

---

# Java™ Power Tools

*John Ferguson Smart*

**ULB Darmstadt**



**16760471**

**O'REILLY®**

Beijing • Cambridge • Farnham • Köln • Paris • Sebastopol • Taipei • Tokyo

---

# Table of Contents

|                    |        |
|--------------------|--------|
| Foreword .....     | xvii   |
| Preface .....      | xix    |
| Introduction ..... | xxxiii |

---

## Part I. Build Tools

|                                                                 |    |
|-----------------------------------------------------------------|----|
| 1. Setting Up a Project Using Ant .....                         | 5  |
| 1.1 Ant in the Build Process .....                              | 5  |
| 1.2 Installing Ant .....                                        | 5  |
| 1.3 A Gentle Introduction to Ant .....                          | 8  |
| 1.4 Compiling Your Java Code in Ant .....                       | 15 |
| 1.5 Customizing Your Build Script Using Properties .....        | 17 |
| 1.6 Running Unit Tests in Ant .....                             | 21 |
| 1.7 Generating Documentation with Javadoc .....                 | 39 |
| 1.8 Packaging Your Application .....                            | 41 |
| 1.9 Deploying Your Application .....                            | 45 |
| 1.10 Bootstrapping Your Build Scripts .....                     | 47 |
| 1.11 Using Maven Dependencies in Ant with the Maven Tasks ..... | 49 |
| 1.12 Using Ant in Eclipse .....                                 | 53 |
| 1.13 Using Ant in NetBeans .....                                | 53 |
| 1.14 Manipulating XML with XMLTask .....                        | 54 |
| 1.15 Conclusion .....                                           | 60 |
| 2. Setting Up a Project Using Maven 2 .....                     | 61 |
| 2.1 Maven and the Development Build Process .....               | 61 |
| 2.2 Maven and Ant .....                                         | 62 |
| 2.3 Installing Maven .....                                      | 62 |
| 2.4 Declarative Builds and the Maven Project Object Model ..... | 64 |
| 2.5 Understanding the Maven 2 Lifecycle .....                   | 77 |

---

|      |                                                       |     |
|------|-------------------------------------------------------|-----|
| 2.6  | The Maven Directory Structure                         | 79  |
| 2.7  | Configuring Maven to Your Environment                 | 80  |
| 2.8  | Dependency Management in Maven 2                      | 82  |
| 2.9  | Looking for Dependencies with MvnRepository           | 91  |
| 2.10 | Project Inheritance and Aggregation                   | 91  |
| 2.11 | Creating a Project Template with Archetypes           | 96  |
| 2.12 | Compiling Code                                        | 100 |
| 2.13 | Testing Your Code                                     | 101 |
| 2.14 | Packaging and Deploying Your Application              | 105 |
| 2.15 | Deploying an Application Using Cargo                  | 107 |
| 2.16 | Using Maven in Eclipse                                | 111 |
| 2.17 | Using Maven in NetBeans                               | 113 |
| 2.18 | Using Plug-Ins to Customize the Build Process         | 113 |
| 2.19 | Setting Up an Enterprise Repository with Archiva      | 122 |
| 2.20 | Setting Up an Enterprise Repository Using Artifactory | 135 |
| 2.21 | Using Ant in Maven                                    | 147 |
| 2.22 | Advanced Archetypes                                   | 153 |
| 2.23 | Using Assemblies                                      | 157 |

---

## Part II. Version Control Tools

|           |                                                          |            |
|-----------|----------------------------------------------------------|------------|
| <b>3.</b> | <b>Setting Up Version Control Using CVS .....</b>        | <b>165</b> |
| 3.1       | An Introduction to CVS                                   | 165        |
| 3.2       | Setting Up a CVS Repository                              | 166        |
| 3.3       | Creating a New Project in CVS                            | 166        |
| 3.4       | Checking Out a Project                                   | 168        |
| 3.5       | Working with Your Files—Updating and Committing          | 170        |
| 3.6       | Resolving a Locked Repository                            | 174        |
| 3.7       | Working with Keyword Substitution                        | 174        |
| 3.8       | Working with Binary Files                                | 175        |
| 3.9       | Tags in CVS                                              | 178        |
| 3.10      | Creating Branches in CVS                                 | 179        |
| 3.11      | Merging Changes from a Branch                            | 180        |
| 3.12      | Viewing Change History                                   | 181        |
| 3.13      | Reverting Changes                                        | 183        |
| 3.14      | Using CVS in Windows                                     | 185        |
| <b>4.</b> | <b>Setting Up Version Control Using Subversion .....</b> | <b>187</b> |
| 4.1       | An Introduction to Subversion                            | 187        |
| 4.2       | Installing Subversion                                    | 191        |
| 4.3       | Subversion Repository Types                              | 191        |
| 4.4       | Setting Up a Subversion Repository                       | 193        |

