

THREADS

- Uma unidade básica de utilização de CPU;
 - Mais “leve” que um processo;
- Procedimento executado dentro de um processo de forma independente;
- Encontra-se dentro de um processo, utilizando seus recursos;
 - Um processo pode ter um ou vários threads!



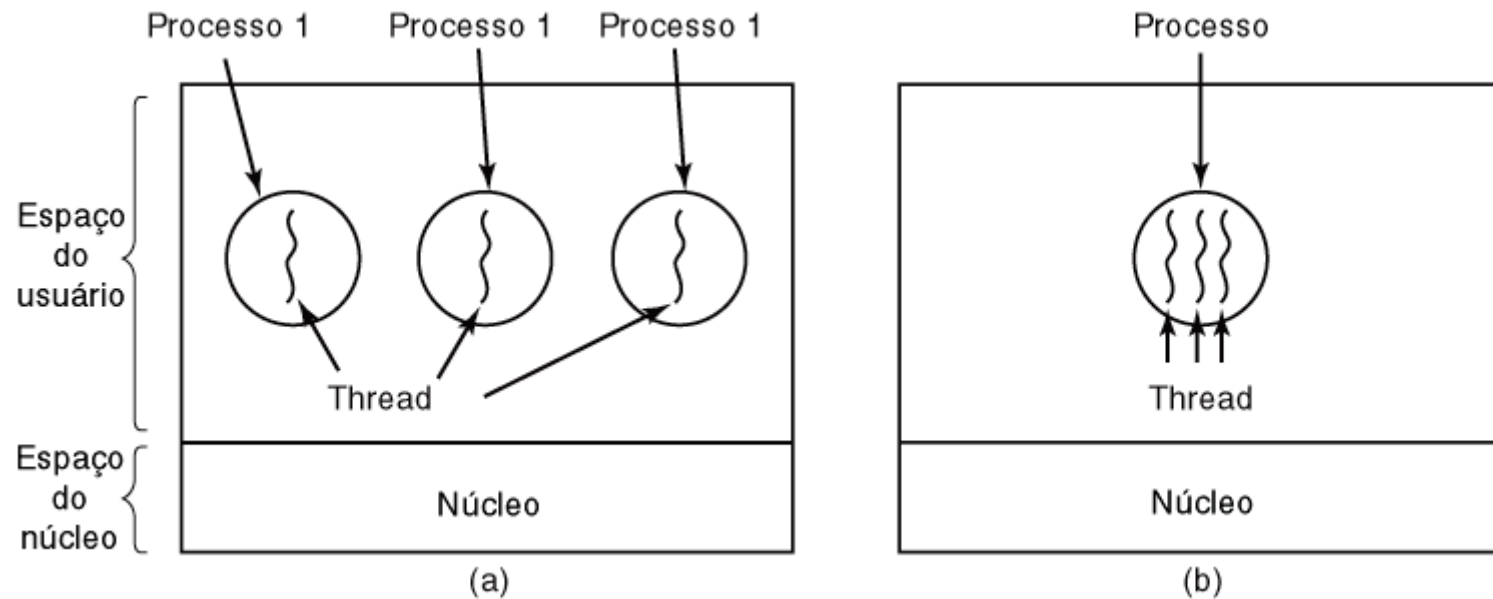
THREADS

- Podem ser usadas para implementar concorrência dentro de um processo:
 - Ex: Processo com requisições de E/S (um servidor de arquivos ou um sistema de bancos de dados):
 - Quando um thread solicita leitura, fica bloqueada até que os dados fiquem disponíveis;
 - Um processo com um único thread não pode executar nada de útil enquanto o disco está sendo acessado;
 - Com threads múltiplos, um thread pode continuar a executar enquanto o outro espera pelo E/S.



