

Working Effectively with Legacy Code

Michael C. Feathers



Prentice Hall Professional Technical Reference
Upper Saddle River, NJ 07458
www.phptr.com

Contents

Foreword by Robert C. Martin.	xv
Preface	xv
Introduction	xxi
PART I: The Mechanics of Change.	1
Chapter 1: Changing Software	3
Four Reasons to Change Software	4
Risky Change	7
Chapter 2: Working with Feedback	9
What Is Unit Testing?	12
Higher-Level Testing	14
Test Coverings	14
The Legacy Code Change Algorithm	18
Chapter 3: Sensing and Separation	21
Faking Collaborators	23
Chapter 4: The Seam Model.	29
A Huge Sheet of Text	29
Seams	30
Seam Types	33
Chapter 5: Tools.	45
Automated Refactoring Tools	45
Mock Objects	47
Unit-Testing Harnesses	48
General Test Harnesses	53

PART II: Changing Software	55
Chapter 6: I Don't Have Much Time and I Have to Change It.	57
Sprout Method	59
Sprout Class	63
Wrap Method	67
Wrap Class	71
Summary	76
Chapter 7: It Takes Forever to Make a Change	77
Understanding	77
Lag Time	78
Breaking Dependencies	79
Summary	85
Chapter 8: How Do I Add a Feature?	87
Test-Driven Development (TDD)	88
Programming by Difference	94
Summary	104
Chapter 9: I Can't Get This Class into a Test Harness	105
The Case of the Irritating Parameter	106
The Case of the Hidden Dependency	113
The Case of the Construction Blob	116
The Case of the Irritating Global Dependency	118
The Case of the Horrible Include Dependencies	127
The Case of the Onion Parameter	130
The Case of the Aliased Parameter	133
Chapter 10: I Can't Run This Method in a Test Harness	137
The Case of the Hidden Method	138
The Case of the "Helpful" Language Feature	141
The Case of the Undetectable Side Effect	144
Chapter 11: I Need to Make a Change. What Methods Should I Test? ...	151
Reasoning About Effects	151
Reasoning Forward	157
Effect Propagation	163
Tools for Effect Reasoning	165
Learning from Effect Analysis	167
Simplifying Effect Sketches	168

Chapter 12: I Need to Make Many Changes in One Area.....	173
Interception Points	174
Judging Design with Pinch Points	182
Pinch Point Traps	184
Chapter 13: I Need to Make a Change, but I Don't Know What Tests to Write	185
Characterization Tests	186
Characterizing Classes	189
Targeted Testing	190
A Heuristic for Writing Characterization Tests	195
Chapter 14: Dependencies on Libraries Are Killing Me	197
Chapter 15: My Application Is All API Calls	199
Chapter 16: I Don't Understand the Code Well Enough to Change It	209
Notes/Sketching	210
Listing Markup	211
Scratch Refactoring	212
Delete Unused Code	213
Chapter 17: My Application Has No Structure	215
Telling the Story of the System	216
Naked CRC	220
Conversation Scrutiny	224
Chapter 18: My Test Code Is in the Way	227
Class Naming Conventions	227
Test Location	228
Chapter 19: My Project Is Not Object Oriented. How Do I Make Safe Changes?.....	231
An Easy Case	232
A Hard Case	232
Adding New Behavior	236
Taking Advantage of Object Orientation	239
It's All Object Oriented	242
Chapter 20: This Class Is Too Big and I Don't Want It to Get Any Bigger .	245
Seeing Responsibilities	249

Other Techniques	265
Moving Forward	265
After Extract Class	268
Chapter 21: I'm Changing the Same Code All Over the Place	269
First Steps	272
Chapter 22: I Need to Change a Monster Method	
and I Can't Write Tests for It	289
Varieties of Monsters	290
Tackling Monsters with Automated Refactoring Support	294
The Manual Refactoring Challenge	297
Strategy	304
Chapter 23: How Do I Know That I'm Not Breaking Anything?	309
Hyperaware Editing	310
Single-Goal Editing	311
Preserve Signatures	312
Lean on the Compiler	315
Chapter 24: We Feel Overwhelmed. It Isn't Going to Get Any Better.	319
PART III: Dependency-Breaking Techniques	323
Chapter 25: Dependency-Breaking Techniques	325
Adapt Parameter	326
Break Out Method Object	330
Definition Completion	337
Encapsulate Global References	339
Expose Static Method	345
Extract and Override Call	348
Extract and Override Factory Method	350
Extract and Override Getter	352
Extract Implementer	356
Extract Interface	362
Introduce Instance Delegator	369
Introduce Static Setter	372
Link Substitution	377
Parameterize Constructor	379
Parameterize Method	383

Primitivize Parameter	385
Pull Up Feature	388
Push Down Dependency	392
Replace Function with Function Pointer	396
Replace Global Reference with Getter	399
Subclass and Override Method	401
Supersede Instance Variable	404
Template Redefinition	408
Text Redefinition	412
Appendix: Refactoring	415
Extract Method	415
Glossary	421
Index	423