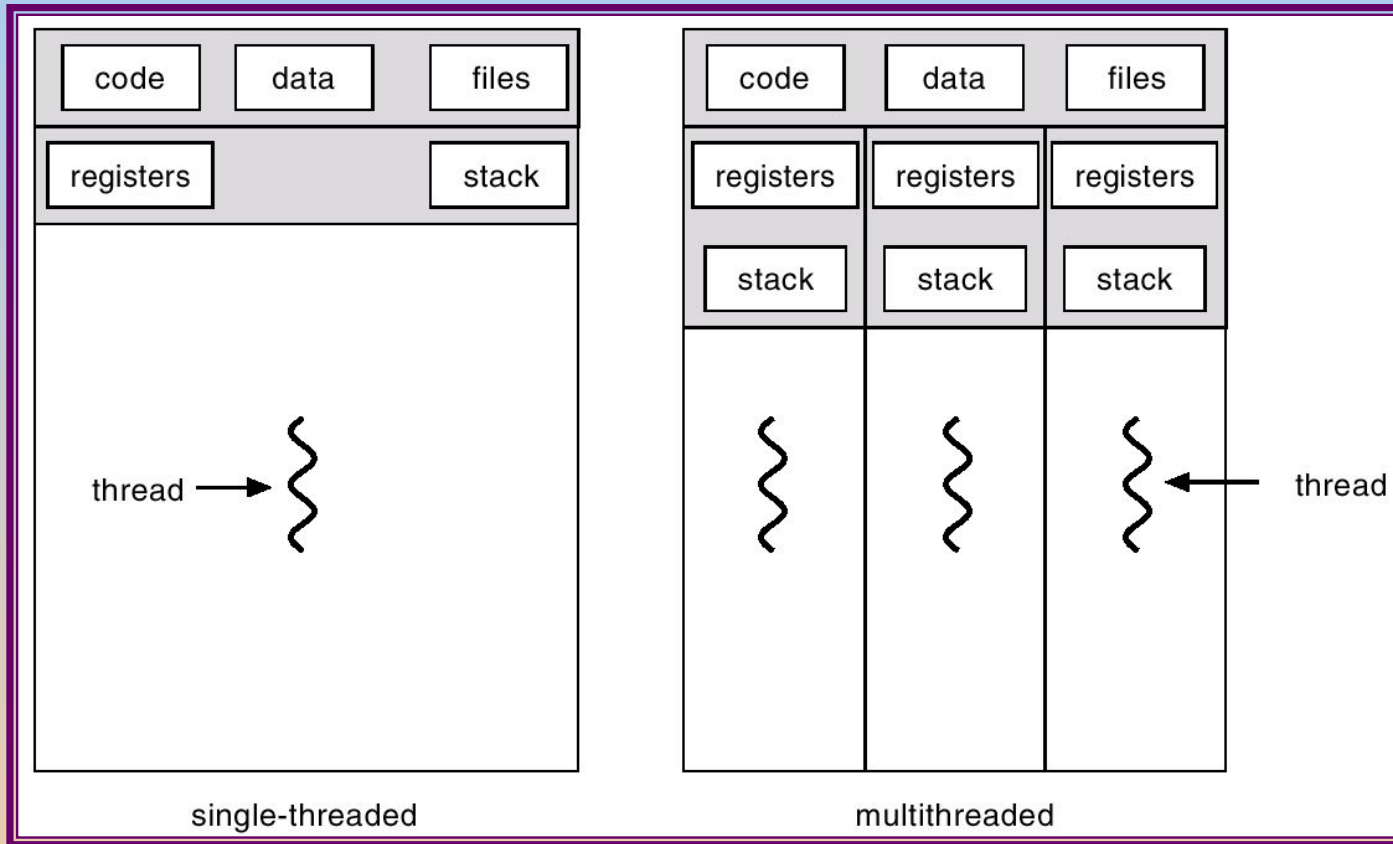# Chapter 5: Threads

- Overview
- Multithreading Models
- Threading Issues
- Pthreads
- Solaris 2 Threads
- Windows 2000 Threads
- Linux Threads
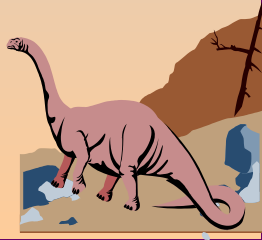- Java Threads