THREADS

O MODELO DE THREAD



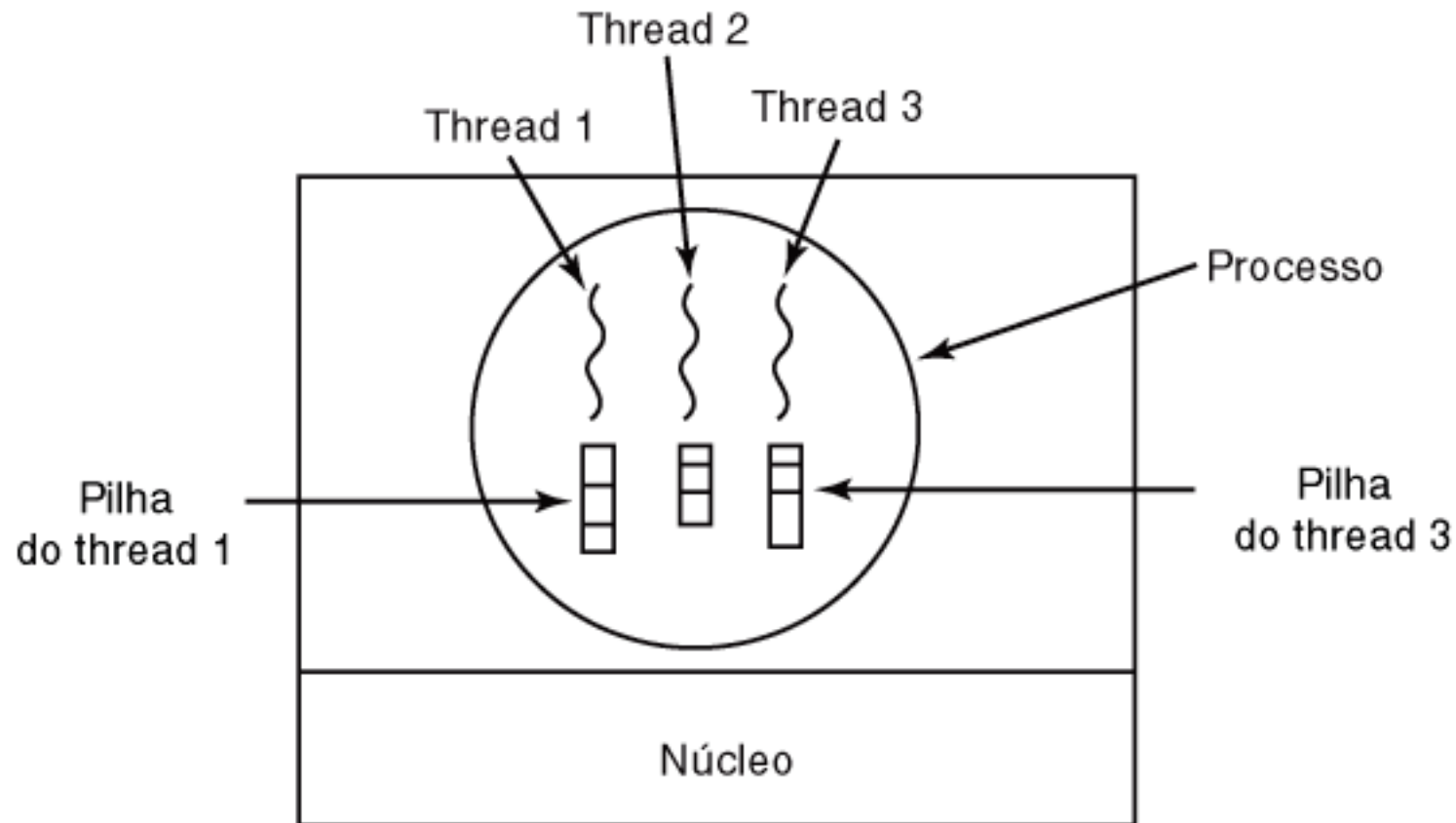
- (a) Três processos cada um com um thread;
- (b) Um processo com três threads.

O MODELO DE THREAD

Itens por processo	Itens por thread
Espaço de endereçamento Variáveis globais Arquivos abertos Processos filhos Alarmes pendentes Sinais e tratadores de sinais Informação de contabilidade	Contador de programa Registradores Pilha Estado



