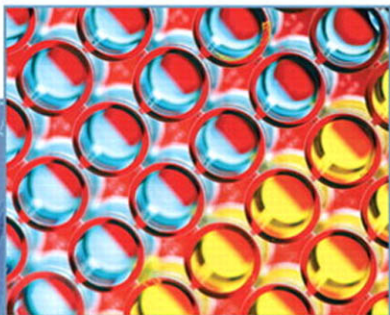




PATTERNS FOR PARALLEL PROGRAMMING



TIMOTHY G. MATTSON
BEVERLY A. SANDERS
BERNA L. MASSINGILL

SOFTWARE PATTERNS SERIES

Patterns for Parallel Programming

Patterns for Parallel Programming

Table of Contents

Contents

Preface

1 A Pattern Language for Parallel Programming

1.1 Introduction

1.2 Parallel Programming

1.3 Design Patterns and Pattern Languages

1.4 A Pattern Language for Parallel Programming

2 Background and Jargon of Parallel Computing

2.1 Concurrency in Parallel Programs Versus Operating Systems

2.2 Parallel Architectures: A Brief Introduction

2.2.1 Flynn's Taxonomy

2.2.2 A Further Breakdown of MIMD

2.2.3 Summary

2.3 Parallel Programming Environments

2.4 The Jargon of Parallel Computing

2.5 A Quantitative Look at Parallel Computation

2.6 Communication

2.6.1 Latency and Bandwidth

2.6.2 Overlapping Communication and Computation and Latency Hiding

2.7 Summary

3 The Finding Concurrency Design Space

3.1 About the Design Space

3.1.1 Overview

Table of Contents

3.1.2 Using the Decomposition Patterns

3.1.3 Background for Examples

3.2 The Task Decomposition Pattern

3.3 The Data Decomposition Pattern

3.4 The Group Tasks Pattern

3.5 The Order Tasks Pattern

3.6 The Data Sharing Pattern

3.7 The Design Evaluation Pattern

3.8 Summary

4 The Algorithm Structure Design Space

4.1 Introduction

4.2 Choosing an Algorithm Structure Pattern

4.2.1 Target Platform

4.2.2 Major Organizing Principle

4.2.3 The Algorithm Structure Decision Tree

4.2.4 Re-evaluation

4.3 Examples

4.3.1 Medical Imaging

4.3.2 Molecular Dynamics

4.4 The Task Parallelism Pattern

4.5 The Divide and Conquer Pattern

4.6 The Geometric Decomposition Pattern

4.7 The Recursive Data Pattern

4.8 The Pipeline Pattern

4.9 The Event-Based Coordination Pattern

5 The Supporting Structures Design Space

5.1 Introduction

5.1.1 Program Structuring Patterns

Table of Contents

5.1.2 Patterns Representing Data Structures

5.2 Forces

5.3 Choosing the Patterns

5.4 The SPMD Pattern

5.5 The Master/Worker Pattern

5.6 The Loop Parallelism Pattern

5.7 The Fork/Join Pattern

5.8 The Shared Data Pattern

5.9 The Shared Queue Pattern

5.10 The Distributed Array Pattern

5.11 Other Supporting Structures

5.11.1 SIMD

5.11.2 MPMD

5.11.3 Client-Server Computing

5.11.4 Concurrent Programming with Declarative Languages

5.11.5 Problem-Solving Environments

6 The Implementation Mechanisms Design Space

6.1 Overview

6.2 UE Management

6.2.1 Thread Creation/Destruction

6.2.2 Process Creation/Destruction

6.3 Synchronization

6.3.1 Memory Synchronization and Fences

6.3.2 Barriers

6.3.3 Mutual Exclusion

6.4 Communication

6.4.1 Message Passing

6.4.2 Collective Communication

6.4.3 Other Communication Constructs

Table of Contents

Appendix A: A Brief Introduction to OpenMP

- A.1 Core Concepts
- A.2 Structured Blocks and Directive Formats
- A.3 Worksharing
- A.4 Data Environment Clauses
- A.5 The OpenMP Runtime Library
- A.6 Synchronization
- A.7 The Schedule Clause
- A.8 The Rest of the Language

Appendix B: A Brief Introduction to MPI

- B.1 Concepts
- B.2 Getting Started
- B.3 Basic Point-to-Point Message Passing
- B.4 Collective Operations
- B.5 Advanced Point-to-Point Message Passing
- B.6 MPI and Fortran
- B.7 Conclusion

Appendix C: A Brief Introduction to Concurrent Programming in Java

- C.1 Creating Threads
- C.2 Atomicity, Memory Synchronization, and the volatile
Keyword
- C.3 Synchronized Blocks
- C.4 Wait and Notify
- C.5 Locks
- C.6 Other Synchronization Mechanisms and Shared Data
Structures

Table of Contents

C.7 Interrupts

Glossary

A

B

C

D

E

F

G

H

I

J

L

M

N

O

P

R

S

T

U

V

W

Bibliography

About the Authors

Index