2020春

LINUX OPERATING SYSTEM

YANG

LINUX操作系统（双语）

双语课→课件内容中英混排

Lecture 5

Threads

本讲内容

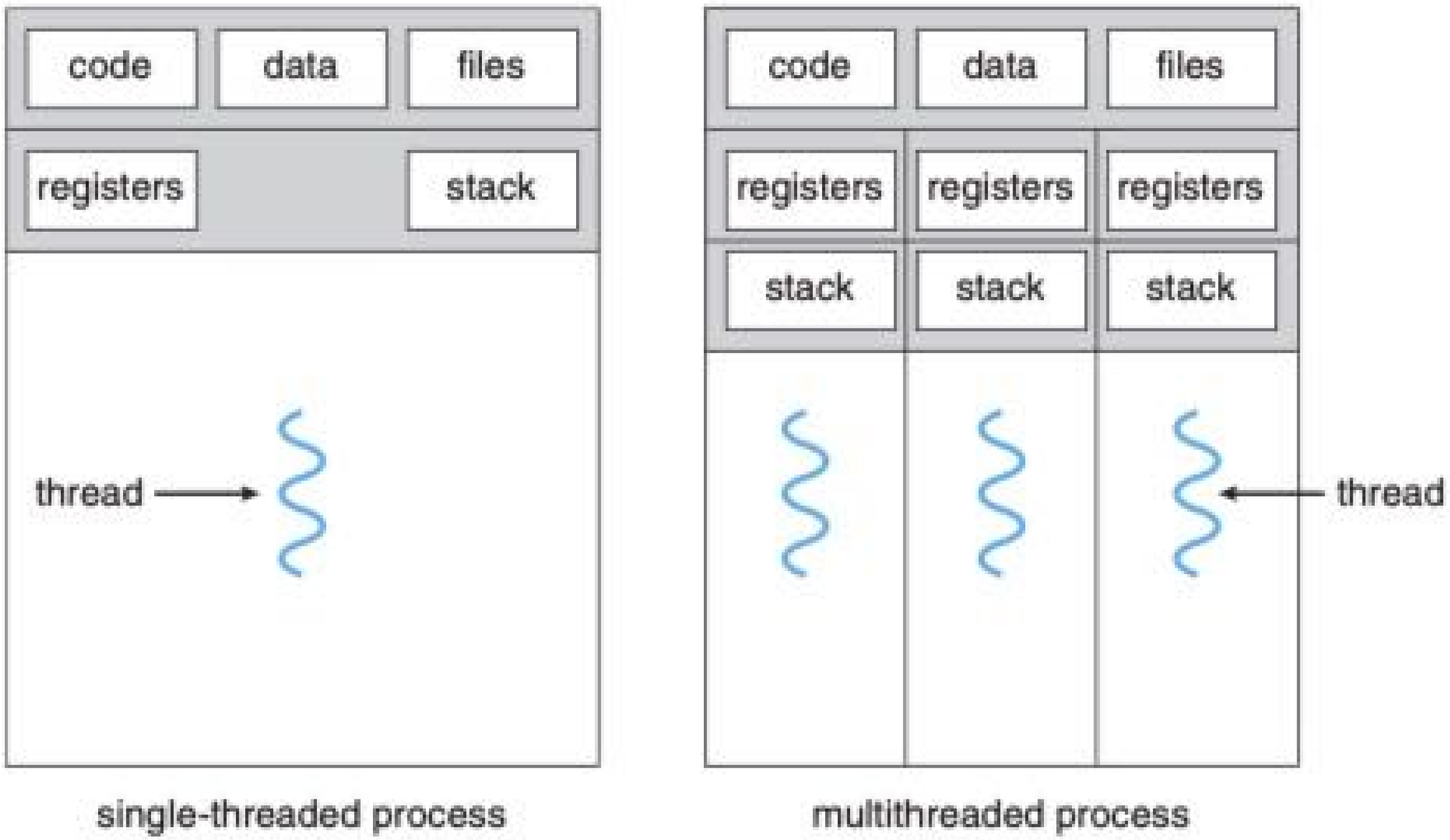
- 线程定义
- 多线程模型

线程定义

动机

- 一个应用通常需要同时处理很多工作，比如一个Web浏览器，可能需要同时处理文字和图片，这些同时执行的任务可称为“执行流”，我们不希望它们是顺序执行的。
- 早期，每个执行流都要创建一个进程来实现，但是进程的创建需要消耗大量的时间和资源。
- 现在，和一个应用相关的所有执行任务都装在一个进程里，这些进程内部的执行任务就是“线程”（Thread）。

