

# Web Tools Platform (WTP) 3.1

## *for the Galileo Simultaneous Release Review*

Full Release Review Materials

June 10, 2009

Prepared by David Williams and sub-project leads

### Table of Contents

Introduction and Purpose.....	2
History.....	2
Project Organization.....	2
PMC Organization.....	3
WTP 3.1 Goals and Plans.....	4
Features.....	5
Common Components.....	5
Server Tools.....	5
Source Editing.....	5
Web Service Tools.....	6
JEE Tools.....	6
EJB Tools.....	7
JSF Tools.....	8
Dali.....	9
Non-Code Aspects.....	10
Developer and API Documentation.....	10
End-User Documentation and Examples.....	10
APIs.....	11
Architectural Issues.....	12
Tool Usability.....	13
End-of-Life.....	14
Quality (Bugzilla).....	15
Focused Quality Activities.....	15
Bugzilla statistics.....	15
Standards.....	16
W3C and OASIS standards.....	16
JCP standards.....	16
UI Usability.....	17
Schedule.....	18
Communities.....	19
Committers and Contributors.....	19
End-User Community.....	20
Adopter Community.....	20
IP Issues.....	21
IP Checks.....	21
IP Log and Documentation.....	21

## Introduction and Purpose

This document is to fulfill the requirements of the [Eclipse Release Review](#)<sup>1</sup> for WTP 3.1 planned for release at the end of June, 2009.

Communication Channel: wtp-dev mailing list (See <https://dev.eclipse.org/mailman/listinfo>).

## History

The Eclipse Web Tools Platform Project was originally proposed in 2004 by ObjectWeb, IBM and others. The Eclipse Foundation creation review was in June, 2004 with full time development since October, 2004. The original code contributions were from IBM and Eteration (“ObjectWeb Lomboz”). Since then several other large contributors have joined the effort, including SAS, BEA, Oracle and SAP, and others and several new subprojects formed, such as Dali and JSF.

### Previous Releases

- WTP 0.7 July, 2005 and subsequent 0.7.1
- WTP 1.0 December 2005 and subsequent 1.0.1, 1.0.2, 1.0.3
- WTP 1.5 June, 2006 and subsequent 1.5.1, 1.5.2, 1.5.3, 1.5.4, 1.5.5
- WTP 2.0 June, 2007 and subsequent 2.0.1, 2.0.2
- WTP 3.0 June, 2009 and subsequent 3.0.1, 3.0.2, 3.0.3, 3.0.4, 3.0.5

## Project Organization

Below is the list of current sub-projects and project leads as of June, 2009, that are part of this release.

Project	Lead
Common: tools and infrastructure not directly related to web tools, but required by Web Tools Platform	Konstantin Komissarchik, Oracle
Dali (JPA Tools): infrastructure and tools for JPA applications	Neil Hauge, Oracle
EJB Tools: EJB creation wizards, preferences, future annotation tools	Kaloyan Raev, SAP
Java EE Tools: Common Project Infrastructure, JEE models, preferences, classpath model, publish api, refactoring	Chuck Bridgham, IBM
JSF Tools: infrastructure and tools for Java Server Faces.	Raghu Srinivasan, Oracle
Server Tools: tools and infrastructure to define and interact with servers.	Tim Deboer, IBM
Source Editing: xml, dtd, xsd, xsl, sse infrastructure, html, css, javascript, jsp	Nitin Dahyabhai, IBM
Web Services: Web services wizards and frameworks, Axis1 & Axis2 support, Web Services Explorer, WSDL Editor	Keith Chong, IBM

<sup>1</sup> [http://www.eclipse.org/projects/dev\\_process/release-review.php](http://www.eclipse.org/projects/dev_process/release-review.php)

## PMC Organization

Our Project Management Committee, as of June, 2009, is made up of 6 members, each having a WTP-wide management role, in addition to what ever project-specific roles they have. In the execution of their tasks within these roles, the PMC members will form groups, organize meetings, etc., to accomplish their goals. In other words, they don't do all the work ... they just manage it!

