Foreword by Douglas C. Schmidt, Inventor of ACE

THE ACE PROGRAMMER'S GUIDE

Practical Design Patterns for Network and Systems Programming

Stephen D. Huston • James CE Johnson
Umar Syyid

The ACE Programmer's Guide

ACE Programmer's Guide, The: Practical Design Patterns for Network and Systems Programming, Portable Documents

Table of Contents

Contents

Illustrations

Tables

Foreword

Preface

Part I: ACE Basics

- 1. Introduction to ACE
  - 1.1 A History of ACE
  - 1.2 ACEs Benefits
  - 1.3 ACEs Organization
  - 1.4 Patterns, Class Libraries, and Frameworks
  - 1.5 Porting Your Code to Multiple Operating Systems
  - 1.6 Smoothing the Differences among C++ Compilers
  - 1.7 Using Both Narrow and Wide Characters
  - 1.8 Where to Find More Information and Support
  - 1.9 Summary
- 2. How to Build and Use ACE in Your Programs
  - 2.1 A Note about ACE Versions
  - 2.2 Guide to the ACE Distribution

Table of Contents

2.3 How to Build ACE

2.4 How to Include ACE in Your Applications

2.5 How to Build Your Applications

2.6 Summary

3. Using the ACE Logging Facility

3.1 Basic Logging and Tracing

3.2 Enabling and Disabling Logging Severities

3.3 Customizing the ACE Logging Macros

3.4 Redirecting Logging Output

3.5 Using Callbacks

3.6 The Logging Client and Server Daemons

3.7 The LogManager Class

3.8 Runtime Configuration with the ACE Logging Strategy

3.9 Summary

4. Collecting Runtime Information

4.1 Command Line Arguments and ACE_Get_Opt

4.2 Accessing Configuration Information

4.3 Building Argument Vectors

4.4 Summary

5. ACE Containers

5.1 Container Concepts

5.2 Sequence Containers

5.3 Associative Containers

5.4 Allocators

5.5 Summary

Part II: Interprocess Communication

6. Basic TCP/IP Socket Use

Table of Contents

6.1 A Simple Client

6.2 Adding Robustness to a Client

6.3 Building a Server

6.4 Summary

7. Handling Events and Multiple I/O Streams

7.1 Overview of the Reactor Framework

7.2 Handling Multiple I/O Sources

7.3 Signals

7.4 Notifications

7.5 Timers

7.6 Using the Acceptor-Connector Framework

7.7 Reactor Implementations

7.8 Summary

8. Asynchronous I/O and the ACE Proactor Framework

8.1 Why Use Asynchronous I/O?

8.2 How to Send and Receive Data

8.3 Establishing Connections

8.4 The ACE_Proactor Completion Demultiplexer

8.5 Using Timers

8.6 Other I/O Factory Classes

8.7 Combining the Reactor and Proactor Frameworks

8.8 Summary

9. Other IPC Types

9.1 Interhost IPC with UDP/IP

9.2 Intrahost Communication

9.3 Summary

Part III: Process and Thread Management

Table of Contents

10. Process Management

10.1 Spawning a New Process

10.2 Using the ACE_Process_Manager

10.3 Synchronization Using ACE_Process_Mutex

10.4 Summary

11. Signals

11.1 Using Wrappers

11.2 Event Handlers

11.3 Guarding Critical Sections

11.4 Signal Management with the Reactor

11.5 Summary

12. Basic Multithreaded Programming

12.1 Getting Started

12.2 Basic Thread Safety

12.3 Intertask Communication

12.4 Summary

13. Thread Management

13.1 Types of Threads

13.2 Priorities and Scheduling Classes

13.3 Thread Pools

13.4 Thread Management Using ACE_Thread_Manager

13.5 Signals

13.6 Thread Start-Up Hooks

13.7 Cancellation

13.8 Summary

14. Thread Safety and Synchronization

14.1 Protection Primitives

14.2 Thread Synchronization

Table of Contents

14.3 Thread-Specific Storage

14.4 Summary

15. Active Objects

15.1 The Pattern

15.2 Using the Pattern

15.3 Summary

16. Thread Pools

16.1 Understanding Thread Pools

16.2 Half-Sync/Half-Async Model

16.3 Leader/Followers Model

16.4 Thread Pools and the Reactor

16.5 Summary

Part IV: Advanced ACE

17. Shared Memory

17.1 ACE_Malloc and ACE_Allocator

17.2 Persistence with ACE_Malloc

17.3 Position-Independent Allocation

17.4 ACE_Malloc for Containers

17.5 Wrappers

17.6 Summary

18. ACE Streams Framework

18.1 Overview

18.2 Using a One-Way Stream

18.3 A Bidirectional Stream

18.4 Summary

19. ACE Service Configurator Framework

19.1 Overview

Pearson

Table of Contents

19.2 Configuring Static Services
19.3 Setting Up Dynamic Services
19.4 Setting Up Streams
19.5 Reconfiguring Services During Execution
19.6 Using XML to Configure Services and Streams
19.7 Configuring Services without svc.conf
19.8 Singletons and Services
19.9 Summary
20. Timers
20.1 Timer Concepts
20.2 Timer Queues
20.3 Prebuilt Dispatchers
20.4 Managing Event Handlers
20.5 Summary
21. ACE Naming Service
21.1 The ACE_Naming_Context
21.2 A Single-Process Naming Context: PROC_LOCAL
21.3 Sharing a Naming Context on One Node: NODE_LOCAL
21.4 Sharing a Naming Context across the Network: NET_LOCAL
21.5 Summary
Bibliography
Index

Pearson