THREAD

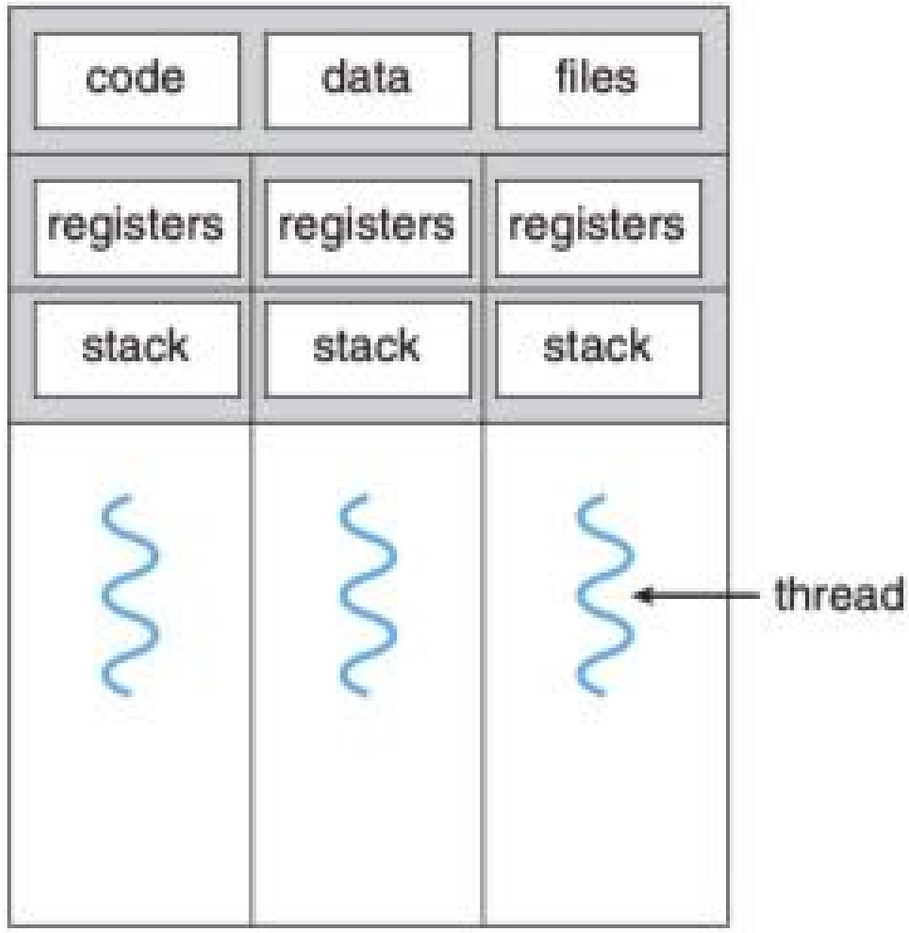


single-threaded process

multithreaded process

采用多线程的优点

- 响应性
- 资源共享
- 经济
- 可伸缩性



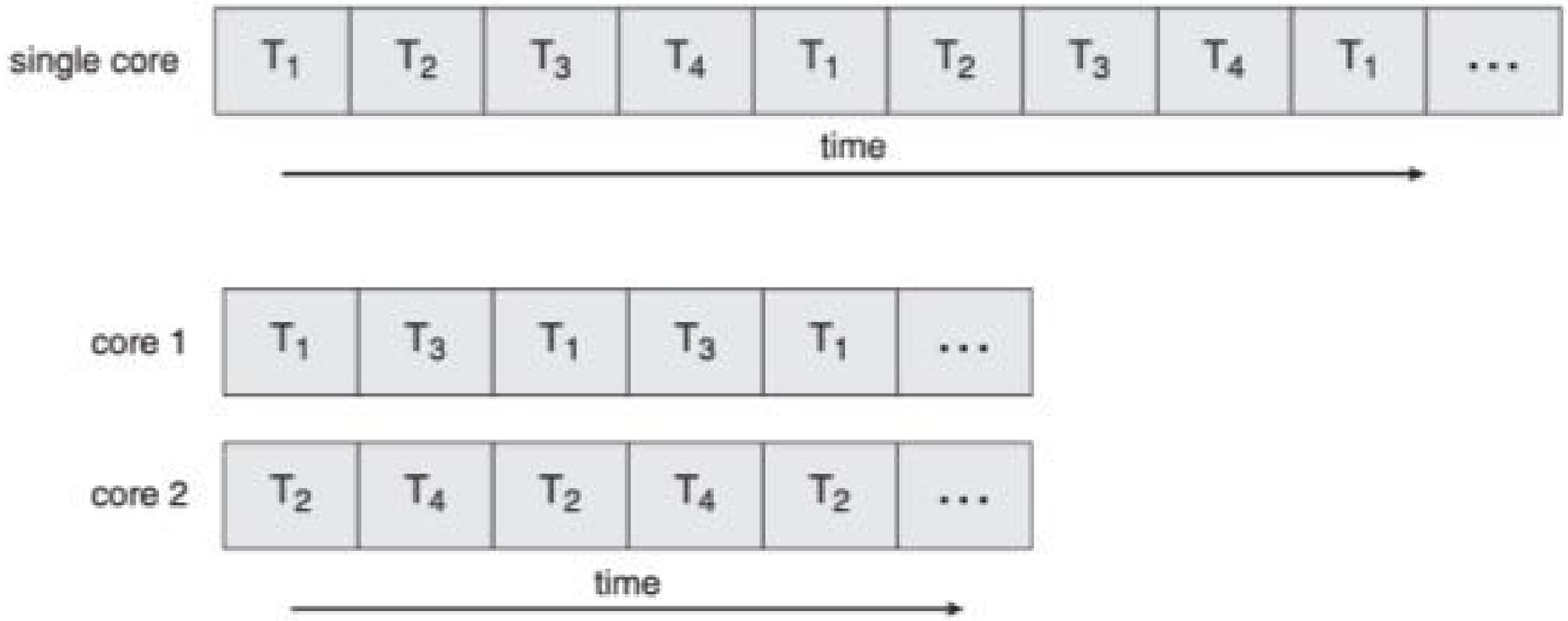
DEFINITION OF THREAD

- A thread is a basic unit of CPU utilization; it comprises a thread id, a program counter, a register set, and a stack.
- It shares with other threads belonging to the same process its code section, data section, and other operating-system resources, such as open files and signals.
- A traditional (or heavyweight) process has a single thread of control. If a process has multiple threads of control, it can perform more than one task at a time.