<b>Member</b>	<b>Role</b>
David Williams, IBM	PMC Lead, and Planning Role
Tim Deboer, IBM	Architecture
Neil Hauge, Oracle	Quality
Kaloyan Raev, SAP	User Experience
Raghu Srinivasan, Oracle	Requirements
Naci Dai, Eteration	Education

## **WTP 3.1 Goals and Plans**

Our Requirements Planning, Process, and Documentation is managed by Raghu Srinivasan, with substantial participation from each project lead.

We provide a standard-format [WTP Project Plan](#) which is updated every milestone as items are delivered or deferred, as part of our iterative development cycle.

Detailed requirements, plans, and progress are tracked (mostly) via Bugzilla, with 'plan' added as a keyword.

Themes and high-level requirements are coordinated through Eclipse Requirements Council and Planning Council representation

## Features

We document all our [new and noteworthy items](#)<sup>2</sup> for each milestone, for every release. The following summarizes the functionality provided by each sub-project with emphasis on what's new this release.

### Common Components

- Facets  
Provides a technique and UI for modeling server and runtime capabilities to associate with projects. This release moved some provisional API to be API, and usability improvements made in the UI for selecting facets for projects.
- Validation Framework  
Provides preferences and APIs to provide domain specific validation for resources and projects. Provided official API this release, instead of only provisional.
- Snippets View
- Internet Cache

### Server Tools

- Support for Application Servers of several types (JEE, HTTPD)
- Configure, publish, start/stop, debug
- Supports deploy, debug, project restart on Java EE runtimes
- Generic server adapter
  - XML based configuration files for quick setup
  - Includes JOnAS, WebLogic, WebSphere, Oracle Application Server, JBoss
- Custom (Java) server adapter for total control
  - Includes Tomcat (included), Geronimo adapters (downloadable)

### Source Editing

- JavaScript  
Substantial improvements in JavaScript tools and editing were made in this release.
- Web Language Tools
  - HTML source editor
  - CSS source editor
  - JSP editor including syntax highlighting, code assist for HTML, Java, JSF, EL's, JavaScript, taglibs, JSR-45 compliant debugging
- XML Language Tools
  - XML source editor
  - XSD editor - Graphical and source editing
  - DTD source editor
  - XSL Editing and Tools. New this release. This component has been incubating for 1.5 years so it quite mature, graduated last April, and is being released at a 1.0 level with WTP 3.1.
- Structured Source Editing (SSE) framework

---

<sup>2</sup> <http://www.eclipse.org/webtools/development/news/main.php>

## Web Service Tools

- Extensible Web Service Wizards
- Creation of servlet based Web service using Axis1 and Axis2 runtimes
- Extension points for finding, creating and testing Web service
- Integrated into Java EE Navigator
- JAX-RPC codegen
- JSR 109 deployment ready
- Web services Ant tasks
- Web Services Explorer - UDDI, WSDL and WSIL pages
- WSDL Editor - graphical and source modes, integrated with XSD Editor
- WS-I Test Tools - Validate WSDL and SOAP for WS-I compliance
- Models for Web services deployment descriptors
- Service policies

## JEE Tools

- New this release:
  - New wizard page in Java EE project wizards for configuring Java facets
  - Better compliance with the UI guidelines for the Dynamic Web and EAR project wizards
  - Improved default layout of the Java EE perspective
  - Java EE preferences page introduced
  - Avoid having the .JETemitter system project in workspace by default
  - Easier creation of new Servlets from existing Servlets and JSP pages
  - Easier creation of new Filters from existing Servlets and JSP pages
- Continuing to support and improve:
  - New wizards for creating Web artifacts: Servlet Filters and Application Lifecycle Listeners
  - Usability enhancements in the toolbar of the Java EE perspective
  - EAR 5 Bundled Libraries support
  - Java EE Deployment Descriptor nodes in the Project Navigator view
  - Improved Java EE classpath management/UI
  - Ability to read Java EE models from binary archives
  - Java EE Core Models and Model Provider Framework
  - Natures and Builders
  - Java EE Views and Navigators
  - Java EE Projects and Modules
  - Support for WAR, EJB-JAR, EAR, etc

- Java EE Navigator view
- Ability to target on different servers

## **EJB Tools**

- New this release:
  - Better compliance with the UI guidelines for the EJB project wizard
- Continuing to support and improve
  - New wizards for creating EJB 3.0 artifacts: Session Beans and Message-Driven Beans

