

**Licenciatura en Sistemas de Información**

**Trabajo Práctico N° 4**

**Procesos e Hilos**

**Cátedra: Sistemas Operativos**

**Alumnos: Rondán Nicolás, Ruiz Mazzocato Nicolás**

**Universidad Autónoma de Entre Ríos**

**Facultad de Ciencia y Tecnología**

**Año: 2019**

## Procesos e Hilos

### 1) ¿A qué se le llama traza de un proceso? ¿y traza combinada?

La traza de un proceso es la ruta de ejecución del mismo a través de uno o más programas. La traza combinada se refiere el cambio de procesos en ejecución en una traza en particular.

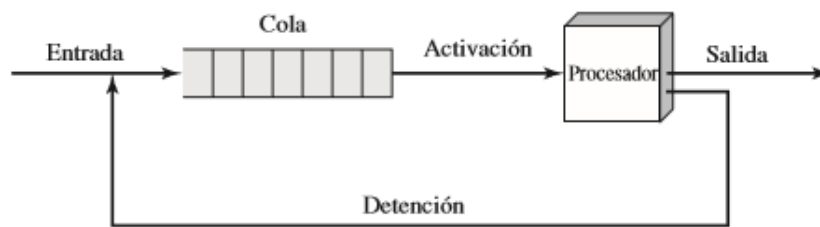
### 2)

0100	P1
0101	
0103	
0105	
7231	P2
7234	
7236	
7238	
8021	P3
8022	
8023	
8024	
0109	
010B	
010D	
723A	
723C	
8025	
E/S	
E/S	
723D	
723F	

3) Dibuje y explique un diagrama de procesos de dos estados. Dibuje y explique cómo queda conformado un diagrama de colas para ejecutar procesos con dos estados.



(a) Diagrama de transiciones de estados



(b) Modelos de colas

#### Diagrama de procesos de dos estados

En este modelo, un proceso puede estar en dos estados: Ejecutando o No Ejecutando. Cuando el sistema operativo crea un nuevo proceso, crea el bloque de control de proceso (BCP) para el nuevo proceso e inserta dicho proceso en el sistema en estado No Ejecutando. El proceso existe, es conocido por el sistema operativo, y está esperando su oportunidad de ejecutar. De cuando en cuando, el proceso actualmente en ejecución se interrumpirá y una parte del sistema operativo, el activador, seleccionará otro proceso a ejecutar. El proceso saliente pasará del estado Ejecutando a No Ejecutando y pasará a Ejecutando un nuevo proceso.

#### Diagrama de colas

Los procesos que no están ejecutando deben estar en una especie de cola, esperando su turno de ejecución. Existe una sola cola cuyas entradas son punteros al BCP de un proceso en particular. Alternativamente, la cola debe consistir en una lista enlazada de bloques de datos, en la cual cada bloque representa un proceso. Podemos describir el comportamiento del activador en términos de este diagrama de colas. Un proceso que se interrumpe se transfiere a la cola de procesos en espera. Alternativamente, si el proceso ha finalizado o ha sido abortado, se descarta (sale del sistema). En cualquier caso, el activador selecciona un proceso de la cola para ejecutar.

#### 4) ¿Cuáles son los eventos que producen la creación de procesos? De un ejemplo de cada uno.

Existen cuatro eventos comunes que llevan a la creación de un proceso:

Nuevo proceso de lotes	El sistema operativo dispone de un flujo de control de lotes de trabajos, habitualmente una cinta un disco. Cuando el sistema operativo está listo para procesar un nuevo trabajo, leerá la siguiente secuencia de mandatos de control de trabajos.
Sesión interactiva	Un usuario desde un terminal entra en el sistema.
Creado por el sistema operativo para proporcionar un servicio	El sistema operativo puede crear un proceso para realizar una función en representación de un programa de usuario, sin que el usuario tenga que esperar (por ejemplo, un proceso para controlar la impresión).
Creado por un proceso existente	Por motivos de modularidad o para explotar el paralelismo, un programa de usuario puede ordenar la creación de un número de procesos.