|      |                                                                |     |
|------|----------------------------------------------------------------|-----|
| 4.5  | Setting Up a New Subversion Project                            | 195 |
| 4.6  | Checking Out Your Working Copy                                 | 197 |
| 4.7  | Importing Existing Files into Subversion                       | 198 |
| 4.8  | Understanding Subversion Repository URLs                       | 200 |
| 4.9  | Working with Your Files                                        | 201 |
| 4.10 | Seeing Where You're At: The Status Command                     | 205 |
| 4.11 | Resolving Conflicts                                            | 208 |
| 4.12 | Using Tags, Branches, and Merges                               | 210 |
| 4.13 | Rolling Back to a Previous Revision                            | 214 |
| 4.14 | Using File Locking with Binary Files                           | 215 |
| 4.15 | Breaking and Stealing Locks                                    | 217 |
| 4.16 | Making Locked Files Read-Only with the svn:needs-lock Property | 219 |
| 4.17 | Using Properties                                               | 220 |
| 4.18 | Change History in Subversion: Logging and Blaming              | 223 |
| 4.19 | Setting Up a Subversion Server with svnserve                   | 224 |
| 4.20 | Setting Up a Secure svnserve Server                            | 228 |
| 4.21 | Setting Up a WebDAV/DeltaV Enabled Subversion Server           | 229 |
| 4.22 | Setting Up a Secure WebDAV/DeltaV Server                       | 234 |
| 4.23 | Customizing Subversion with Hook Scripts                       | 235 |
| 4.24 | Installing Subversion As a Windows Service                     | 236 |
| 4.25 | Backing Up and Restoring a Subversion Repository               | 238 |
| 4.26 | Using Subversion in Eclipse                                    | 239 |
| 4.27 | Using Subversion in NetBeans                                   | 245 |
| 4.28 | Using Subversion in Windows                                    | 250 |
| 4.29 | Defect Tracking and Change Control                             | 260 |
| 4.30 | Using Subversion in Ant                                        | 262 |
| 4.31 | Conclusion                                                     | 265 |

---

## Part III. Continuous Integration

|           |                                                                        |            |
|-----------|------------------------------------------------------------------------|------------|
| <b>5.</b> | <b>Setting Up a Continuous Integration Server with Continuum .....</b> | <b>271</b> |
| 5.1       | An Introduction to Continuum                                           | 271        |
| 5.2       | Installing a Continuum Server                                          | 271        |
| 5.3       | Manually Starting and Stopping the Server                              | 275        |
| 5.4       | Checking the Status of the Server                                      | 276        |
| 5.5       | Running the Continuum Server in Verbose Mode                           | 276        |
| 5.6       | Adding a Project Group                                                 | 277        |
| 5.7       | Adding a Maven Project                                                 | 277        |
| 5.8       | Adding an Ant Project                                                  | 280        |
| 5.9       | Adding a Shell Project                                                 | 280        |
| 5.10      | Managing Your Project Builds                                           | 281        |