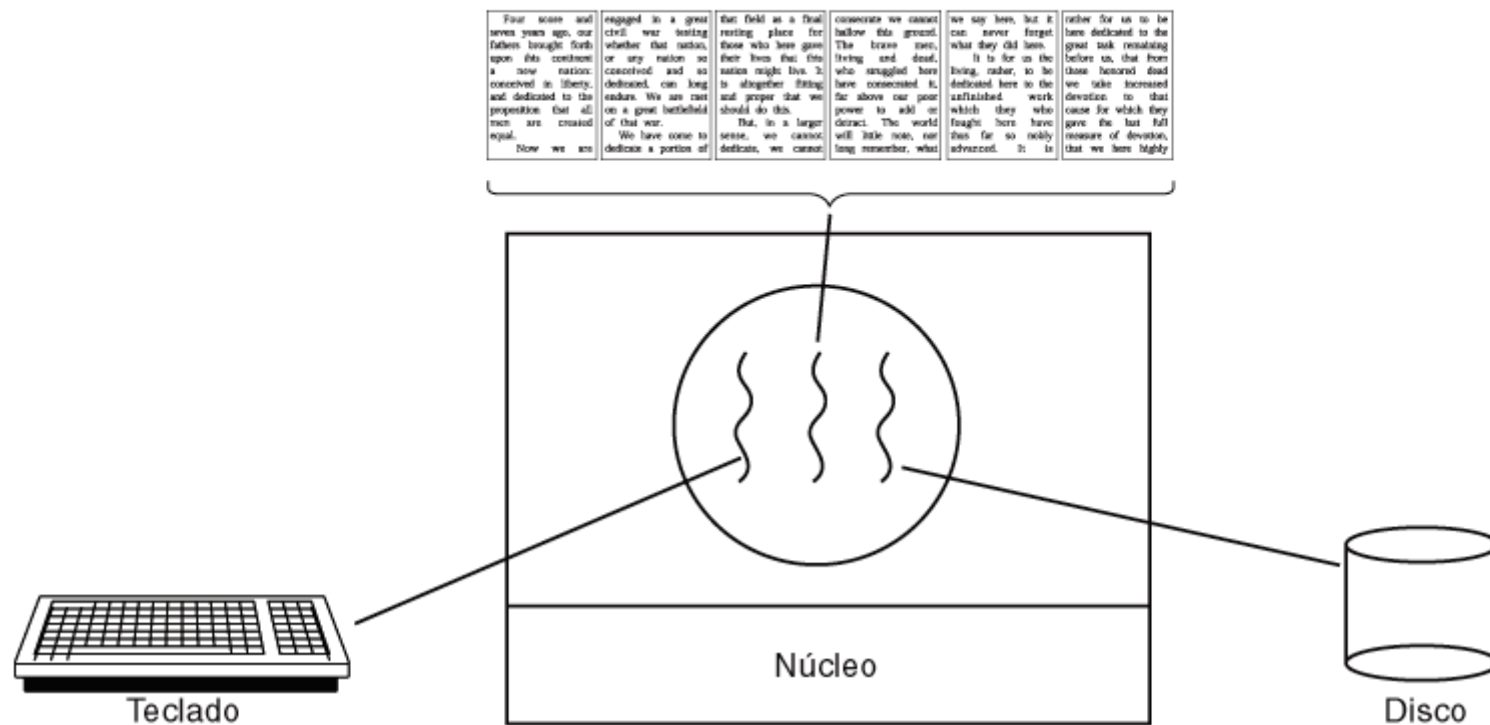
O MODELO DE THREAD



Cada thread tem sua própria pilha



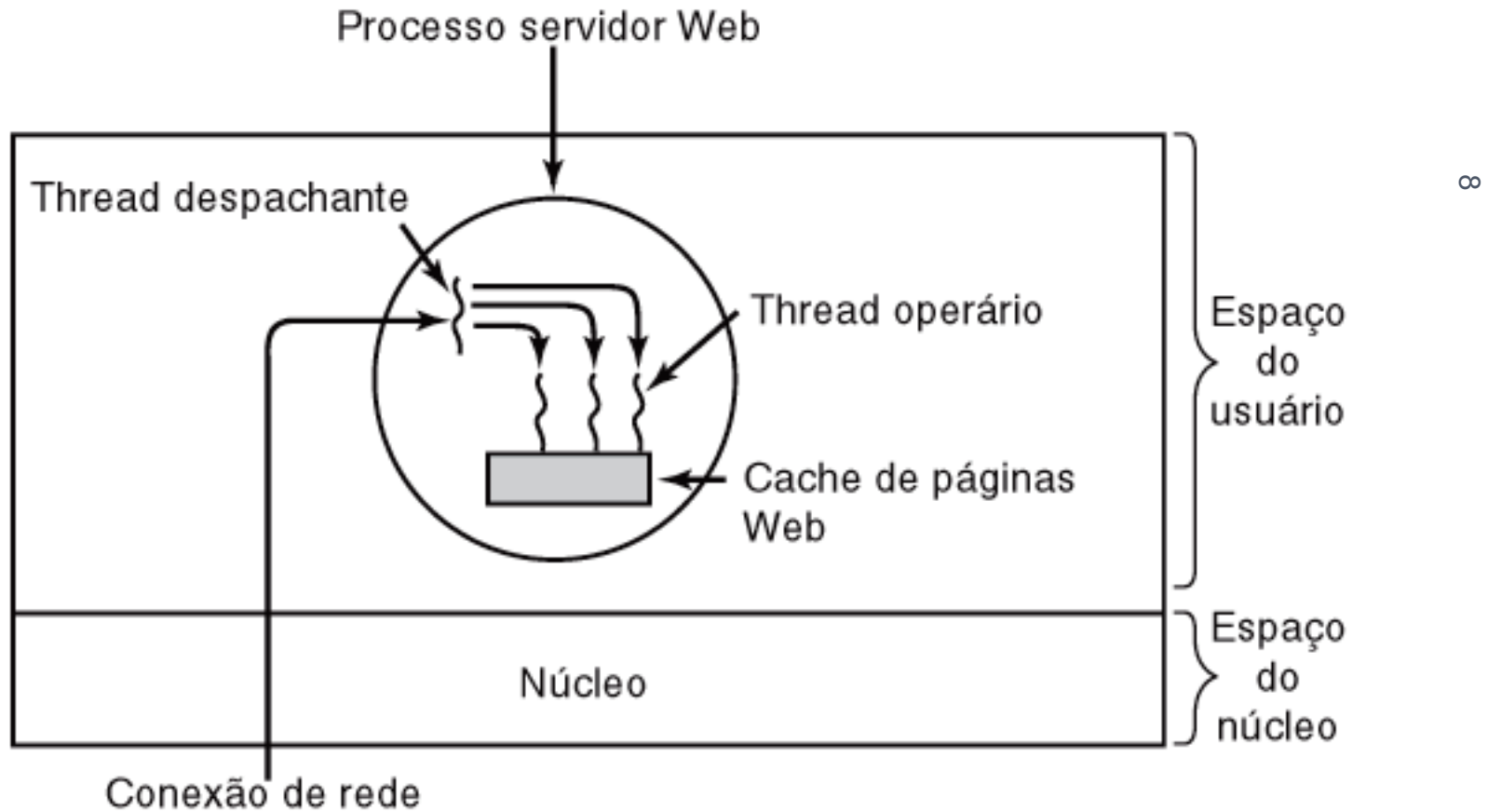
USO DE THREAD



Um processador de texto com três threads



USO DE THREAD