#### 5) ¿Cómo se llama al hecho de que un proceso sea creado por otro? ¿Qué nombre recibe el proceso creador y el proceso creado?

Cuando un sistema operativo crea un proceso a petición explícita de otro proceso, dicha acción se denomina creación del proceso. Cuando un proceso lanza otro, al primero se le denomina proceso padre, y al proceso creado se le denomina proceso hijo. Habitualmente, la relación entre procesos necesita comunicación y cooperación entre ellos.

#### 6) De ejemplos de terminación de procesos.

- Finalización normal: El proceso ejecuta una llamada al sistema operativo para indicar que ha completado su ejecución
- Límite de tiempo excedido: El proceso ha ejecutado más tiempo del especificado en un límite máximo. Existen varias posibilidades para medir dicho tiempo. Estas incluyen el tiempo total utilizado, el tiempo utilizado únicamente en ejecución, y, en el caso de procesos interactivos, la cantidad de tiempo desde que el usuario realizó la última entrada.
- Memoria no disponible: El proceso requiere más memoria de la que el sistema puede proporcionar.
- Violaciones de frontera: El proceso trata de acceder a una posición de memoria a la cual no tiene acceso permitido.
- Error de protección: El proceso trata de usar un recurso, por ejemplo, un fichero, al que no tiene permitido acceder, o trata de utilizarlo de una forma no apropiada, por ejemplo, escribiendo en un fichero de sólo lectura.
- Error aritmético: El proceso trata de realizar una operación de cálculo no permitida, tal como una división por 0, o trata de almacenar números mayores de los que la representación hardware puede codificar.
- Límite de tiempo: El proceso ha esperado más tiempo que el especificado en un valor máximo para que se cumpla un determinado evento.
- Fallo de E/S: Se ha producido un error durante una operación de entrada o salida, por ejemplo, la imposibilidad de encontrar un fichero, fallo en la lectura o escritura después de un límite

máximo de intentos (cuando, por ejemplo, se encuentra un área defectuosa en una cinta), o una operación inválida (la lectura de una impresora en línea).

- Instrucción no válida: El proceso intenta ejecutar una instrucción inexistente (habitualmente el resultado de un salto a un área de datos y el intento de ejecutar dichos datos).
- Instrucción privilegiada: El proceso intenta utilizar una instrucción reservada al sistema operativo.
- Uso inapropiado de datos: Una porción de datos es de tipo erróneo o no se encuentra inicializada.
- Intervención del operador: Por alguna razón, el operador o el sistema operativo ha finalizado el por el sistema operativo proceso (por ejemplo, se ha dado una condición de interbloqueo).
- Terminación del proceso: Cuando un proceso padre termina, el sistema operativo puede automáticamente finalizar todos los procesos hijos descendientes de dicho padre.
- Solicitud del proceso padre: Un proceso padre habitualmente tiene autoridad para finalizar sus propios procesos descendientes.

**7) Al pasar de un modelo de dos estados a uno de 5 estados uno de los estados se divide en dos, Listo y Bloqueado. ¿Cuál es ese estado? ¿Cuál es la razón por la cual se divide este estado en dos?**

