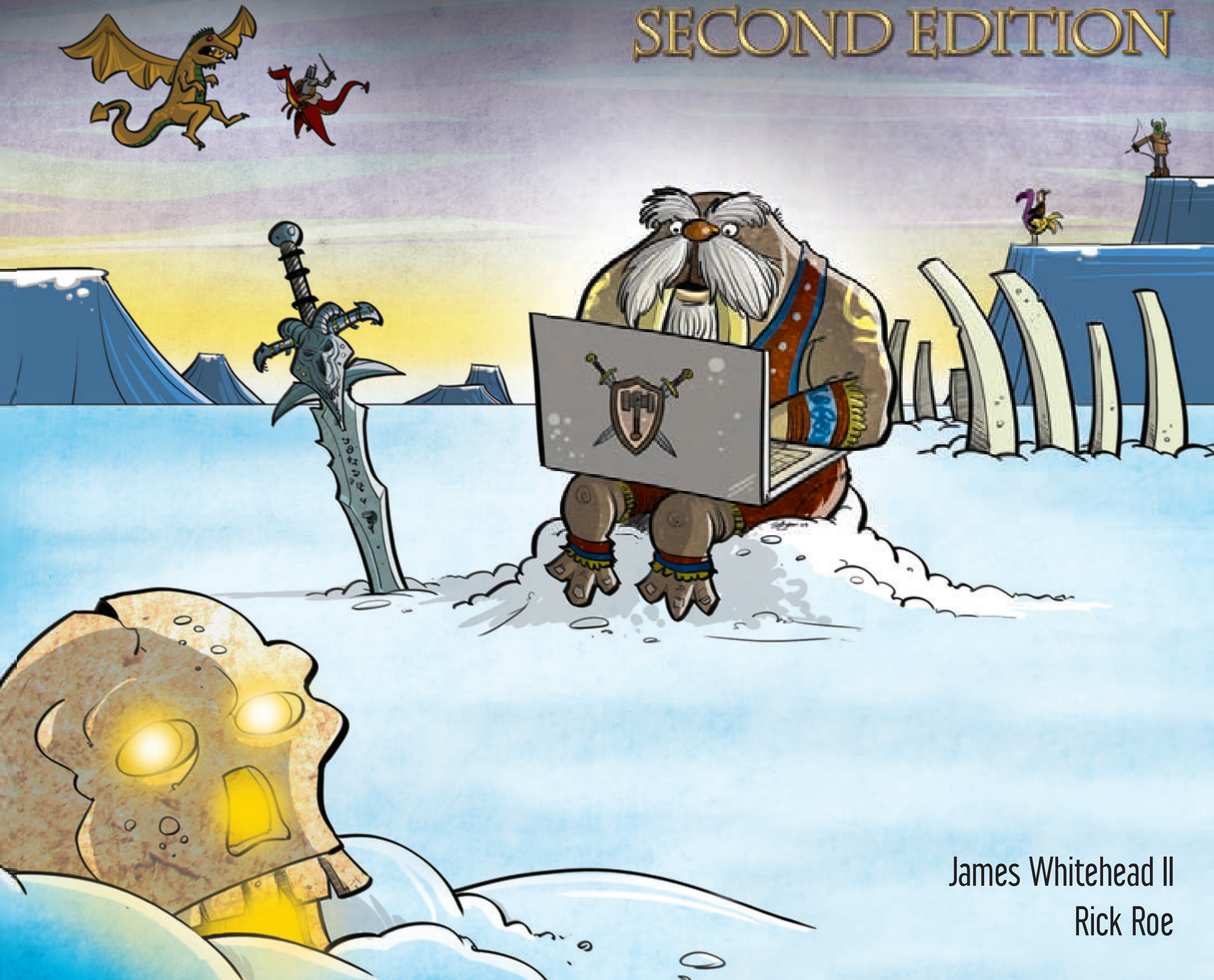
A Guide and Reference for Creating WoW Addons

World of Warcraft® Programming

Second Edition



James Whitehead II

Rick Roe

World of Warcraft®
Programming

World of Warcraft®
Programming

A Guide and Reference for Creating WoW Addons

Second Edition

James Whitehead II
Rick Roe

WILEY
Wiley Publishing, Inc.

