CONSULTORIA ESPECIALIZADA: SISTEMAS DISTRIBUÍDOS

CONCEITOS BÁSICOS DE THREADS

PROF. CARLOS PAES.

PUC-SP

CONCORRÊNCIA EM SO

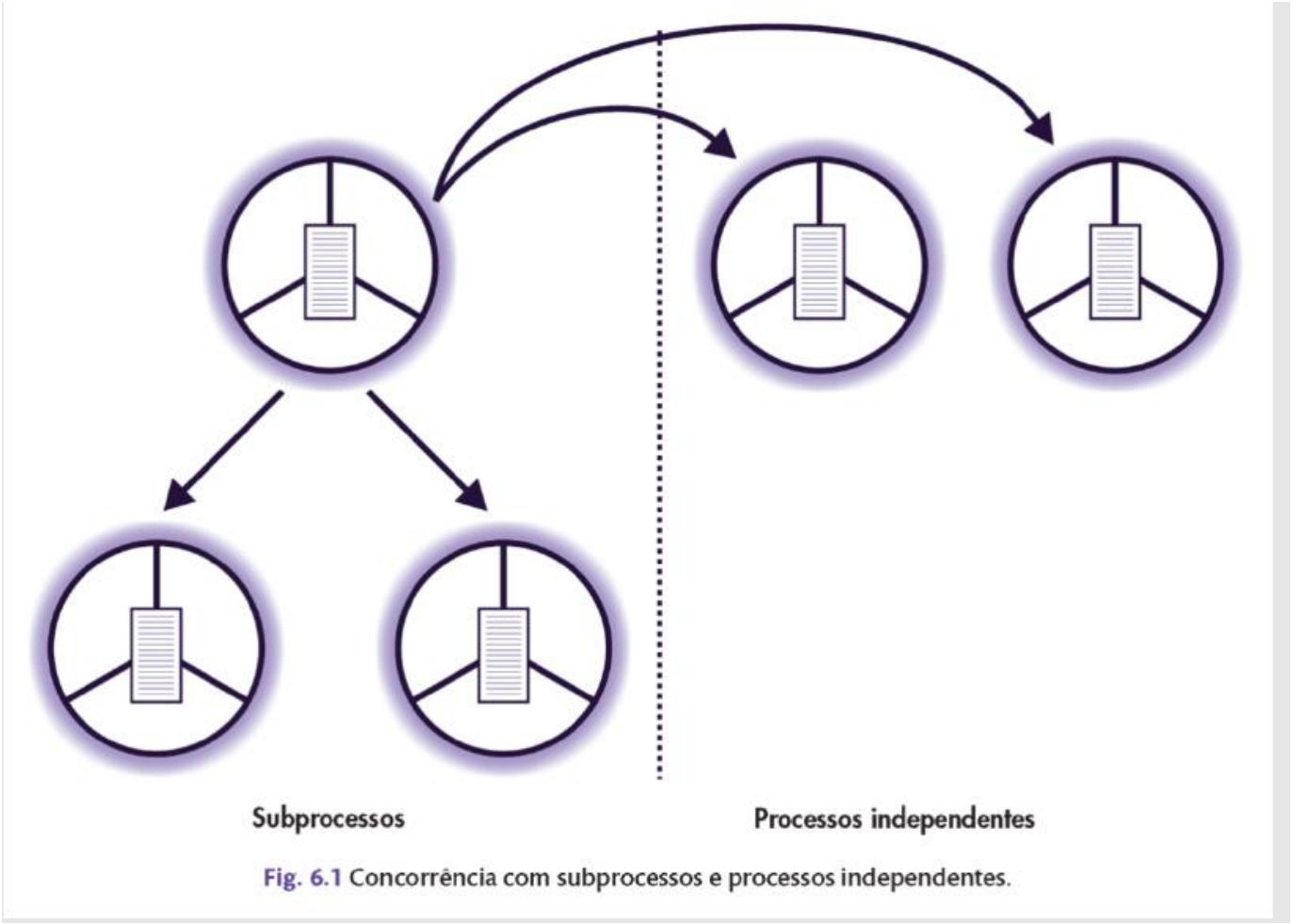
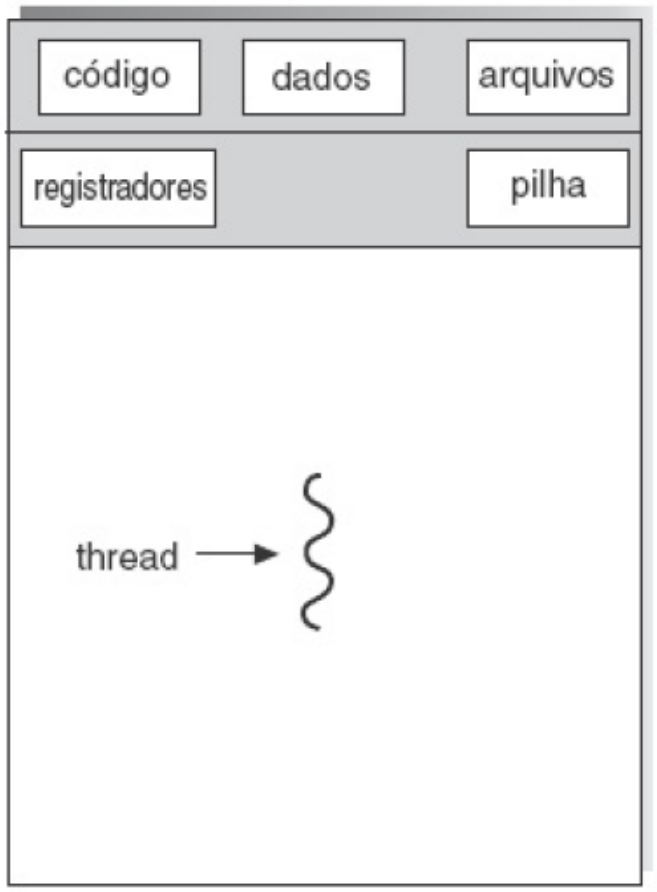
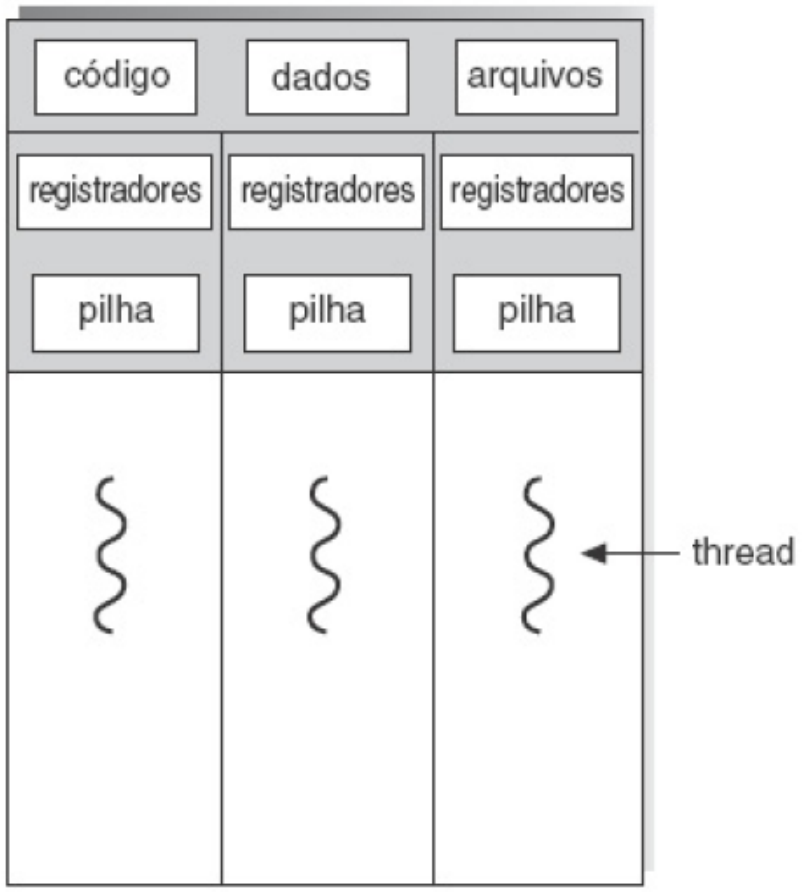