|           |                                                                            |            |
|-----------|----------------------------------------------------------------------------|------------|
| 5.11      | Managing Users                                                             | 282        |
| 5.12      | Setting Up Notifiers                                                       | 283        |
| 5.13      | Configuring and Scheduling Builds                                          | 285        |
| 5.14      | Debugging Your Builds                                                      | 288        |
| 5.15      | Configuring the Continuum Mail Server                                      | 288        |
| 5.16      | Configuring the Continuum Web Site Ports                                   | 289        |
| 5.17      | Automatically Generating a Maven Site with Continuum                       | 290        |
| 5.18      | Configuring a Manual Build Task                                            | 291        |
| 5.19      | Conclusion                                                                 | 293        |
| <b>6.</b> | <b>Setting Up a Continuous Integration Server with CruiseControl .....</b> | <b>295</b> |
| 6.1       | An Introduction to CruiseControl                                           | 295        |
| 6.2       | Installing CruiseControl                                                   | 296        |
| 6.3       | Configuring an Ant Project                                                 | 297        |
| 6.4       | Keeping People Notified with Publishers                                    | 303        |
| 6.5       | Setting Up a Maven 2 Project in CruiseControl                              | 309        |
| 6.6       | The CruiseControl Dashboard                                                | 311        |
| 6.7       | Third-Party Tools                                                          | 312        |
| 6.8       | Conclusion                                                                 | 314        |
| <b>7.</b> | <b>LuntBuild—A Web-Based Continuous Integration Server .....</b>           | <b>315</b> |
| 7.1       | An Introduction to LuntBuild                                               | 315        |
| 7.2       | Installing LuntBuild                                                       | 315        |
| 7.3       | Configuring the LuntBuild Server                                           | 317        |
| 7.4       | Adding a Project                                                           | 319        |
| 7.5       | Using Project Variables for Version Numbering                              | 326        |
| 7.6       | Build Results Diagnostics                                                  | 328        |
| 7.7       | Using LuntBuild with Eclipse                                               | 331        |
| 7.8       | Reporting on Test Coverage in Luntbuild Using Cobertura                    | 333        |
| 7.9       | Integrating Luntbuild with Maven                                           | 341        |
| 7.10      | Conclusion                                                                 | 347        |
| <b>8.</b> | <b>Continuous Integration with Hudson .....</b>                            | <b>349</b> |
| 8.1       | An Introduction to Hudson                                                  | 349        |
| 8.2       | Installing Hudson                                                          | 349        |
| 8.3       | Managing the Hudson Home Directory                                         | 350        |
| 8.4       | Installing Upgrades                                                        | 351        |
| 8.5       | Configuring Hudson                                                         | 352        |
| 8.6       | Adding a New Build Job                                                     | 354        |
| 8.7       | Organizing Your Jobs                                                       | 359        |
| 8.8       | Monitoring Your Builds                                                     | 360        |
| 8.9       | Viewing and Promoting a Particular Build                                   | 361        |
| 8.10      | Managing Users                                                             | 362        |

|           |                                                                     |            |
|-----------|---------------------------------------------------------------------|------------|
| 8.11      | Authentication and Security                                         | 364        |
| 8.12      | Viewing Changes                                                     | 365        |
| 8.13      | Hudson Plug-Ins                                                     | 365        |
| 8.14      | Keeping Track of Test Results                                       | 366        |
| 8.15      | Keeping Track of Code Metrics                                       | 367        |
| 8.16      | Reporting on Code Coverage                                          | 368        |
| <b>9.</b> | <b>Setting Up an Instant Messaging Platform with Openfire .....</b> | <b>373</b> |
| 9.1       | Instant Messaging in a Development Project                          | 373        |
| 9.2       | Installing Openfire                                                 | 373        |
| 9.3       | Setting Up Users and Accounts on Openfire                           | 374        |
| 9.4       | Authenticating Users in an External Database                        | 374        |
| 9.5       | Authenticating Users Against a POP3 Server                          | 377        |
| 9.6       | Virtual Team Meetings with the Group Chat                           | 378        |
| 9.7       | Extended Functionality with Openfire Plug-Ins                       | 379        |
| 9.8       | Using Openfire with Continuum                                       | 379        |
| 9.9       | Using Openfire with CruiseControl                                   | 382        |
| 9.10      | Using Openfire with Luntbuild                                       | 382        |
| 9.11      | Sending Jabber Messages from a Java Application Using the Smack API | 382        |
| 9.12      | Detecting Presence Using the Smack API                              | 385        |
| 9.13      | Receiving Messages Using the Smack API                              | 385        |