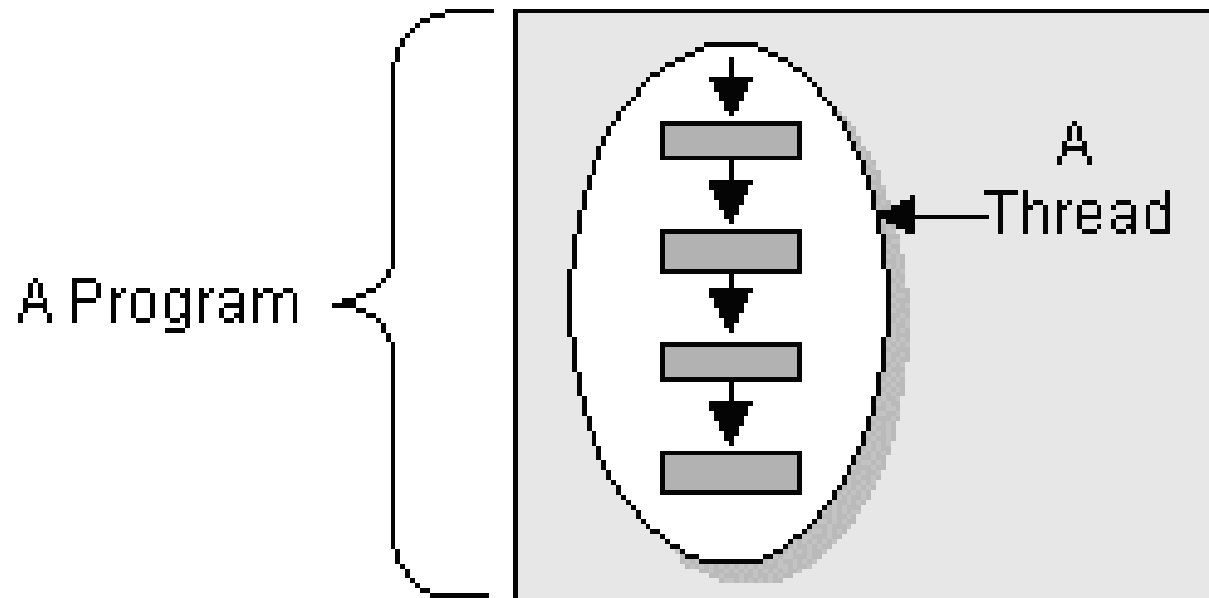


Um servidor web com múltiplos threads



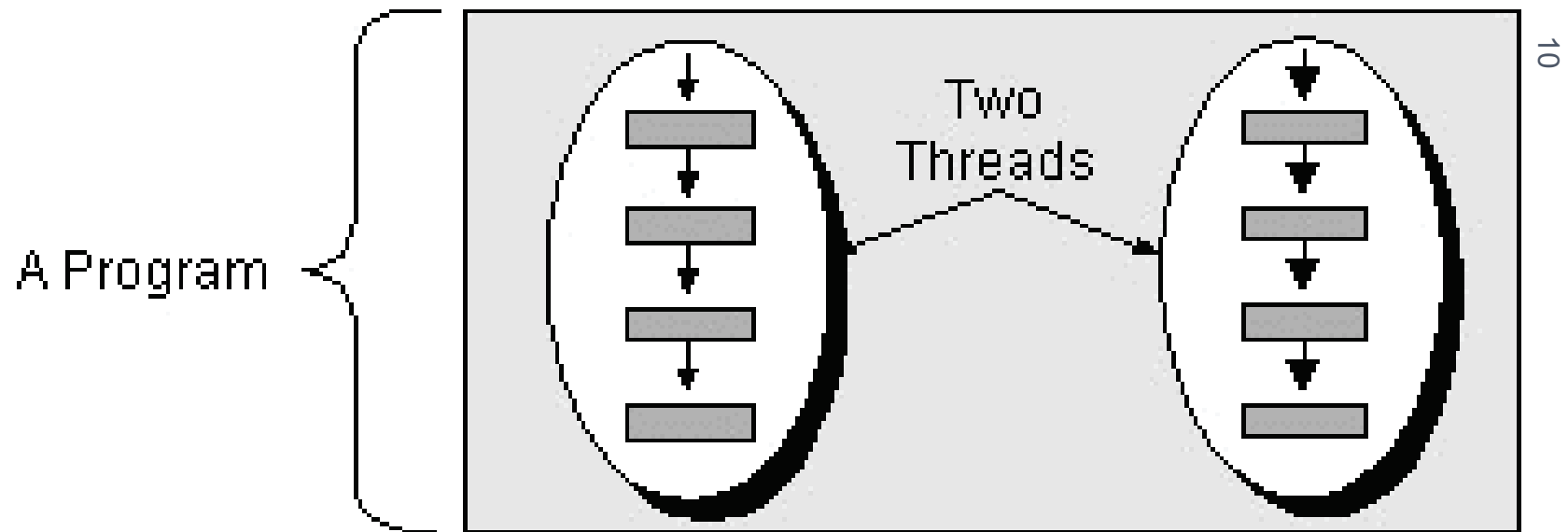
FLUXO SEQUENCIAL DE THREADS

- Fluxo sequencial único de controle dentro de um programa



Um Processo com 1 thread

