

Universidad Autónoma
de Nuevo León

Facultad de Ingeniería
Mecánica y Eléctrica

EQUIPO 2

SISTEMAS OPERATIVOS

Docente: Norma Edith Marín Martínez

Grupo: 001 Salón: 1105

Multitarea y Control de Concurrencia

Integrantes

Matricula	Nombre	Carrera	Semestre
1950074	José Amhed Vela Canales	IAS	6to
1953575	Emmanuel Sánchez Aranda	ITS	6to
1959360	Jonathan Francisco Galvan Villanueva	ITS	7mo
2109542	Marco Antonio Arreola de Leon	ITS	6to
1961463	Ángel David Gómez González	ITS	7mo
1895340	Jorge Alberto Morales Reyes	ITS	6to

Índice



Introducción----- 4



Procesos e hilos ----- 5



Tipos de procesos concurrentes ----- 20



Tipos de interacciones dentro de la concurrencia----- 23



Procesos concurrentes ----- 26



Referencias bibliográficas ----- 30



Introducción

La multitarea es un concepto fundamental en los sistemas operativos modernos que permite a una computadora realizar múltiples tareas simultáneamente o de manera aparentemente simultánea. En lugar de ejecutar una sola tarea a la vez, un sistema operativo con capacidad de multitarea permite que varias aplicaciones o procesos se ejecuten en paralelo, lo que mejora significativamente la eficiencia y la utilidad de una computadora.

En la siguiente presentación se hablará a profundidad como la multitarea y el control de concurrencia son fundamentales en un sistema operativo.

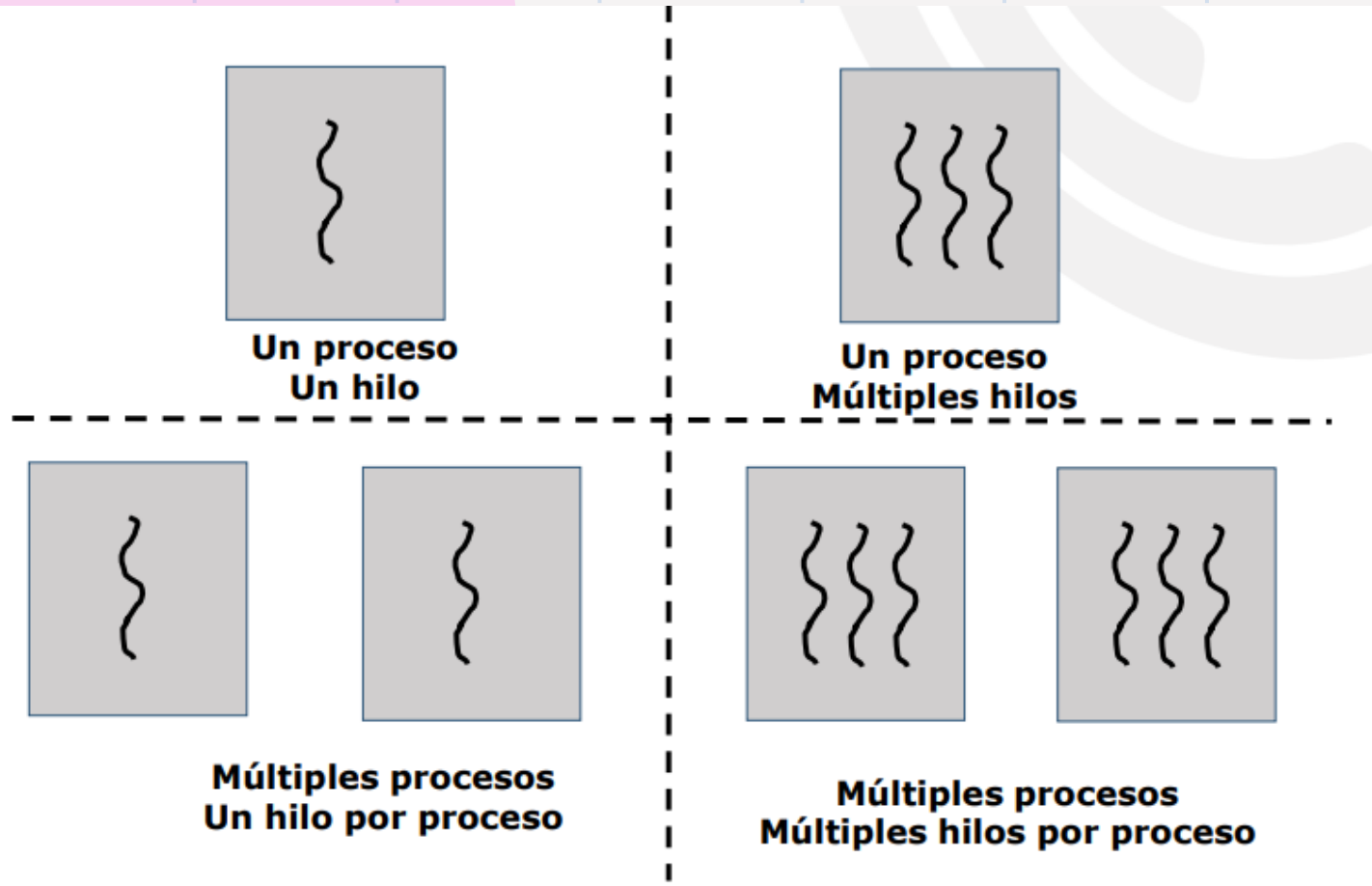
¿Qué es un proceso?

Instancia de un programa en ejecución que está aislado de otros procesos en la misma máquina.



