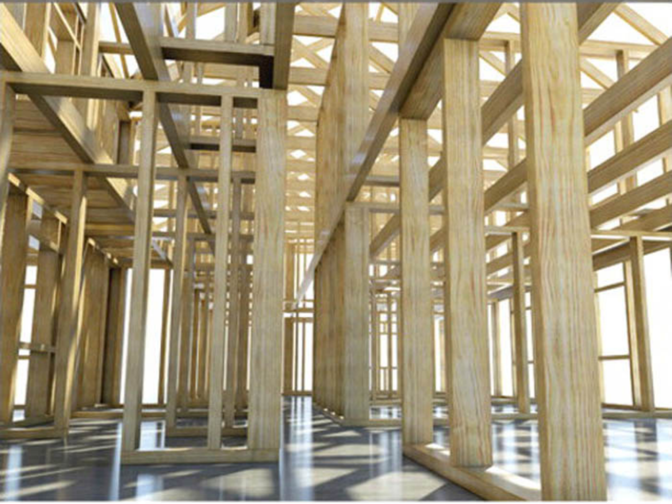
Software Build Systems

Principles and Experience



Peter Smith

Praise for
Software Build Systems

"This book represents a thorough and extensive treatment of the software build process including the choices, benefits, and challenges of a well designed build process. I recommend it not only to all software build engineers but to all software developers since a well designed build process is key to an effective software development process."

—**Kevin Bodie**, Director Software Development, Pitney Bowes Inc.

"An excellent and detailed explanation of build systems, an important but often overlooked part of software development projects. The discussion of productivity as related to build systems is, alone, well worth the time spent reading this book."

—**John M. Pantone**, Objectech Corporation, VP,
IT Educator and Course Developer

"Peter Smith provides an interesting and accessible look into the world of software build systems, distilling years of experience and covering virtually every type of tool in the build engineer's toolbox. Well organized, well written, and very thorough; I would recommend this book to anyone with a build system under their responsibility."

—**Jeff Overbey**, Project Co-Lead, Photran

"*Software Build Systems* teaches how to think about building software. It surveys the tools and techniques for building software products and the ways things go wrong. This book will appeal to those new to build systems as well as experienced build system engineers."

—**Monte Davidoff**, Software Development Consultant,
Alluvial Software, Inc.

Software Build Systems: Principles and Experience

Table of Contents

CONTENTS

PREFACE

Why Do Build Systems Become Complex?

The True Cost of a Build System

The Focus of This Book

Who Should Read This Book?

How This Book Is Organized

Summary

ACKNOWLEDGMENTS

ABOUT THE AUTHOR

PART I: THE BASICS

Chapter 1 BUILD SYSTEM OVERVIEW

What Is a Build System?

Components of a Build System

The Build Process and Build Description

How a Build System Is Used

Build System Quality

Summary

Chapter 2 A MAKE-BASED BUILD SYSTEM

Calculator Example

Creating a Simple Makefile

Simplifying the Makefile

Pearson

Table of Contents

Additional Build Targets

Using a Framework

Summary

Chapter 3 THE RUNTIME VIEW OF A PROGRAM

Executable Programs

Libraries

Configuration and Data Files

Distributed Programs

Summary

Chapter 4 FILE TYPES AND COMPILATION TOOLS

C/C++

Java

C#

Other File Types

Summary

Chapter 5 SUBTARGETS AND BUILD VARIANTS

Building Subtargets

Building Different Editions of the Software

Building Different Target Architectures

Summary

PART II: THE BUILD TOOLS

Chapter 6 MAKE

The GNU Make Programming Language

Real-World Build System Scenarios

Praise and Criticism

Similar Tools

Summary

Chapter 7 ANT

The Ant Programming Language

Table of Contents

Real-World Build System Scenarios

Praise and Criticism

Similar Tools

Summary

Chapter 8 SCons

The SCons Programming Language

Real-World Build System Scenarios

Praise and Criticism

Similar Tools

Summary

Chapter 9 CMAKE

The CMake Programming Language

Real-World Build System Scenarios

Praise and Criticism

Similar Build Tools

Summary

Chapter 10 ECLIPSE

The Eclipse Concepts and GUI

Real-World Build System Scenarios

Praise and Criticism

Similar Build Tools

Summary

PART III: ADVANCED TOPICS

Chapter 11 DEPENDENCIES

The Dependency Graph

The Problem with Bad Dependencies

Step 1: Computing the Dependency Graph

Step 2: Determining Which Files Are Out-of-Date

Step 3: Sequencing the Compilation Steps

Table of Contents

Summary

Chapter 12 BUILDING WITH METADATA

Debugging Support

Profiling Support

Coverage Support

Source Code Documentation

Unit Testing

Static Analysis

Adding Metadata to a Build System

Summary

Chapter 13 SOFTWARE PACKAGING AND INSTALLATION

Archive Files

Package-Management Tools

Custom-Built GUI Installation Tools

Summary

Chapter 14 VERSION MANAGEMENT

What Should Be Version-Controlled

What Should Not Be in the Source Tree

Version Numbering

Summary

Chapter 15 BUILD MACHINES

Native and Cross-Compilation

Centralized Development Environments

Open-Source Development Environments

GNU Autoconf

Summary

Chapter 16 TOOL MANAGEMENT

Rules for Managing Tools

Writing Your Own Compilation Tools

Table of Contents

Summary

PART IV: SCALING UP

Chapter 17 REDUCING COMPLEXITY FOR END USERS

Build Frameworks

Reasons to Avoid Supporting Multiple Variants

Various Ways to Reduce Complexity

Scheduling and Staffing Build System Changes

Summary

Chapter 18 MANAGING BUILD SIZE

The Problem with Monolithic Builds

Component-Based Software

People and Process Management

Apache Ivy

Chapter 19 FASTER BUILDS

Measuring Build System Performance

Build Avoidance: Eliminating Unnecessary Rebuilds

Parallelism

Reducing Disk Usage

Summary

REFERENCES

INDEX