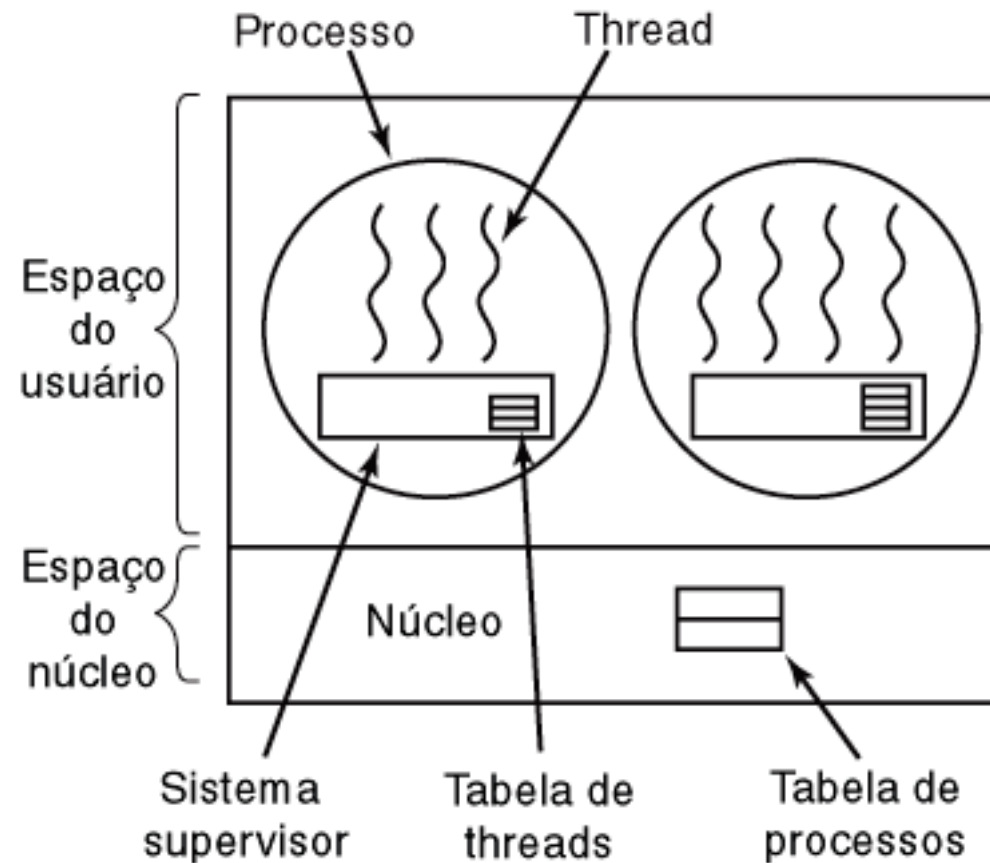
FLUXO SEQUENCIAL DE THREADS



Um Processo com 2 threads

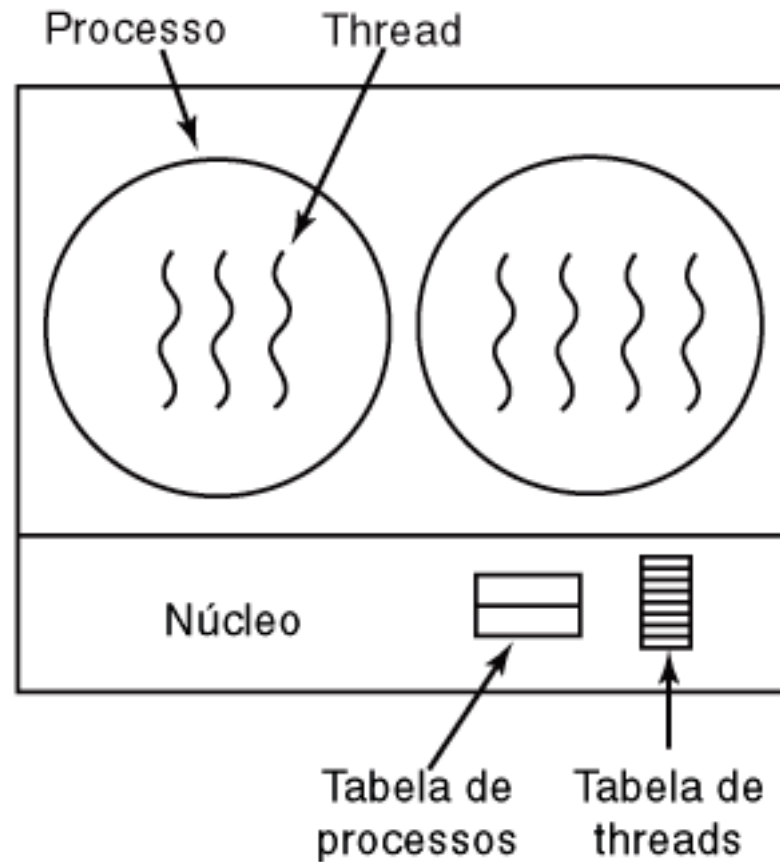


IMPLEMENTAÇÃO DE THREADS DE USUÁRIO



Um pacote de threads de usuário

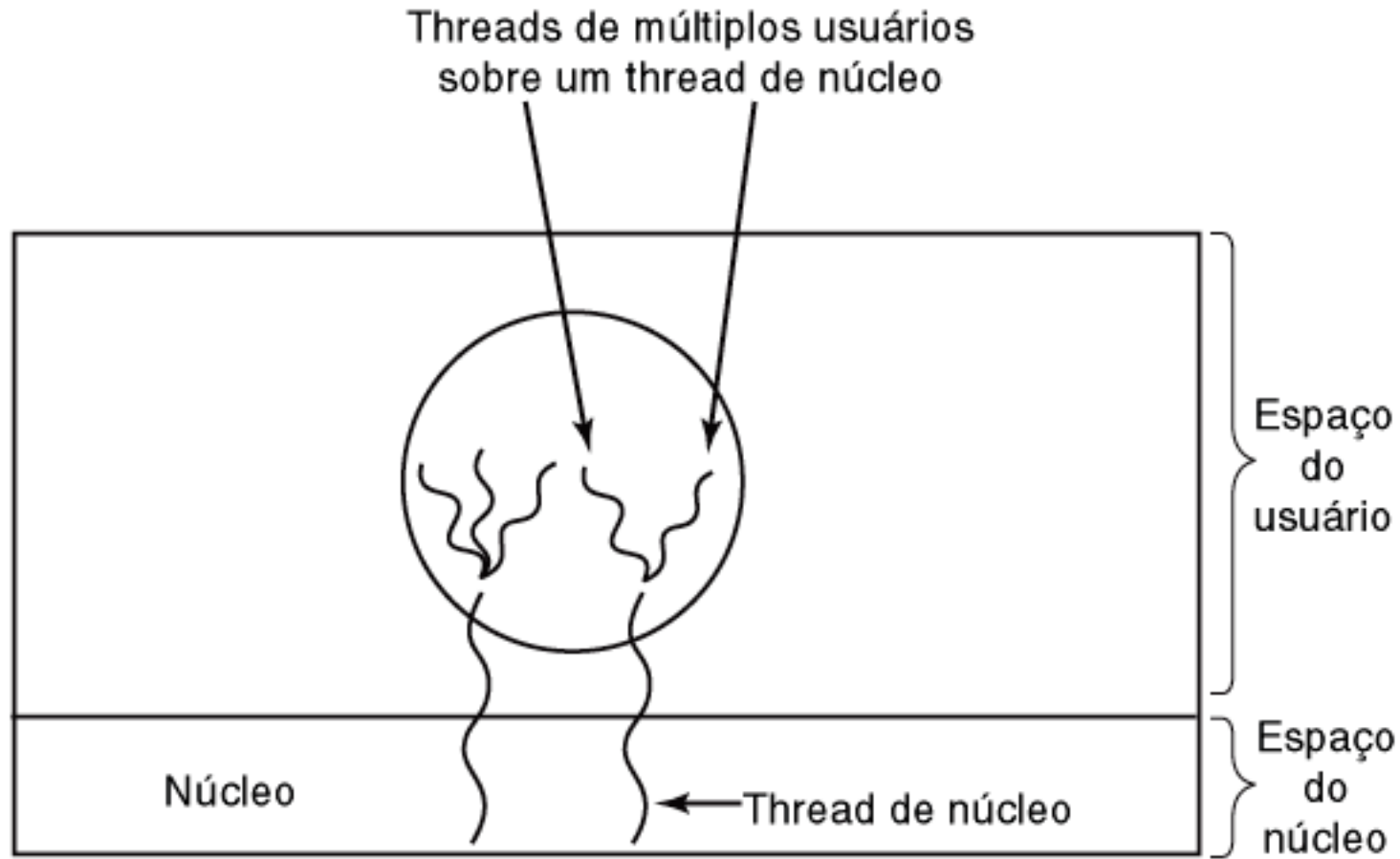
IMPLEMENTAÇÃO DE THREADS DE NÚCLEO



Um pacote de threads gerenciado pelo núcleo



IMPLEMENTAÇÕES HÍBRIDAS



Multiplexação de threads de usuário sobre threads de núcleo

THREADS

- Também chamadas de *processos leves* porque:
 - Criação de threads é mais “barato” computacionalmente que a de processos;
 - A troca de threads não demora tanto quanto a troca de processos;
 - A comunicação entre threads é mais rápida/fácil do que entre processos.