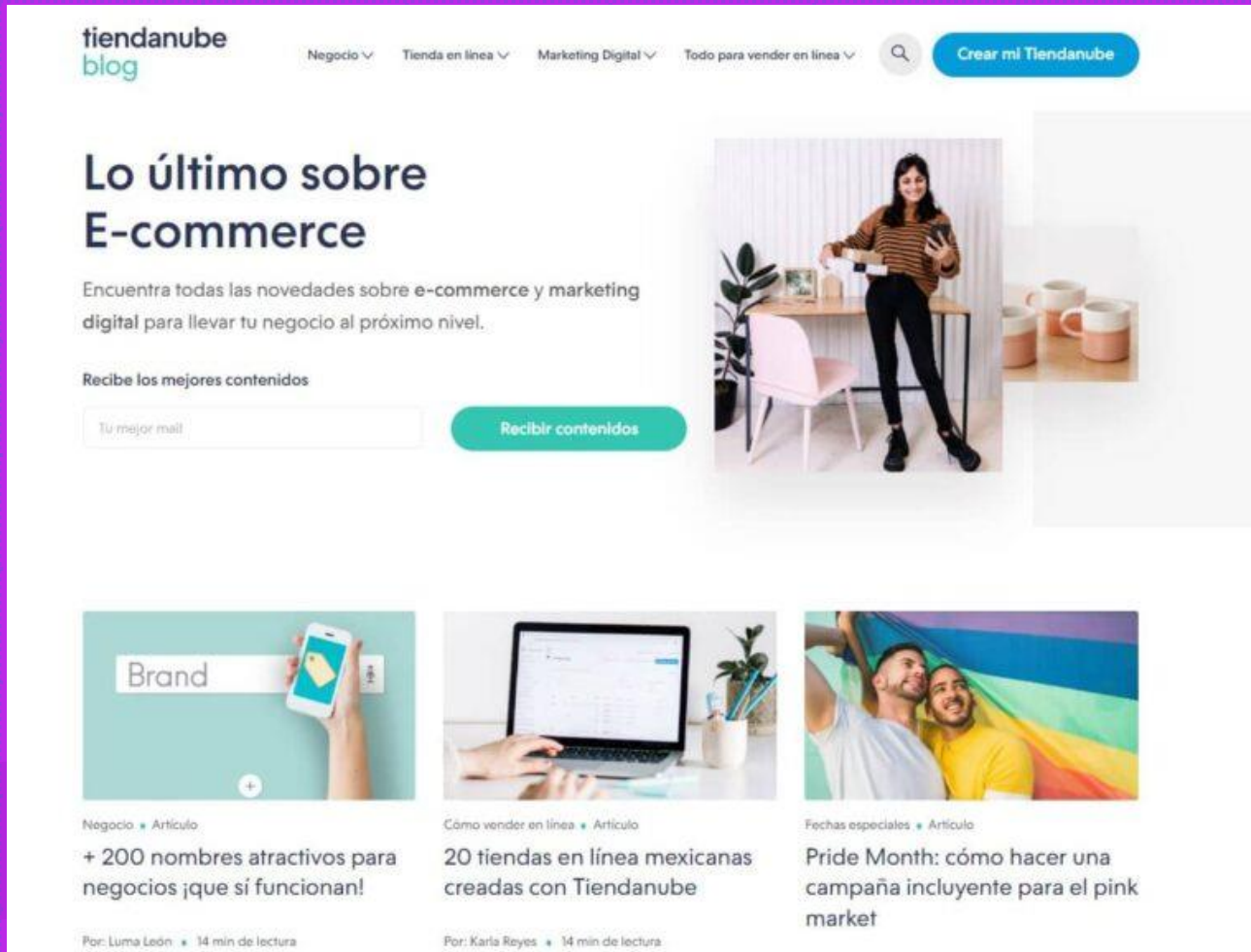


¿Qué es un hilo?



Un hilo (thread o proceso de peso liviano) es una unidad básica de utilización de CPU, consiste de:

- contador de programa
- conjunto de registros
- espacio de pila



HTML



CSS



JAVASCRIPT



EVENTOS DEL MOUSE



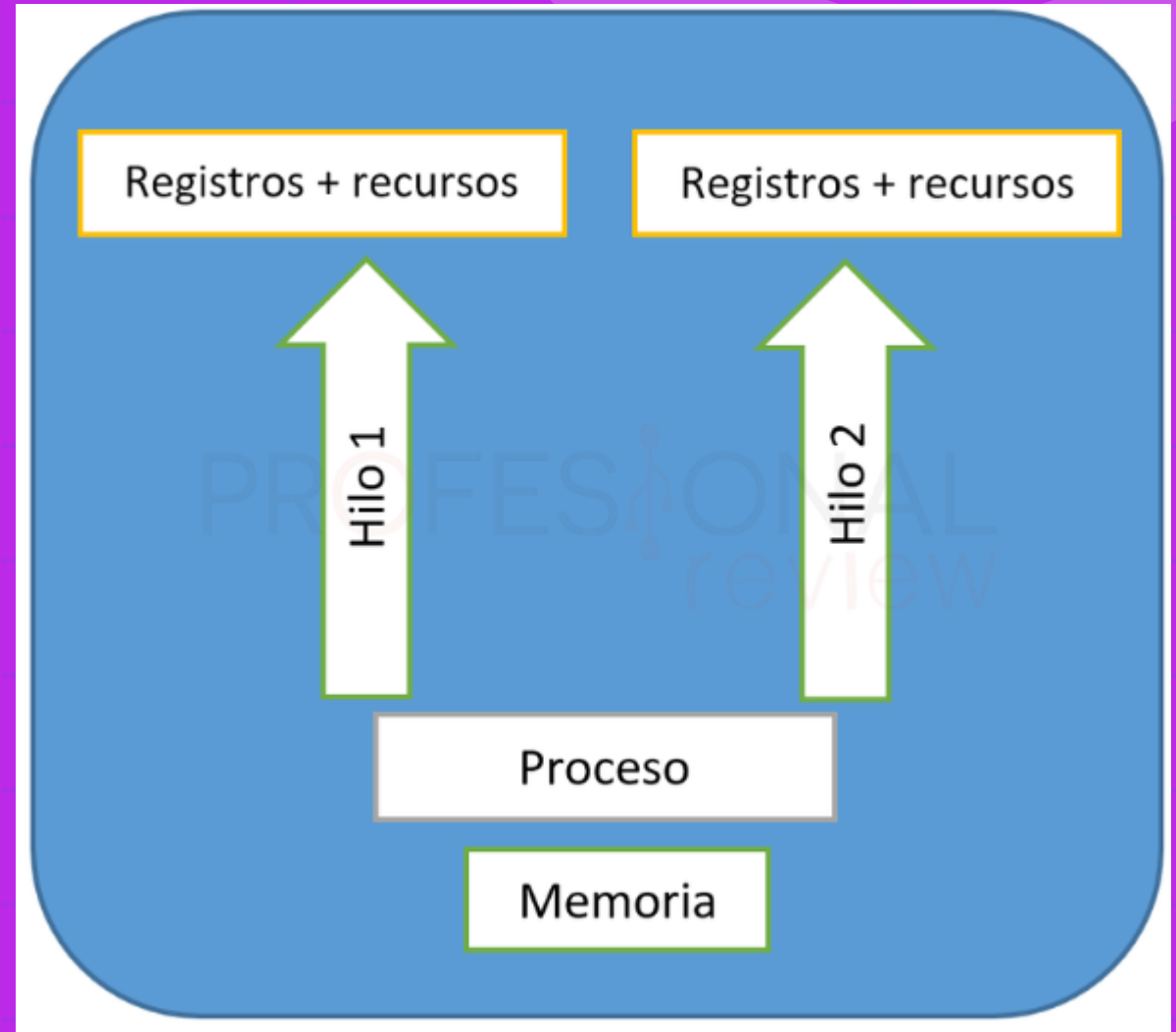
EVENTOS DEL TECLADO

Espacio propio de memoria

Asignación de recursos

Planificación de procesos

Control de acceso



Estados de los procesos

NUEVO



El proceso está siendo creado

LISTO



Listo para ser ejecutado, pero aún no se le ha asignado tiempo de CPU.

EJECUCIÓN



En ejecución y utiliza la CPU para realizar su trabajo.

BLOQUEADO



El proceso está temporalmente detenido

TERMINADO

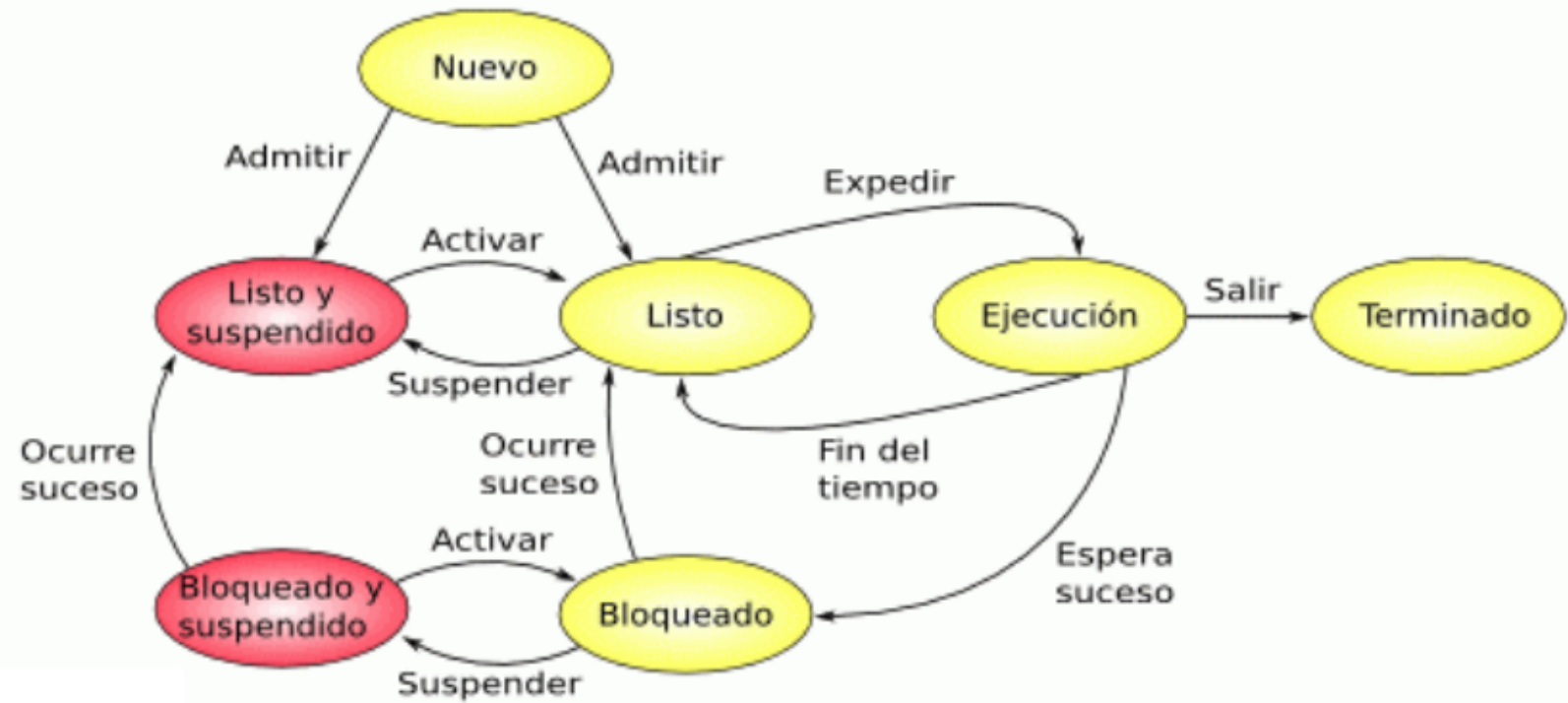


El proceso ha finalizado su ejecución y está siendo eliminado



Transiciones de Estados

Representan las diferentes fases por las que pasa un proceso desde su inicio hasta su finalización.



dreamstime.

Estados de energía

Características sistema S1:

Consumo de energía

Menor consumo que en S0 y mayor que en los demás estados de suspensión. El reloj del procesador está apagado y los relojes de bus se detienen.

Reanudación de software

El control se reinicia donde lo dejó.

Latencia de hardware

Normalmente, no más de dos segundos.

Contexto de hardware del sistema

Todo el contexto retenido y mantenido por el hardware.

Características:

Sistema S2:

El estado de energía del sistema S2 es similar a S1, salvo que el contexto de CPU y el contenido de la memoria caché del sistema se pierden porque el procesador pierde energía. El estado S2 tiene las siguientes características:

Consumo de energía

Menor consumo que en el estado S1 y mayor que en S3. El procesador está desactivado. Se detienen los relojes de bus; algunos buses pueden perder energía.