Fig. 6.1 Concorrência com subprocessos e processos independentes.

PROCESSO COM THREAD ÚNICA X MULTITHREADED

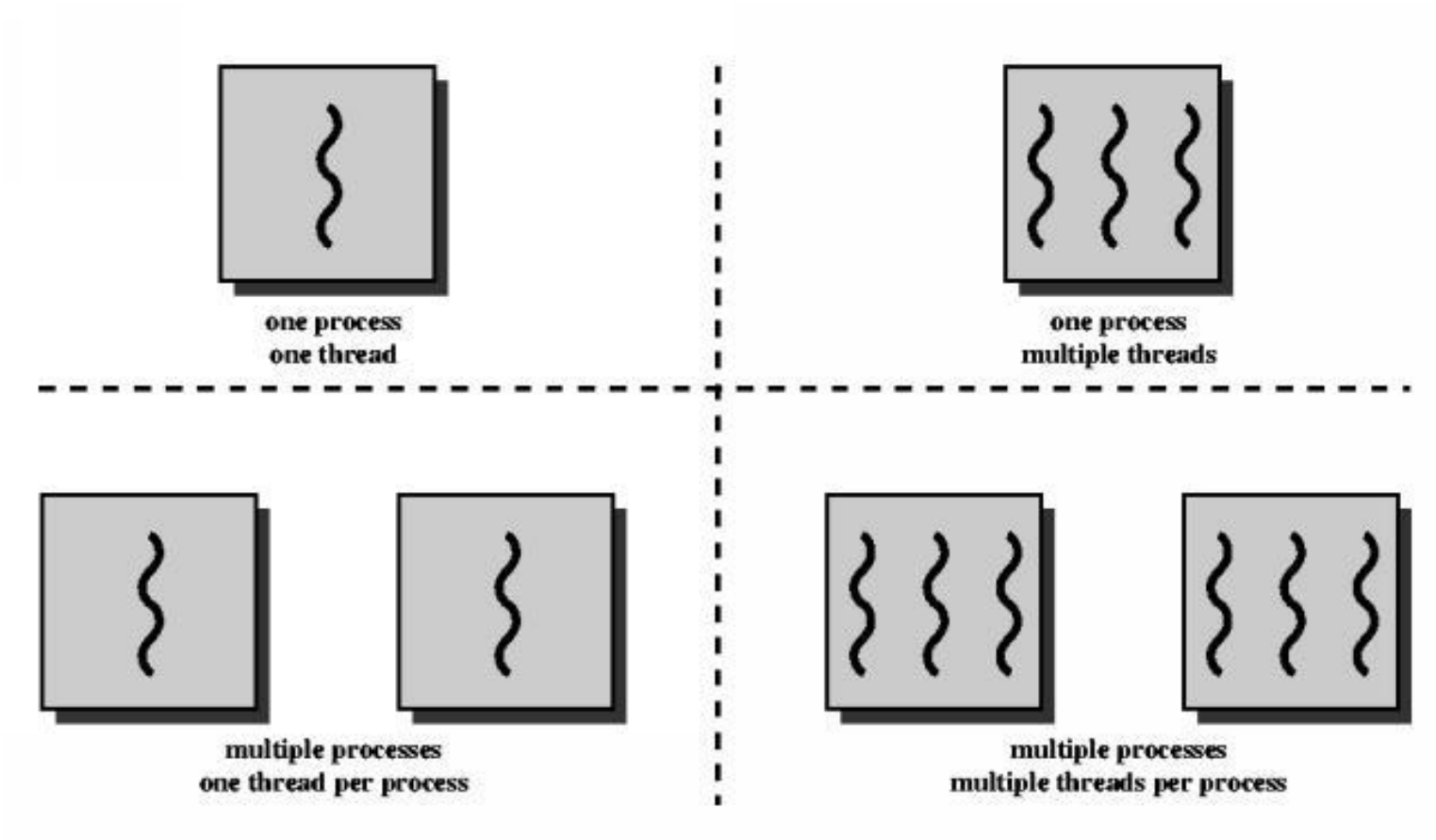


processo dotado de única thread

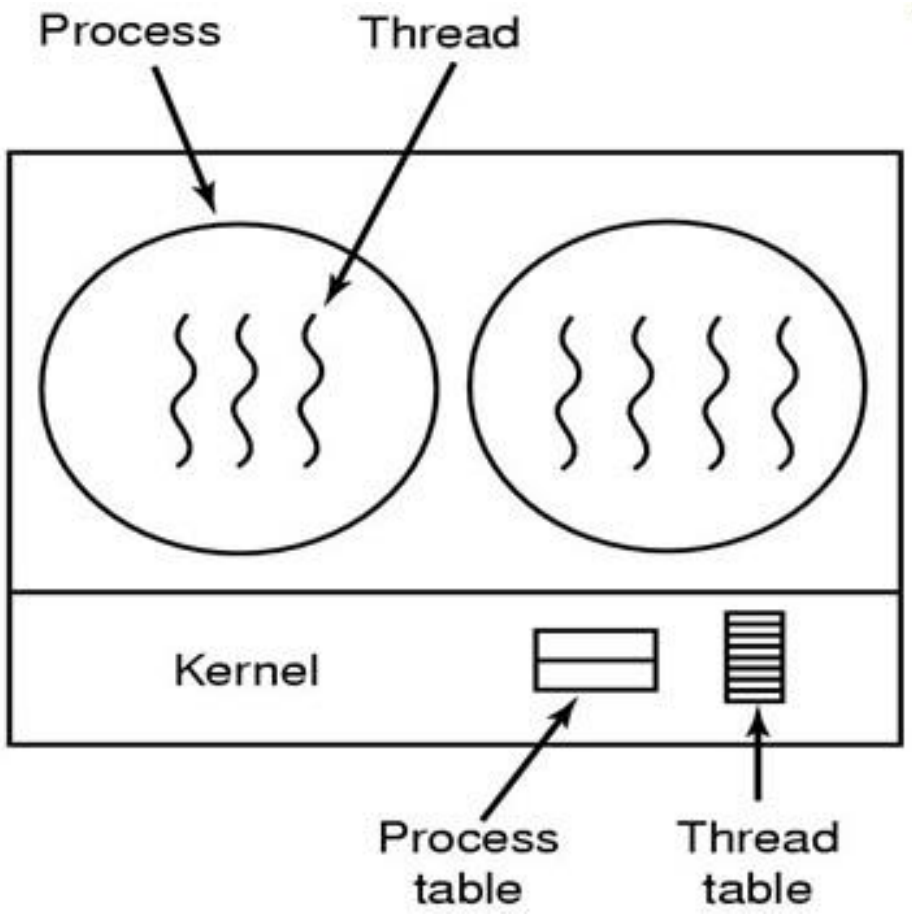


processo dotado de múltiplas threads

PROCESSO COM THREAD ÚNICA X MULTITHREADED



PROCESSOS X THREADS



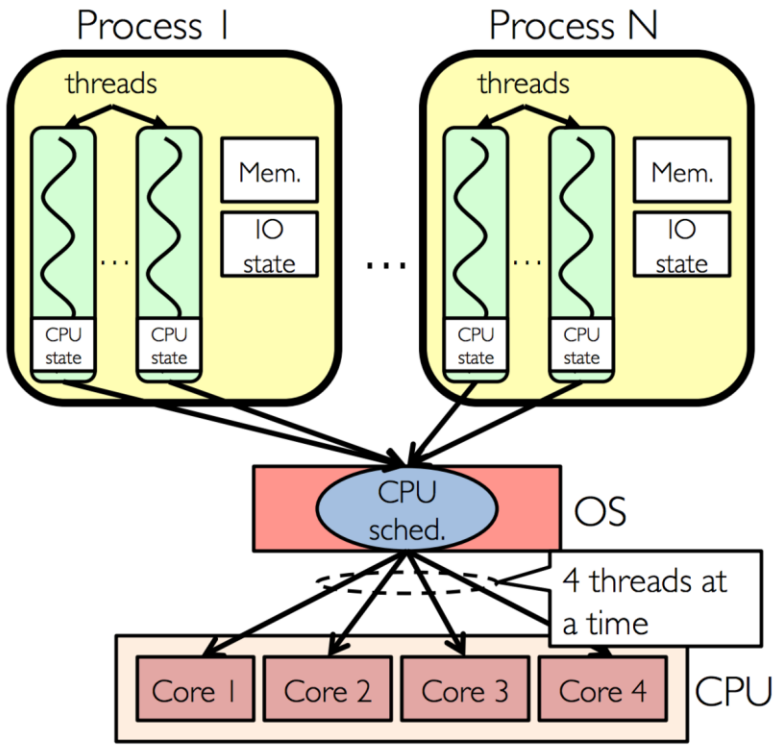
THREADS

- Fluxo de execução ou linha de execução
- Multitarefa preemptiva
- Conhecida também como lightweight process(LWP)
- Modelo de programação concorrente/paralela
- Existems outras formas de implementação de concorrência:
  - fiber/corotina (multitarefa cooperativa) (baixo suporte do SO)