World of Warcraft® Programming: A Guide and Reference for Creating WoW Addons, Second Edition

Published by
Wiley Publishing, Inc.
10475 Crosspoint Boulevard
Indianapolis, IN 46256
www.wiley.com

Copyright © 2010 by Wiley Publishing, Inc., Indianapolis, Indiana

Published simultaneously in Canada

ISBN: 978-0-470-48128-8

Manufactured in the United States of America

10 9 8 7 6 5 4 3 2 1

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, except as permitted under Sections 107 or 108 of the 1976 United States Copyright Act, without either the prior written permission of the Publisher, or authorization through payment of the appropriate per-copy fee to the Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923, (978) 750-8400, fax (978) 646-8600. Requests to the Publisher for permission should be addressed to the Permissions Department, John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, (201) 748-6011, fax (201) 748-6008, or online at http://www.wiley.com/go/permissions.

Limit of Liability/Disclaimer of Warranty: The publisher and the author make no representations or warranties with respect to the accuracy or completeness of the contents of this work and specifically disclaim all warranties, including without limitation warranties of fitness for a particular purpose. No warranty may be created or extended by sales or promotional materials. The advice and strategies contained herein may not be suitable for every situation. This work is sold with the understanding that the publisher is not engaged in rendering legal, accounting, or other professional services. If professional assistance is required, the services of a competent professional person should be sought. Neither the publisher nor the author shall be liable for damages arising herefrom. The fact that an organization or Web site is referred to in this work as a citation and/or a potential source of further information does not mean that the author or the publisher endorses the information the organization or Web site may provide or recommendations it may make. Further, readers should be aware that Internet Web sites listed in this work may have changed or disappeared between when this work was written and when it is read.

For general information on our other products and services please contact our Customer Care Department within the United States at (877) 762-2974, outside the United States at (317) 572-3993 or fax (317) 572-4002.

Wiley also publishes its books in a variety of electronic formats. Some content that appears in print may not be available in electronic books.

Library of Congress Control Number: 2009933378

Trademarks: Wiley and the Wiley logo are trademarks or registered trademarks of John Wiley & Sons, Inc. and/or its affiliates, in the United States and other countries, and may not be used without written permission. All other trademarks are the property of their respective owners. Wiley Publishing, Inc. is not associated with any product or vendor mentioned in this book.

About the Authors

James Whitehead II is the author of a number of popular addons for World of Warcraft, including LightHeaded, TomTom, Clique, PerfectRaid, and many other small but useful addons. He has been an active member of both the WoW UI and Lua communities since the World of Warcraft Beta began and has been writing addons ever since. When he actually has time to play the game, you can find him playing one of his many characters on the Emerald Dream (EU) server.

Jim currently resides in Oxford, England where he is pursuing his DPhil (PhD) in Computer Science at the Computing Laboratory. In his spare time he enjoys all things rowing, hacking on his Android phone, knitting, crocheting, and spending time with his friends.

Rick Roe—known in the WoW UI community as the zany goblin tinker Gazmik Fizzwidget—is the author of several popular addons including Feed-O-Matic and FactionFriend, as well as TrackMenu and a couple of others so useful that Blizzard made them obsolete by rolling their functionality into the default UI. When not slaving away for their goblin master, Rick's alter egos can often be found adventuring on Cenarius US.

Rick currently resides in Vancouver, Washington, with his wife and cats. His time outside of Azeroth is split between working to finish a computer science degree at Washington State University and building Mac and iPhone applications as an independent software developer.

About the Technical Editors

Daniel Stephens—more widely known in the WoW addon community as Iriel—was the Blizzard WoW UI Forum's first MVP. He has been helping others develop addons for several years, creating a few of his own along the way. His addons include DevTools (recently rolled into the base WoW UI) and he has made significant contributions to secure handlers and a number of other utilities. As somewhat of an altaholic, he has characters spread all over the realms, but considers Silver Hand (US) his original WoW home.

Daniel lives in the San Francisco bay area with his wife, cats, and camera. He spends his ''not free'' time doing systems design, architecture, and occasionally development work.

Esteban Santana Santana, known online as MentalPower, is both Lead Developer for the Auctioneer AddOns Package and one of the Administrators of Norganna's AddOns. He's been part of the WoW UI community since mid-2005 and has helped many people via the IRC channels and the various game and UI-related forums. When he logs into World of Warcraft, you can find him on the US-Alleria realm trying to level his various characters on the Emerald Dream guild.

Esteban currently resides in Carolina, Puerto Rico, and is a jack-of-all-trades IT person for Liberty Cablevision. In his spare time, he enjoys thrashing his buddies in a good game of Rock Band on the XBox 360.