Reanudación de software

Después de la reactivación, el control se inicia desde el vector de restablecimiento del procesador.

Latencia de hardware

Sistema S3

Consumo de energía

Menos consumo que en el estado S2. El procesador está apagado y algunos chips de la placa base también podrían estar apagados.

Reanudación de software

Después del evento de reactivación, el control se inicia desde el vector de restablecimiento del procesador.

Latencia de hardware

Casi indistinguible de S2.

Contexto de hardware del sistema

Sistema 4

Consumo de energía

Desactivado, excepto en el caso de la corriente complicada del botón de encendido y dispositivos similares.

Reanudación de software

El sistema se reinicia desde el archivo de hibernación guardado. Si no se puede cargar el archivo de hibernación, es necesario reiniciar. La reconfiguración del hardware mientras el sistema está en el estado S4 podría dar lugar a cambios que impiden que el archivo de hibernación se cargue correctamente.

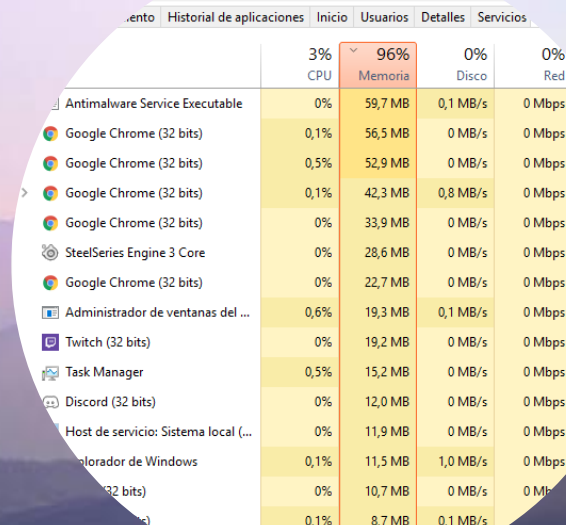


Transiciones

Las transiciones son animaciones que se usan para mantener a los usuarios orientados durante los cambios de estado de la interfaz de usuario (UI) y las manipulaciones de objetos, y hacer que esos cambios se sientan suaves en lugar de jarr. Las buenas transiciones se sienten naturales, a menudo dando la ilusión de que los usuarios interactúan con objetos reales.

Control y Gestión de Windows

1. El sistema de Windows lleva a cabo su control y gestión desde el administrador de tareas, el administrador de tareas es una aplicación integrada en los sistemas operativos de Windows con la cual podremos obtener información de los programas y procesos que se ejecutan en el equipo, además de proporcionar los indicadores de rendimientos más utilizados.



The screenshot shows the Windows Task Manager Performance tab. At the top, there are tabs for 'Historial de aplicaciones', 'Inicio', 'Usuarios', 'Detalles', and 'Servicios'. Below these, a summary bar shows: CPU at 3%, Memoria at 96% (highlighted in red), Disco at 0%, and Red at 0%. The main table lists running processes with their CPU, Memory, Disk, and Network usage.

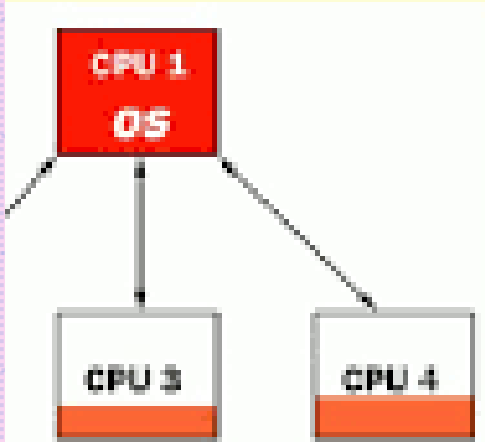
	3% CPU	96% Memoria	0% Disco	0% Red
Antimalware Service Executable	0%	59,7 MB	0,1 MB/s	0 Mbps
Google Chrome (32 bits)	0,1%	56,5 MB	0 MB/s	0 Mbps
Google Chrome (32 bits)	0,5%	52,9 MB	0 MB/s	0 Mbps
Google Chrome (32 bits)	0,1%	42,3 MB	0,8 MB/s	0 Mbps
Google Chrome (32 bits)	0%	33,9 MB	0 MB/s	0 Mbps
SteelSeries Engine 3 Core	0%	28,6 MB	0 MB/s	0 Mbps
Google Chrome (32 bits)	0%	22,7 MB	0 MB/s	0 Mbps
Administrador de ventanas del ...	0,6%	19,3 MB	0,1 MB/s	0 Mbps
Twitch (32 bits)	0%	19,2 MB	0 MB/s	0 Mbps
Task Manager	0,5%	15,2 MB	0 MB/s	0 Mbps
Discord (32 bits)	0%	12,0 MB	0 MB/s	0 Mbps
Host de servicio: Sistema local (...)	0%	11,9 MB	0 MB/s	0 Mbps
Explorador de Windows	0,1%	11,5 MB	1,0 MB/s	0 Mbps
Google Chrome (32 bits)	0%	10,7 MB	0 MB/s	0 Mbps
Google Chrome (32 bits)	0,1%	8,7 MB	0,1 MB/s	0 Mbps

Multiprocesamiento simetrico

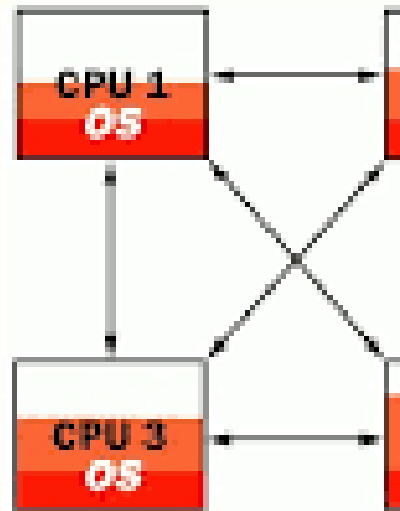
1. El multiprocesamiento simétrico es aquel en el que todo el procesador ejecuta las tareas en el sistema operativo. No tiene una relación maestro-esclavo como multiprocesamiento asimétrico. Todos los procesadores aquí, se comunican utilizando la memoria compartida.