多线程模型

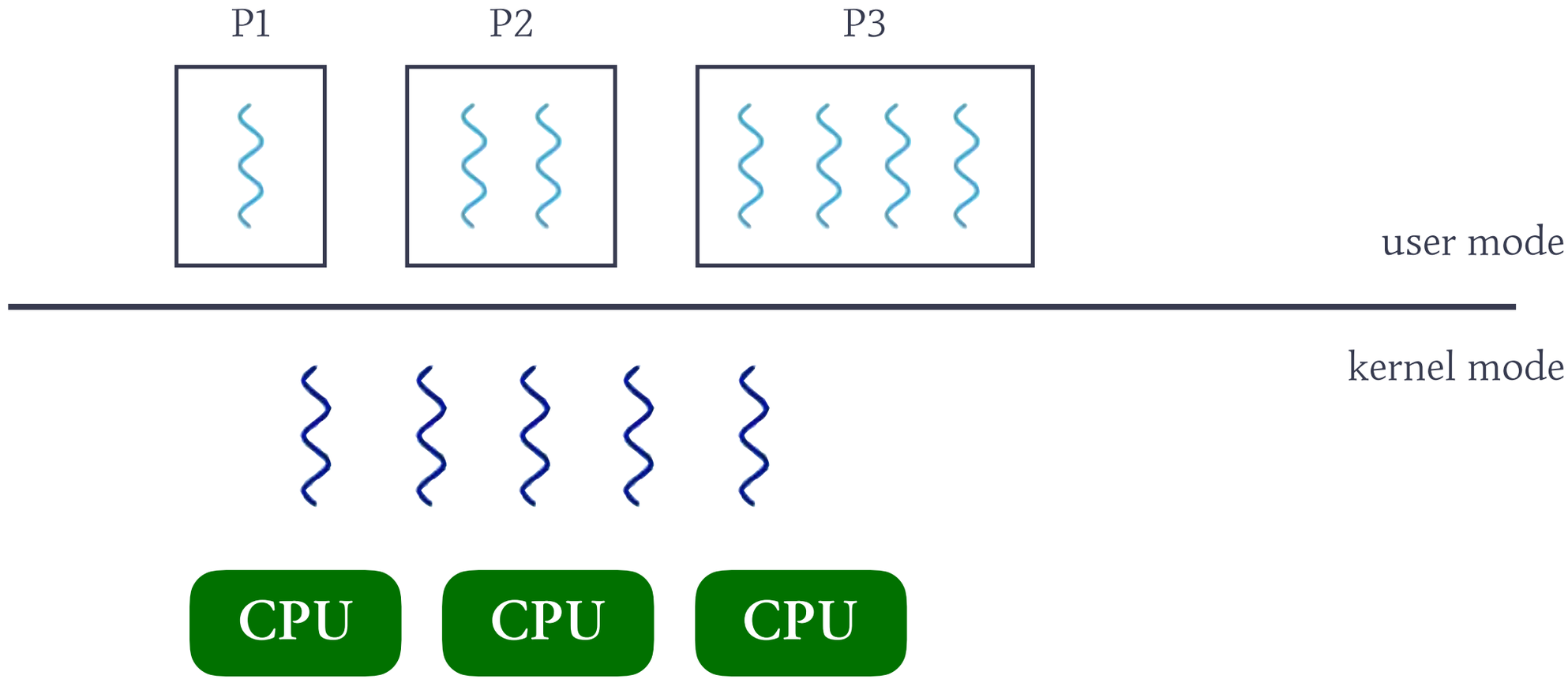
多核编程

- 在多处理器系统中，多核编程机制让应用程序可以更有效地将自身的多个执行任务（并发的线程）分散到不同的处理器上运行，以实现并行计算。

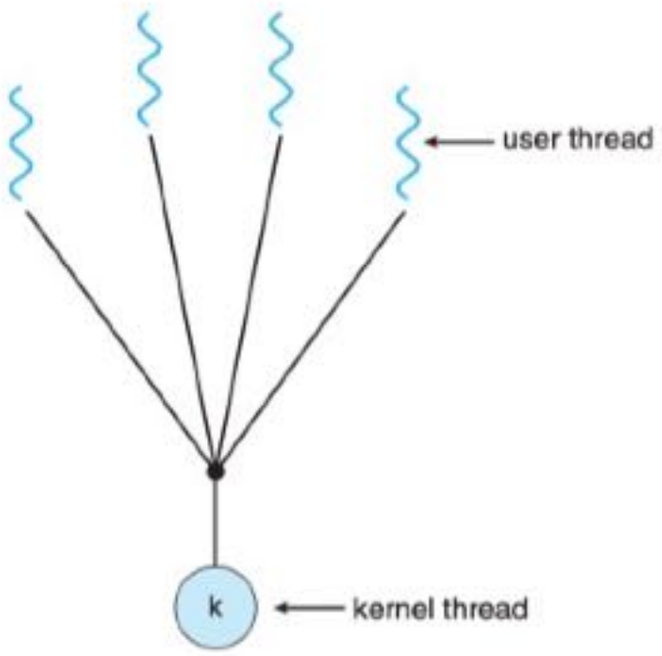


