad de información implicada en un cambio de contexto es muy grande
- Desperdicio *burocrático* de recursos
- Una respuesta: *procesos ligeros* (*Lightweight processes*, LWP)
  - Diversos *hilos de ejecución* dentro de un mismo proceso

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

Extrapolando

- Un proceso que no usa hilos es *un sólo hilo*
- Un sistema operativo que no ofrece soporte para hilos agendaría a nuestro proceso como a cualquier otro

La diferencia desde dentro

- Entre diferentes procesos, cada uno tiene ilusión de *exclusividad virtual* sobre la computadora
  - Espacio de direccionamiento de memoria exclusivo
  - Descriptores de archivos y dispositivos independientes
- Los hilos *son y deben ser conscientes* de su coexistencia
  - Comparten memoria, descriptores de archivos y dispositivos
  - Pueden tener *variables* locales y globales

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

¿Qué tanto se *aligera* el PCB?

- Estado del proceso
- Cada hilo se ejecuta de forma aparentemente secuencial
  - Tiene su propio *contador de programa*
  - Y su propia pila de llamadas (*stack*)
- Conjunto de variables locales
  - Puede ser implementado de forma muy sencilla
  - p.ej. con arreglos indexados por ID de hilo
- Información de planeación *interna*
- Pueden llevar información de contabilidad
- No requerimos:
  - Registros del CPU
  - Estado de E/S

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

Patrones de trabajo: Jefe / trabajador

- Un hilo tiene una tarea distinta de todos los demás
- El hilo *jefe* genera o recopila tareas
- Los hilos *trabajadores* efectúan el trabajo.
- Modelo común para procesos servidor, GUIs que procesan eventos
- El jefe puede llevar la contabilidad de los trabajos realizados

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

Patrones de trabajo: Jefe / trabajador

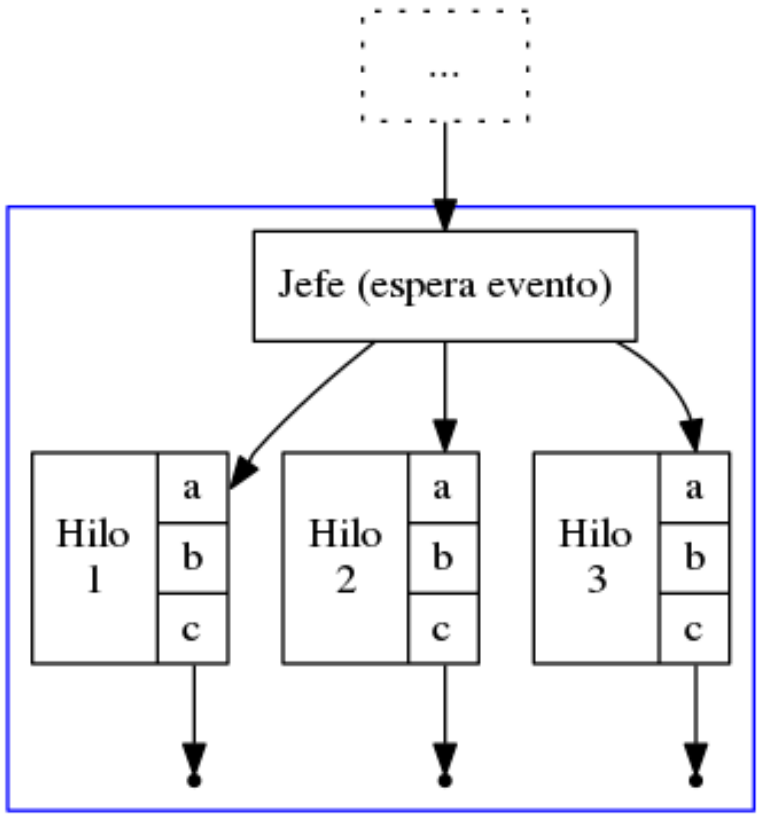

Figura: Patrón de hilos jefe/trabajador

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

Patrones de trabajo: Equipo de trabajo

- A partir de hilos idénticos
- Realizarán las mismas tareas sobre diferentes datos
- Muy frecuentemente utilizado para cálculos matemáticos (p.ej. criptografía, render).
- Puede combinarse con jefe/trabajador para generar previsualizaciones (la tarea se realiza progresivamente)