---

## Part IV. Unit Testing

|            |                                                  |            |
|------------|--------------------------------------------------|------------|
| <b>10.</b> | <b>Testing Your Code with JUnit .....</b>        | <b>389</b> |
| 10.1       | JUnit 3.8 and JUnit 4                            | 389        |
| 10.2       | Unit Testing with JUnit 4                        | 390        |
| 10.3       | Setting Up and Optimizing Your Unit Test Cases   | 392        |
| 10.4       | Simple Performance Testing Using Timeouts        | 393        |
| 10.5       | Checking for Exceptions the Easy Way             | 394        |
| 10.6       | Using Parameterized Tests                        | 395        |
| 10.7       | Using assertThat and the Hamcrest Library        | 398        |
| 10.8       | JUnit 4 Theories                                 | 400        |
| 10.9       | Using JUnit 4 with Maverr2                       | 402        |
| 10.10      | Using JUnit 4 with Ant                           | 403        |
| 10.11      | Selectively Running JUnit 4 Tests in Ant         | 406        |
| 10.12      | Integration Tests                                | 408        |
| 10.13      | Using JUnit 4 in Eclipse                         | 409        |
| <b>11.</b> | <b>Next-Generation Testing with TestNG .....</b> | <b>413</b> |
| 11.1       | Introducing TestNG                               | 413        |

|                                                                       |                                                               |            |
|-----------------------------------------------------------------------|---------------------------------------------------------------|------------|
| 11.2                                                                  | Creating Simple Unit Tests with TestNG                        | 413        |
| 11.3                                                                  | Defining TestNG Test Suites                                   | 415        |
| 11.4                                                                  | The TestNG Eclipse Plug-In                                    | 416        |
| 11.5                                                                  | Using TestNG in Ant                                           | 420        |
| 11.6                                                                  | Using TestNG with Maven 2                                     | 424        |
| 11.7                                                                  | Managing the Test Lifecycle                                   | 425        |
| 11.8                                                                  | Using Test Groups                                             | 430        |
| 11.9                                                                  | Managing Dependencies                                         | 432        |
| 11.10                                                                 | Parallel Testing                                              | 435        |
| 11.11                                                                 | Test Parameters and Data-Driven Testing                       | 436        |
| 11.12                                                                 | Checking for Exceptions                                       | 437        |
| 11.13                                                                 | Handling Partial Failures                                     | 437        |
| 11.14                                                                 | Rerunning Failed Tests                                        | 438        |
| <b>12.</b>                                                            | <b>Maximizing Test Coverage with Cobertura .....</b>          | <b>441</b> |
| 12.1                                                                  | Test Coverage                                                 | 441        |
| 12.2                                                                  | Running Cobertura from Ant                                    | 442        |
| 12.3                                                                  | Checking the Code Coverage of TestNG Tests                    | 445        |
| 12.4                                                                  | Interpreting the Cobertura Report                             | 447        |
| 12.5                                                                  | Enforcing High Code Coverage                                  | 450        |
| 12.6                                                                  | Generating Cobertura Reports in Maven                         | 451        |
| 12.7                                                                  | Integrating Coverage Tests into the Maven Build Process       | 453        |
| 12.8                                                                  | Code Coverage in Eclipse                                      | 456        |
| 12.9                                                                  | Conclusion                                                    | 458        |
| <hr/>                                                                 |                                                               |            |
| <b>Part V. Integration, Functional, Load, and Performance Testing</b> |                                                               |            |
| <b>13.</b>                                                            | <b>Testing a Struts Application with StrutsTestCase .....</b> | <b>463</b> |
| 13.1                                                                  | Testing a Struts Application                                  | 464        |
| 13.2                                                                  | Introducing StrutsTestCase                                    | 465        |
| 13.3                                                                  | Mock Tests Using StrutsTestCase                               | 465        |
| 13.4                                                                  | Testing Struts Error Handling                                 | 470        |
| 13.5                                                                  | Customizing the Test Environment                              | 471        |
| 13.6                                                                  | First-Level Performance Testing                               | 471        |
| 13.7                                                                  | Conclusion                                                    | 472        |
| <b>14.</b>                                                            | <b>Integration Testing Databases with DbUnit .....</b>        | <b>473</b> |
| 14.1                                                                  | Overview                                                      | 473        |
| 14.2                                                                  | DbUnit Structure                                              | 475        |
| 14.3                                                                  | Example Application                                           | 478        |
| 14.4                                                                  | Priming the Database                                          | 479        |
| 14.5                                                                  | Verifying the Database                                        | 488        |