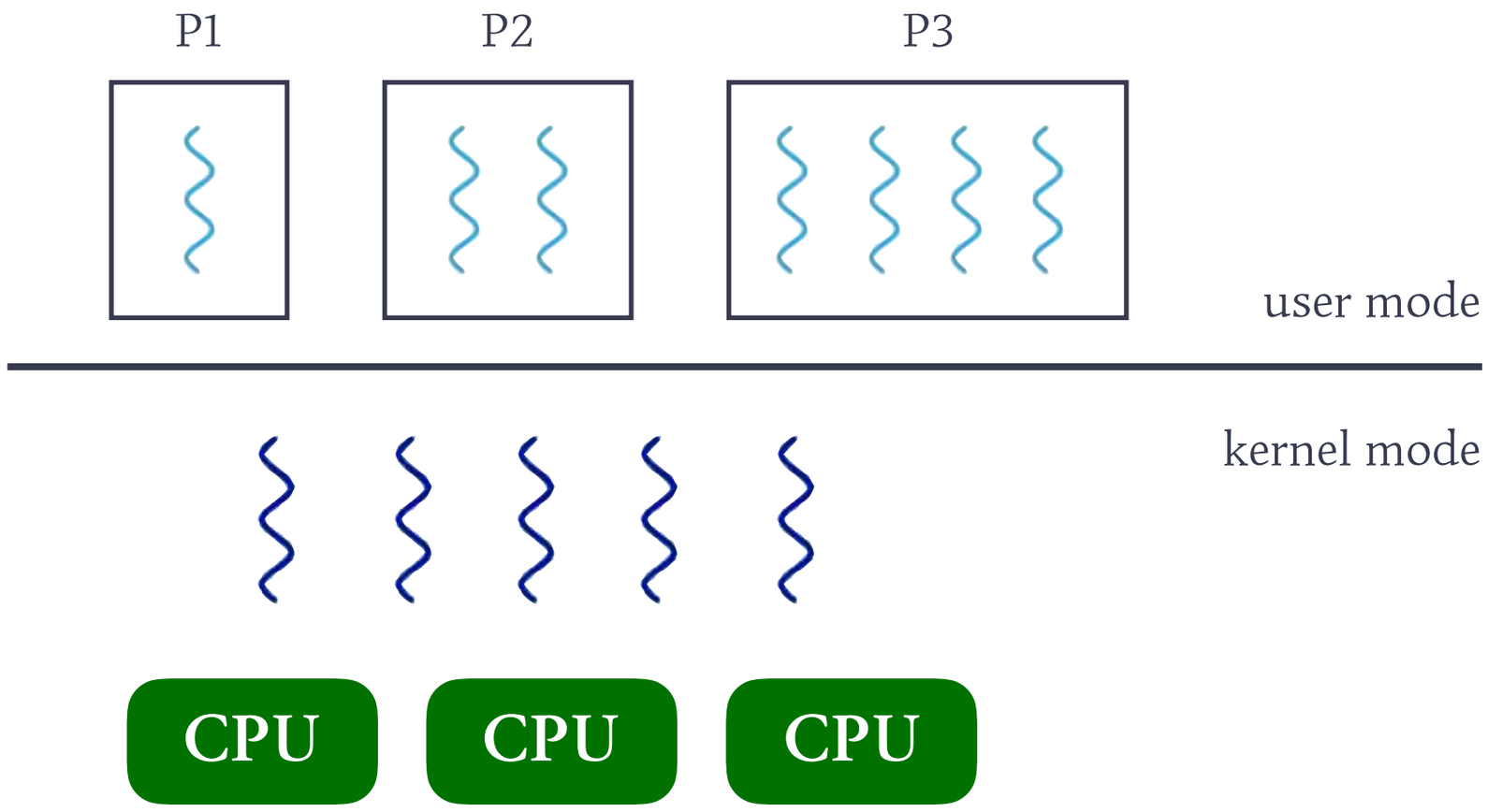
多线程模型

- 用户线程ULT（User Level Thread）
  - ULT在user mode下运行，它的管理无需内核支持。
- 内核线程KLT（Kernel Level Thread）
  - KLT在kernel mode下运行，由操作系统支持与管理。