Bases para la comparación	Multiprocesamiento simétrico
BASIC	Cada procesador ejecuta las tareas en el sistema operativo.
Proceso	El procesador toma los procesos de una cola preparada común o puede haber una cola preparada para cada procesador.
Arquitectura	Todo el procesador en multiprocesamiento simétrico tiene la misma arquitectura.
Comunicación	Todos los procesadores se comunican con otro procesador mediante una memoria compartida.
Fracaso	Si un procesador falla, la capacidad de cálculo del sistema se reduce.
Facilitar	El multiprocesador simétrico es complejo ya que todos los procesadores deben estar sincronizados para mantener el equilibrio de la carga.

Multiproceso OS (continuación)



Asimétrico



Simétrico

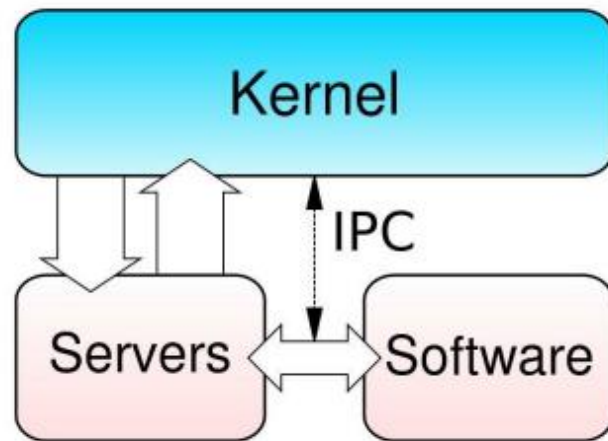
Diferencias clave entre multiprocesamiento simétrico y asimétrico

El punto más distinguible entre el multiprocesamiento simétrico y asimétrico es que las tareas en el sistema operativo son manejadas solamente por el procesador maestro en el multiprocesamiento asimétrico. Por otro lado, todos los procesadores en multiprocesamiento simétrico ejecutan las tareas en el sistema operativo.

En el multiprocesamiento simétrico, cada procesador puede tener su propia cola privada de procesos listos, o pueden tomar procesos de una cola lista común. Pero, en el multiprocesamiento asimétrico, el procesador maestro asigna procesos a los procesadores esclavos.

Micronucleos

- 1 Un micronúcleo (en inglés, microkernel o μ kernel) es un tipo de núcleo de un sistema operativo que provee un conjunto de primitivas o llamadas mínimas al sistema para implementar servicios básicos como espacios de direcciones, comunicación entre procesos y planificación básica. Todos los otros servicios (gestión de memoria, sistema de archivos, operaciones de E/S, etc.), que en general son provistos por el núcleo, se ejecutan como procesos servidores en espacio de usuario.



Ejemplos de micronúcleos

- AIX
- La familia de micronúcleos L4
- El micronúcleo Mach, usado en GNU Hurd y en Mac OS X
- BeOS
- Minix
- MorphOS
- QNX
- RadiOS
- VSTa
- Hurd