Operating System

Single and Multithreaded Processes

single-threaded   multithreaded

Operating System

Benefits

- Responsiveness

- Resource Sharing

- Economy

- Utilization of MP Architectures

Operating System

User Threads

- Thread management done by user-level threads library

- Examples
  - POSIX *Pthreads*
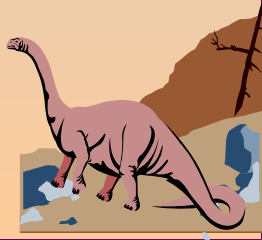 - Mach *C-threads*
  - Solaris *threads*

Operating System
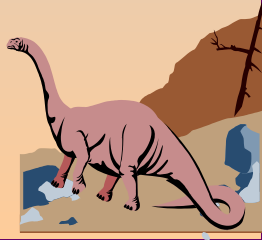
Kernel Threads

- Supported by the Kernel

- Examples
  - Windows 95/98/NT/2000
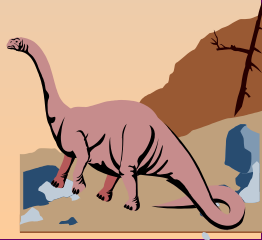 - Solaris
  - Tru64 UNIX
  - BeOS
  - Linux

Operating System

Multithreading Models
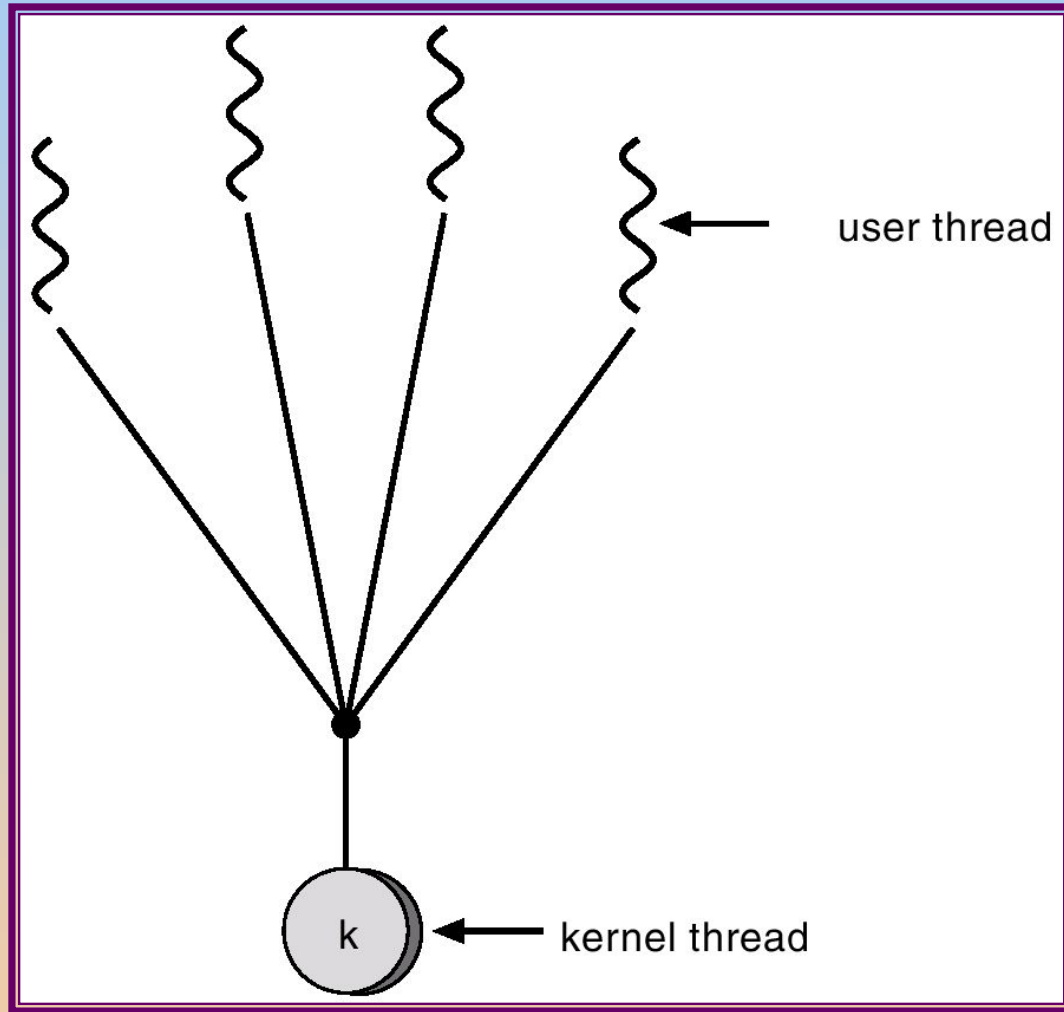
- Many-to-One

- One-to-One

- Many-to-Many

Operating System

Many-to-One

- Many user-level threads mapped to single kernel thread.
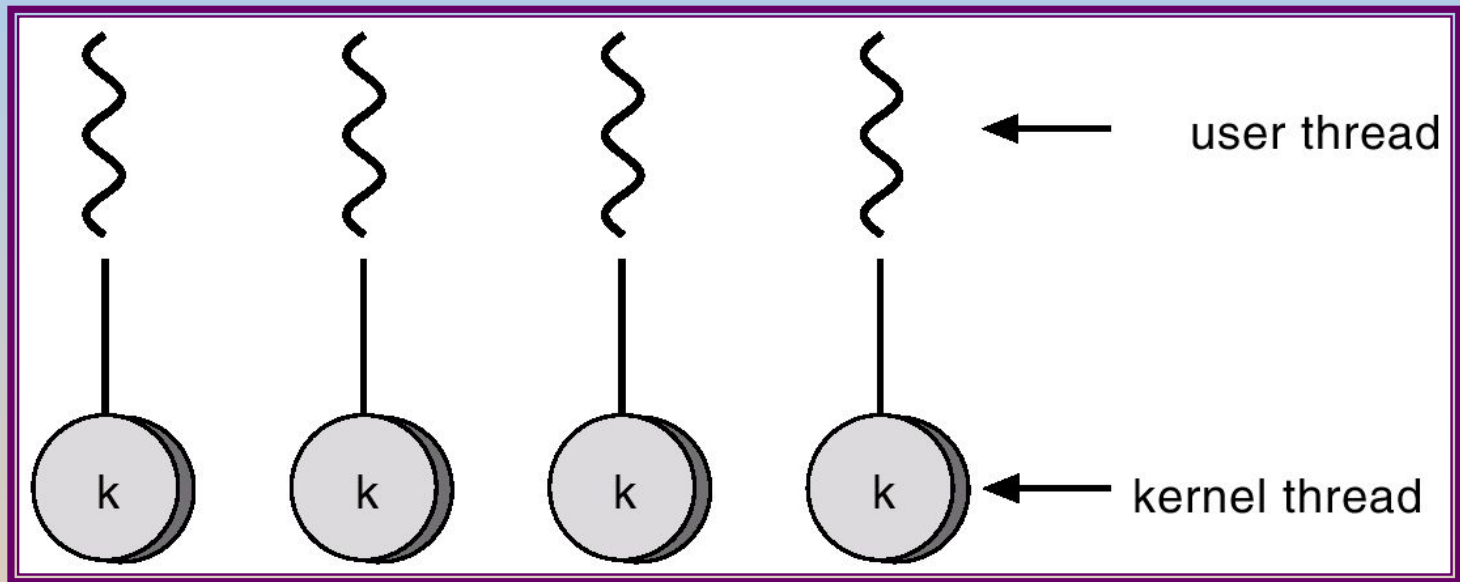
- Used on systems that do not support kernel threads.

Silberschatz, Galvin and Gagne ©2002

Operating System

Many-to-One Model

user thread

kernel thread

Operating System

One-to-One

- Each user-level thread maps to kernel thread.