- Amplamente adotada pelo sistemas operacionais modernos e linguagens de programação (interpretadas)

THREADS (ORIGEM)

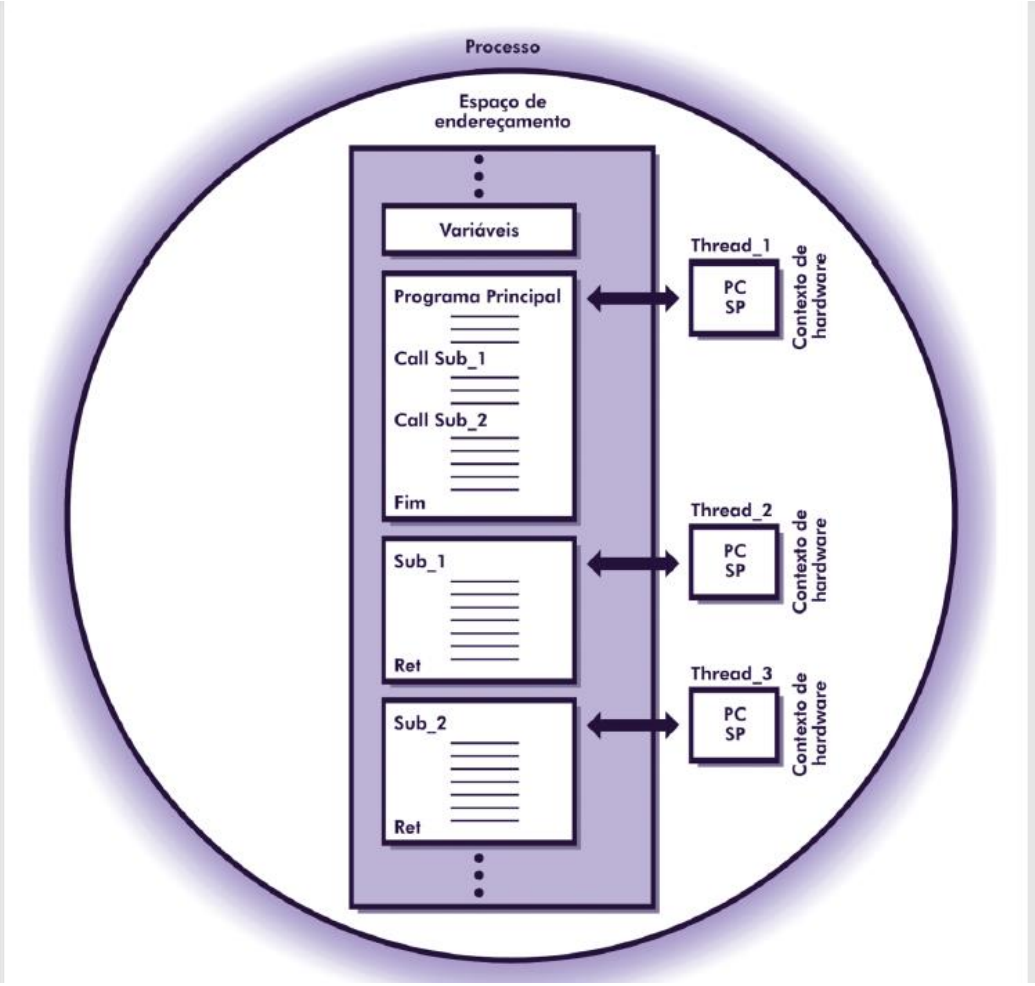
- 1979, Sistema Operacional Toth
  - Foi introduzido o conceito de processos lightweight (peso leve), onde o espaço de endereçamento de um processo era compartilhado por vários programas.
- Apesar do conceito revolucionário, a ideia não foi utilizada comercialmente e somente em meados de 1980, com o desenvolvimento do sistema operacional Mach, na Universidade de Carnegie Mellon, ficou clara a separação entre o conceito de processo e thread.
- A partir do conceito de múltiplos threads (multithread) é possível projetar e implementar aplicações concorrentes