## JSF Tools

- New this release:
  - Support for Apache MyFaces Trinidad components
  - New Library Management facility
- Continuing to support and improve:
  - Web Page Editor
    - Multi-page Editor
    - Visual JSF-JSP Page Designer
    - Support for JSF RI components
    - Extensibility framework to simplify adding support for other component libraries
    - Preview Page
  - Enhanced Source Editor gives content assists and provides both syntax and semantic validations
  - JSF Validation
  - Faces Configuration Model, Editor and Wizards
    - Multi-page Editor
    - Graphical diagram editor for navigational rules
    - EMF model of the application configuration resource file
  - Extensible Frameworks
    - Design-time Meta-data Framework
    - Design-time Tag Processor
    - Design-time Application Manager
    - JSF Application Configuration Manager
  - Support for JavaServer Faces 1.1 and 1.2 versions



## Dali

- New and Improved this release
  - Multiple XML Mapping File support
  - New and improved Entity Generation from Tables wizard
  - Improved Joining Strategy UI
  - Table-per-concrete-class inheritance support
  - New Library Management facility
  - EclipseLink features
    - New Platform for EclipseLink 1.1
    - Caching support
    - Entity and Attribute mapping support
    - Converter support
    - EclipseLink XML Mapping file support
- Continuing to support:
  - Comprehensive, rich UI and validation for JPA metadata
  - Provisional API for JPA model
  - Extensible persistence.xml Editor
  - Project Explorer Content for JPA projects
  - New XML Mapping File wizard
  - New Entity Wizard
  - WTP (Facet) integration
  - DTP Integration for DB metadata
  - XML Mapping Descriptor editing support (JPA1.0 orm.xml)
  - Code completion for JPA annotation values
  - Annotated Java and ORM XML context based defaulting

## Non-Code Aspects

### Developer and API Documentation

Project-wide architectural overview (on website)

- Describes nature and relationship of components
- Presentations and tutorials provide drill down on selected topics (such as server definition)

Component overview (on website)

- Describes operation of an individual component and relationships among its parts, lifecycle issues, and other emergent properties of component

JavaDoc Package documentation

- Describes contents and inter-relationship of package contents

Per-file JavaDoc

- Conventional JavaDoc guidelines apply; scope is the class/interface being documented and its immediate surface area

Extension point documentation

- Provided with and as part of the API and JavaDoc documentation.

### End-User Documentation and Examples

- Tutorials and presentation materials available on our [WTP Project Community website](http://www.eclipse.org/webtools)<sup>3</sup>. Some major additions were added this release and provided as re-usable education materials to teach web development.
- Contributed to conference presentations (EclipseCon, JavaOne, EclipseWorld), articles, etc.
- Two known books: Pro Eclipse JST, Eclipse Web Tools Platform.
- WTP downloads and installations includes end-user documentation. Additionally, this same documentation will be available on the internet, via an Eclipse info-center provided by the Eclipse Foundation.

---

<sup>3</sup> <http://www.eclipse.org/webtools>

## APIs

In general we provide APIs according to [Eclipse Quality API standards](#)<sup>4</sup>. But, we are aware that we (WTP) still do not provide enough APIs and still have too large a “provisional debt”.

We have published an [API Policy](#)<sup>5</sup> document that describes how we protect some non-API so that adopters can invest with some assurance of continuity, but also detail the limits to that policy, so that eventually we can provide complete API.

---

4 [http://www.eclipse.org/projects/dev\\_process/eclipse-quality.php](http://www.eclipse.org/projects/dev_process/eclipse-quality.php)

5 [http://wiki.eclipse.org/WTP\\_API\\_Policy](http://wiki.eclipse.org/WTP_API_Policy)

## Architectural Issues

- Tim Deboer manages WTP Architecture for this release, and has periodically held [work-group meetings](#)<sup>6</sup> to discuss and decide specific architectural.
- We still have a few cases where we rely on non-API from the Eclipse Platform and JDT, this will be addressed in future releases.

---

6 [http://wiki.eclipse.org/WTP\\_Architecture\\_Working\\_Group](http://wiki.eclipse.org/WTP_Architecture_Working_Group)

## Tool Usability

WTP provides a lot of functionality to web developers as indicated by the number of downloads and the traffic on the webtools newsgroup. It does emphasize Java based web development, even though it has some tools for pure HTML, CSS, XML and JavaScript development.