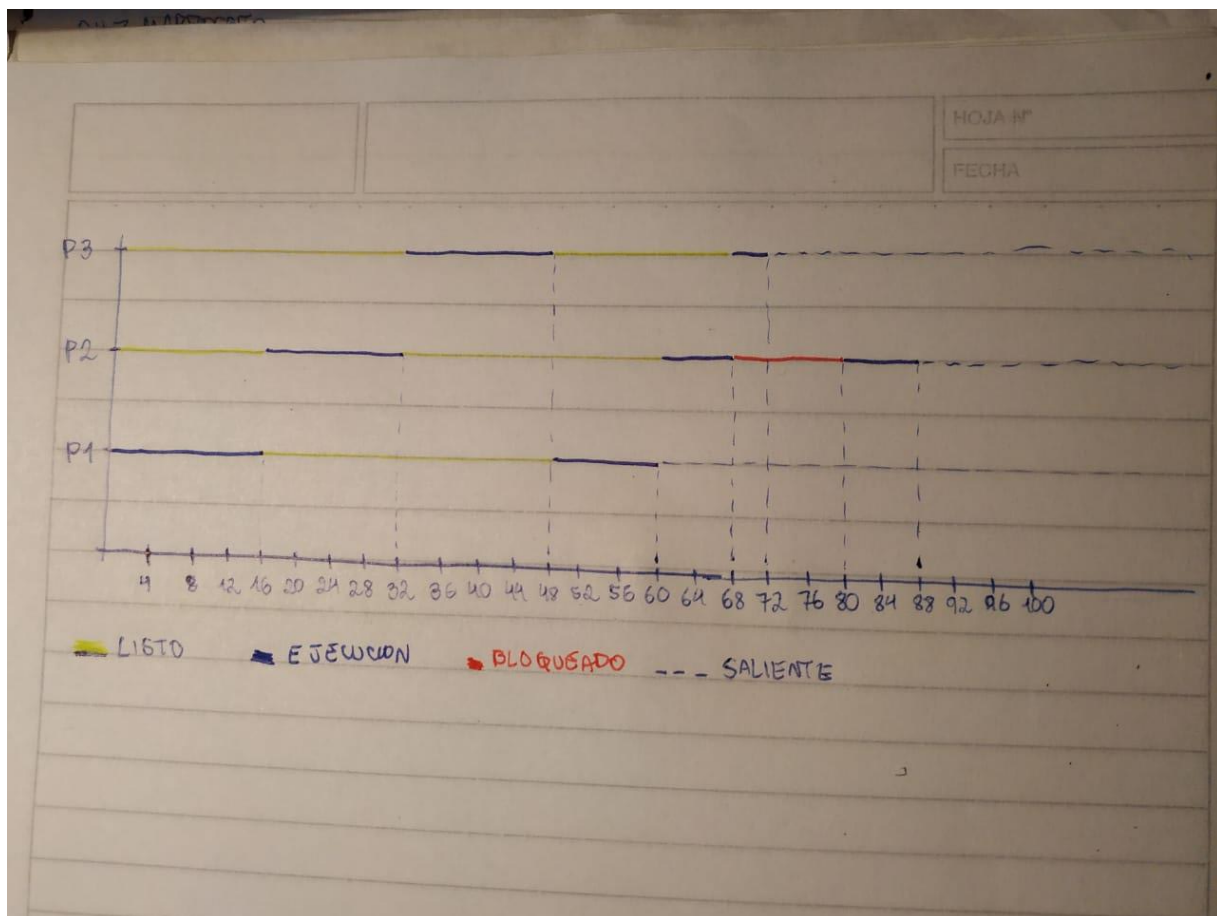
Se refiere al estado No Ejecutado. Algunos procesos que están en el estado de No Ejecutando están listos para ejecutar, mientras que otros están bloqueados, esperando a que se complete una operación de E/S. Por tanto, utilizando una única cola (para los No Ejecutados), el activador no puede seleccionar únicamente los procesos que lleven más tiempo en la cola. En su lugar, debería recorrer la lista buscando los procesos que no estén bloqueados y que lleven en la cola más tiempo. Una forma más natural para manejar esta situación es dividir el estado de No Ejecutando en dos estados, Listo y Bloqueado.