VANTAGENS



- Responsividade
- Compartilhamento de recursos
- Economia
- Utilização de arquiteturas multiprocessadas e Multicore (vários núcleos)

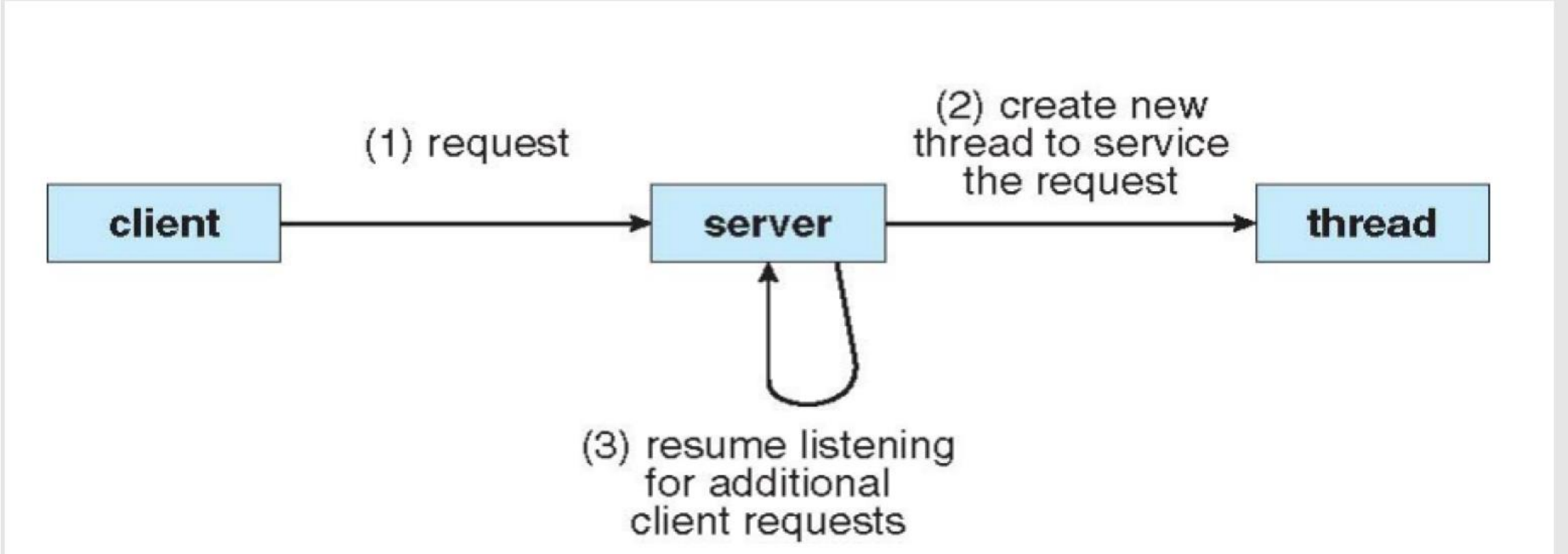
PROGRAMA MULTITHREADED



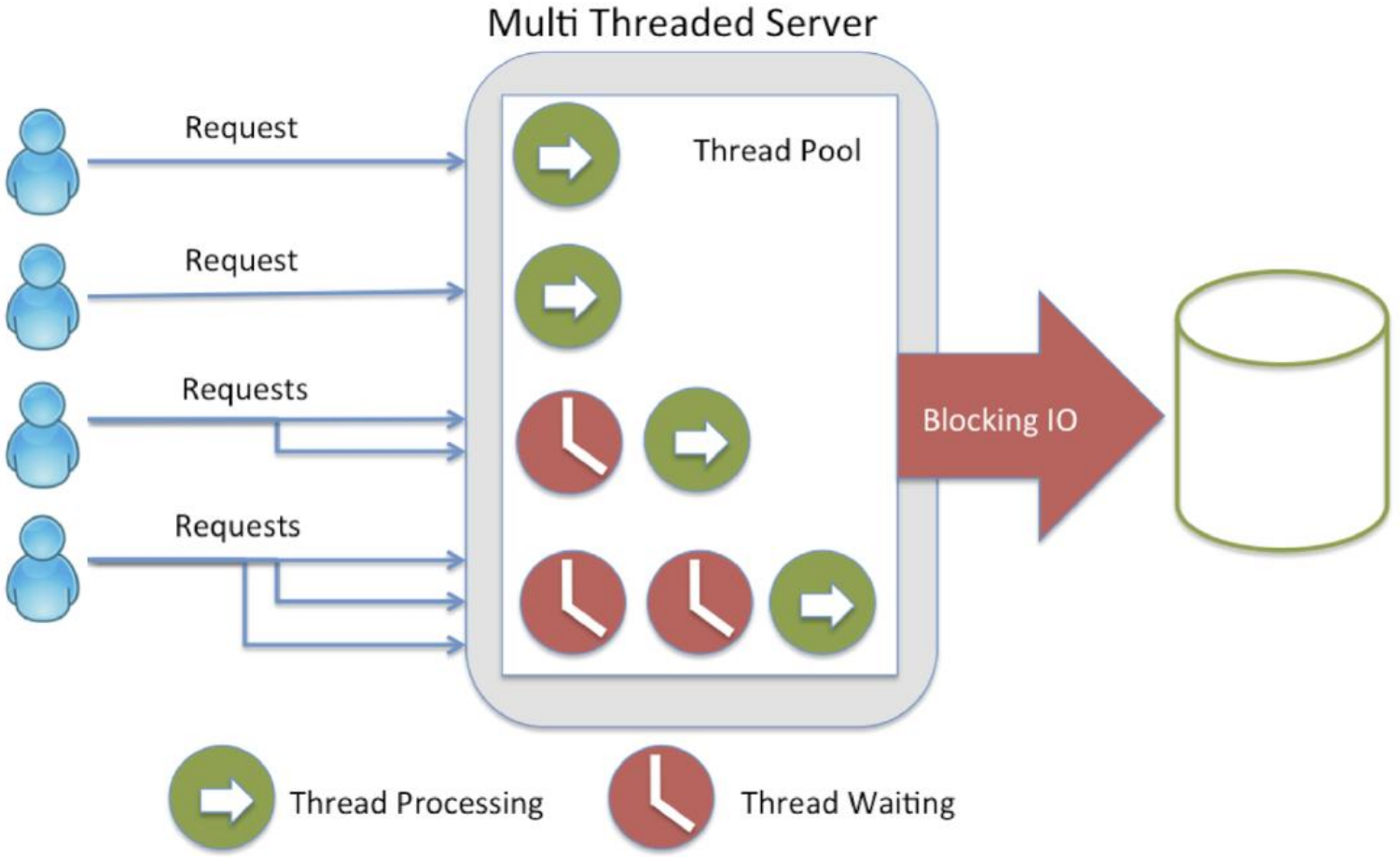
PROGRAMAÇÃO MULTICORE

- Sistemas multicore pressionam os programadores, os desafios incluem:
  - Divisão em tarefas/atividades(especificação da arquitetura) (visões de concorrência e paralelismo)
  - Balanceamento da carga de processamento
  - Divisão de dados
  - Dependência de dados
  - Sincronização
  - Anomalias como Deadlock
  - Teste e depuração

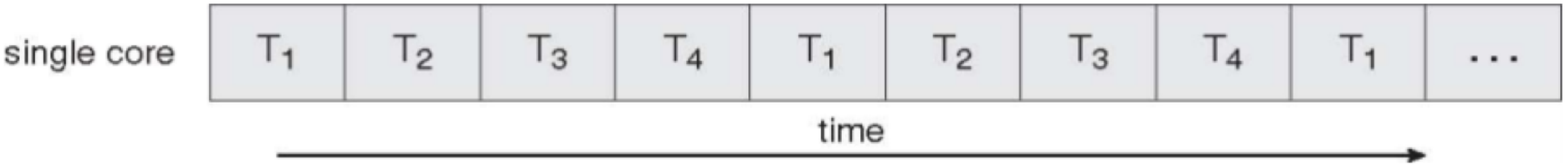
ARQUITETURA SERVIDOR MULTITHREAD



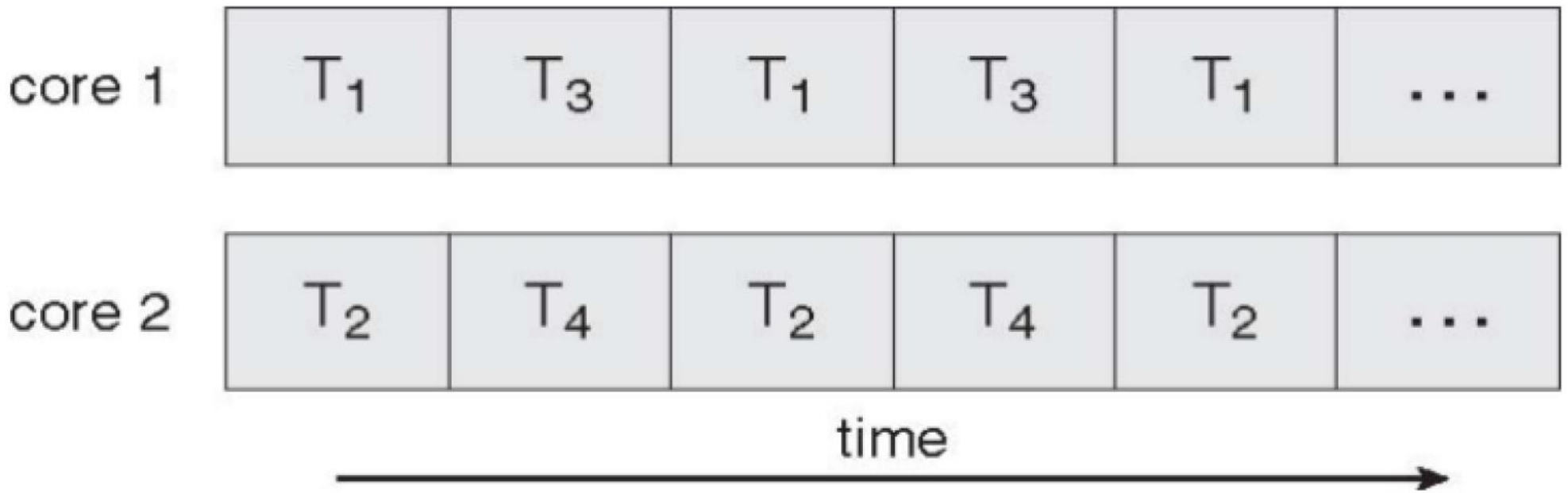
ARQUITETURA POOL DE THREADS



EXECUÇÃO CONCORRENTE SINGLE-CORE



EXECUÇÃO PARALELA MULTICORE



TIPOS DE THREAD

- Threads de usuário
  - Conhecidas também como green thread
- Threads de sistema
  - conhecidas também como threads de kernel

THREADS DE USUÁRIO

- Também conhecidas como Green threads
- Gerenciadas por bibliotecas de tempo de execução (runtime library) ou por máquina virtual nível de usuário
- Três bibliotecas de threads principais:
  - POSIX Pthreads (foco SO Unix)
  - Threads Java
  - Threads Win32 (Microsoft Windows)
  - Python Threads

THREADS DE KERNEL

- Implementação de threads no nível de Sistema Operacional
- O SO gerencia processos (HWP) e threads (LWP)
- Modelo implementado por vários SOs modernos
- Pode ter controle do número de threads por processo de usuário
- Exemplos de SOs
  - Microsoft Windows
  - Solaris
  - Linux
  - Tru64 UNIX
  - Mac OS X

COMPARATIVO HWP E LWP

Tabela 6.1 Latência de processos e threads (Vahalia, 1996)