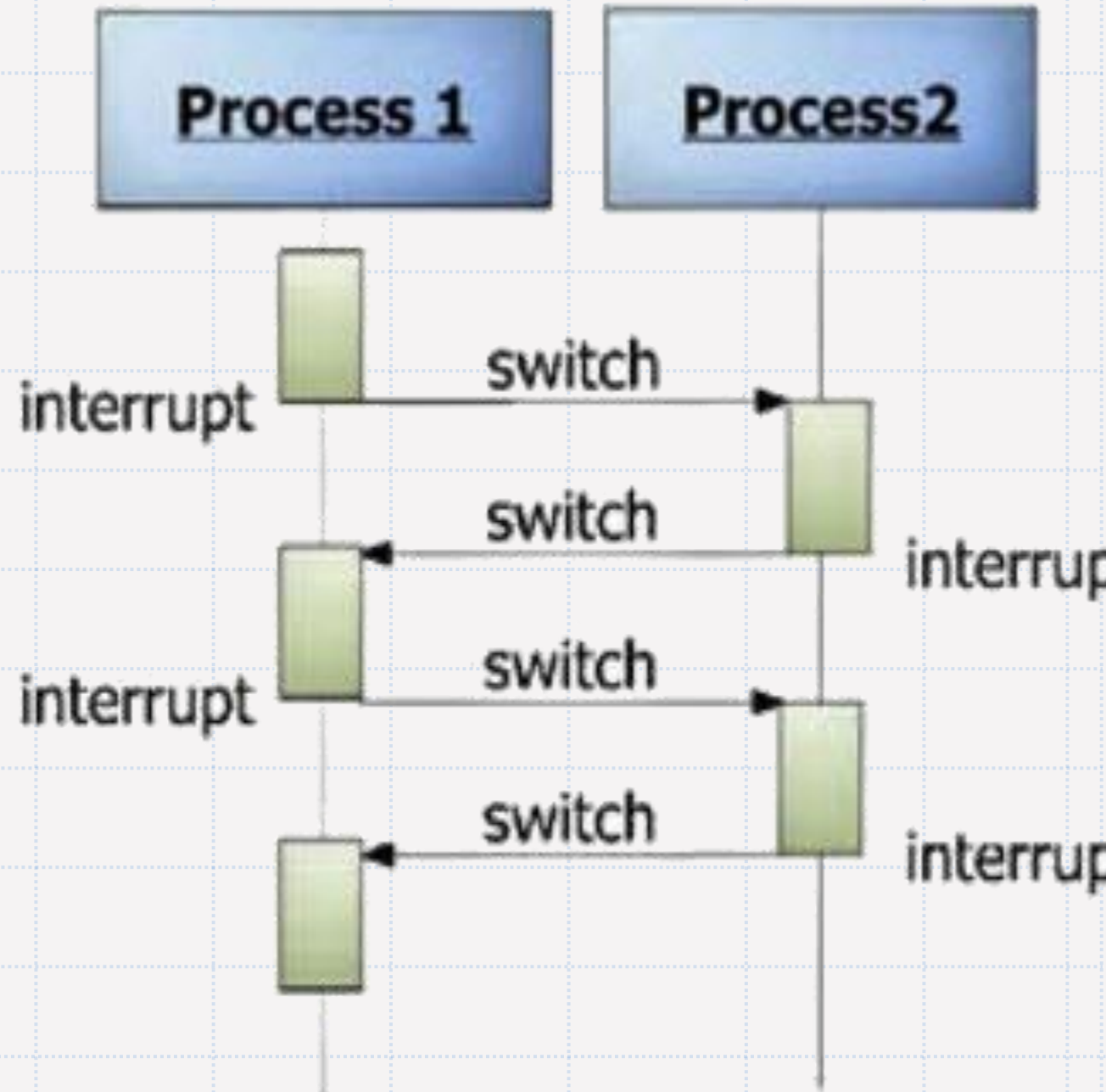


Gestion de hilos

1. Existen dos maneras de poder gestionar los hilos en un sistema operativo:
2. **Hilos de usuario (ULT):** Mediante este método una aplicación realiza todo el trabajo de la gestión de hilos mientras el kernel no es consciente de la existencia de estos.
3. **Hilos de Kernel (KLT):** Mediante este método los hilos son gestionados directamente por el kernel y en la aplicación no existe ningún código para la gestión de estos. Este método es usado en Windows 2000 y algunas versiones de Linux

Multiprocesos

Microsoft Windows permite la implementación de multitareas preventivas, para esto, Windows divide el tiempo de procesador disponible entre los diferentes procesos o hilos que lo requieran, entonces cuando un hilo se suspende debido a que su parte del tiempo se agota esto permite la ejecución de otro hilo diferente. El sistema al cambiar de un hilo a otro guarda el contexto del hilo reemplazado y restaura el contexto guardado del siguiente subproceso en la cola. Debido a que cada hilo se ejecuta en un tiempo aproximado de 20 milisegundos, el sistema da la sensación de ejecutar múltiples hilos a la vez.

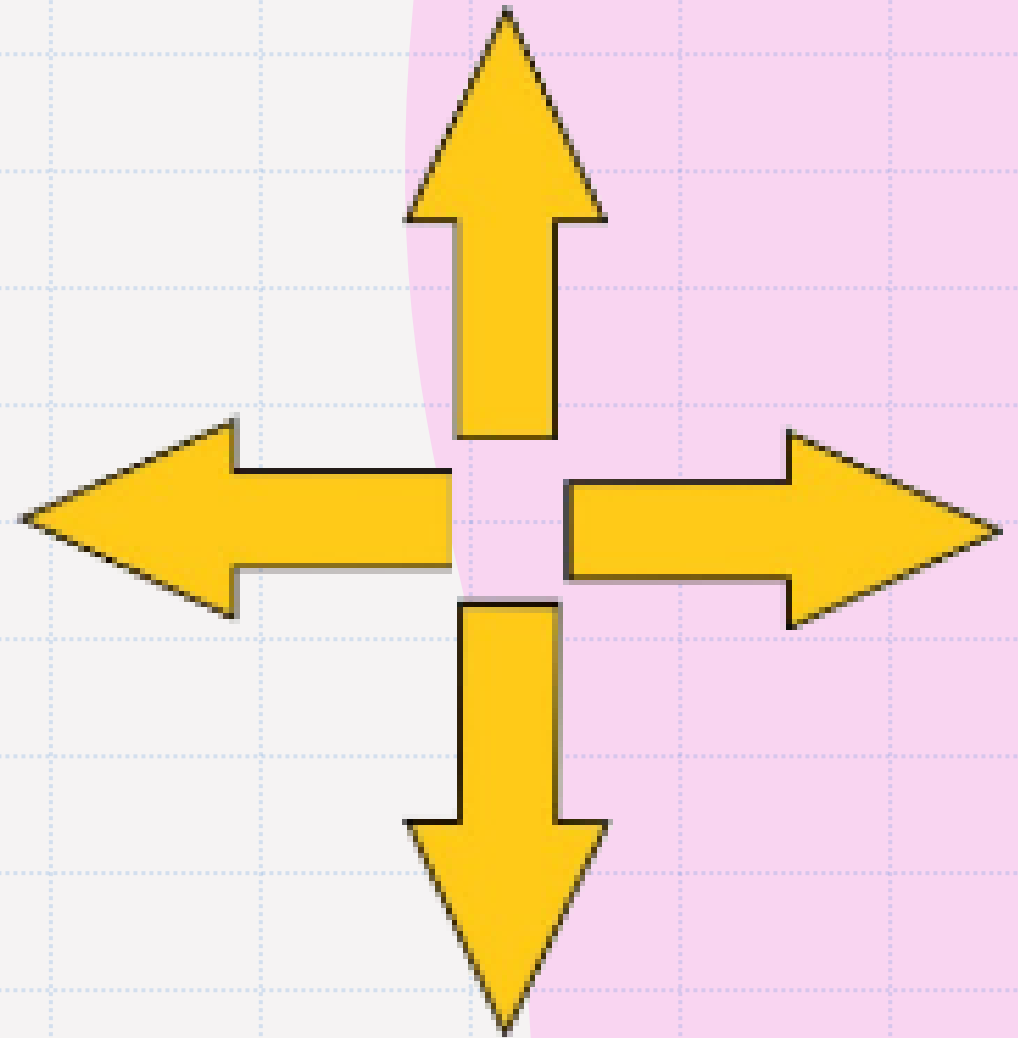


Tipos de procesos concurrentes

Los procesos que se ejecutan de manera concurrente en un sistema operativo se pueden clasificar en dos tipos:

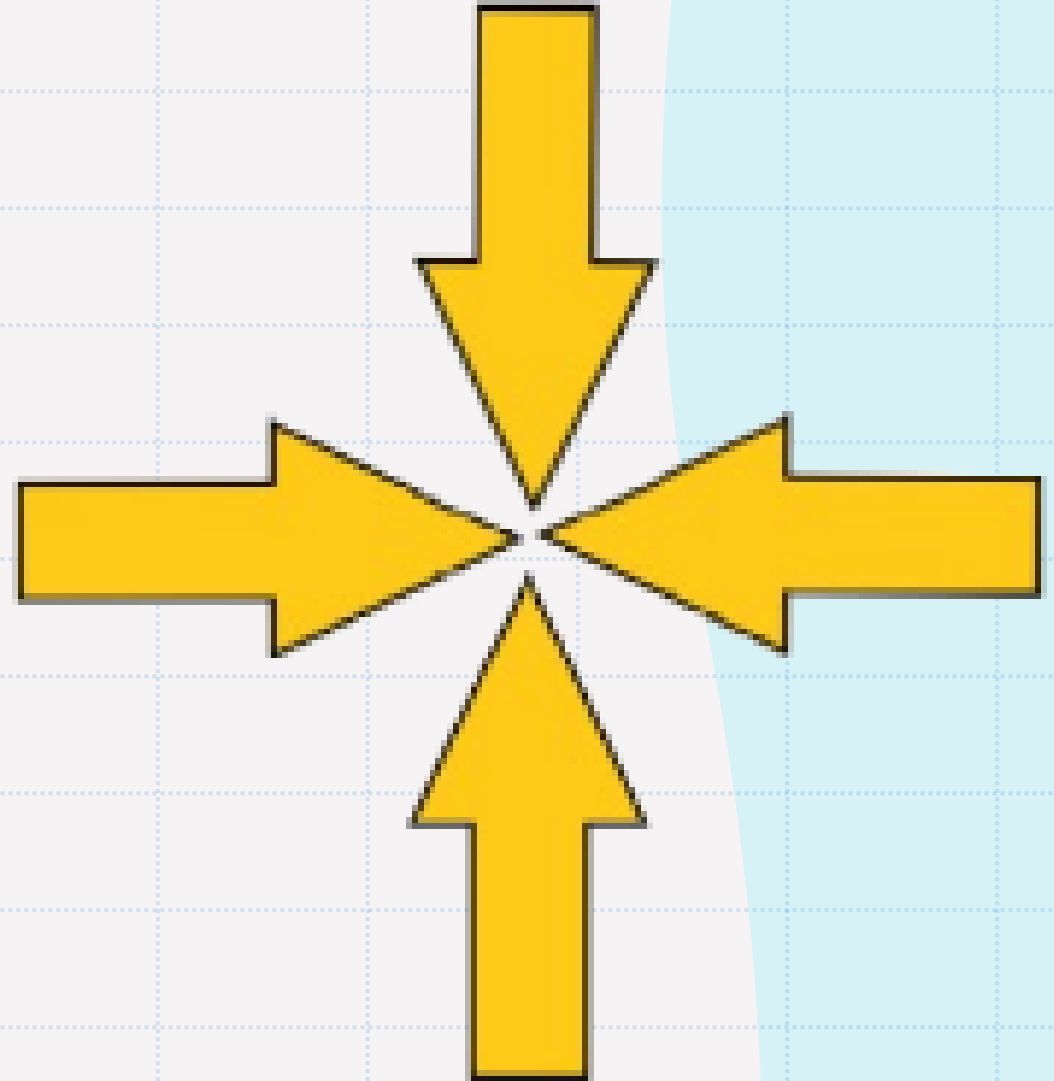
Procesos independientes

- Este tipo de procesos se ejecutan sin necesidad de recibir ayuda de otro proceso, por lo cual no puede afectar ni ser afectado por otros procesos que se ejecuten en el sistema, a su vez estos procesos compiten por el uso de los recursos. Un ejemplo de este tipo de procesos puede ser un programa que calcula 1000 cifras decimales de π .



Procesos cooperativos

Este tipo de procesos están diseñados para que trabajen de manera conjunta con otros procesos para la realización de una actividad. Estos procesos deben ser capaces de intercambiar datos entre ellos. Un ejemplo de procesos cooperantes sería un proceso que escribe en la terminal la cadena “abc” y en otro la cadena “cba”.





Tipos de interacciones dentro de la concurrencia

Primero comunicacion

Si tenemos varios procesos ejecutando y queremos que interactúen entre ellos para solucionar un determinado problema, y por tanto, necesitan intercambiar información, es decir necesitan comunicarse.

Mediante la comunicación se consigue que la ejecución de un proceso influya en la ejecución de otro.

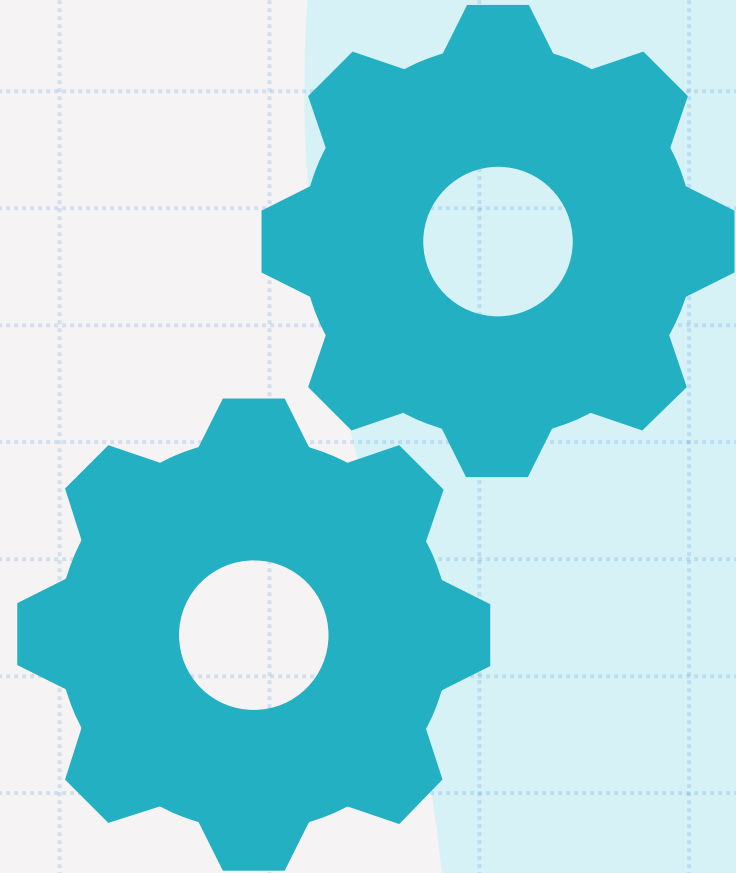




Segunda sincronización

- La sincronización (sinónimo de espera) es habitualmente necesaria cuando dos procesos se comunican puesto que la velocidad a la que se ejecutan no es predecible
- La necesidad de contar con comunicación y sincronización puede verse en términos de competencia y cooperación entre procesos.
- La competencia ocurre cuando el proceso requiere el uso exclusivo de un recurso.
- La cooperación ocurre cuando dos procesos trabajan en distintas partes del mismo problema y por lo general incluyen a la comunicación y a la sincronización.