|            |                                                                           |            |
|------------|---------------------------------------------------------------------------|------------|
| 14.6       | Replacing Values                                                          | 493        |
| 14.7       | Alternative Dataset Formats                                               | 498        |
| 14.8       | Dealing with Custom Data Types                                            | 502        |
| 14.9       | Other Applications                                                        | 506        |
| <b>15.</b> | <b>Performance Testing with JUnitPerf</b>                                 | <b>517</b> |
| 15.1       | Introducing JUnitPerf                                                     | 517        |
| 15.2       | Measuring Performance with TimedTests                                     | 518        |
| 15.3       | SimulatingLoad with LoadTests                                             | 520        |
| 15.4       | Load-Testing Tests That Are Not Thread-Safe                               | 523        |
| 15.5       | Separating Performance Tests from Unit Tests in Ant                       | 523        |
| 15.6       | Separating Performance Tests from Unit Tests in Maven                     | 524        |
| <b>16.</b> | <b>Load and Performance Testing with JMeter</b>                           | <b>527</b> |
| 16.1       | Installing JMeter                                                         | 528        |
| 16.2       | Testing a Simple Web Application                                          | 528        |
| 16.3       | Structuring Your Test Case                                                | 534        |
| 16.4       | Recording and Displaying Test Results                                     | 537        |
| 16.5       | Using the JMeter Proxy to Record a Test Case                              | 541        |
| 16.6       | Testing Using Variables                                                   | 543        |
| 16.7       | Testing on Multiple Machines                                              | 544        |
| <b>17.</b> | <b>Testing Web Services with SoapUI</b>                                   | <b>547</b> |
| 17.1       | An Introduction to SoapUI                                                 | 547        |
| 17.2       | Installing SoapUI                                                         | 549        |
| 17.3       | Installing a Local Web Service                                            | 549        |
| 17.4       | Testing Web Services with SoapUI                                          | 551        |
| 17.5       | Load-Testing with SoapUI                                                  | 557        |
| 17.6       | Running SoapUI from the Command Line                                      | 561        |
| 17.7       | Running SoapUI from Ant                                                   | 563        |
| 17.8       | Running SoapUI from Maven                                                 | 564        |
| 17.9       | Continuous Testing                                                        | 565        |
| 17.10      | Conclusion                                                                | 566        |
| <b>18.</b> | <b>Profiling and Monitoring Java Applications Using the Sun JDK Tools</b> | <b>569</b> |
| 18.1       | The Sun JDK Profiling and Monitoring Tools                                | 569        |
| 18.2       | Connecting To and Monitoring a Java Application with jConsole             | 569        |
| 18.3       | Monitoring a Remote Tomcat Application with jConsole                      | 572        |
| 18.4       | Detecting and Identifying Memory Leaks with the JDK Tools                 | 574        |
| 18.5       | Diagnosing Memory Leaks Using Heap Dumps, jmap, and jhat                  | 579        |
| 18.6       | Detecting Deadlocks                                                       | 582        |



|                                                                 |            |
|-----------------------------------------------------------------|------------|
| <b>19. Profiling Java Applications in Eclipse .....</b>         | <b>585</b> |
| 19.1 Profiling Applications from Within an IDE                  | 585        |
| 19.2 The Eclipse Test & Performance Tools Platform              | 585        |
| 19.3 Installing TPTP                                            | 587        |
| 19.4 TPTP and Java 6                                            | 587        |
| 19.5 Basic Profiling with TPTP                                  | 587        |
| 19.6 Studying Memory Use with the Basic Memory Analysis Results | 593        |
| 19.7 Analyzing Execution Time                                   | 595        |
| 19.8 Displaying Coverage Statistics                             | 597        |
| 19.9 Using Filters to Refine Your Results                       | 597        |
| 19.10 Profiling a Web Application                               | 599        |
| 19.11 Conclusion                                                | 600        |
| <b>20. Testing Your User Interfaces .....</b>                   | <b>603</b> |
| 20.1 Testing Your Web Application with Selenium                 | 603        |
| 20.2 Testing Swing GUIs with FEST                               | 633        |
| 20.3 Conclusion                                                 | 642        |