**8) Dibuje un modelo de procesos de 5 estados. Explique cada estado. ¿Cuáles son los eventos por los cuales se pasa de un estado a otro?**



- Ejecutando: El proceso está actualmente en ejecución. Para este capítulo asumimos que el computador tiene un único procesador, de forma que sólo un proceso puede estar en este estado en un instante determinado.
- Listo: Un proceso que se prepara para ejecutar cuando tenga oportunidad.
- Bloqueado: Un proceso que no puede ejecutar hasta que se cumpla un evento determinado o se complete una operación E/S.
- Nuevo: Un proceso que se acaba de crear y que aún no ha sido admitido en el grupo de procesos ejecutables por el sistema operativo. Típicamente, se trata de un nuevo proceso que no ha sido cargado en memoria principal, aunque su bloque de control de proceso (BCP) si ha sido creado.
- Saliente: Un proceso que ha sido liberado del grupo de procesos ejecutables por el sistema operativo, debido a que ha sido detenido o que ha sido abortado por alguna razón.

9) Suponga que los procesos pueden estar solo en los estados Listo, ejecución y Bloqueado. Dibuje un diagrama de estado de proceso para la traza del Ej. 2) donde el eje de las X son los tiempos de ejecución de una instrucción y el eje Y son los procesos P1, P2 y P3.



**10) Explique la razón de incorporar un estado suspendido a un modelo de estados de procesos.**

Una de las razones de incorporar un estado suspendido es la necesidad de intercambio (swapping), ya que el sistema operativo necesita liberar suficiente memoria principal para cargar un proceso que está listo para ejecutarse.

Cuando todos los procesos en memoria principal se encuentran en estado Bloqueado, el sistema operativo puede suspender un proceso poniéndolo en el estado Suspendido y transfiriéndolo a disco. El espacio que se libera en memoria principal puede usarse para traer a otro proceso.

Existen otras razones, como, por ejemplo:

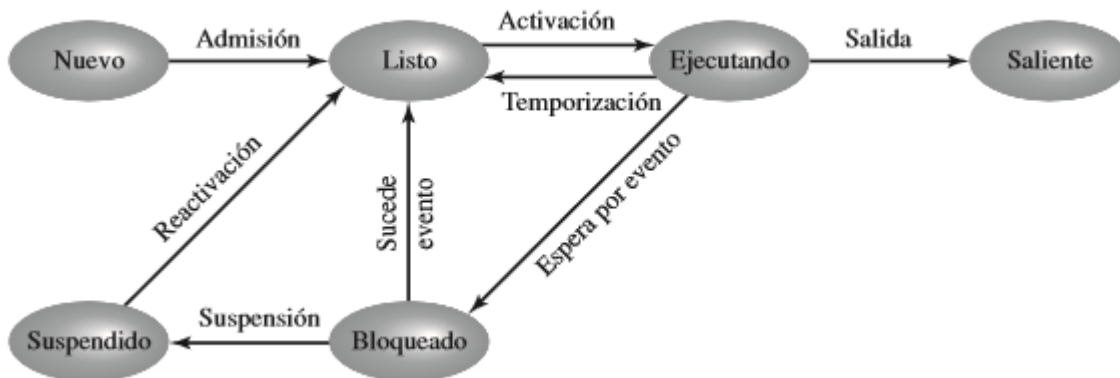
Otras razones del sistema operativo: El sistema operativo puede suspender un proceso en segundo plano o de utilidad o un proceso que se sospecha puede causar algún problema.

Solicitud interactiva del usuario: Un usuario puede desear suspender la ejecución de un programa con motivo de su depuración o porque está utilizando un recurso.

Temporización: Un proceso puede ejecutarse periódicamente (por ejemplo, un proceso monitor de estadísticas sobre el sistema) y puede suspenderse mientras espera el siguiente intervalo de ejecución.

Solicitud del proceso padre: Un proceso padre puede querer suspender la ejecución de un descendiente para examinar o modificar dicho proceso suspendido, o para coordinar la actividad de varios procesos descendientes.

**11) ¿Cuáles son los eventos que pueden suceder para que un proceso en estado bloqueado pase a estado de ejecución en un modelo de estado con un solo estado de suspensión?**



Según indica el gráfico, para pasar del estado Bloqueado al estado Ejecutando puede ocurrir:

- Que suceda el evento que estaba esperando, pase a Listo y se active.
- Que pase al estado Suspendido, sea reactivado y pase a Listo, y finalmente se active.

