Thread Pools Solutions

Copyright © James Raynard 2025

Thread Pool Motivation

- What is the motivation for thread pools?
  - Re-using threads avoids the overhead of creating and taking down threads
  - It makes full use of all available cores
  - Avoids scaling problems caused by starting more threads than there are cores

Copyright © James Raynard 2025

Thread Pool Structure

- Explain the basic structure of a thread pool in C++
  - A fixed-size container of thread objects
  - Tasks are represented by callable objects
  - Pending tasks are queued up
  - When a thread becomes idle, it takes the next task from the queue
  - The thread performs the task by invoking the callable object

Copyright © James Raynard 2025

Advantages of Thread Pool

- What are the advantages of using a thread pool?
  - It makes efficient use of system resources
  - No scaling concerns
- What are the disadvantages of using a thread pool?
  - Requires a concurrent queue (not currently provided by the C++ library)
  - Queuing the tasks requires synchronization, which adds overhead
- Which problems are best suited to thread pools?
  - Problems which involve a large number of short, non-blocking tasks

Copyright © James Raynard 2025