---

## Part VI. Quality Metrics Tools

|                                                                                                |            |
|------------------------------------------------------------------------------------------------|------------|
| <b>21. Detecting and Enforcing Coding Standards with Checkstyle .....</b>                      | <b>647</b> |
| 21.1 Using Checkstyle to Enforce Coding Standards                                              | 647        |
| 21.2 Using Checkstyle in Eclipse                                                               | 649        |
| 21.3 Customizing Checkstyle Rules in Eclipse                                                   | 652        |
| 21.4 Customizing Checkstyle Rules Using the XML Configuration Files                            | 655        |
| 21.5 Customizing Checkstyle: Common Rules That You Can Do Without, and Some That You Could Use | 657        |
| 21.6 Defining Rules for Source Code Headers with Checkstyle                                    | 660        |
| 21.7 Suppressing Checkstyle Tests                                                              | 662        |
| 21.8 Using Checkstyle with Ant                                                                 | 663        |
| 21.9 Using Checkstyle with Maven                                                               | 664        |
| <b>22. Preemptive Error Detection with PMD .....</b>                                           | <b>667</b> |
| 22.1 PMD and Static Code Analysis                                                              | 667        |
| 22.2 Using PMD in Eclipse                                                                      | 667        |
| 22.3 Configuring PMD Rules in Eclipse                                                          | 670        |
| 22.4 More on the PMD Rulesets                                                                  | 670        |
| 22.5 Writing Your Own PMD Ruleset                                                              | 674        |
| 22.6 Generating a PMD Report in Eclipse                                                        | 675        |
| 22.7 Suppressing PMD Rules                                                                     | 676        |
| 22.8 Detecting Cut-and-Paste with CPD                                                          | 677        |

|            |                                                                            |            |
|------------|----------------------------------------------------------------------------|------------|
| 22.9       | Using PMD in Ant                                                           | 679        |
| 22.10      | Using PMD in Maven                                                         | 681        |
| <b>23.</b> | <b>Preemptive Error Detection with FindBugs .....</b>                      | <b>685</b> |
| 23.1       | FindBugs: A Specialized Bug Killer                                         | 685        |
| 23.2       | Using FindBugs in Eclipse                                                  | 687        |
| 23.3       | Selectively Suppressing Rules with FindBug Filters                         | 688        |
| 23.4       | Using FindBugs Annotations                                                 | 691        |
| 23.5       | Using FindBugs in Ant                                                      | 692        |
| 23.6       | Using FindBugs in Maven                                                    | 695        |
| 23.7       | Conclusion                                                                 | 696        |
| <b>24.</b> | <b>Inspecting the Results—Semiautomated Code Review with Jupiter .....</b> | <b>697</b> |
| 24.1       | Introducing Jupiter—A Code Review Tool for Eclipse                         | 697        |
| 24.2       | Installing Jupiter in Eclipse                                              | 698        |
| 24.3       | Understanding the Jupiter Code Review Process                              | 699        |
| 24.4       | Conducting Personal Code Reviews                                           | 700        |
| 24.5       | Configuration                                                              | 701        |
| 24.6       | Setting Up Default Configuration Values                                    | 706        |
| 24.7       | Individual Reviews                                                         | 706        |
| 24.8       | Team Review                                                                | 708        |
| 24.9       | Rework Phase                                                               | 712        |
| 24.10      | Jupiter Behind the Scenes                                                  | 714        |
| 24.11      | Conclusion                                                                 | 715        |
| <b>25.</b> | <b>Sharpen Your Focus with Mylyn .....</b>                                 | <b>717</b> |
| 25.1       | Introduction to Mylyn                                                      | 717        |
| 25.2       | Installing Mylyn                                                           | 718        |
| 25.3       | Tracking Tasks and Issues                                                  | 719        |
| 25.4       | Interacting with Task Repositories                                         | 722        |
| 25.5       | Focusing on a Task with Context Management                                 | 726        |
| 25.6       | Using the Eclipse Change Sets                                              | 728        |
| 25.7       | Sharing Context with Other Developers                                      | 729        |
| 25.8       | Conclusion                                                                 | 731        |
| <b>26.</b> | <b>Monitoring Build Statistics .....</b>                                   | <b>733</b> |
| 26.1       | QALab                                                                      | 733        |
| 26.2       | Source Code Management Metrics with StatSCM                                | 741        |
| 26.3       | Statistics in Ant with StatSVN                                             | 742        |