Credits

Acquisitions Editor
Scott Meyers

Contributing Author
Nevin Flanagan

Project Editor
Maryann Steinhart

Technical Editors
Daniel Stephens
Rick Roe
Esteban Santana Santana

Production Editor
Rebecca Anderson

Copy Editor
Kim Cofer

Editorial Director
Robyn B. Siesky

Editorial Manager
Mary Beth Wakefield

Associate Director of Marketing
David Mayhew

Production Manager
Tim Tate

Vice President and Executive Group Publisher
Richard Swadley

Vice President and Executive Publisher
Barry Pruett

Associate Publisher
Jim Minatel

Project Coordinator, Cover
Lynsey Stanford

Proofreaders
Josh Chase and Nelson Kim, Word One

Indexer
J & J Indexing

Cover Image
Scott Johnson, FrogPants Studios LLC

Cover Designer
Michael E. Trent

Acknowledgments

James and Lee Whitehead, thank you for walking alongside your children as we journey down the winding road of life; we're blessed to have the two of you in our lives. Michelle Hastings, thank you for being such a role model of strength and determination for your little brother.

Robert Whitehead, thank you for always being yourself and making sure I don't stray far from who I am. Gregory Whitehead, thank you for being there for me whenever I need to ''geek'' out, I don't know many people that can get as excited as I do about silly things. Tom Harper, thank you for what you give me every single day. Everything about you makes me feel like the luckiest person alive.

Jamie Anderson, Edward Wilman, Amelia Earl, Rhianedd Jewell, Erika Nitsch, Daniel Jordan, and all my other friends at Oxford, thank you for keeping me busy, for helping me make excuses, and for being such a bad influence.

To Karen Hobson and everyone at WowInterface who have put up with me for five years now, thank you for all of your efforts in organizing all three books. To Mike, Kevin, Tom, Jacob, Sam, and everyone at Blizzard, thank you for creating such an amazing game and supporting us in our documentation efforts. To everyone at Wiley who helped bring these books into existence, thank you for your efforts to provide resources for the WoW user interface community. To Rick, thank you for stepping in when we needed you the most; your work has been instrumental in making this book what it is today. To Daniel, Nevin, and Esteban, thank you for all of your help in shaping the edges of this edition.

Finally, thank you to the World of Warcraft user interface community for everything you do.

— Jim

I’d first like to thank my coauthor, Jim, for offering me the opportunity to ‘‘graduate’’ from tech editing on the first edition to authoring on this second version. Crazy as the schedules and deadlines may have been, I’m still happier having been able to write my part instead of worrying about mucking with someone else’s work in order to satisfy my nitpicky tendencies. Thanks also for all your infrastructure work—without your website and database I’d have been a scribe without paper.

To Daniel and Esteban fell the unenviable task of performing the role I did on the first edition—catching all the silly code errors and obtuse explanations we dumb authors make—and with it my sympathy and gratitude. I can but hope I haven’t made your work too hard.

Thanks to Karen for playing den mother to the rowdy WoW UI community and giving us all a place to hook up; if it weren’t for your efforts I’d never have found my way into this project. Thanks as well to Scott, Maryann, and everyone at Wiley for making the project happen!

Thanks (again) to Daniel and Jim not just for your work on the book but for providing development tools without which my tasks would’ve been a whole lot harder. And of course, thanks to my family, Karen, Doug, and Brad, for putting up with me for a couple decades and making me the person I am, and to my wonderful wife Anne: I can’t imagine life without you, much less without the loving support you give for whatever crazy ideas I set myself to.

Finally, a very special thank you to Mike, Jacob, Sam, and Tom at Blizzard, without whose patience and willingness to answer oblique questions outside a normal work schedule we wouldn’t have been able to figure out several important chunks of the API we’re documenting. Next time I’m in SoCal, your drinks are on my tab.