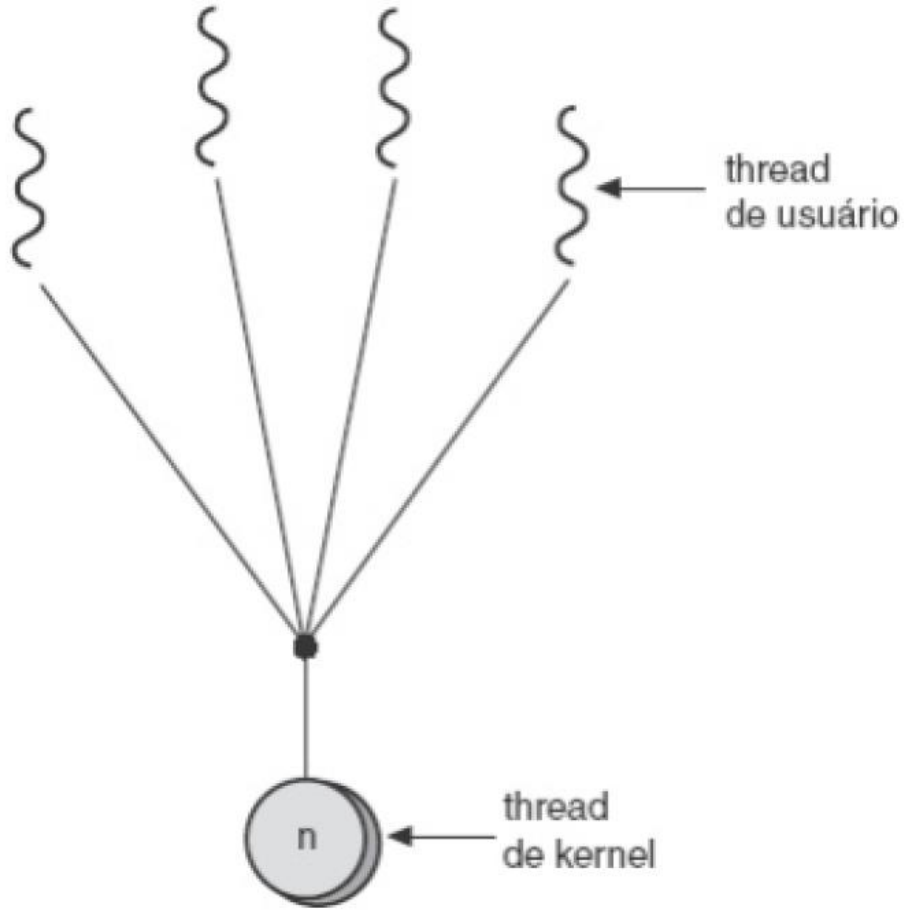
Implementação	Tempo de criação (µs)	Tempo de sincronização (µs)
Processo	1700	200
Processo Lightweight	350	390
Thread	52	66

- VAHALIA, U. 1996. UNIX internals: the new frontiers. Prentice Hall.

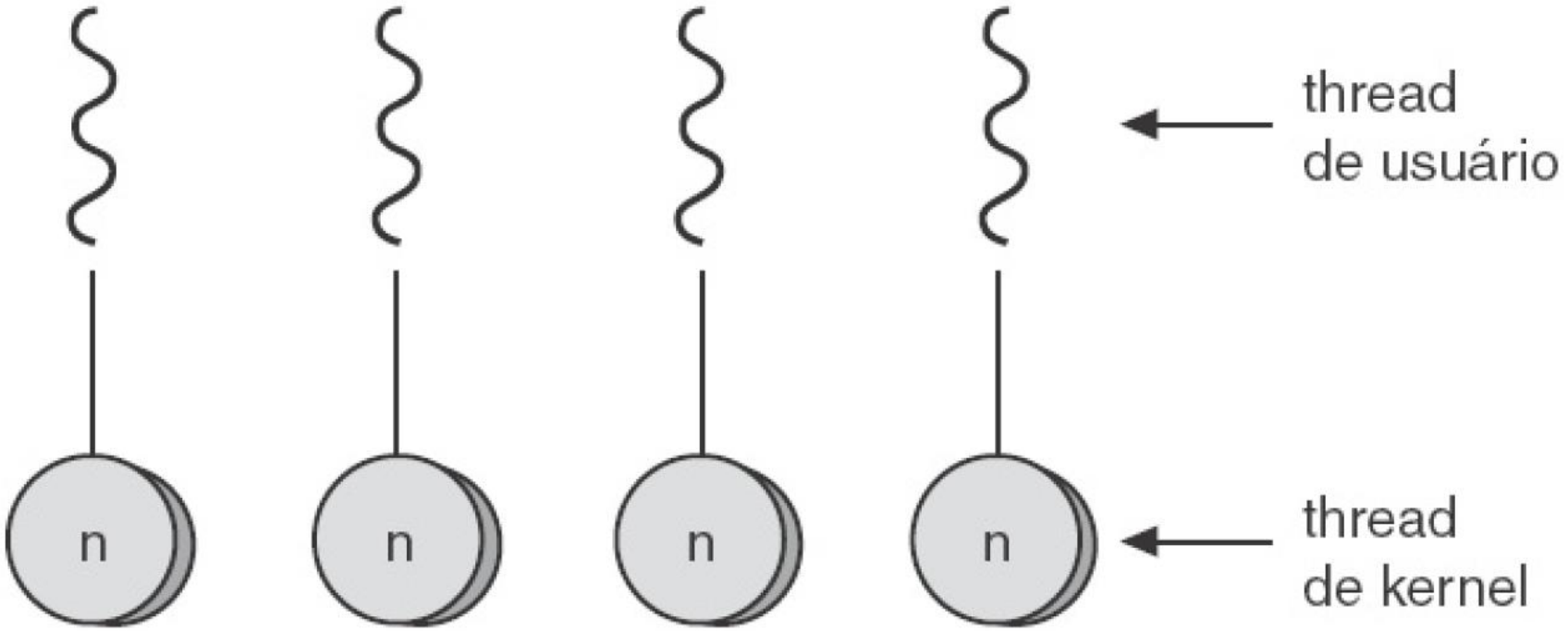
MODELO MULTITHREADED

- Muitos-para-um
  - Muitas threads no nível do usuário associadas a uma única thread de kernel
  - Ex: Solaris Green Threads e GNU Portable Threads
- Um-para-um
  - Cada thread do usuário associada a uma thread de kernel
  - Exemplos: Microsoft Windows, Linux Solaris 9, Mac OS
- Muitos-para-muitos
  - Permite que muitas threads no nível do usuário sejam associadas a muitas threads no nível do kernel
  - Exemplos: Solaris antes da versão 9, Microsoft Windows com o recurso ThreadFiber

