

# Preface

---

Our goal in writing this book was to help build the community around the Eclipse Web Tools Platform (WTP) Project. We decided to write this book soon after WTP was approved by the Eclipse Foundation. At that time, the project was in its formative stages and there was virtually nothing written about WTP. We believed that a book on how to use and extend WTP would help promote its adoption.

We naively hoped that we would have this book finished soon after WTP 0.7 was released in July 2005. However, since we were all actively engaged in developing WTP, work on this book got delayed. Also, many significant changes in the design of WTP were planned, so we felt it was better to have the book describe the next major version, WTP 1.5, which was part of the Eclipse 3.2 Callisto simultaneous release in June 2006.

Allowing WTP to mature also gave us more time to develop and refine the material in this book. Much of the material in this book has been test-driven at several major software development conferences including EclipseCon, EclipseWorld, Rational Software Development Conference, and Colorado Software Summit. Attendees at those events provided valuable feedback that has improved the content of this book.

Since the WTP 1.5 release, there has been increasing adoption of WTP by both commercial and Open Source tool developers. This activity has generated a stream of maintenance releases. As we went into production, this book accurately reflected the content of WTP 1.5.2, but by the time it appears in print, the latest release should be WTP 1.5.3. However, each maintenance release should only contain bug fixes and not affect the user interface. This book should therefore also be accurate for WTP 1.5.3 and future maintenance releases. And although WTP 2.0, which is planned for June 2007, will certainly contain many enhancements, we expect that most of the content of this book will still be valid.

## About This Book

This book is divided into four parts: Getting Started, Java Web Application Development, Extending WTP, and Products and Plans.

In Part I, Getting Started, we introduce you to WTP. We give a brief overview of the history and architecture of the project and discuss how you can contribute to its development. By being an active contributor as well as a user, you can help improve WTP and ensure its long-term success. We also introduce you to League Planet, a fictitious amateur sport Web site, which serves as the inspiration for the programming examples in the rest of the book. Next we take you on a Quick Tour of WTP in which you build a simple Web application that includes dynamic content generated by servlets and JSPs running on Tomcat, JDBC database access to Derby, and Web services running on Axis. We conclude with a detailed discussion of how to install WTP and tailor it to your needs using its many preferences. At the end of this part, you'll be able to start building your own Java Web applications with WTP.

Part II, Java Web Application Development, is for Java Web application developers. We describe the architecture of Java Web applications and how to build them using WTP. We start with a discussion of how to set up your project, including the use of Maven for automated builds. We then discuss architecture in some detail. Java Web applications have a multi-tiered architecture, and each of the presentation, business logic, and persistence tiers is addressed in its own chapter. The presentation tier chapter covers tools for HTML, CSS, JavaScript, XML, DTD, and XSLT. The business logic tier chapter discusses tools for EJBs and XDoclet. The persistence tier chapter describes tools for SQL. Next we focus on developing Web services, including tools for SOAP, WSDL, XSD, and UDDI. We close with a discussion of testing, including JUnit, Cactus, HttpUnit, and the Eclipse Test and Performance Tools Platform (TPTP).

In Part III, Extending WTP, we shift attention to developing Eclipse plug-ins that extend WTP. This part of the book is aimed at tool developers. WTP contains many plug-ins and extension points, so the coverage here serves mainly to illustrate the process. A comprehensive treatment of all the APIs in WTP would itself fill several books. We start with the important example of adding a new server runtime to WTP, and illustrate this by adding support for GlassFish, the reference implementation for Java Enterprise Edition 5 (Java EE 5). Next, we show how to add support for new file types and do so for DocBook, the XML format used for authoring books (such as this one). We follow that by describing how to support new WSDL extensions and add a new SOAP binding as an example. We conclude this subject by extending the URI resolution framework, which enables XML processors to locate resources.

The book wraps up with Part IV, Products and Plans. We begin with a brief survey of commercial and Open Source Eclipse-based Web development products

that can be used with WTP. Although WTP contains a core set of useful tools, it is also a platform intended to be built on by others. After you master WTP, you may find that your tool needs are not fully satisfied. Perhaps you want to develop with Struts, Hibernate, or Spring. Or you may want to use a different Web development language, such as PHP, Python, or Ruby, in conjunction with Java. Fortunately, there are many products available to round out your Web development IDE. We end the book with a preview of functions we expect to be added to WTP in future releases. WTP is currently hosting subprojects for JavaServer Faces (JSF), Java Persistence Architecture (Dali), and AJAX (ATF). In addition, WTP is planning tighter integration with other Eclipse projects, as well as support for Java EE 5. Of course, the future of WTP largely depends on you. By becoming an active user and contributor, you will influence the continuing support and evolution of WTP.

## Audience

This book is primarily written for Java Web application developers. We assume that you have a working knowledge of Java programming and some experience using Eclipse. There are many excellent books available that cover both topics. Some experience in Java Web application development is also desirable. We have made an attempt to introduce the subject of Java Web application development in addition to describing the tools available in WTP. Although this book deals with WTP, it will also be of use to users of products built on WTP. And remember, one of the best ways you can contribute to WTP is by reporting bugs. If you hit a bug while using WTP, please report it to the Eclipse Bugzilla system at

[https://bugs.eclipse.org/bugs/enter\\_bug.cgi?product=web+tools](https://bugs.eclipse.org/bugs/enter_bug.cgi?product=web+tools)

This book also includes material for Eclipse plug-in developers who want to extend WTP. Experience in plug-in development is assumed. Several available books cover the topic of Eclipse plug-in development for those who need some background information. Although we expect commercial and Open Source projects to extend WTP, we also expect individuals to do so. If you develop a cool plug-in that fits within the scope of the WTP charter, please consider contributing it to WTP. To do so, start by sending a note to the WTP developers mailing list at

[<wtp-dev@eclipse.org>](mailto:wtp-dev@eclipse.org)

## Sample Code

The Web site for this book is located at

<http://www.eclipsewtp.org>

All of the example code used throughout this book can be downloaded from there. The site will also provide an errata list, and other news related to the book.

The following Eclipse components are required to run the examples in this book:

- Eclipse Software Development Kit (SDK), Version 3.2
- Eclipse Modeling Framework (EMF), Version 2.2
- Graphical Editing Framework (GEF), Version 3.2
- Java Edit Model (JEM), Version 1.2
- Web Tools Platform (WTP), Version 1.5

All of the above are available from

<http://www.eclipse.org/webtools/downloads>

## Conventions

We use a sans serif font for user interface elements such as menu items, buttons, and labels. We use a monospace font for programmatic elements such as file names, source code listings, URLs, package names, and XML content. Examples of these conventions are listed below.

- buttons, e.g., **Submit**
- class names, e.g., `LoginServlet`
- code, e.g., `out.println("Hello, world");`
- email addresses, e.g., `<feedback@eclipsewtp.org>`
- file names, e.g., `web.xml`
- labels, e.g., **Servers**
- menu items, e.g., **File ▶ New ▶ Project**
- method names, e.g., `getParameter()`
- URLs, e.g., `http://eclipse.org`

## Feedback

We've set up an e-mail address to receive feedback about this book. Please send your comments on this book to

`<feedback@eclipsewtp.org>`