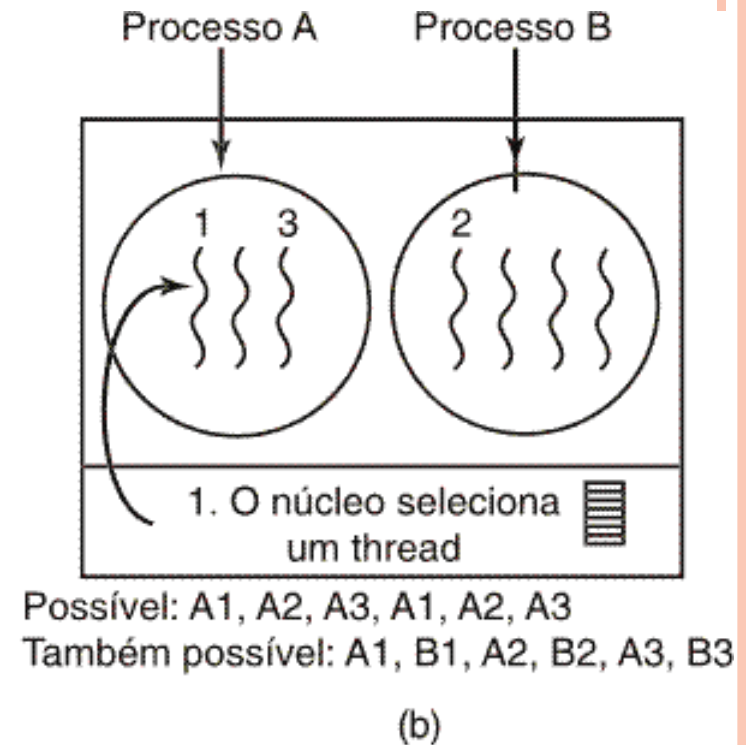
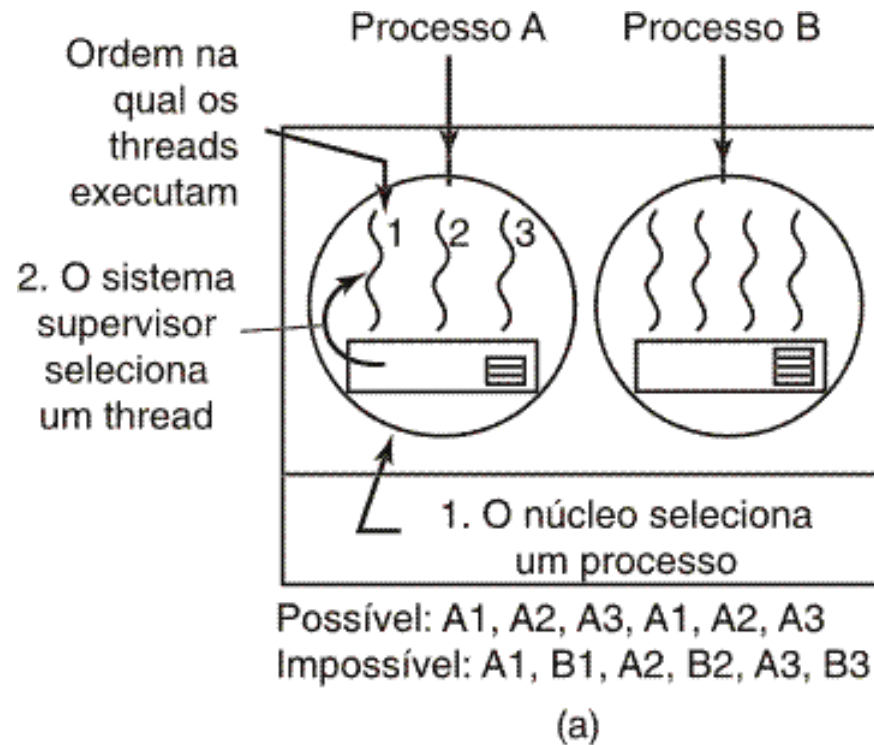


ESTADOS DOS THREADS

Estado	Significado
Executando	A thread está rodando.
Pronto	A thread está pronta para rodar.
Suspenso	A atividade da thread está parada temporariamente.
Continue	O ato da thread retomar sua atividade.
Bloqueado	A execução da thread está bloqueada à espera de um recurso.
Terminado	A execução da thread foi encerrada, não podendo ser retomada.



ESCALONAMENTO DE THREADS



Possível escalonamento de threads de usuário

- Processo com quantum de 50-mseg;
- Threads executam 5 mseg por surto de CPU.

PROCESSOS *VERSUS* THREADS

- Os processo estão isolados entre si;
- Os threads partilham os recursos entre si;
- A criação de threads resultam em menos sobrecarga para o sistema operacional do que a criação de processos;
- A gestão de threads necessita de menos recursos que a gestão dos processos.



PROCESSOS *VERSUS* THREADS

- Todas os threads em um processo partilham o mesmo espaço de endereçamento;
- A comunicação entre threads é na maior parte das situações mais fácil do que a comunicação entre processos.



VANTAGENS

- As aplicações com threads são mais eficientes quando se refere a:
 - Ganhos nas performances dos programas;
 - Podem não desperdiçar CPU com operações de entrada/saída de dados.



DÚVIDAS?!