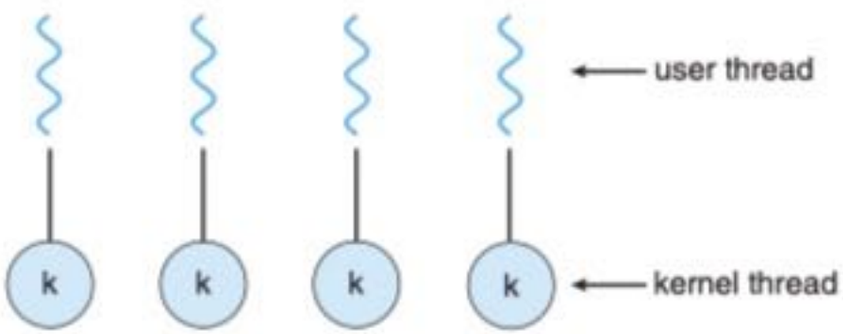


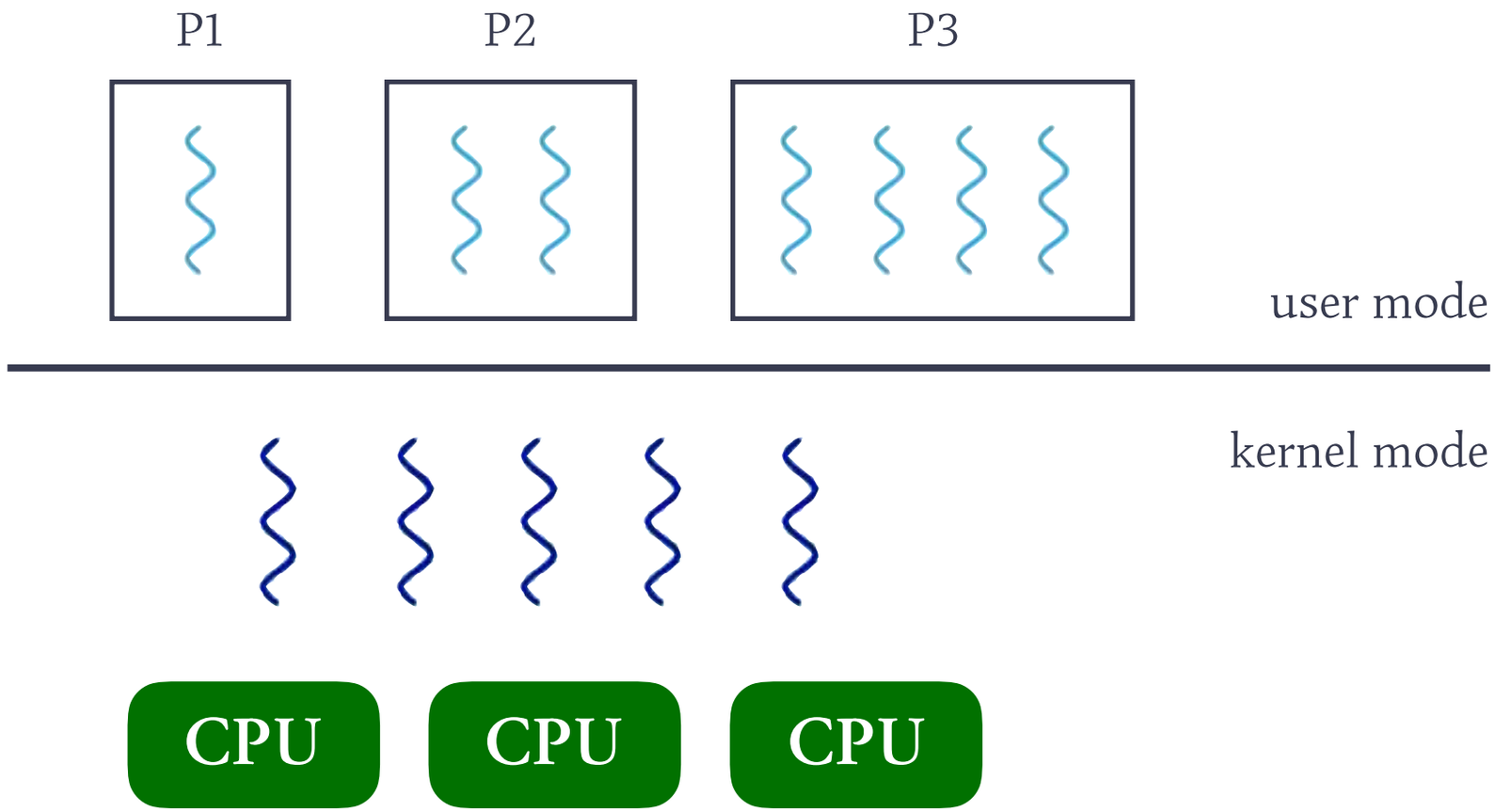
M:1模型


user thread
k
kernel thread

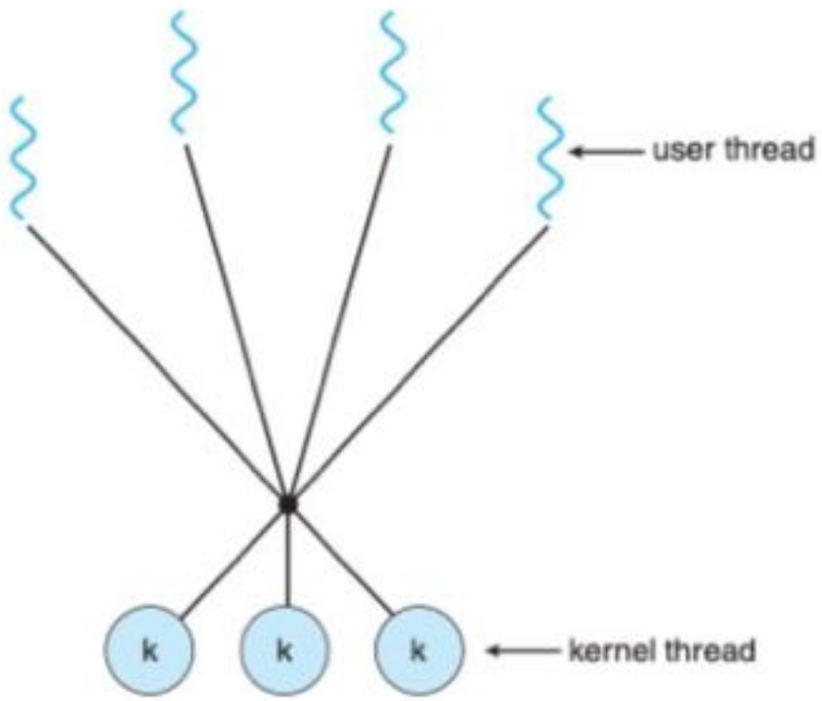

P1
P2
P3
user mode
kernel mode
CPU
CPU
CPU

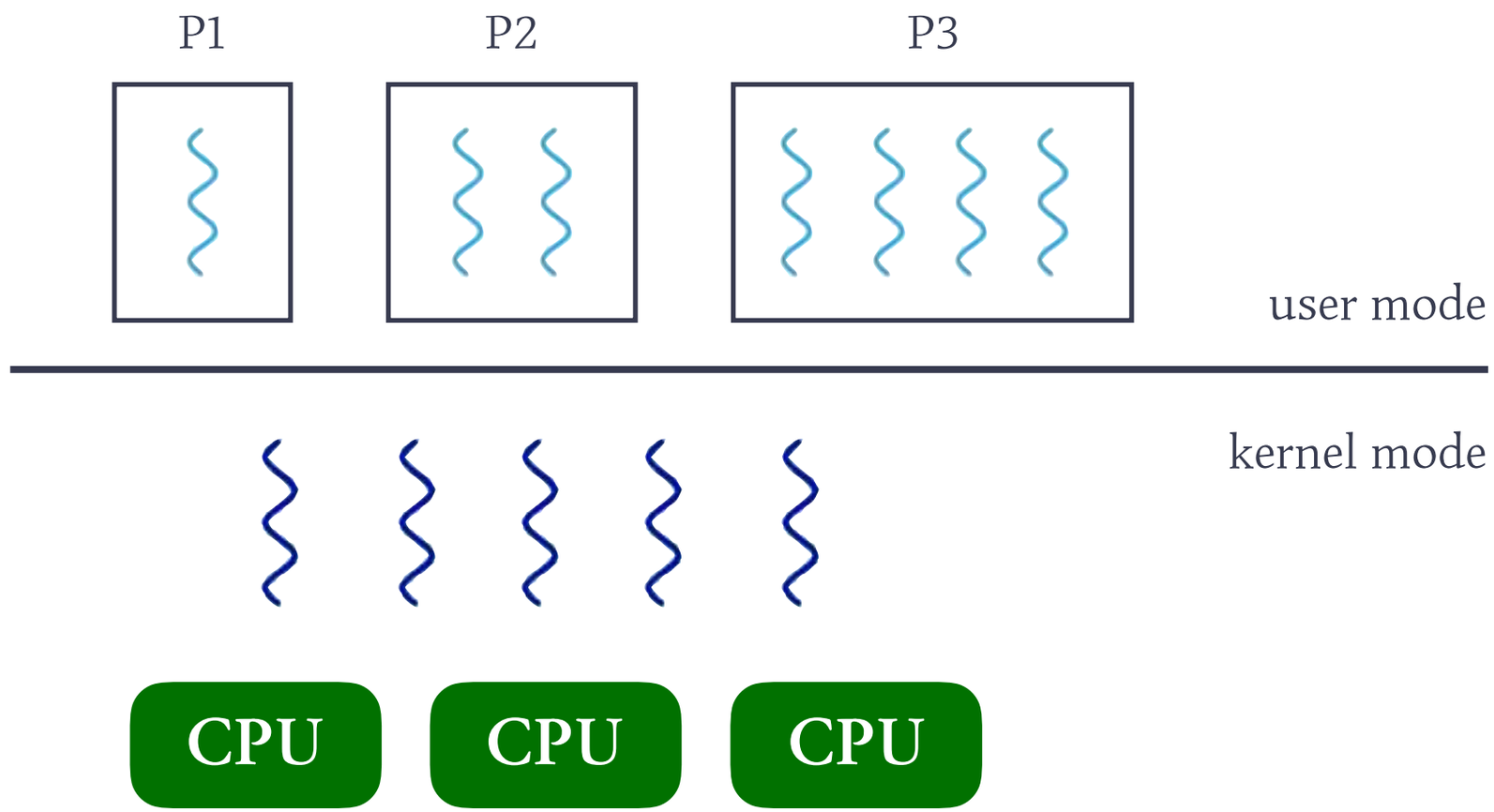
1:1模型


user thread
k
k
k
k
kernel thread


P1
P2