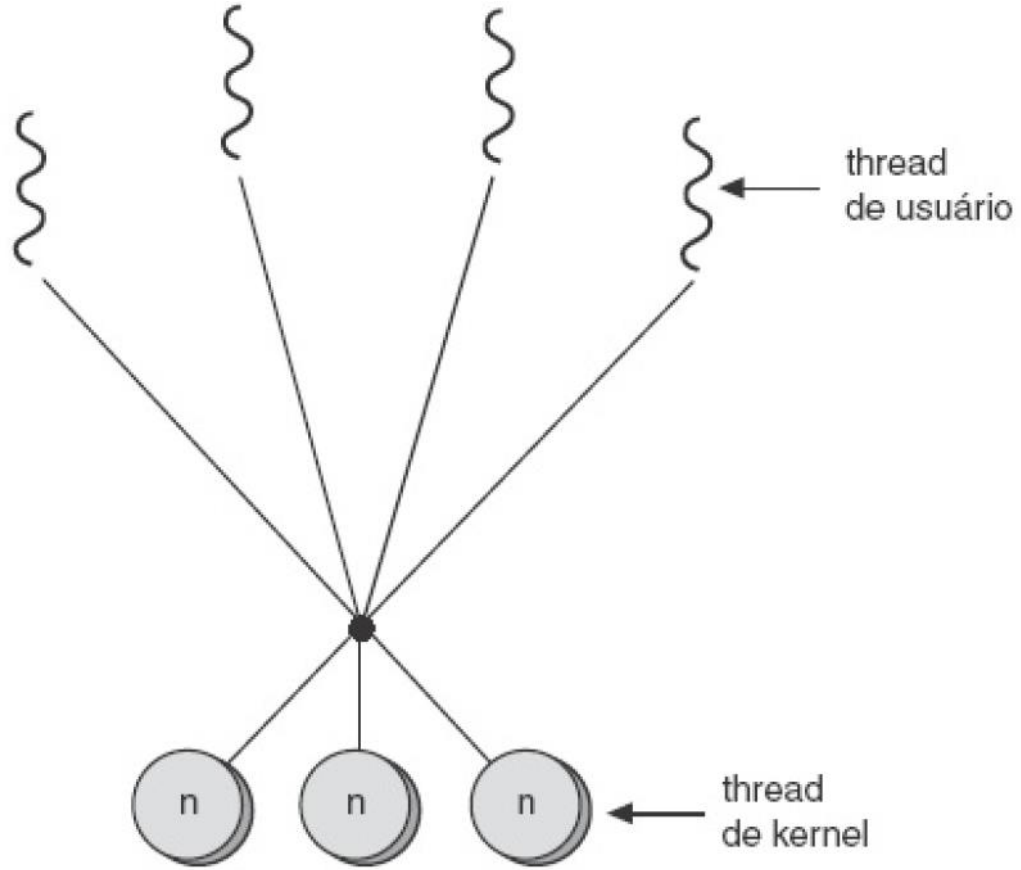
MUITOS-PARA-UM



UM-PARA-UM



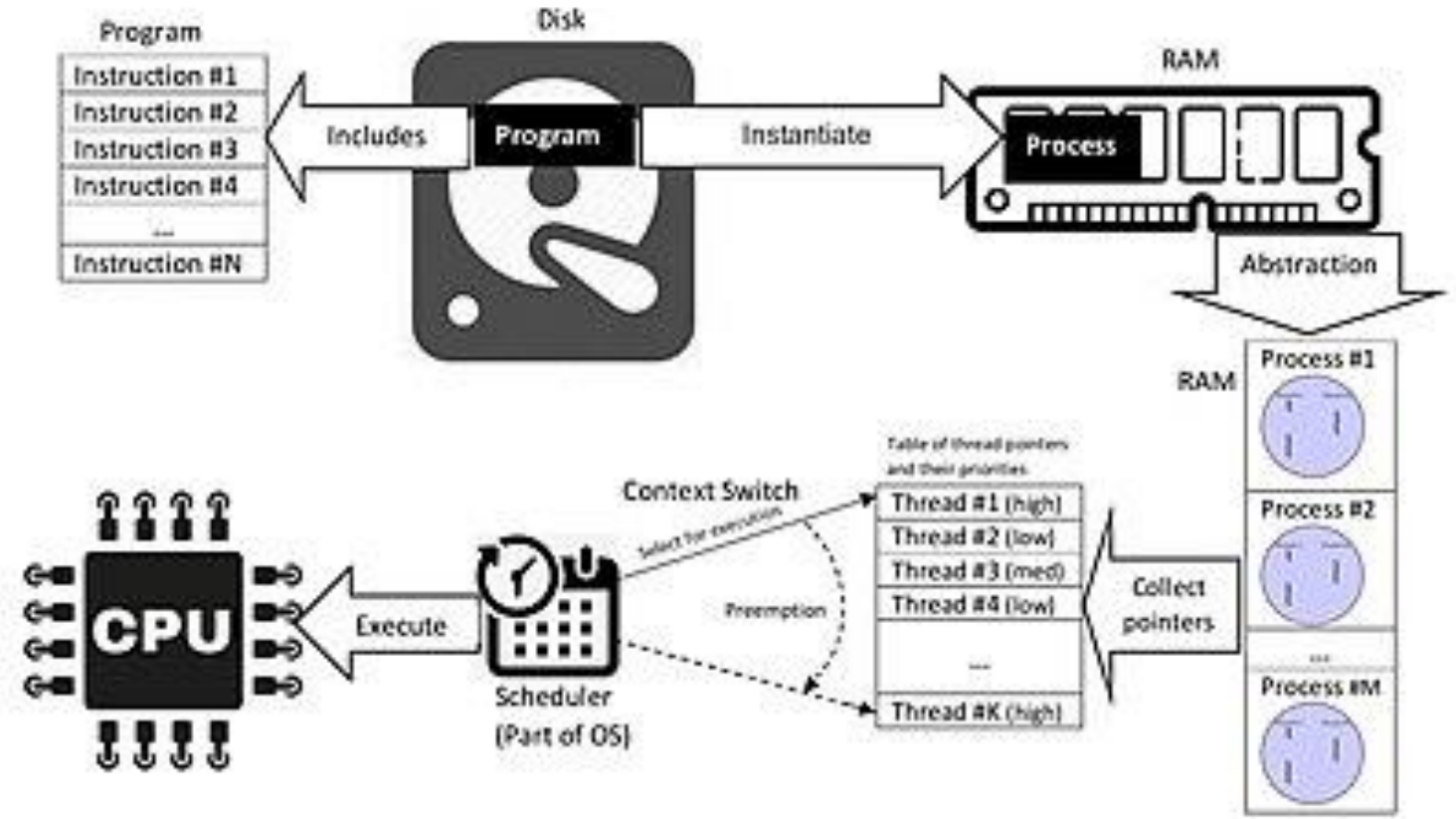
MUITOS-PARA-MUITOS

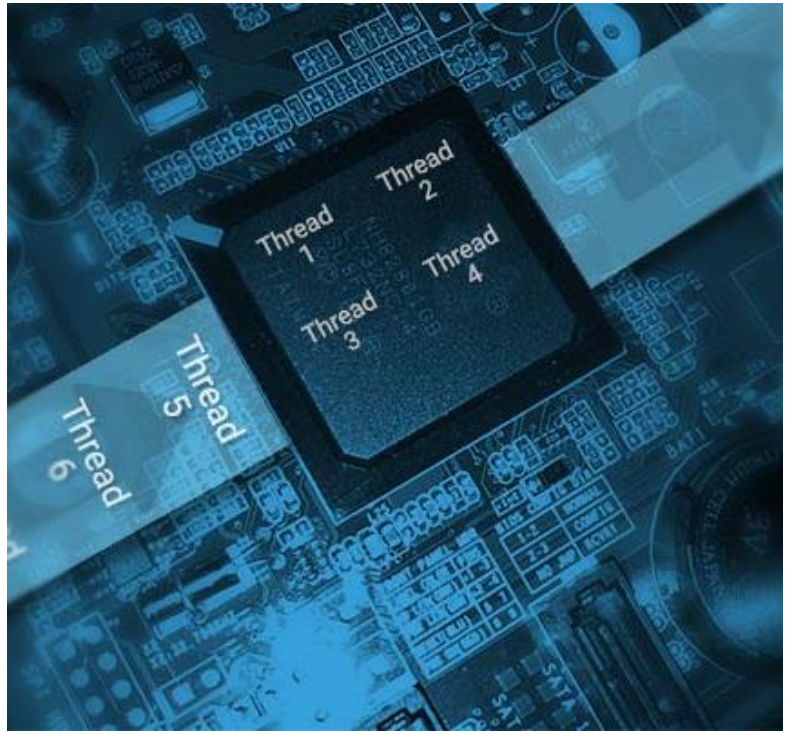


ASPECTOS DE USO DE THREADS

- Semântica das chamadas de sistema fork() e exec()
- Cancelamento da thread
- Tratamento de sinais
- Bancos de threads
- Dados específicos da thread