---

## Part VII. Issue Management Tools

|                                                           |                |
|-----------------------------------------------------------|----------------|
| <b>27. Bugzilla .....</b>                                 | <b>747</b>     |
| 27.1 An Introduction to Bugzilla                          | 747            |
| 27.2 Installing Bugzilla                                  | 747            |
| 27.3 Setting Up Your Bugzilla Environment                 | 751            |
| 27.4 Managing User Accounts                               | 752            |
| 27.5 Restricting Access Using User Groups                 | 754            |
| 27.6 Configuring a Product                                | 756            |
| 27.7 Tracking Progress with Milestones                    | 758            |
| 27.8 Managing Groups of Products with Classifications     | 758            |
| 27.9 Searching for Bugs                                   | 759            |
| 27.10 Creating a New Bug                                  | 761            |
| 27.11 The Lifecycle of a Bugzilla Bug                     | 762            |
| 27.12 Scheduling Notifications (Whining)                  | 764            |
| 27.13 Customizing Fields in Bugzilla                      | 766            |
| 27.14 Conclusion                                          | 767            |
| <br><b>28. Trac—Lightweight Project Management .....</b>  | <br><b>769</b> |
| 28.1 An Introduction to Trac                              | 769            |
| 28.2 Installing Trac                                      | 770            |
| 28.3 Setting Up a Trac Project                            | 772            |
| 28.4 Running Trac on the Standalone Server                | 774            |
| 28.5 Setting Up Trac As a Windows Service                 | 775            |
| 28.6 Installing Trac on an Apache Server                  | 776            |
| 28.7 Administrating the Trac Site                         | 777            |
| 28.8 Managing User Accounts                               | 779            |
| 28.9 Tailoring the Trac Web Site: Using the Wiki Function | 783            |
| 28.10 Using the Trac Ticket Management System             | 787            |
| 28.11 Updating Trac Issues from Subversion                | 790            |
| 28.12 Customizing Trac Ticket Fields                      | 792            |
| 28.13 Setting Up Email Notifications                      | 793            |
| 28.14 Reporting Using Trac Queries and Reports            | 795            |
| 28.15 Managing Progress with Trac Roadmaps and Timelines  | 797            |
| 28.16 Browsing the Source Code Repository                 | 799            |
| 28.17 Using RSS and ICalendar                             | 802            |
| 28.18 Customizing a Wiki Page with Python                 | 803            |
| 28.19 Conclusion                                          | 804            |

---

## Part VIII. Technical Documentation Tools

|                                                                       |            |
|-----------------------------------------------------------------------|------------|
| <b>29. Team Communication with the Maven 2 Project Web Site .....</b> | <b>807</b> |
| 29.1 The Maven 2 Project Web Site As a Communication Tool             | 807        |
| 29.2 Setting Up a Maven Site Project                                  | 808        |
| 29.3 Integrating Reports into Your Site                               | 813        |
| 29.4 Creating a Dedicated Maven Site Project                          | 818        |
| 29.5 Defining the Site Outline                                        | 819        |
| 29.6 The Maven Site Generation Architecture                           | 821        |
| 29.7 Using Snippets                                                   | 825        |
| 29.8 Customizing the Look and Feel of Your Site                       | 826        |
| 29.9 Distributing Your Site                                           | 829        |
| <b>30. Automatically Generating Technical Documentation .....</b>     | <b>831</b> |
| 30.1 Visualizing a Database Structure with SchemaSpy                  | 831        |
| 30.2 Generating Source Code Documentation with Doxygen                | 840        |
| 30.3 Embedding UML Diagrams in Your Javadoc with UmlGraph             | 849        |
| 30.4 Conclusion                                                       | 853        |
| <b>Bibliography .....</b>                                             | <b>855</b> |
| <b>Index .....</b>                                                    | <b>857</b> |