

A Practitioner's Guide to Software Test Design



Lee Copeland



A Practitioner's Guide to Software Test Design

by Lee Copeland

ISBN:158053791x

Artech House © 2004

This text presents all the important test design techniques in a single place and in a consistent, and easy-to-digest format. It enables you to choose the best test case design, find software defects, develop optimal strategies, and more.

Table of Contents

[A Practitioner's Guide to Software Test Design](#)

[Preface](#)

[Chapter 1](#) - The Testing Process

[Chapter 2](#) - Case Studies

Section I - Black Box Testing Techniques

[Chapter 3](#) - Equivalence Class Testing

[Chapter 4](#) - Boundary Value Testing

[Chapter 5](#) - Decision Table Testing

[Chapter 6](#) - Pairwise Testing

[Chapter 7](#) - State-Transition Testing

[Chapter 8](#) - Domain Analysis Testing

[Chapter 9](#) - Use Case Testing

Section II - White Box Testing Techniques

[Chapter 10](#) - Control Flow Testing

[Chapter 11](#) - Data Flow Testing

Section III - Testing Paradigms

[Chapter 12](#) - Scripted Testing

[Chapter 13](#) - Exploratory Testing

[Chapter 14](#) - Test Planning

Section IV - Supporting Technologies

[Chapter 15](#) - Defect Taxonomies

[Chapter 16](#) - When to Stop Testing

Section V - Some Final Thoughts

[Appendix A](#) - Brown & Donaldson Case Study

[Appendix B](#) - Stateless University Registration
System Case Study

[Bibliography](#)

[Index](#)

[List of Figures](#)

[List of Tables](#)

[List of Examples](#)

Back Cover

Here's a comprehensive, up-to-date and practical introduction to software test design. This invaluable book presents all the important test design techniques in a single place and in a consistent, and easy-to-digest format. An immediately useful handbook for test engineers, developers, quality assurance professionals, and requirements and systems analysts, it enables you to: choose the best test case design, find software defects in less time and with fewer resources, and develop optimal strategies that help reduce the likelihood of costly errors. It also assists you in estimating the effort, time and cost of good testing.

Numerous case studies and examples of software testing techniques are included, helping you to fully understand the practical applications of these techniques. From well-established techniques such as equivalence classes, boundary value analysis, decision tables, and state-transition diagrams, to new techniques like use case testing, pairwise testing, and exploratory testing, the book is an indispensable resource for testing professionals seeking to improve their skills and an excellent reference for college-level courses in software test design.

About the Author

Lee Copeland is an internationally known consultant in software testing, with over 30 years of experience as an information systems professional. He has held a number of technical and managerial positions with commercial and nonprofit organizations in the areas of software development, testing, and process improvement. He has taught seminars and consulted extensively throughout the United States and internationally.

A Practitioner's Guide to Software Test Design

Lee Copeland



Artech House Publishers
Boston • London

Library of Congress and British CIP information available on request

685 Canton Street
Norwood, MA 02062
(781) 769-9750
www.artechhouse.com

46 Gillingham Street
London SW1V 1AH
+44 (0)20 7596-8750

Copyright © 2004 STQE Publishing

All rights reserved. No part of this book shall be reproduced, stored in a retrieval system, or transmitted by any means, electronic, mechanical, photocopying, recording, or otherwise without written permission from the publisher.

International Standard Book Number: 1-58053-791-X

Printed in the United States of America

First Printing: November 2003

Trademarks

All terms mentioned in this book that are known to be trademarks or service marks have been appropriately capitalized. Artech House

Publishers and STQE Publishing cannot attest to the accuracy of this information. Use of a term in this book should not be regarded as affecting the validity of any trademark or service mark.

Warning and Disclaimer

Every effort has been made to make this book as complete and accurate as possible, but no warranty or fitness is implied. The information provided is on an "as is" basis. The authors and the publisher shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this book.

Dedication

To my wife Suzanne, and our wonderful children and grandchildren

Shawn and Martha

Andrew and Cassandra

David

Cathleen

Katelynn and Kiley

Melissa and Jay

Ross, Elizabeth, and Miranda

Brian and Heather

Cassidy and Caden

Thomas and Jeni

Carrie

Sundari

Rajan

and to Wayne, Jerry, Dani, Ron, and Rayanne for their encouragement over the years.



Lee Copeland is an internationally known consultant in software testing, with over 30 years of experience as an information systems professional. He has held a number of technical and managerial positions with commercial and nonprofit organizations in the areas of software development, testing, and process improvement. He has taught seminars and consulted extensively throughout the United States and internationally.

As a consultant for Software Quality Engineering, Lee travels the world promoting effective software testing to his clients. In addition, he is the program chair for STAREAST and STARWEST, the world's premier conferences on software testing.

Preface

A Practitioner's Guide to Software Test Design contains today's important current test design approaches in one unique book. Until now, software testers had to search through a number of books, periodicals, and Web sites to locate this vital information.

Importance of Test Design

"The act of careful, complete, systematic, test design will catch as many bugs as the act of testing. ... Personally, I believe that it's far more effective."

- Boris Beizer

The book focuses only on software test design, not related subjects such as test planning, test management, test team development, etc. While those are important in software testing, they have often overshadowed what testers really need—the more practical aspects of testing, specifically test case design. Other excellent books can guide you through the overall process of software testing. One of my favorites is *Systematic Software Testing* by Rick Craig and Stefan Jaskiel.

A Practitioner's Guide to Software Test Design illustrates each test design approach through detailed examples and step-by-step instructions. These lead the reader to a clear understanding of each test design technique.

Today's Testing Challenges

For any system of interesting size it is impossible to test all the different logic paths and all the different input data combinations. Of the infinite number of choices, each one of which is worthy of some level of testing, testers can only choose a very small subset because of resource constraints. The purpose of this book is to help you analyze, design, and choose such subsets, to implement those tests that are most likely to discover defects.

It is vital to choose test cases wisely. Missing a defect can result in significant losses to your organization if a defective system is placed into production.

A Practitioner's Guide to Software Test Design describes a set of key test design strategies that improve both the efficiency and effectiveness of software testers.