- Ativações de escalonador

VISÃO GERAL DO AMBIENTE DE EXECUÇÃO NO SO



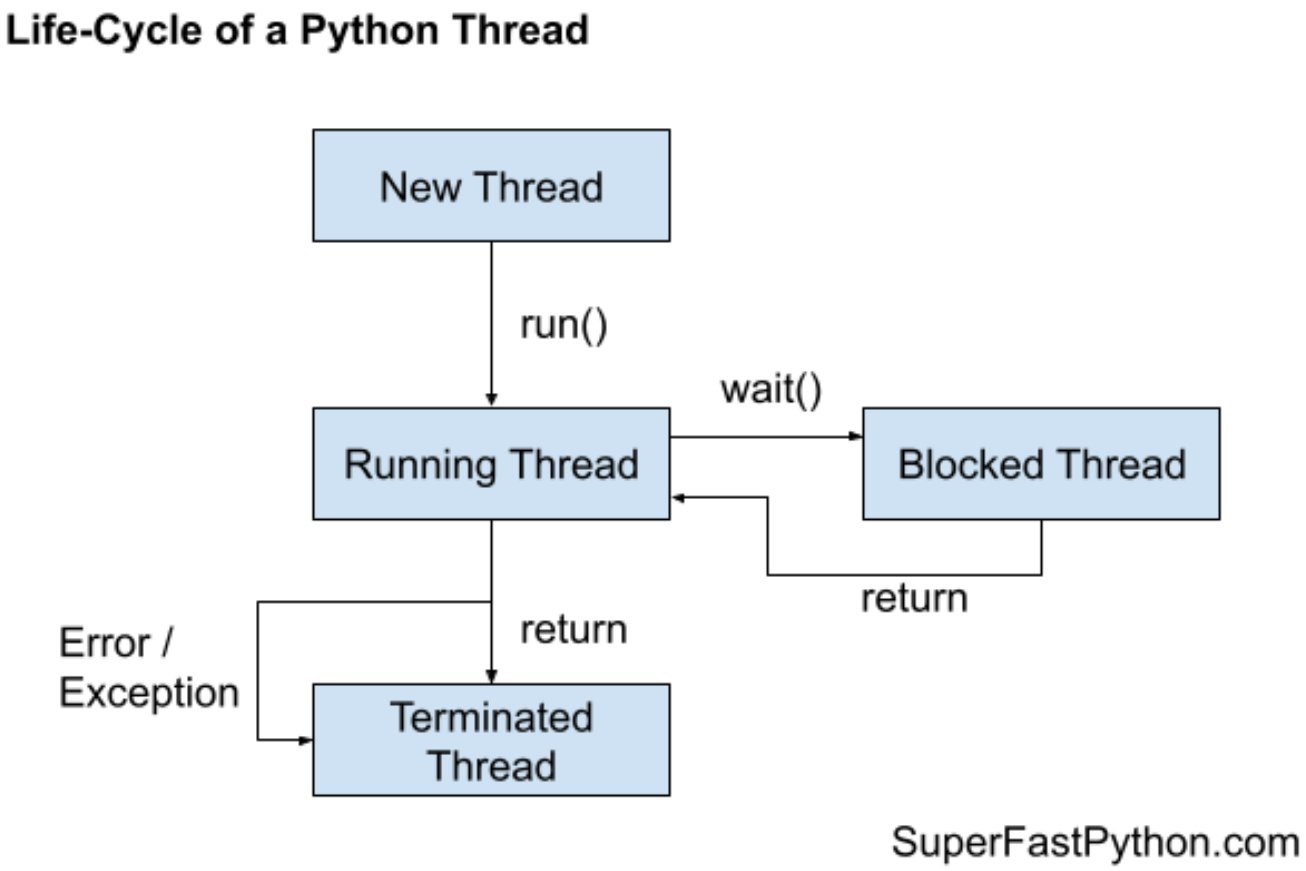


THREADS NO PYTHON

THREADS NO PYTHON

- São gerenciadas pelo interpretador python
- Duas formas de implementação
  - Extensão da classe Thread (POO)
  - Criação da função e instanciação

ESTADOS DAS THREADS NO PYTHON



OBRIGADO

CARLOSP@PUCSP.BR