## 12) ¿Cuáles son las estructuras de control más importantes del SO? ¿Para qué se utilizan cada una de ellas?

Las estructuras de control más importantes del SO son las tablas de memoria, E/S, ficheros y procesos.

- Las **tablas de memoria** se usan para mantener un registro tanto de la memoria principal (real) como de la secundaria (virtual).
- las **tablas de E/S** gestionan los dispositivos de E/S y los canales del computador.
- las **tablas de ficheros** proporcionan información sobre la existencia de ficheros, su posición en almacenamiento secundario, su estado actual y otros atributos.

Por último, las **tablas de procesos** se utilizan para gestionar procesos.

## 13) ¿Cómo está formada la “información del estado del proceso”?

La información de estado de proceso indica los contenidos de los registros del procesador. Cuando un proceso está ejecutando, esta información está, por supuesto, en los registros. Cuando un proceso se interrumpe, toda la información de los registros debe salvaguardarse de forma que se pueda restaurar cuando el proceso continúe con su ejecución. La naturaleza y el número de estos registros depende del diseño del procesador. Normalmente, el conjunto de registros incluye registros visibles por usuario, registros de control y de estado, y punteros de pila.

## 14) ¿Que realiza el SO cuando crea un proceso?

Una vez que el sistema operativo decide, por cualquier motivo, crear un proceso procederá de la siguiente manera:

1. **Asignar un identificador de proceso único al proceso.** En este instante, se añade una nueva entrada a la tabla primaria de procesos, que contiene una entrada por proceso.
2. **Reservar espacio para proceso.** Esto incluye todos los elementos de la imagen del proceso. Para ello, el sistema operativo debe conocer cuánta memoria se requiere para el espacio de direcciones privado (programas y datos) y para la pila de usuario. Estos valores se pueden asignar por defecto basándonos en el tipo de proceso, o pueden fijarse en base a la solicitud de creación del trabajo remitido por el usuario. Si un proceso es creado por otro proceso, el proceso padre puede pasar los parámetros requeridos por el sistema operativo como parte de la solicitud de la creación de proceso. Si existe una parte del espacio direcciones compartido por este nuevo proceso, se fijan los enlaces apropiados. Por último, se debe reservar el espacio para el bloque de control de proceso (BCP).
3. **Inicialización del bloque de control de proceso.** La parte de identificación de proceso del BCP contiene el identificador del proceso, así como otros posibles identificadores, tal como el indicador del proceso padre. En la información de estado de proceso del BCP, habitualmente se inicializa con la mayoría de entradas a 0, excepto el contador de programa (fijado en el punto entrada del programa) y los punteros de pila de sistema (fijados para definir los límites de la pila del proceso). La parte de información de control de procesos se inicializa en base a los valores por omisión, considerando también los atributos que han sido solicitados para este proceso. Por ejemplo, el estado del proceso se puede inicializar normalmente a Listo o Listo/Suspendido. La prioridad se puede fijar, por defecto, a la prioridad más baja, a menos que una solicitud explícita la eleve a una prioridad mayor.



Inicialmente, el proceso no debe poseer ningún recurso (dispositivos de E/S, ficheros) a menos que exista una indicación explícita de ello o que haya sido heredados del padre.

4. **Establecer los enlaces apropiados.** Por ejemplo, si el sistema operativo mantiene cada cola del planificador como una lista enlazada, el nuevo proceso debe situarse en la cola de Listos o en la cola de Listos/Suspendidos.

5. **Creación o expansión de otras estructuras de datos.** Por ejemplo, el sistema operativo puede mantener un registro de auditoría por cada proceso que se puede utilizar posteriormente a efectos de facturación y/o de análisis de rendimiento del sistema.