P3
user mode
kernel mode
CPU
CPU
CPU

M:M模型


user thread
k
k
k
kernel thread


P1
P2
P3
user mode
kernel mode
CPU
CPU
CPU

线程库

- Thread Library为程序员提供创建和管理线程的API
  - POSIX Pthreads：用户线程库和内核线程库
  - Windows Threads：内核线程库
  - Java Threads：依据所依赖的操作系统而定

PTHREADS

- Pthreads是POSIX标准定义的线程创建与同步API。不同的操作系统对该标准的实现不尽相同。
- 本节实验所用的API是Linux对Pthreads的实现。

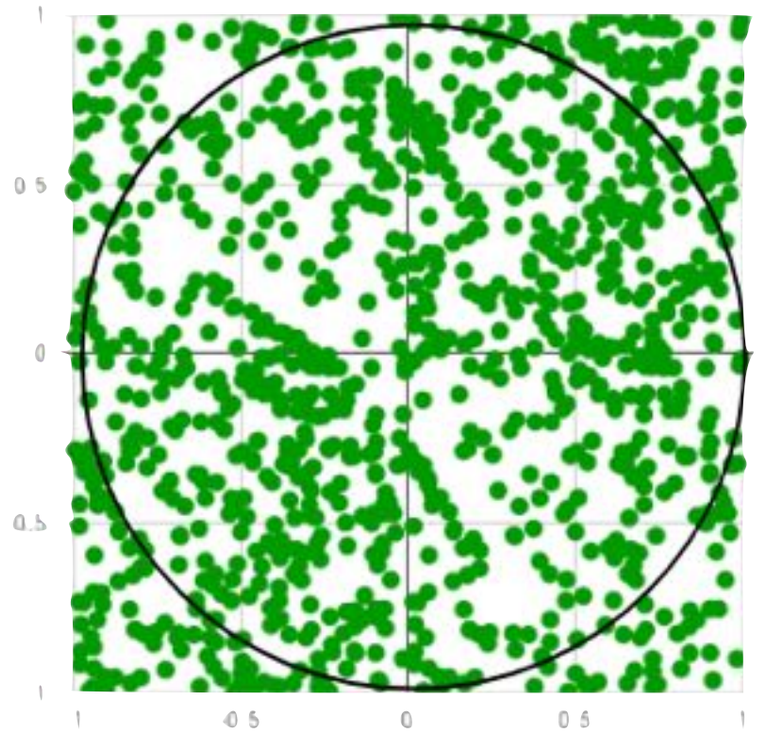
TAKE A BREAK

实验2 多线程实验

实验任务

- 使用Pthreads库创建多个线程，并观察线程的并发执行现象以及数据共享关系。
- Monte Carlo技术计算π值（多线程）：

$$\pi = 4 *(\text{圆内点数})/（\text{总的点数}）$$



Lecture 5

The End

下期预告

- 下次直播时间：2月24日 上午9:30
- 课程内容
  - 处理器调度策略