- Examples
  - Windows 95/98/NT/2000
  - OS/2

Operating System
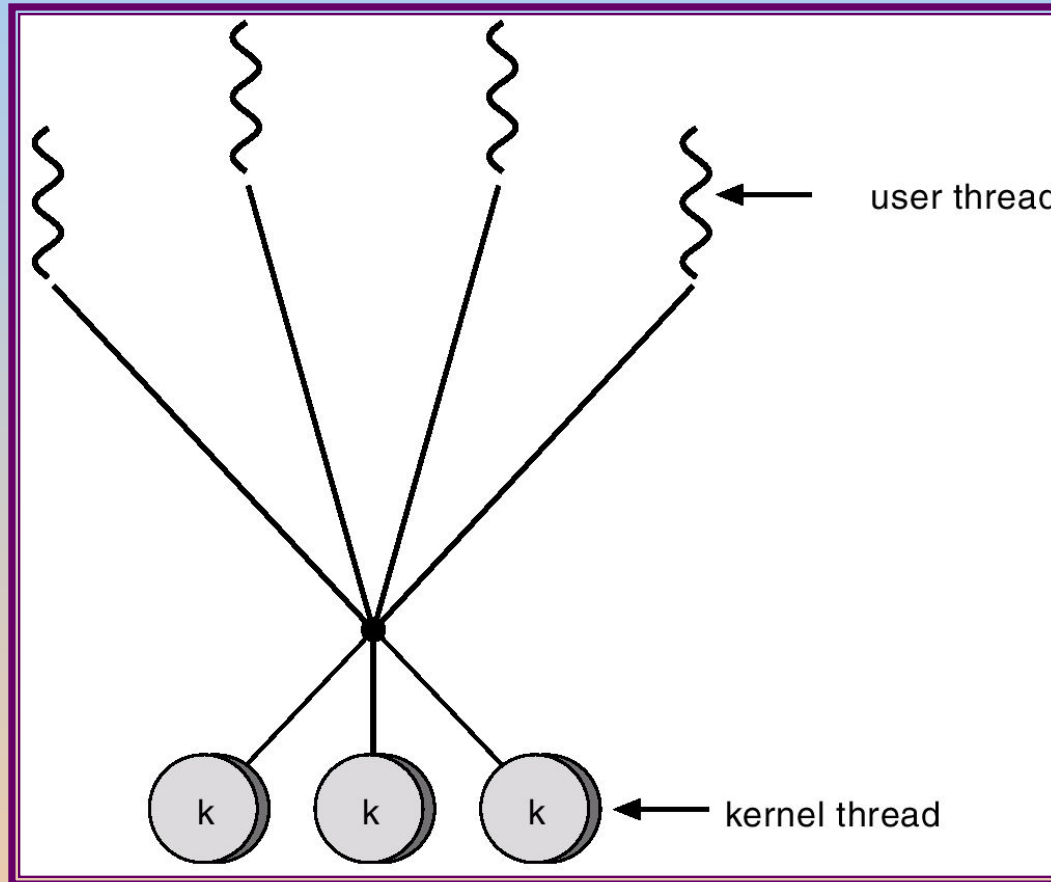
One-to-one Model

Operating System

Many-to-Many Model

- Allows many user level threads to be mapped to many kernel threads.

- Allows the operating system to create a sufficient number of kernel threads.

- Solaris 2

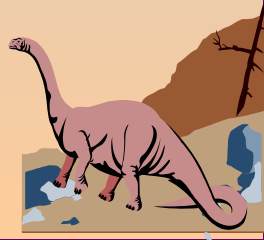- Windows NT/2000 with the *ThreadFiber* package

