GPars - Groovy Parallel Systems

Jon Kerridge

Version 1.0, 2015-10-01

Table of Contents

Let’s Do It in Parallel

Groovy Parallel Programming for Java Developers

Jon Kerridge

School of Computing Edinburgh Napier University, Scotland

email: j.kerridge@napier.ac.uk [Jon Kerridge]

Tel. +44 (0)131 455 2777

Room D30, Merchiston Campus

10 Colinton Road

Edinburgh, Scotland EH10 5DT

Jon Kerridge

- Faculty of Engineering, Computing & Creative Industries
- Job Title: Professor
- School: School of Computing
- Subject Group: Software Engineering
- Edinburgh Napier University, Scotland
- http://www.napier.ac.uk

Introduction

The aim of this book is to show both students and practitioners that concurrent and parallel programming does not need to be as hard as it is often portrayed and in fact is often easier that building the equivalent sequential system. This will be achieved by presenting a set of example systems that demonstrate the underlying principles of parallel system design based upon real world examples. Each chapter will discuss the complete implementation of such a system, rather than presenting fragments of solutions. The approach will therefore be founded in principled engineering rather than a detailed exploration of the scientific underpinning. The science has been explored in many books but these have not demonstrated the engineering aspects of actually designing and building parallel systems.

For the purposes of this book; `Concurrent` means a system built from a set of processes that execute on a single processor. `Parallel` means that more than one processor is used to execute the processes and these communicate over some form of network. Within a parallel system, it is likely that some of the processors will run some processes concurrently.

The book will use, as its underpinning parallel environment, a package called **JCSP** (`Communicating Sequential Processes for Java`) that is available under the **LGPL** software licence from the University of Kent, Canterbury UK. This package implements the `Communicating Sequential Process` concepts developed by Professor Hoare some 25 years ago in a form that makes them easily accessible to the programmer. The book's emphasis is on the engineering of parallel systems using these well-defined concepts without delving into their detailed theoretical aspects. The **JCSP** package essentially hides Java's underlying thread model from the programmer in a manner that allows easy implementation of concurrent and parallel systems. It is immaterial whether a process is executed concurrently or in parallel, the process definition remains the same.

Understanding the principles behind parallel processing is an increasingly important skill with the advent of multi-core processors. Much effort has been made by processor manufacturers to hide the underlying parallel design techniques by providing tools that will take existing code and extract some parallelism from it. This hides the real need to actually design and build parallel systems from the outset. Far too many people have been put off parallel processing because they believe that they have to understand the underlying thread model supplied as part of the language or operating system environment. The goal of the book is to dispel all these misconceptions and show that parallel system can be built quite easily with a very few simple design patterns and that such parallel systems can be easily implemented on a single processor or a collection of networked processors. Furthermore, the advent of multi- core processors means that we can now start to build genuinely parallel systems for the commonest desktop workstations in which we can exploit the inherent parallelism more easily once we have the tools required to place a process on a specific core. The extension to a network of multi-core processors becomes even easier. Equally important is that the same design principles can be used to build mobile systems that permit interactions between mobile devices and fixed services using wireless and Bluetooth technology.

Background

The book results from a module taught during the spring semester to master's students, though the approach would be applicable to senior undergraduates and professional programmers. As part of the module, students were asked to complete a practical portfolio and are included in the book. The source coding for all the examples and for solutions to the practical problems is also available. A set of **PowerPoint** slides is also available for instructors.

Please see org.jcsp.lang in GitHub to test code samples.

Book Index

Redirecting Channels: A Self Monitoring Process Ring

Chapter 20

Mobile Agents and Processes: Process Discovery

Chapter 21

Concluding Remarks: Why Use GroovyParallel and JCSP

Chapter 22

Licencing