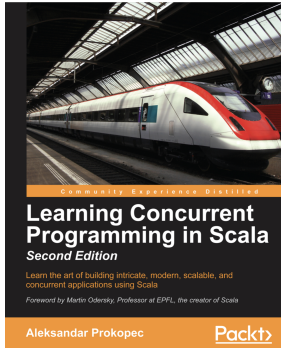


Concurrent Programming in Scala



Concurrent Programming¹

- ▶ Concurrent programming: separate overlapping threads of execution, not necessarily run on separate processors/cores
- ▶ Parallel programming: separate threads of execution running on separate threads/cores
- ▶ Distributed programming: a program running on separate machines

Issues: coordination, communication

¹<https://www.packtpub.com/eu/application-development/learning-concurrent-programming-scala-second-edition>

Concurrency Fundamentals

Programs are executed by OS in of of two primary ways

- ▶ Cooperative multitasking: processes yield the processor when they don't need it, freeing up processor resources for other processes
- ▶ Preemptive multitasking: OS schedules processes/threads on processors. When a process/thread executes, how much is executed before getting suspended again is under OS control

Modern OSes use preemptive multitasking.

Processes and Threads

- ▶ A process is an instance of a computer program that is being executed.
- ▶ Threads are independent computations occurring in the same process. In a typical operating system, there are many more threads than processors.

JVM Threads

JVM starts a `main` thread for every application.

Application may create and run (technically, schedule for execution) any number of additional threads.

Example code: <https://gitlab.com/cs2340/scala-concurrency>

Thread Communication

- ▶ Each thread has its own stack.
- ▶ A process has a single region of dynamic memory often called “heap” memory. All threads share the heap.

Threads communicate by accessing shared memory.

Monitors and Synchronization

Race condition: one thread is updating shared memory, another is reading shared memory. Should happen in a specific order, but thread execution is governed by operating system.

- ▶ Solution: ensure only one thread can access a particular piece of shared memory and that regions of code accessing that memory execute atomically (don't get split up by operating system).