WORKING OUTLINE

Title: *Parallel Programming with Visual C++: Design Patterns for Decomposition and Coordination on Multicore Architectures*

Foreword

Preface

Who This Book Is For

Why This Book is Pertinent Now

What You Need To Use This Code

How To Use This Book

Introduction

Parallelism with Control Dependencies Only

Parallelism with Control and Data Dependencies

Dynamic Task Parallelism and Asynchronous Agents

Supporting Material

What Is Not Covered

Goals

Acknowledgments

About the Authors

Chapter 1: Introduction

The Importance of Potential Parallelism

Decomposition, Coordination and Scalable Sharing

Understanding Tasks

Coordinating Tasks

Scalable Sharing of Data

Design Approaches

Selecting the Right Pattern

A Word about Terminology

The Limits of Parallelism

A Few Tips

Exercises

For More Information

Chapter 2: Parallel Loops

The Basics

An Example

Variations

Breaking Out of Loops Early

External Loop Cancellation

Exception Handling

Special Handling of Small Loop Bodies

Controlling the Degree of Parallelism

Using Task Local State in a Loop Body

Anti-Patterns

Step Size Other than One

Hidden Loop Body Dependencies

Small Loop Bodies with Few Iterations

Processor Oversubscription and Undersubscription

Duplicates in the Input Enumeration

Design Notes

Adaptive Partitioning

Adaptive Concurrency

Related Patterns

Exercises

Further Reading

Chapter 3: Parallel Tasks

The Basics

An Example

Variations

Canceling a Task

Handling Exceptions

Waiting for the First Task to Complete

Speculative Execution

Creating Tasks with Custom Scheduling

Anti-Patterns

Variables Captured by Closures

Disposing a Resource Needed by a Task

Avoid Thread Abort

Exercises

Further Reading

Chapter 4: Parallel Aggregation

The Basics

An Example

Variations

Using Parallel Loops for Aggregation

Design Notes

Related Patterns

Exercises

Further Reading

Chapter 5: Futures

The Basics

Futures

Continuations

Example: The Adatum Financial Dashboard

The Business Objects

The Analysis Engine

Loading External Data

Merging

Normalizing

Analysis and Model Creation

Processing Historical Data

Comparing Models

View and View Model

Variations

Canceling Futures and Continuation Tasks

Continue When "At Least One" Antecedent Completes

Removing Bottlenecks

Modifying the Graph at Run Time

Design Notes

Decomposition into Futures and Continuation Tasks

Functional Style

Related Patterns

Pipeline Pattern

Master/Worker Pattern

Dynamic Task Parallelism Pattern

Discrete Event Pattern

Exercises

Further Reading

Chapter 6: Dynamic Task Parallelism

The Basics

An Example

Variations

Parallel While-Not-Empty

Task Chaining with Parent/Child Tasks

Implementation Notes

Design Notes

Exercises

Further Reading

Chapter 7: Asynchronous Agents

The Basics

An Example

Sequential Image Processing

The Image Pipeline using Agents

Performance Characteristics

Variations

Canceling Asynchronous Agents

Handling Exceptions

Load Balancing Using Multiple Producers

Pipelines and Streams

Asynchronous Pipelines

Anti-Patterns

Thread Starvation

Infinite Blocking Collection Waits

Using Other Producer/Consumer Collections

Design Notes

Related Patterns

Exercises

Further Reading

Appendix A: Concurrency Runtime and Scheduling

[Outline TBD]

Tasks and Threads

Task Life Cycle

Relationship Between Data Parallelism and Task Parallelism

Scheduling

The Thread Pool

Decentralized Scheduling Techniques

Work Stealing

Thread Injection

Bypassing the Thread Pool

Appendix B: Debugging and Profiling Parallel Programs

The Parallel Tasks and Parallel Stacks Windows

The Concurrency Visualizer

Visual Patterns

Oversubscription

Lock Contention and Serialization

Load Imbalance

Further Reading

Appendix C: Technology Roadmap

Further Reading

Glossary

References

Other Online Sources

Index