



Release Review: AJDT 1.4.1 and AspectJ 1.5.3

Matt Chapman,
Andrew Clement
October 10, 2006



Introduction

- Co-ordinated release of:
 - AJDT 1.4.1 for Eclipse 3.2
 - AspectJ 1.5.3
- AJDT includes AspectJ
- AspectJ also made available separately for command-line and build system use (Ant)
- Additional maintenance releases to update the version of AspectJ in AJDT releases for older Eclipse versions:
 - AJDT 1.3.3 for Eclipse 3.1
 - AJDT 1.2.3 for Eclipse 3.0



Features

- AJDT:
 - Ability to use PDE build process with AspectJ plug-ins
 - Show crosscutting relationships when binary weaving
 - Participate in Java type rename refactorings to update references in aspects
 - Bug fixes
- AspectJ:
 - Pipelining compiler to reduce memory usage
 - For source compilation, memory reduced by 60%
 - Load time weaving enhancements
 - Improved serviceability – trace infrastructure put in place
 - Bug fixes



Non-Code aspects

- Existing documentation and tutorials etc have been updated where necessary for these releases
- AJDT user strings are externalized via Eclipse message bundles



APIs

- AJDT:
 - Mainly internal API
 - Small number of public packages:
 - Few clients
 - Considered provisional / incomplete
- AspectJ:
 - Public interfaces for IDEs to use for integration:
 - Compilation invocation, feedback on weaving, Ast manipulation, AspectJ code parsing



Tool Usability

- The tools are mature and have been used in a number of production environments already



Architectural Issues

- AJDT
 - No architectural changes are being made in this release
- AspectJ
 - Moved from a 'compile everything/weave everything' model to a pipeline model 'compile one file, weave one file'
 - Drastic reduction in maximum memory required
- The projects are extensible in a number of ways:
 - AJDT:
 - Several tools have been built on top of AJDT
 - The Visualiser and Cross References view components are highly extensible and can be reused independently
 - AspectJ:
 - Message processing can be handed off to a user defined class
 - New pointcuts can be defined through a registration mechanism



End-of-Life

- No APIs or significant user features from previous releases are being end-of-life'd in this release



Bugzilla

- AJDT since previous 1.4 release:
 - 64 Bugs raised
 - 45 Bugs resolved (27 involving a code fix)
 - Current state: 0 Blocking 0 Critical
- AspectJ since previous 1.5.2 release:
 - 86 Bugs raised (and 31 enhancements)
 - 54 Bugs resolved (and 13 enhancements implemented)
 - Current state: 0 Blocking 0 Critical



Standards

- J2SE
 - AJDT runs on J2SE 1.4 and 1.5
 - The AspectJ compiler will run on JS2E 1.3 and higher
 - The code created by the AspectJ compiler will run on J2SE 1.1 and higher



UI Usability

- Accessibility:
 - No review for this release, but a previous major version was given an accessibility review and all issues found then have been resolved since (http://www.eclipse.org/ajdt/accessibility1_3.html)
- We follow the User Interface Guidelines

Schedule



- Release was scheduled for October 2006



Process

- These releases been developed using open, transparent, permeable, and inclusive processes
- Use of Bugzilla, the AJDT newsgroup, the AspectJ users mailing list, and the developer mailing lists



Communities

- Continuing to foster active community:
 - Regular monitoring of AJDT newsgroup, AspectJ users mailing list, and Bugzilla
 - The Aspects Blog, AJDT Eclipse Corner article, InfoQ articles
 - Integration in Spring 2



IP Issues

- IP process followed
- IP logs:
 - AJDT: <http://www.eclipse.org/ajdt/project-info/ip-log.txt>
 - AspectJ: <http://www.eclipse.org/aspectj/project-info/ip-log.txt>
- Released under EPL



Project Plan

- Future releases:
 - AspectJ 1.6 + AJDT 1.4.2 (for Eclipse 3.2)
 - AspectJ 1.6.1 + AJDT 1.5 (for Eclipse 3.3)



Release Review Version

- These slides are based on the following version of the Release Review document:
 - version 032 - January 15, 2006