Silberschatz, Galvin and Gagne ©2002

Operating System

Many-to-Many Model

Operating System

Threading Issues

- Semantics of fork() and exec() system calls.
- Thread cancellation.
- Signal handling
- Thread pools
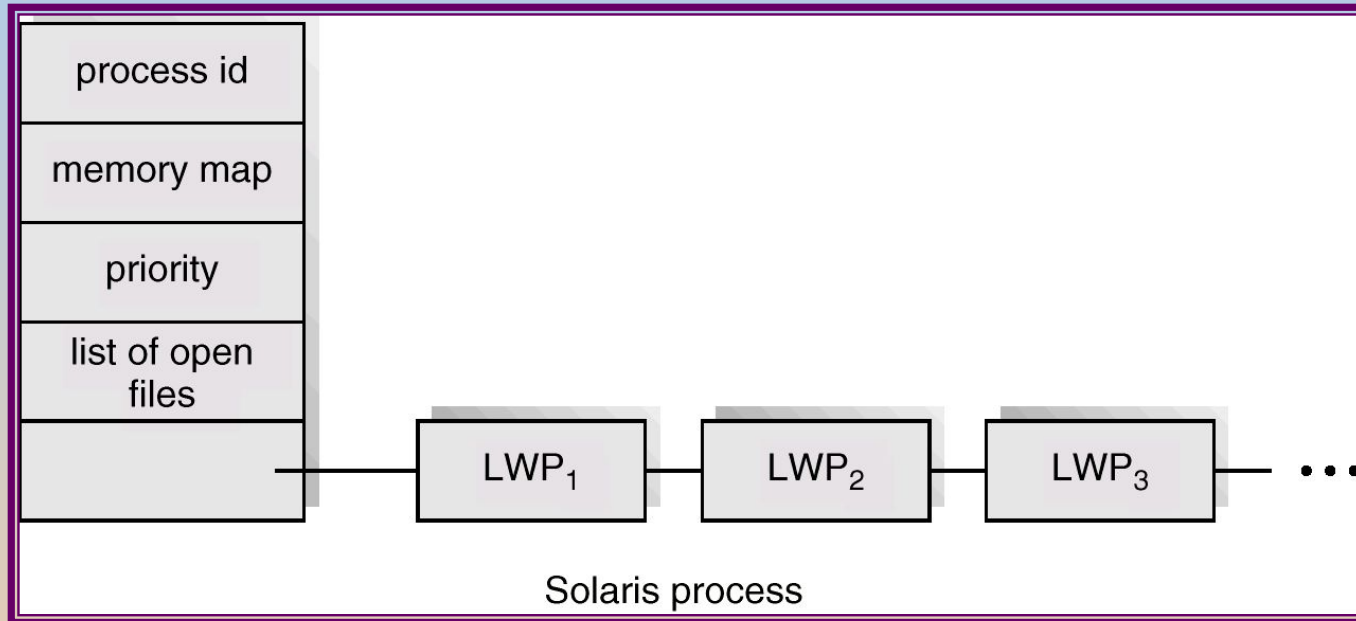- Thread specific data

Operating System

Pthreads

- a POSIX standard (IEEE 1003.1c) API for thread creation and synchronization.

- API specifies behavior of the thread library, implementation is up to development of the library.

- Common in UNIX operating systems.