— Rick

Contents at a Glance

Introduction xxxvii

Part I Learning to Program 1

Chapter 1 Programming for World of Warcraft 3

Chapter 2 Exploring Lua Basics 13

Chapter 3 Basic Functions and Control Structures 39

Chapter 4 Working with Tables 53

Chapter 5 Advanced Functions and Control Structures 77

Chapter 6 Lua Standard Libraries 91

Chapter 7 Learning XML 111

Part II Programming in World of Warcraft 123

Chapter 8 Anatomy of an Addon 125

Chapter 9 Working with Frames, Widgets, and Other Graphical Elements 143

Chapter 10 Saving Time with Frame Templates 171

Chapter 11 Exploring the World of Warcraft API 187

Chapter 12 Interacting with Widgets 207

Chapter 13 Responding to Game Events 243

Chapter 14 Tracking Damage with CombatTracker 267

Part III Advanced Addon Techniques 283

Chapter 15 Taking Action with Secure Templates 285

Chapter 16 Binding Keys and Clicks to Addon Code 309

Chapter 17 Creating Slash Commands 337

Chapter 18 Responding to Graphic Updates with OnUpdate 351

Chapter 19 Altering Existing Behavior with Function Hooking 359

Chapter 20 Creating Custom Graphics 373

Chapter 21 Responding to the Combat Log and Threat Information 387

Chapter 22 Creating Scroll Frames 413

Chapter 23 Creating Dropdown Menus 431

Chapter 24 Scanning and Constructing Tooltips 451

Chapter 25 Taking Protected Action in Combat 463

Chapter 26 Creating Unit Frames with Group Templates 501

Part IV Reference 537

Chapter 27 API Reference 539

Chapter 28 API Categories 1025

Chapter 29 Widget Reference 1121

Chapter 30 Events Reference 1277

Part V Appendixes 1303

Appendix A Best Practices 1305

Appendix B Utilizing Addon Libraries 1329

Appendix C Tracking History Using Version Control Systems 1339

Appendix D Addon Author Resources 1349

Index 1357

Contents

Introduction **xxxvii**

Part I Learning to Program **1**

Chapter 1 Programming for World of Warcraft **3**

Customizing the User Interface 3

What Is an Addon? 4

What Can Addons Do? 4

Exploring Your AddOns Directory 7

Blizzard Addons 8

Custom Addons 10

Creating Your First Addon: HeyThere 10

Creating Files and Directories 10

Loading and Testing the Addon 11

Summary 12

Chapter 2 Exploring Lua Basics **13**

Downloading and Installing Lua 14

Downloading and Installing WowLua 14

Using Lua on the Web 15

Downloading and Installing a Lua Interpreter 16

Microsoft Windows 16

Mac OS X 16

Using the Lua Interpreter 17

Running Commands 18

Understanding Error Messages 18

Using History to Make Changes 19
Quitting the Interpreter 19
Microsoft Windows 19
Mac OS X 19
Working with Numbers 20
Basic Arithmetic Operations 20
Scientific Notation 21
Hexadecimal Notation 21
Understanding Floating Point 22
Understanding Values and Variables 23
Exploring Values and Their Types 23
Primitive Types 23
Using the type() Function 23
Using Variables 24
Valid Variable Names 25
Assigning Variables 25
Assigning Multiple Variables 26
Comparing Values 26
Working with Strings 27
Comparing Strings 27
Concatenating Multiple Strings 28
Converting Numbers to Strings 28
Converting Strings to Numbers 29
Quoting Strings 29
Single Quote (′) 29
Double Quote (′′) 30
Bracket Quote ([[ ]]) 30
Escaping Special Characters 31
Getting a String′s Length 32
Boolean Values and Operators 33
Using the and Operator 33
Using the or Operator 34
Negation Using the not Operator 34
Understanding the nil Value 35
Exploring Scope 35
Blocks 36
Chunks 37
Summary 37

Chapter 3 Basic Functions and Control Structures 39

Using Functions 39
Creating a Function 39
Local Functions 40

Function Arguments and Returns 41
Converting Celsius to Fahrenheit 41
Empty Arguments 42
No Return Values 42
Functions as Lua Values 42
Making Decisions with the if Statement 43
Simple Conditionals 43
Complex Expressions 44
Extended Conditionals 44
Displaying a Personalized Greeting 45
Repeating Actions with the while Statement 46
Computing Factorials 46
Differences Between while and repeat 47
Looping with the Numeric for Statement 48
Computing Factorials 50
Evaluation of Loop Conditions 50
Variable Scope in for Loops 50
Summary 51

Chapter 4 Working with Tables 53
Storing Data Using Tables 53
Creating and Indexing Tables 54
Clearing an Element from a Table 54
Shortcuts for String Keys 55
Creating Populated Tables 55
Using Tables as Arrays 56
Creating an Array 57
Getting the Length of an Array 57
Adding Elements to an Array 58
Removing Elements from an Array 60
Sorting the Elements of an Array 61
Using Tables as Namespaces 61
Creating a Namespace of Utility Functions 62
Adding Functions to a Namespace 62
Storing an Existing Function 62
Defining a New Function 63
Object-Oriented Programming with Tables 63
Creating a Non-Object-Oriented Counter 63
Using Tables as Simple Objects 64
Using : to Call Object Methods 65
Defining Functions Using : 66
Making a Better Counter 67