Procesos concurrentes



PROCESOS CONCURRENTES

Un programa concurrente es aquel en el que ciertas unidades de ejecución internamente secuenciales (procesos o threads), se ejecutan paralela o simultáneamente.

PROCESO CONCURRENTE INDEPENDIENTE

es aquel que ejecuta sin requerir la ayuda o cooperación de otros procesos.

Un claro ejemplo de procesos independientes son los diferentes intérpretes de mandatos que se ejecutan de forma simultánea en un sistema

PROCESO CONCURRENTE COOPERATIVO

Idea secundaria

Idea secundaria

– Interacción motivada porque los procesos se comunican y sincronizan entre sí para alcanzar un objetivo común. Por ejemplo, los **procesos compilador y ensamblador** descritos anteriormente son dos procesos que deben comunicarse y sincronizarse entre ellos con el fin de producir código en lenguaje máquina

– **Interacción motivada** porque los procesos comparten o compiten por el acceso a recursos físicos o lógicos. Esta situación aparece en los distintos tipos de procesos anteriormente comentados. Por ejemplo, dos procesos totalmente independientes pueden competir por el acceso al disco.

Conclusión

Todos los mecanismos de control de concurrencia deben asegurar la consistencia de los objetos y cada transacción atómica será completada en un tiempo finito. Un método de control de concurrencia es correcto si es serializable, es decir existe una secuencia equivalente en que las operaciones de cada transacción aparecen antes o después de otra transacción pero no entremezcladas. Una ejecución serial de transacciones es siempre correcta.

Se debe sincronizar las transacciones concurrentes de los usuarios, extendiendo los argumentos para la serializabilidad y los algoritmos de control de concurrencia para la ejecución en ambientes distribuidos. La finalidad del control de concurrencia es asegurar la consistencia de los datos al ejecutar transacciones, y que cada acción atómica sea completada en un tiempo finito.

Conclusiones Individuales

Emmanuel Sánchez Aranda

Me sorprende la complejidad detrás de algo tan simple como abrir un navegador web o reproducir música. Aunque todavía soy un estudiante y no tengo un profundo conocimiento técnico, me emociona de seguir aprendiendo sobre estos temas y cómo puedo aplicarlos en mi futuro como desarrollador de software o profesional de la tecnología. También puedo ver cómo estos conceptos de procesos, hilos, concurrencia y transiciones son fundamentales para entender cómo funcionan los sistemas operativos y las aplicaciones en mi computadora.

Ángel David Gómez González

Gracias a este trabajo que realizamos me pude dar cuenta de lo cuan importante es la concurrencia en las páginas web, ya que estas nos ayudan a poder partir las páginas en "partes" y así poder ir las haciendo a diferentes pasos, también con esto que pudimos ver e investigar, podemos ver detrás de las páginas web que tanto se hace, ya sea los códigos, algoritmos y los diferentes tipos de lenguaje que se llevan acabo para realizar dicha página

Marco Antonio Arreola de Leon

Durante esta actividad aprendi con claridad como es que ocurre los procesos internos de una computadora asi como su funcionamiento. Me parece muy interesante e increible que todo esto se haya podido desarrollar en menos de un siglo y que haya avanzado tan rapido, sin dudas creo que todos estos conocimientos me seran sumamente

- **Jonathan Francisco Galvan Villanueva**

- Al investigar sobre el tema, y el subtema que me toco, me doy cuenta de los pequeños pero importantes procesos que el procesador hace dentro de su o sus nucleos, sobre la funcion e importancia de los hilos y el como estos ayudan a organizar y ejecutar los procesos, asi mismo en mi caso pude aprender lo que son los procesos concurrentes y sus dos tipos de procesos.

- **Jorge Alberto Morales Reyes**

- Cuando investigue para poder presentar este tema me sorprendir a pensar que en realidad se muy poco de sistemas operativos y que hacen internamente aunque yo uso una computadora todos los dias. Tambien aprecio mas a todos los programadores y ingenieeros que trabajan en desarrollar sistemas operativos que es mas complejo que ser desarrollador de paginas web.

Enlace al video

https://uanledu.sharepoint.com/sites/presentacionSO/Shared%20Documents/General/Recordings/Nueva%20reuni%C3%B3n%20de%20canal-20240215_232949-Grabaci%C3%B3n%20de%20la%20reuni%C3%B3n.mp4

Referencias bibliograficas

H. (2022, 12 mayo). *Animaciones y transiciones - Win32 apps*. Microsoft Docs. Recuperado 26 de agosto de 2022, de <https://docs.microsoft.com/es-es/windows/win32/uxguide/vis-animations> Mario Alejandro Moreno Coronel

WEBEDIA BRAND SERVICES. (2014). El administrador de tareas de Windows: qué es y cómo funciona. Julio 14, 2014, de xatakawindows Sitio web: <https://www.xatakawindows.com/bienvenidoawindows8/el-administrador-de-tareas-de-windows-que-es-y-como-funciona>

Manuel M., J. (2019, 30 septiembre). *Diferencias entre Multiprocesamiento Simétrico y Asimétrico*. PC Solución. Recuperado 26 de agosto de 2022, de <https://pc-solucion.es/2018/04/18/diferencias-entre-multiprocesamiento-simetrico-y-asimetrico/>

Kare G. 11/05/2021 España
https://www.google.com/url?sa=t&source=web&rct=j&url=https://docs.microsoft.com/es-es/windows/win32/power/system-power-states&ved=2ahUKEwiMIJ_2uOX5AhWekWoFHez-CPcQFnoECBUQAQ&usg=AOvVaw2USNdulk-Qp87pR5tIVAMR