#### 15) Explique detalladamente que es el intercambio de procesos.

El swapping (memoria de intercambio) implica mover parte o todo el proceso de memoria principal al disco. Cuando ninguno de los procesos en memoria principal se encuentra en estado Listo, el sistema operativo intercambia uno de los procesos bloqueados a disco, en la cola de Suspendidos. Esta es una lista de procesos existentes que han sido temporalmente expulsados de la memoria principal, o suspendidos. El sistema operativo trae otro proceso de la cola de Suspendidos o responde a una solicitud de un nuevo proceso. La ejecución continúa con los nuevos procesos que han llegado.

#### 16) ¿Cuándo se puede producir un intercambio de procesos?

Un intercambio de procesos puede ocurrir en cualquier instante en el que el sistema operativo obtiene el control sobre el proceso actualmente en ejecución. El siguiente cuadro indica los posibles momentos en los que el sistema operativo puede tomar dicho control:

Mecanismo	Causa	Uso
Interrupción	Externa a la ejecución del proceso actualmente en ejecución.	Reacción ante un evento externo asíncrono.
Trap	Asociada a la ejecución de la instrucción actual.	Manejo de una condición de error o de excepción.
Llamada al sistema	Solicitud explícita.	Llamada a una función del sistema operativo.

#### 17)18) ¿Cuál es la diferencia entre un proceso y un hilo? ¿Qué son “unidad propietaria de recursos” y “unidad de ejecución”?

Hasta ahora el concepto de un proceso abarcaba dos características:

- **Propiedad de recursos:** Un proceso incluye un espacio de direcciones virtuales para el manejo de la imagen del proceso (colección de programa, datos, pila y atributos definidos en el bloque de control del proceso). De vez en cuando a un proceso se le puede asignar control o propiedad de recursos tales como la memoria principal, canales E/S, dispositivos E/S y archivos. El sistema operativo realiza la función de protección para evitar interferencias no deseadas entre procesos en relación con los recursos.
- **Planificación/ejecución:** La ejecución de un proceso sigue una ruta de ejecución (traza) a través de uno o más programas. Esta ejecución puede estar intercalada con ese u otros procesos. De esta manera, un proceso tiene un estado de ejecución (Ejecutando, Listo, etc.) y una prioridad de activación y ésta es la entidad que se planifica y activa por el sistema operativo.

En la mayor parte de los sistemas operativos tradicionales, estas dos características son, realmente, la esencia de un proceso. Sin embargo, estas dos características son independientes y podrían ser tratadas como tales por el sistema operativo. Así se hace en diversos sistemas operativos, sobre todo en los desarrollados recientemente. Para distinguir estas dos características, la unidad que se activa (**unidad de ejecución**) se suele denominar **hilo** (thread), o proceso ligero, mientras que la **unidad de propiedad de recursos** se suele denominar **proceso** o tarea.

### 19) ¿Que significa multihilo? ¿Cuáles son las diferencias entre modelo multihilo y monohilo?

Multihilo se refiere a la capacidad de un sistema operativo de dar soporte a múltiples hilos de ejecución en un solo proceso. El enfoque tradicional de un solo hilo de ejecución por proceso, en el que no se identifica con el concepto de hilo, se conoce como estrategia monohilo.

En un modelo de proceso monohilo (es decir, no existe el concepto de hilo), la representación de un proceso incluye su bloque de control de proceso y el espacio de direcciones de usuario, además de las pilas de usuario y núcleo para gestionar el comportamiento de las llamadas/retornos en la ejecución de los procesos. Mientras el proceso está ejecutando, los registros del procesador se controlan por ese proceso y, cuando el proceso no se está ejecutando, se almacena el contenido de estos registros. En un entorno multihilo, sigue habiendo un único bloque de control del proceso y un espacio de direcciones de usuario asociado al proceso, pero ahora hay varias pilas separadas para cada hilo, así como un bloque de control para cada hilo que contiene los valores de los registros, la prioridad, y otra información relativa al estado del hilo. De esta forma, todos los hilos de un proceso comparten el estado y los recursos de ese proceso, residen en el mismo espacio de direcciones y tienen acceso a los mismos datos. Cuando un hilo cambia determinados datos en memoria, otros hilos ven los resultados cuando acceden a estos datos. Si un hilo abre un archivo con permisos de lectura, los demás hilos del mismo proceso pueden también leer ese archivo.