Extending Tables with Metatables 68
Adding a Metatable 68
Defining Metamethods 69
Defining Basic Arithmetic Using __add, __sub, __mul, and __div 70
Defining Negation Using __unm 71
Creating Meaningful Output with __tostring 71
Concatenating Tables Using __concat 72
Exploring Fallback Tables with __index 72
Catching Creation of Keys with __newindex 74
Bypassing Metatables 75
value = rawget(tbl, key) 75
rawset(tbl, key, value) 76
Summary 76

Chapter 5 Advanced Functions and Control Structures 77
Multiple Return Values 77
Converting Hex to RGB 77
Assigning Multiple Values 78
Missing Return Values? 79
Multiple Return Values in World of Warcraft 79
Using a Dummy Variable 80
Using the select() Function 81
Accepting a Variable Number of Arguments 81
Declaring a Vararg Function 82
Using select() with . . . 83
Generic for Loops and Iterators 84
Syntax of Generic for 84
Traversing the Array Part of a Table 85
Traversing an Entire Table 86
Clearing a Table 86
Using Other Iterators 87
Sorting an Array of Table Data 87
Define Example Data 88
Default Sort Order 88
Creating a Comparison Function 88
Creating a More Complex Sort Function 89
Summary 90

Chapter 6 Lua Standard Libraries 91
Table Library 92
str = table.concat (table [, sep [, i [, j]]]) 92
table.insert (table, [pos,] value) 92
max = table.maxn (table) 93

value = table.remove (table [, pos]) 93
table.sort (table [, comp]) 93
String Utility Functions 94
Formatting New Strings 95
Pattern Matching 98
Character Classes 98
Pattern Items 100
Pattern Captures 101
Pattern Anchors 102
Pattern Examples 102
Pattern Matching Functions 102
string.gmatch(s, pattern) 103
string.gsub(s, pattern, repl [, n]) 103
string.match(s, pattern [, init]) 104
string.find(s, pattern [, init [, plain]]) 104
Math Library 105
World of Warcraft Additions to Lua 108
Function Aliases 109
Summary 110

Chapter 7 Learning XML 111

XML as a Markup Language 111
XML's Relationship to HTML 112
Components of XML 112
XML Tags 113
XML Elements 113
XML Attributes 113
XML Entities 114
Creating Well-Formed XML 114
Validating an XML Document 115
Example Schema Definition 115
Example XML Document 116
Exploring the Schema 116
XML in World of Warcraft 117
Using a GradientType 118
Exploring Blizzard's XML User Interface Customization Tool 119
Summary 121

Part II Programming in World of Warcraft 123

Chapter 8 Anatomy of an Addon 125

Exploring an Addon's Files and Folders 125
Table of Contents (.toc) File 125
Interface: 126

Title: 127
Notes: 128
Dependencies:, ## RequiredDeps: 128
OptionalDeps: 129
LoadOnDemand: 129
LoadsWith: 129
DefaultState: 130
LoadManager: 130
SavedVariables: 130
SavedVariablesPerCharacter: 131
X-Label Directives 131
Addon Categories 131
XML Files 132
Lua Script Files 133
Media Files 133
Music 133
Graphics 133
Localizing Your Addons 134
Valid Locales 135
Reasons for Providing Localization 135
Encouraging Users to Contribute 136
Implementing Localization 136
Add a File for Each Locale 136
Create a Global Table Containing the Base Strings 136
Using the Localization Table 137
Adding New Locales 137
Handling Partial Translations 138
Introducing Frames, Widget Scripts, and Events 138
Frames, FontStrings, and Textures 138
Displaying Text with FontStrings 139
Showing Graphics and Colors with Textures 139
Anchoring Objects On-Screen 139
Responding to Interaction with Widget Scripts 139
Responding to Game Events 139
Loading of an Addon 141
Summary 142

Chapter 9 Working with Frames, Widgets, and Other Graphical Elements 143

Introducing BagBuddy 143
Creating an Addon Skeleton 144
Creating a Frame 144
Parenting 145

Giving Objects Sizes 146
Absolute Dimensions 146
Relative Dimensions 146
Anchoring Objects 147
Sticky Anchors 148
SetAllPoints 148
Anchor Examples 148
Using Lua to Create Frames 149
Adding Layers of Textures and Font Strings 150
Layering Frames and Graphics 150
Frame Strata 150
Frame Levels 151
Graphical Layers 152
BagBuddy Frame Design 153
Finding Graphics 155
TexBrowser AddOn 155
ArtBrowser on Wowprogramming.com 155
Adding Textures 155
Defining BagBuddy's Background Textures 157
Coloring Textures 158
Using Solid Colors 158
Creating a Gradient 159
Adding the Portrait Texture 160
Creating Textures in Lua 162
Creating Text using FontStrings 164
Further Customization 165
Using Font Definitions 165
Creating FontStrings in Lua 166
Understanding Object Visibility 166
Finding Existing Frames 167
Summary 167
The Code 168