All WTP subprojects have “Ease of Use” in their plan for this release to address usability issues.

Several WTP subprojects made [UI walk-throughs](#) in front of the UI Best Practices Working Group. The feedback was useful for improving the usability in this release.

Another way we contribute to the overall usefulness experience to Eclipse end-users is to participate in the EPP packaging project, by “owning” the Eclipse IDE for Java EE Developers package. We have improved the usability of this package this release, by adding a custom welcome page which links to more information and samples.

## End-of-Life

- Two old, previously deprecated areas were removed this release:
  - Cactus unit test support (no equivalent functionality in WTP, though TPTP has similar functionality).
  - The old 'JavaScript' component (It has been replaced by the JSDT component).

## Quality (Bugzilla)

Neil Hauge manages the general, overall measurement and monitoring of our quality and bug handling.

### Focused Quality Activities

In order to make sure we focused on quality of particular types, that are important to our project's health, we devoted part of each of our weekly status meetings to discussing and reducing our bug backlog in specific areas.

- First we focused on older bugs that had a severity of “Major” or higher to ensure that these potentially serious issues were being addressed, or at least correctly categorized.
- Next we focused on “bugs with patches attached” to make sure we were being responsive to non-committers efforts to improve parts of WTP that was important enough to them to contribute a patch.
- Finally we focused on reviewing newer bugs that had a severity of “Major” or higher to ensure that these potentially serious issues were being addressed, or at least correctly categorized.

### Bugzilla statistics

The statistics in this table reflect overall Bugzilla entries (since project inception) with current snapshot, obtained on May 29, 2009 (4 weeks before release).

Total bugs	20963
Total Resolved/Closed bugs	17157
Total Open bugs	3806
Blocker/Critical	8
Major	204

The statistics in this table reflect activity since the previous release (for the period May 20th, 2008 to May 29, 2009)

Bugs opened	3773
Bugs resolved	3117
Fixed	2318
Invalid	157
Wontfix	133
Duplicate	313
Worksforme	173
Not Eclipse	23

We interpret these statistics as demonstrating an alive-and-well project: many bugs open, many fixed, approximately 60 per week. Our backlog increased this year by about 21%, although 1/3 of this new backlog is attributed to Enhancement requests.

## Standards

### W3C and OASIS standards

- HTML 4.01, XHTML 1.0 / 1.1, XML Catalog 1.0, CSS 2.0, ECMAScript 262
- XML 1.0, XSD 1.0, WSDL 1.1, WS-I Basic Profile 1.1
- SOAP 1.1, WS-I Attachment Profile 1.0.

### JCP standards

- J2EE 1.2 / 1.3 / 1.4: Servlet, JSP, EJB, JAX-RPC, JSR109, JSR045, JSR109, JSR921
- EE5: Basic support
- JDBC 2.1
- JSR 220: EJB 3.0
- JSR 127: JSF 1.1
- JSR 252: JSF 1.2



## UI Usability

We are familiar with, and follow, the [Eclipse User Interface Guidelines](#) .<sup>7</sup>

We do have some contributors that often open bugs for Accessibility and National Language issues so we can fix those bugs before end-users encounter them.

We participated in several UI walk throughs, this release.

---

<sup>7</sup> [http://wiki.eclipse.org/index.php/User\\_Interface\\_Guidelines](http://wiki.eclipse.org/index.php/User_Interface_Guidelines)

## **Schedule**

WTP followed the simultaneous release schedule, and delivered all milestones on time.

We plan to participate in simultaneous maintenance as well.

WTP may elect to deliver additional maintenance, tech preview milestones, or off-cycle releases in addition, if adopter or user requests warrant.

In addition to providing predictable milestones and releases, we also frequently “step up” to building and testing with our prerequisite software (usually weekly) so that we can find bugs early and get fixes in the platform and other prerequisite on behalf of the whole release train.