Operating System

Solaris 2 Threads

Operating System

Solaris Process

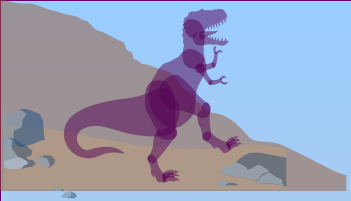Solaris process

Operating System
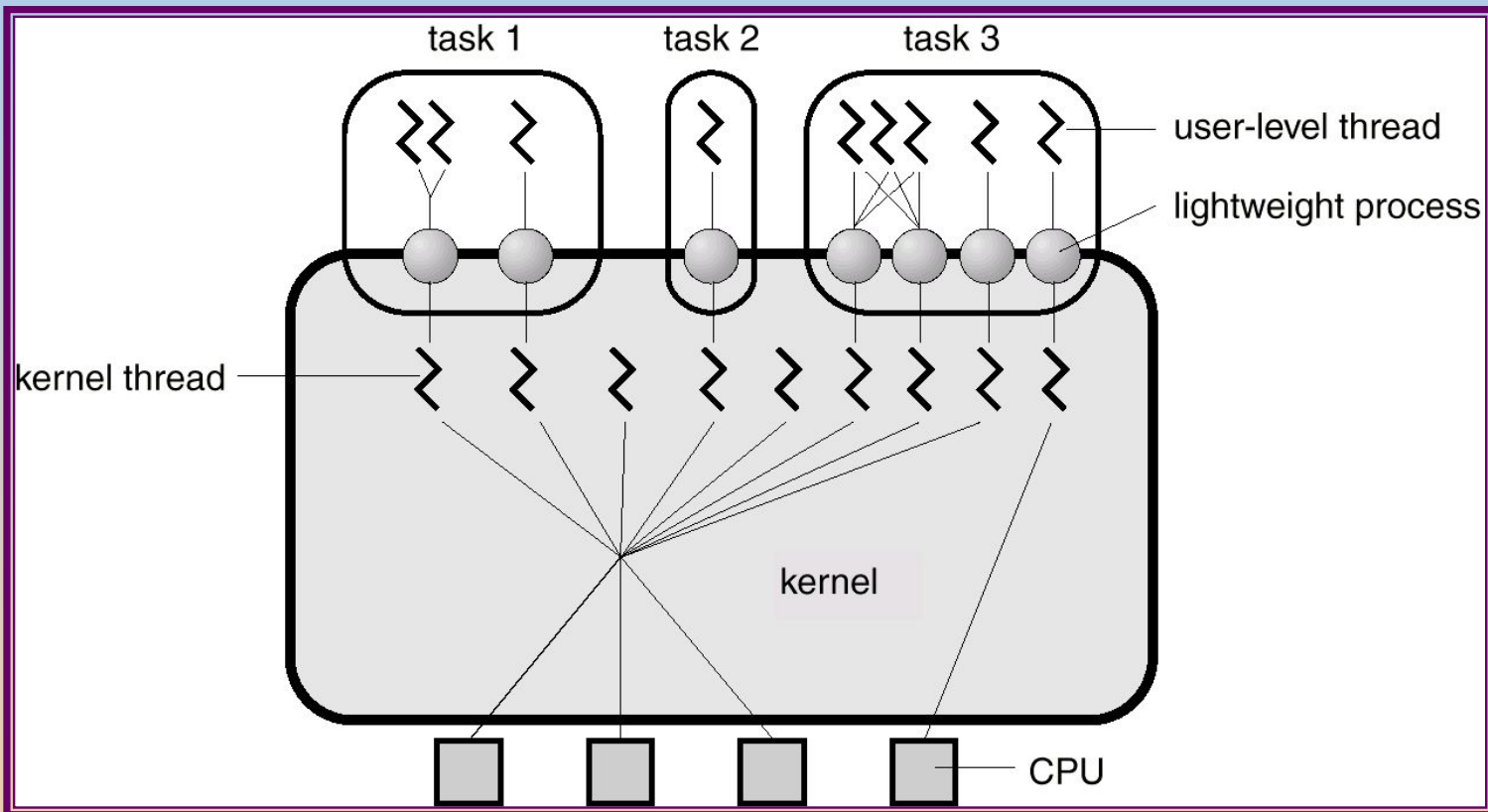
Windows 2000 Threads

- Implements the one-to-one mapping.
- Each thread contains

  - a thread id

  - register set

  - separate user and kernel stacks

  - private data storage area

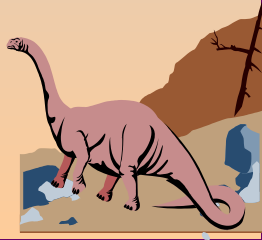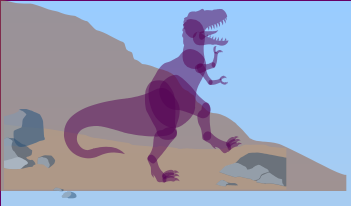Silberschatz, Galvin and Gagne ©2002

Operating System

Linux Threads

- Linux refers to them as *tasks* rather than *threads*.

- Thread creation is done through clone() system call.

- Clone() allows a child task to share the address space of the parent task (process)

Silberschatz, Galvin and Gagne ©2002

Operating System
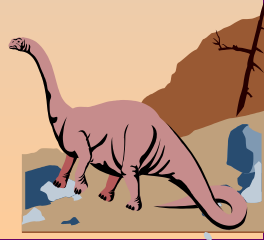
Java Threads

- Java threads may be created by:

  - Extending Thread class
  - Implementing the Runnable interface

- Java threads are managed by the JVM.

Operating System

Java Thread States

Operating System