Chapter 10 Saving Time with Frame Templates 171
Understanding Templates 171
Advantages of Using Templates 173
Naming Elements Using $parent 173
Setting Keys Using the parentKey Attribute 174
Creating a Template for BagBuddy's Item Buttons 174
Setting Button Textures 175
Creating New Frames with Your Template 176
Exploring Font Definitions 177
Altering a Font Definition 178

Investigating UIPanelTemplates 179
UIPanelButtonTemplate 180
UIPanelCloseButton 180
UIPanelScrollBarTemplate 181
InputBoxTemplate 181
UICheckButtonTemplate 182
TabButtonTemplate 183
UIRadioButtonTemplate 183
Summary 183
The Code 184

Chapter 11 Exploring the World of Warcraft API 187
Understanding the WoW API 187
Normal APIs 188
Library-like APIs 188
FrameXML Functions 189
Protected Functions 189
Unit Functions Up Close 190
Querying Item Information for BagBuddy 193
Scanning Bags with the Container API 193
Querying Detailed Item Information 194
Item Identifiers 195
Using the Item API 197
Writing a Bag Scanner 198
Sorting the Player's Inventory 199
Displaying the Inventory 199
Testing the Update Function 200
Finding the Right API Functions 201
Exploring the API Categories 201
Examining the FrameXML Code 202
Looking at Another Addon 203
Asking for Help! 203
Summary 203
The Code 204

Chapter 12 Interacting with Widgets 207
Making BagBuddy's Buttons Interactive 207
Setting Frame Scripts via XML 208
Using the function Attribute 209
Setting Frame Scripts Using Lua 209
Showing Item Tooltips Using OnEnter and OnLeave 210
Adding Clickable Buttons to BagBuddy 212
Introducing the OnClick Handler 212

Creating a Close Button Using Templates 213
Creating Clickable Filter Buttons 214
Creating the Filter Buttons Dynamically 216
Adding Custom Tooltips 217
Making the Filter Buttons Clickable 217
Updating the Results 218
Navigating Multiple Pages 219
Adding XML Definitions for Buttons and Status Text 220
Writing OnClick Handlers for Navigation Buttons 221
Altering the Update Function for Pages 221
Enabling and Disabling Navigation Buttons 222
Creating and Updating Status Text 223
Final Changes to Support Navigation 224
Adding a Name Filter to BagBuddy 224
Creating an EditBox 225
Filtering by Name 226
Exploring Widget Types 227
Button 227
CheckButton 228
ColorSelect 228
EditBox 229
GameTooltip 229
MessageFrame 229
Minimap 229
Model 230
ScrollingMessageFrame 231
ScrollFrame 231
SimpleHTML 231
Slider 232
StatusBar 232
Summary 233
The Code 233

Chapter 13 Responding to Game Events 243

Understanding Events 243
Registering for Events 244
Responding to Events with OnEvent 244
Query Events 246
Tracking Changes to Inventory for BagBuddy 246
Examining the BAG_UPDATE Event 246
Tracking New Inventory Items 246
Writing a New Sorting Function 248
Altering BagBuddy_Update 248

Adding an OnEvent Handler 249
Cleaning Up 250
Adding a Slash Command 251
Storing Data with SavedVariables 251
Registering a New Saved Variable 252
Saved Variables and ADDON_LOADED 252
Using Items from BagBuddy 253
Finding the Right Event Using /eventtrace 254
Summary 255
The Code 255

Chapter 14 Tracking Damage with CombatTracker 267
Defining Specifications 267
CombatTracker User Experience 267
Finding the Right Game Events 268
PLAYER_REGEN_DISABLED 268
PLAYER_REGEN_ENABLED 268
UNIT_COMBAT 269
Creating the Addon's Skeleton 269
Defining CombatTracker's XML Frame 270
Defining a Backdrop 271
Adding a Font String 272
Testing CombatTrackerFrame 272
Adding Script Handlers to CombatTrackerFrame 273
Adding Functions to CombatTracker.lua 275
CombatTracker_OnLoad(frame) 275
CombatTracker_OnEvent 276
PLAYER_REGEN_ENABLED 276
PLAYER_REGEN_DISABLED 277
UNIT_COMBAT 277
CombatTracker_UpdateText() 277
CombatTracker_ReportDPS() 278
Testing CombatTracker 278
Frame Dragging 279
Right-Click Reporting: Part I 279
Testing Combat Tracking 280
Right-Click Reporting: Part II 280
Summary 281