# Communities

## Committers and Contributors

Many active committers from several companies (including individuals)

- Committer elections and removals have followed charter principles
- Continuing to recruit additional contributors (organizations and individuals)
- Many, diverse, Contributing Organizations (current and previous): BEA, Eteration, Exadel, IBM, Innoopract, JBoss, ObjectWeb, Oracle, SAS, Thales, University of Karlsruhe, SAP, Sybase

Open communications via mailing lists and newsgroups

- Mailing lists: PMC, wtp-dev, wtp-releng, JSF, ATF, Dali,
- Meetings, meeting, meetings – numerous, open, and documented
- Weekly PMC, weekly dev status, requirements as needed, bi-weekly architecture
- PMC minutes available on website: [http://www.eclipse.org/webtools/development/index\\_pmc\\_call\\_notes.php](http://www.eclipse.org/webtools/development/index_pmc_call_notes.php)
- Weekly status telecon minutes available on website: [http://wiki.eclipse.org/WTP\\_Development\\_Status\\_Meetings](http://wiki.eclipse.org/WTP_Development_Status_Meetings)

Open and inclusive release planning and tracking processes

- Bugzilla used to request and track all defects, enhancements, and milestone plans
- Additional reports (defect summaries, test stats, etc) used to enhance planning / tracking

All contributions made directly to Eclipse CVS

- Continuous, weekly integration builds, and release builds available to the community

Coordination/cooperation with other Eclipse projects

- Platform, JDT, DTP, TPTP, EMF, GEF, PDT

Leverage other open source technologies in the project

- Participate in Orbit

## End-User Community

Substantial WTP download activity for milestones and releases

- WTP is one of the most popular downloads

Substantial website content

- Download links, New & Noteworthy, mailing lists, presentation DB
- Tutorials, documentation, presentation summary,

Evangelism and outreach in the market & broader community

- Website lists WTP events (conferences, etc.)
- Multiple commercial implementations
- Working with Eclipse and industry press to promote WTP
- Presence at EclipseCon, Eclipse World, others

## Adopter Community

Many, known commercial adopters

- IBM – Rational Application Developer
- Oracle WebLogic Studio
- Eteration – Lomboz
- Genuitec – MyEclipse
- Innoopract – Yoxos
- Exadel – Exadel Studio
- Jboss – JBoss Developer Studio
- SAP – NetWeaver Studio

Many Server Adapters available, another sign of diversity:

Open Source

- Apache Tomcat
- Apache Geronimo
- JBoss
- Jetty
- ObjectWeb JONAS
- Glassfish

Commercial

- IBM WebSphere
- Pramati Server
- Oracle WebLogic Application Server

## IP Issues

### IP Checks

Project Leads have all proof-read, double checked and confirmed the following:

- About files and use licenses are in place as per the Guidelines to Legal Documentation.
- All contributions (code, documentation, images, etc) has been committed by individuals who are either Members of the Foundation, or have signed the appropriate Committer Agreement. In either case, these are individuals who have signed, and are abiding by, the Eclipse IP Policy.
- All significant contributions have been reviewed by the Foundation's legal staff. Include references to the IPZilla numbers of all clearances.
- All non-Committer code contributions, including third-party libraries, have been documented in the release and reviewed by the Foundation's legal staff. Include references to the IPZilla numbers of all clearances.
- All Contribution Questionnaires have been completed
- The "provider" field of most feature is set to "Eclipse Web Tools Platform" (Note: many were left as the previous "Eclipse.org" so as not to change too much at the last minute, but these fields will be made consistent next release.
- The "copyright" field of each feature is set to the copyright owner (the Eclipse Foundation is rarely the copyright owner).
- Any third-party logos or trademarks included in the distribution (icons, help file logos, etc) have been licensed under the EPL.
- Any fonts or similar third-party images included in the distribution (e.g. in PDF or EPS files) have been licensed under the EPL.

### IP Log and Documentation

<In Progress>

- A list of third party software distributed with WTP, including information on the license and a link to the WTP CQ.
- The name of every committer for this release
- The name of every non-committer who contributed code via Bugzilla entries, with bug numbers.

Our IP Log contains a detailed description of our dependencies on third party software that is not re-distributed with WTP. In summary:

- Users can install their own Application Servers, where server adapters have been provided.
- EJB 2.1 developers can install Xdoclet for "old style" annotations
- Axis2 developers will need to install their own Axis2 runtime, if it is not part of their application server.
- JSF developers need to provide a JSF runtime and component libraries, if it is not part of their application server.
- JPA developers need to provide their own JPA runtime, if it is not part of their application server.