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

Patrones de trabajo: Equipo de trabajo

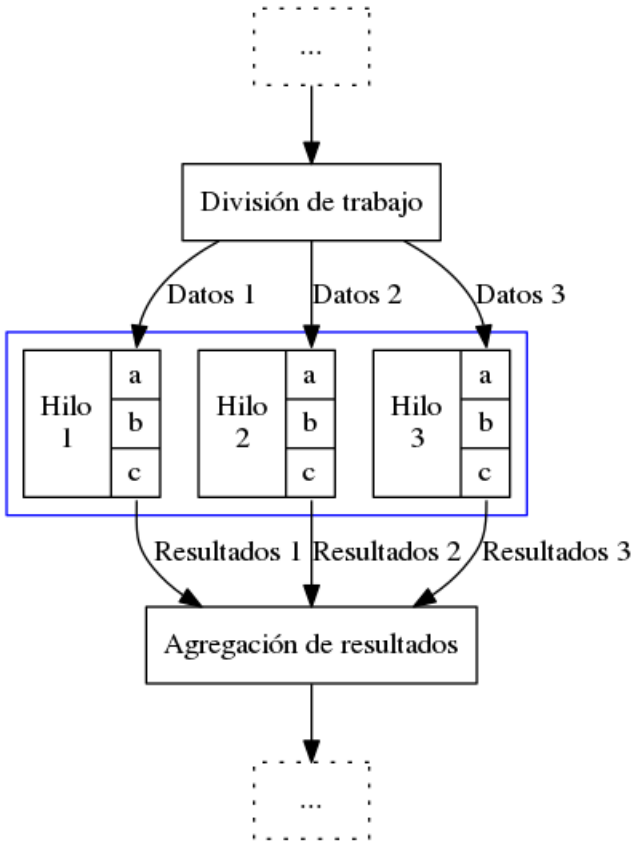

Figura: Patrón de hilos *Equipo de trabajo*

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

Patrones de trabajo: Línea de ensamblado

- Una tarea larga que puede dividirse en pasos
- Cada hilo se enfoca en una sóla tarea
- Pasa los datos a otro hilo conforme va terminando
- Ayuda a mantener las rutinas simples de comprender
- Permite continuar procesando datos cuando hay hilos esperando E/S

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

Patrones de trabajo: Línea de ensamblado

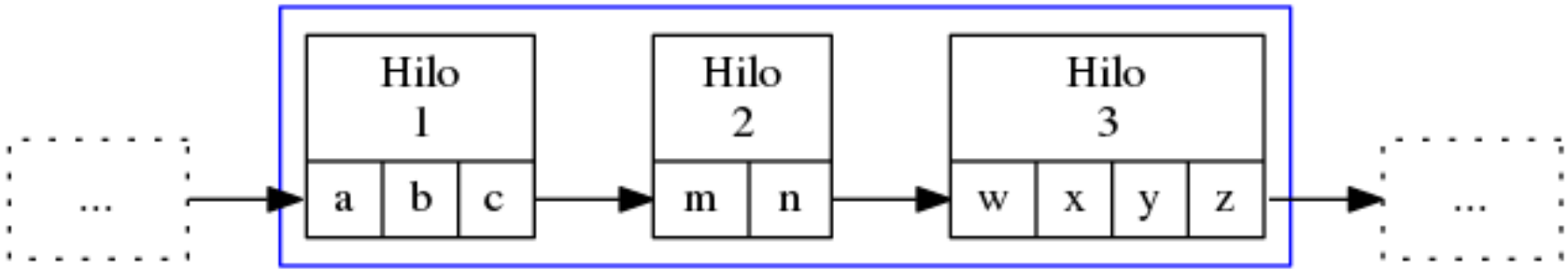

Figura: Patrón de hilos *Línea de ensamblado*

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

Índice

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

Hilos de usuario (o hilos verdes)

- Los hilos pueden implementarse 100 % con las facilidades del proceso
  - Caso extremo: Programas multihilos en sistemas operativos mínimos (o directo sobre el hardware)
- Mayor portabilidad
- A través de alguna biblioteca *del lenguaje/entorno de programación*
- Típicamente multitarea *interna* cooperativa

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

Pros y contras de los hilos de usuario

Ganamos:

- Espacio de memoria compartido sin intervención del sistema operativo
- Mayor rapidez para realizar trabajos cooperativos o multiagentes