Part III Advanced Addon Techniques 283

Chapter 15 Taking Action with Secure Templates 285
Why Are Secure Templates Necessary? 285
Protected Frames 286
Controlling Secure Frames Using Attributes 288

Using Secure Templates 288
Defining Behaviors for Action Buttons 289
Casting a Spell 289
Looking Under the Hood 290
Specifying Units to Affect 291
Other Types and Their Uses 291
Making Simple Choices 296
Working with Modified Attributes 296
Delegating Attribute Responsibility 298
Choosing an Action by Hostility 298
Applying Action Buttons in Practice 299
Modifying an Existing Frame 299
A Complex Action Button 300
Understanding Taint and Working Safely Around Secure Code 302
Enabling Taint Logging 303
Execution Taint 304
Variable Taint 305
Creeping Taint 307
Summary 308

Chapter 16 Binding Keys and Clicks to Addon Code 309
Defining Bindings in XML 310
Creating Your Own Binding Actions 312
Binding Keys to Actions 314
Building a Simple Binding UI 315
Defining Basic Behaviors 318
Using SetBinding() 321
Working with Existing Bindings 324
Displaying an Action's Bindings 325
Understanding Binding Storage 326
Binding Keys to Secure Actions 327
Working with Click Bindings 328
Creating Secure Bindings in XML 329
Summary 329
The Code 330
BindingTest 330
ClickBindingTest 334

Chapter 17 Creating Slash Commands 337
Creating Basic Slash Commands 337
Tokenizing Strings 339
Tokenizing with Patterns 341
Setting Up the Patterns 341

Preparing for the Tokenization 342
Parsing the Formula 343
Using a Command Table 345
Summary 347
The Code 347
SlashCalc 347

Chapter 18 Responding to Graphic Updates with OnUpdate 351
Understanding Graphic Updates 351
Delaying Code Using OnUpdate 352
Grouping Events to Avoid Over-Processing 354
Grouping Multiple Events 355
Repeating Code with OnUpdate 356
Considering Performance with OnUpdate Scripts 357
Summary 357

Chapter 19 Altering Existing Behavior with Function Hooking 359
What Is Function Hooking? 359
Modifying Return Values 360
Using a Variable Argument Function 361
Using Utility Functions capture() and release() 361
Hooking Widget Scripts 362
Hooking a Function Securely 364
Hooking Scripts Securely 365
Deciding When to Hook 365
Understanding the Hook Chain 365
You Can't Rely on Order 366
There Is No ''Unhook'' 366
Hooking Hits Performance 366
Finding Alternatives 367
Designing an Addon: MapZoomOut 367
Creating a Timer Frame 368
Initial Setup 368
Create the Function Hook 369
Writing the Timer Code 369
Final Setup 370
Testing MapZoomOut 370
Summary 370
The Code 371
MapZoomOut 371

Chapter 20 Creating Custom Graphics 373
Common Rules for Creating Graphics 373
The GIMP 374

Create a New Image 374
Adding Graphical Components 375
Saving Textures 376
Adobe Photoshop 376
Create a New Image 376
Adding Graphical Components 377
Creating an Alpha Channel 377
Saving an Image 378
Paint Shop Pro 379
Creating a New Image 380
Adding Graphical Components 380
Creating an Alpha Channel 381
Saving an Image 382
Testing Your Texture 383
No Button Appears 384
A Green Box Appears 384
XML Texture Definition 384
Lua Texture Definition 385
Summary 385

Chapter 21 Responding to the Combat Log and Threat Information 387

Understanding the Combat Log 387
Event Arguments 387
Combat Sub-Events 388
Combat Event Prefix 389
Bit Fields and Spell Schools 389
Combat Event Suffix 390
Spell-Only Suffixes 393
Special Combat Events 395
Unit GUIDs 396
Format of GUIDs 397
Unit Flags 398
COMBATLOG_OBJECT_TYPE_MASK 398
COMBATLOG_OBJECT_CONTROL_MASK 398
COMBATLOG_OBJECT_REACTION_MASK 399
COMBATLOG_OBJECT_AFFILIATION_MASK 399
COMBATLOG_OBJECT_SPECIAL_MASK 399
Using CombatLog_Object_IsA 400
Writing CombatStatus 401
Creating the Basic Addon Structure 401
Initializing CombatStatus 402
Updating Pet Mappings 405
Storing Damage and Healing Information 405
Taking ‘‘Snapshots’’ of Damage and Healing 407