### 20) ¿Un hilo puede estar en estado suspendido? Explique.

No. Porque no tiene sentido aplicarlo, ya que dicho estado es un concepto de nivel de proceso. En particular, si se expulsa un proceso, todos sus hilos se deben expulsar porque comparten el espacio de direcciones del proceso.

### 21) ¿Qué tipos de hilos existen? Detalle cada uno (a nivel de núcleo y a nivel de usuario).

Existen dos amplias categorías de implementación de hilos: hilos de nivel de usuario (user-level threads, ULT) e hilos de nivel de núcleo (kernel-level threads, KLT). Los últimos son también conocidos en la literatura como hilos soportados por el núcleo (kernel-supported threads) o procesos ligeros (lightweight processes).

#### Hilos de nivel de usuario

En un entorno ULT puro, la aplicación gestiona todo el trabajo de los hilos y el núcleo no es consciente de la existencia de los mismos. Cualquier aplicación puede programarse para ser multihilo a través del uso de una biblioteca de hilos, que es un paquete de rutinas para la gestión de ULT. La biblioteca de hilos contiene código para la creación y destrucción de hilos, para paso de mensajes y datos entre los hilos, para planificar la ejecución de los hilos, y para guardar y restaurar el contexto de los hilos. Por defecto, una aplicación comienza con un solo hilo y ejecutando en ese hilo. Esta aplicación y su hilo se

localizan en un solo proceso gestionado por el núcleo. En cualquier momento que la aplicación esté ejecutando (el proceso está en estado Ejecutando), la aplicación puede crear un nuevo hilo a ejecutar dentro del mismo proceso. La creación se realiza llamando a la utilidad de creación en la biblioteca de hilos. Se pasa el control a esta utilidad a través de una llamada a procedimiento. La biblioteca de hilos crea una estructura de datos para el nuevo hilo y luego pasa el control a uno de los hilos de ese proceso que esté en estado listo, utilizando algún algoritmo de planificación. Cuando se pasa el control a la biblioteca, se almacena el contexto del hilo actual, y cuando se pasa el control de la biblioteca al hilo, se recupera el contexto de ese hilo. El contexto tiene esencialmente el contenido de los registros del usuario, el contador que programa, y los punteros de pila. Toda la actividad descrita en el párrafo anterior tiene lugar en el espacio de usuario y dentro de un solo proceso. El núcleo no es consciente de esta actividad. El núcleo continúa planificando el proceso como una unidad y asigna al proceso un único estado (Listo, Ejecutando, Bloqueado, etc.).

### Hilos a nivel de núcleo

En un entorno KLT puro, el núcleo gestiona todo el trabajo de gestión de hilos. No hay código de gestión de hilos en la aplicación, solamente una interfaz de programación de aplicación (API) para acceder a las utilidades de hilos del núcleo. Windows es un ejemplo de este enfoque.

Cualquier aplicación puede programarse para ser multihilo. Todos los hilos de una aplicación se mantienen en un solo proceso. El núcleo mantiene información de contexto del proceso como una entidad y de los hilos individuales del proceso. La planificación realizada por el núcleo se realiza a nivel de hilo. Este enfoque resuelve los dos principales inconvenientes del enfoque ULT. Primero, el núcleo puede planificar simultáneamente múltiples hilos de un solo proceso en múltiples procesadores. Segundo, si se bloquea un hilo de un proceso, el núcleo puede planificar otro hilo del mismo proceso. Otra ventaja del enfoque KLT es que las rutinas del núcleo pueden ser en sí mismas multihilo.

La principal desventaja del enfoque KLT en comparación con el enfoque ULT es que la transferencia de control de un hilo a otro del mismo proceso requiere un cambio de modo al núcleo.