Perdemos:

- Cualquier llamada al sistema interrumpe a *todos los hilos*
- No aprovechan el multiprocesamiento

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

Hilos nativos (o hilos de kernel)

- A través de bibliotecas de sistema que *informan* al sistema
- El núcleo se encarga de la multitarea *preventiva* entre los hilos
- El proceso *puede* pedir al sistema un mayor número de procesadores
  - Logrando ejecución verdaderamente paralela
- El sistema puede priorizar de diferente manera a un proceso multihilo

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

Índice

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

Referencia bibliográfica

Para este tema, recomiendo fuertemente revisen «The little book of semaphores» de Allen Downey (2008) (disponible para su descarga como PDF).

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

¿Qué es concurrencia?

- Quedó ya claro que (para nosotros) concurrencia no significa que dos o más eventos ocurran al mismo tiempo
- Nos referimos a dos o más eventos *cuyo órden no es determinista*
- No podemos predecir el órden relativo en que ocurrirán

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

¿Cuándo hablamos de concurrencia?

- Dos o más hilos del mismo proceso
- Dos o más procesos en la misma computadora
- Dos o más procesos en computadoras separadas conectadas por red
- Dos o más procesos que ocurran sin conexión alguna y posteriormente requieran sincronización

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

La sincronización implica involucramiento del programador

El sistema operativo brinda la *ilusión* a cada proceso de estar ejecutando en una computadora dedicada, pero...

- El proceso puede depender de datos obtenidos en fuentes externas
- Puede ser necesario esperar a que otro proceso haya pasado cierto punto en su ejecución
  - ...Que tengamos ya los resultados parciales de un cómputo paralelo
  - ...O que no más de m o menos de n procesos estén en determinado punto
  - ...O notificar a otro proceso de nuestro avance, o...

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

La sincronización como comunicación entre procesos (IPC)

- En esta sección veremos ejemplos de *primitivas de sincronización*
- Todas ellas son modalidades de *comunicación entre procesos* (IPC)
- Nos centraremos principalmente en los *semáforos*

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

Problemas clásicos para ir pensando

- Problema productor-consumidor
- Problema lectores-escritores
- La cena de los filósofos
- Los fumadores compulsivos

. . . Y hay $\approx$ 20 problemas más explicados en *The little book of semaphores*.

¡Revísenlo!

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

Problema productor-consumidor

- Pensemos un entorno multihilo como una línea de ensamblado
  - Algunos hilos *producen* ciertas estructuras (p.ej. eventos en un GUI)
  - Otros hilos las *consumen* (procesan los eventos)
- Los eventos se *apilan* en un buffer disponible para todos los hilos
- ¿Cómo podemos asegurar que dos hilos no modifiquen al buffer al mismo tiempo?
- ¿Cómo podemos evitar que los consumidores hagan *espera activa*?

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

Problema lectores-escritores

- Muchos procesos *lectores* pueden usar simultáneamente una estrucura
  - Si algún proceso *escritor* la requiere, debemos evitar que cualquier *lector* esté activo
- Requisitos de sincronización:
  - Sin límite en la cantidad de *lectores activos*
  - Los escritores deben tener *acceso exclusivo* a la *sección crítica*
  - Evitar *inanición* de escritores por exceso de lectores

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

La cena de los filósofos (Edsger Dijkstra, 1965)

- Una mesa redonda
- Tazón de arroz al centro
- Cinco platos, cinco filósofos, cinco palillos chinos
- Los filósofos piensan hasta tener hambre.
  - Cuando tienen hambre, buscan comer
  - No hablan entre sí
- Sólo un filósofo puede sostener un palillo a la vez
- ¿Qué puede salir mal?
  - ¿Cómo evitarlo con sólo primitivas de sincronización?

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM

Los fumadores compulsivos (Suhas Patil, 1971)

- Tres fumadores empedernidos
  - Con cantidades ilimitadas de uno de tres ingredientes cada uno: Tabaco, papel, cerillos
- Un *agente* que consigue ingredientes independientemente
- ¿Cómo asegurarse de que los recursos se utilizan siempre, *tan pronto como estén disponibles*?
  - Sin que el agente sepa quién tiene qué ingrediente

Gunnar Wolf • sistop@gwolf.org • http://sistop.gwolf.org
Sistemas Operativos (1554) • Facultad de Ingeniería UNAM