Writing an OnUpdate Function 408
Responding to Events 408
COMBAT_LOG_EVENT_UNFILTERED 409
PARTY_MEMBERS_CHANGED 409
UNIT_PET 409
PLAYER_REGEN_DISABLED 409
PLAYER_REGEN_ENABLED 410
Creating the Frame Display 410
Updating the Frame Display 410
Future Additions 412
Summary 412

Chapter 22 Creating Scroll Frames 413
Using Scroll Frames 414
Adding a Scroll Child 415
Manipulating a ScrollFrame 416
Adding Scroll Bars 417
Creating Faux Scroll Frames 419
Adding Scroll Bars 422
Scrolling with the Mouse Wheel 423
Problems with Slider Precision 423
Summary 424
The Code 424
ScrollFrameTest 424
MacroIconTest 426

Chapter 23 Creating Dropdown Menus 431
Creating a Basic Dropdown 431
Adding a Toggle Button 432
Creating a Dropdown Frame 433
Initializing the Dropdown 433
Adding Buttons to the Dropdown 433
Calling UIDropDownMenu_Initialize() 434
Toggling the Dropdown Menu 434
Testing the Dropdown 435
Creating Multilevel Dropdowns 436
Adding Functionality to Dropdowns 437
Customizing Text Elements 438
Function Menu Items 440
CheckButton Menu Items 440
ColorPicker Menu Items 441
Using Dropdowns for Selection 443
Automating Menu Creation with EasyMenu 445
Creating Dynamic Menus 447
Summary 449

Chapter 24 **Scanning and Constructing Tooltips** **451**
Understanding the Tooltip System 451
Different Types of Tooltips 452
Contextual Tooltips 452
Static Tooltips 453
Tooltip Contents 453
Custom Text in a Tooltip 453
Game Element Tooltips 455
Adding Information to the Tooltip 458
Loading the Tooltip with Item Information 458
Getting Information from Tooltips 460
Accessing Individual Tooltip Lines 460
Checking Soulbound Status 461
Using Global Strings for Localization 461
Replacing a Script Instead of Hooking a Script 461
Summary 462

Chapter 25 **Taking Protected Action in Combat** **463**
Snippets: The Basis of Secure Action 463
How Can Addon Code Be Secure? 463
Writing a Snippet 464
Secure Handler Frames 464
Handler Template Reference 466
Integrating a Click Handler with a Secure Action Button 468
Preserving State and Controlling Information 473
Private Global Environments 474
Secure API Functions 475
The control Object 476
Frame Handles 477
Allowed Actions 479
Additional or Changed Actions 479
Wrapping Frame Scripts 482
A Simple Click Wrapper 483
Using a Post-Hook 484
Script Wrapper Reference 485
Triggered Changes 486
State Drivers 486
State Responders 487
Responding to Show/Hide 487
Responding to Attribute and State Changes 488
State Conditionals 490
Target Specifiers and Unit Conditions 491
State Variables 492
Unit Conditions 492
General Conditions 493

Summary 496
The Code 496
BlessedMenus 496

Chapter 26 Creating Unit Frames with Group Templates 501

Configuring a SecureGroupHeader 501
Configuration Options 502
Initial Configuration Function 505
Creating SquareUnitFrames 506
Constructing the Template 506
Creating a Header Template 508
Setting Name and Status Bars 509
Nudging Frame Levels 511
Responding to Events and Clicks 511
Targeting the Unit on Left-Click 511
Moving the Header 512
Health Update Events 513
Power Update Events 514
Responding to Name Changes 516
Enhancing SquareUnitFrames 516
Highlighting Units on Mouseover 516
Showing the Targeted Unit 517
Displaying Threat Levels 518
Showing Dead Players 519
Displaying Unit Names 521
Adding Pets to SquareUnitFrames 523
Creating a SecureGroupPetHeaderTemplate 526
Summary 526
The Code 526
SquareUnitFrames 526

Part IV Reference 537

Chapter 27 API Reference 539

API Reference Conventions 539
Function Signatures 539
Optional Arguments 540
Argument Choices 540
Argument and Return Listings 540
Common API Flags 541
API Meta-Types 542
1nil 542
actionID 542
ah-list-type 543