

Contents

Foreword	xiii
Preface	xv
<i>Chapter 1 The Big Picture</i>	<i>1</i>
1.1 Book Goals	3
1.2 Plug-In	5
1.3 Eclipse in a Nutshell	5
Part I ○ Circle Zero: Hello World	9
<i>Chapter 2 Setting Up Eclipse for Plug-In Development</i>	<i>11</i>
2.1 Setting Up a Workspace	11
2.2 Browsing and Searching Source	12
<i>Chapter 3 Hello World</i>	<i>15</i>
3.1 Declaration/Implementation Split	16
3.2 Hello Button	18
3.3 Saying “Hello”	25
Part II ○ Circle One: Basic Plug-In	33
<i>Chapter 4 Der Plan</i>	<i>35</i>
4.1 JUnit by Example	35
4.2 JUnit Integration	37

<i>Chapter 5</i>	<i>Contributing a Menu Item to Run Tests</i>	<i>39</i>
<i>Chapter 6</i>	<i>Implementing the Menu Item Behavior</i>	<i>47</i>
<i>Chapter 7</i>	<i>Displaying the Results</i>	<i>59</i>
<i>Chapter 8</i>	<i>Defining an Extension Point</i>	<i>65</i>
<i>Chapter 9</i>	<i>Notifying Extensions</i>	<i>77</i>
<i>Chapter 10</i>	<i>Publishing</i>	<i>79</i>
10.1	Package the Plug-In	79
10.2	Bundling the Plug-In into a Feature	83
10.3	Contributing	86
<i>Chapter 11</i>	<i>Closing Circle One</i>	<i>89</i>

Interlude

<i>Chapter 12</i>	<i>Test-Driven Plug-In Development</i>	<i>91</i>
12.1	PDE JUnit	92
12.2	A Test Project Fixture	98
12.3	Testing the Contributed JUnit Plug-In	100
12.4	And Now...	104

Part III ○ Circle Two: The Rest of the Plug-In 105

<i>Chapter 13</i>	<i>Viewing Results</i>	<i>107</i>
13.1	Contributing a View	107
13.2	Listening to Testing Progress	111
13.3	Changing Colors	113
<i>Chapter 14</i>	<i>Menu Contributions</i>	<i>119</i>
14.1	Creating Context Menus	119
14.2	Contributing Menu Items	124
<i>Chapter 15</i>	<i>Failed Tests Are Compile Errors</i>	<i>129</i>
<i>Chapter 16</i>	<i>Test Failures as Markers</i>	<i>131</i>
16.1	Test for a Marker	131
16.2	Passing the Project	134
16.3	Creating Markers	136

16.4	Deleting Markers	140
16.5	Marker Images	141
16.6	Marker Resolution	142
<i>Chapter 17</i>	<i>Finding Tests</i>	<i>147</i>
<i>Chapter 18</i>	<i>Builders and Natures</i>	<i>151</i>
18.1	Resource Listeners Versus Builders	152
18.2	Using Natures to Configure Builders	154
<i>Chapter 19</i>	<i>Auto-Test Property</i>	<i>161</i>
<i>Chapter 20</i>	<i>Exception Handling</i>	<i>171</i>
20.1	IStatus, CoreException	172
20.2	Presenting Exceptions in an Error Dialog	173
20.3	Logging Errors	175
<i>Chapter 21</i>	<i>Tracing—Instrumenting a Plug-In</i>	<i>179</i>
<i>Chapter 22</i>	<i>Marker Resolution—Invoking a Long-Running Operation</i>	<i>185</i>
22.1	Testing Marker Resolution	186
22.2	Build and Rerun	186
22.3	Showing Progress	188
22.4	Rerunning the Test	191
<i>Chapter 23</i>	<i>Test Report View—Using JFace</i>	<i>193</i>
23.1	TestResult	195
23.2	The Test	196
23.3	The View	197
23.4	TestReportLabelProvider	201
23.5	TestReportContentProvider	204
23.6	Handling Events	210
<i>Chapter 24</i>	<i>A Simple Editor to Exclude Tests</i>	<i>217</i>
24.1	Contributing an Editor	219
24.2	Contributing a Contributor	224
<i>Chapter 25</i>	<i>ResultView Revisited—Observing Changes</i>	<i>231</i>
25.1	Testing Color	231
25.2	Observing Changes	232
25.3	Reacting to Changes	235

Chapter 26	<i>Perspectives</i>	239
26.1	Views in Perspective	239
26.2	Show View Menu	242
Chapter 27	<i>Help</i>	247
27.1	Top-Level Help	247
27.2	Integrated Help	250
27.3	Context-Sensitive Help	251
Chapter 28	<i>Internationalization and Accessibility</i>	257
28.1	Externalizing Strings from the Manifest Files	257
28.2	Externalizing Strings from Code	258
28.3	Accessibility	261
Chapter 29	<i>Publishing a Plug-In for Other Programmers</i>	263
29.1	Defining the API	263
29.2	Exporting Classes	264
29.3	Separating Published from Internal Packages	266
29.4	Separating Core from UI	268
29.5	Publishing an Extension Point—Extension Point Schemas	268
Chapter 30	<i>Closing Circle Two</i>	275
30.1	Contributing	277
30.2	Redeploying the Plug-In	278
30.3	Where to Go Next?	280
Part IV	<i>Circle Three: Pattern Stories</i>	281
Chapter 31	<i>Core Runtime—IAdaptable</i>	283
31.1	Extension Object/Extension Interface	285
31.2	Surfacing Interfaces Using IAdaptable	288
31.3	AdapterFactory—Adding Interfaces to Existing Types	289
Chapter 32	<i>Core Workspace—Resources</i>	295
32.1	Accessing File-System Resources—Proxy and Bridge	296
32.2	The Workspace—Composite	300
32.3	Traversing the Resource Tree—Visitor	301
32.4	Tracking Resource Changes—Observer	302
32.5	Batching Changes—Execute Around Method	305

Chapter 33	<i>Java Core</i>	307
33.1	From Resources to Java Elements—Adapter	308
33.2	Java Elements—(Virtual) Proxy	309
33.3	The Java Element Tree—Composite	313
33.4	Type Hierarchies—Objectifying an Association	314
33.5	Traversing the Java Model	315
33.6	Tracking Java Element Changes—Observer	316
33.7	Collecting Results—Builder	317
33.8	Abstract Syntax Tree Analysis—Visitor	319
Chapter 34	<i>Standard Widget Toolkit—SWT</i>	325
34.1	Composing Widgets—Composite	326
34.2	Defining the Layout—Strategy	330
34.3	Responding to Events—Observer	333
Chapter 35	<i>JFace—User Interface Frameworks</i>	335
35.1	Viewers: Connecting a Widget to a Model— Pluggable Adapter	336
35.2	Viewers: Customizing a Viewer without Subclassing—Strategy	341
35.3	Actions—Command	342
Chapter 36	<i>UI Workbench</i>	345
36.1	Implementing the Lazy Loading Rule— Virtual Proxies	346
36.2	Persisting UI State—Memento	347
36.3	Workbench Services—IAdaptable	351
Chapter 37	<i>Closing Circle Three</i>	355
37.1	Final Forward Pointers	356
37.2	An Invitation to Contribute	359
Part V	<i>Appendices</i>	361
Appendix A	<i>TestRunner Details</i>	363
A.1	TestRunner	364
A.2	SocketTestRunner	368
Appendix B	<i>The TestProject Fixture</i>	371

Appendix C AutoTestBuilder with Exclusion Support	375
References	379
Index	381