AJDT 2.1.0 Release Review

Planned Review Date: June 23, 2010

Communication Channel: ajdt-dev@eclipse.org, eclipse.technology.ajdt

Andrew Eisenberg, Andy Clement

Introduction

- AJDT is the Eclipse tool support for the AspectJ language

- This release review is for 2.1.0

Terms

- Intertype Declaration (ITD)

  - AspectJ language construct that allows aspects to declare methods, fields, or cosntructors in other classes

- AJDT-PDE build support

  - AJDT's reworking of PDE wizards and scripts so that plug-in projects can be compiled using ajc (the AspectJ compiler) instead of a Java compiler.

Overview

- The focus of this release has been on enhancing support for intertype declarations. This is because they are growing in popularity and importance in the AspectJ and Spring communities. For example see the Spring Roo project:

  http://www.springsource.org/roo

  which makes heavy use of ITDs.

Features (summary)

- ITD-aware search Bug 273335

- Refactoring

  - ITD rename refactoring Bug 303558

  - ITD-aware rename refactoring participant Bug 302462

  - Pull-out refactoring Bug 277055

- AspectJ-aware PDE export and headless build Bug 303960

- Modernize the build process Bug 298913

Feature: ITD-aware search

- Java search for ITDs (declarations and references)

  - Results appear in the Java search view

- Java search inside of ITDs

Feature: Refactoring

- ITD rename refactoring

  - Use ITD-aware search to find references

  - Functionality and UI based on JDT rename method refactoring

- Rename participant for renaming Getters and Setters declared by ITDs

  - Seamlessly hooks into JDT rename field wizard

  - Enabled only when rename Getter/Setter check boxes are enabled.

- Pull out refactoring

  - pull out fields and methods into an aspect

  - analog to Push in refactoring

Feature: AspectJ-aware PDE support (3.6 stream only)

- PDE now supports the ability to specify a CompilerAdapter (Bug 303960)

- AJDT supplies AJDTCompilerAdapter

- Previous AJDT-PDE build support is now superfluous and removed

Feature: Modernize build process

- Move to Bamboo-based build

- More flexible and extensible

- Browser-based

  - community can now view and track AJDT builds

AspectJ Features

- Notable changes that AJDT picks up by embedding the latest AspectJ (1.6.9):

  - Reduced memory consumption

  - Faster weaving

  - Better incremental compilation experience (resorting to a full build less frequently)

  - Improvements in performance of woven code (especially when annotation binding)

- More information in the AspectJ readmes:

  - http://www.eclipse.org/aspectj/doc/released/README-167.html

  - http://www.eclipse.org/aspectj/doc/released/README-169.html

Non-Code Aspects

- Documentation is up to date

  - Developer documentation has been moved to the Eclipse wiki

- Many strings have been externalized

  - Not yet translated

APIs

- APIs are in a good state

- No new APIs introduced since 2.0.0

- All existing APIs are unchanged and backwards compatible

Architectural Issues

- AJDT relies heavily on internals of JDT and PDE (3.5 only) in order to be as integrated as possible.

    - Must maintain one branch of AJDT for each minor release of Eclipse

        - 3.5 and 3.6 streams currently supported

        - 3.4 stream no longer supported

    - Due to AJDT's new use of PDE (3.6 stream only) some outdated and superfluous code was removed

    - Between between versions 3.5 and 3.6 AJDT's use of JDT has not changed much

Architectural Issues (cont.)

- AspectJ weaver and runtime

    - Distributed with AJDT

    - Can be used separately from AJDT

- Equinox Weaving

    - Distributed with AJDT

    - Can be used separately from AJDT

Architectural Issues (cont.)

- Some areas in UI should be in core
  - See bugs 268110 and 268116
  - Consider refactoring in future releases

Tool Usability

- AJDT is mature and is being used to develop many large scale applications including AJDT itself.

End-of-Life

- Removal of AJDT-PDE (3.6 stream only)

  - Made redundant due to the resolution of bug 303960

  - Easier maintenance going forward

    - this area of code required a rewrite on each new minor release of Eclipse.

Bugzilla

- Bugs closed for this release (12 bugs)

- Outstanding bugs for 2.1.1 and onward (38 bugs)

- Bug closed since previous release review (2.0.0) (70 bugs)

- New, unresolved bugs since last release review (26 bugs)

- All open bugs in AJDT (126 bugs)

Standards

- AJDT runs on J2SE 5 and 6

- Version 2.1.0 requires Eclipse 3.5 or 3.6

  - Separate update sites for 3.5 related artifacts and 3.6 related artifacts

  - Content will be similar except for internal dependencies on JDT and PDE

  - Use 35x and 36x in qualifier to specify target version of Eclipse

UI Usability

- No UI review for this release. Most recent review can be found here:

  - http://www.eclipse.org/ajdt/accessibility1_3.html

- All issues from this review have been resolved.

- Where possible and appropriate, AJDT uses JDT as a guidance for UI

  - Refactoring wizards
  - Aspect and In path pages

Schedule

- No delays expected for final release
  - to coincide with Helios release

Communities

- Continuing to foster active community:
    - Regular monitoring of AJDT newsgroup, AspectJ/AJDT users/dev mailing lists, and Bugzilla
    - Heavily used by Spring – we regularly monitor the Spring AOP and STS related forums

IP Log

- IP process followed

- Available at:

  - http://www.eclipse.org/projects/ip_log.php?projectid=tools.ajdt

IP Issues

- None.

Project Plan

- Future releases:

  - Inter-type inner-type support (Bug 316554, new to AspectJ 1.6.9)

- Documented at

  - http://www.eclipse.org/projects/project-plan.php?projectid=tools.ajdt

Release Review Version

- These slides are based on the following version of the Release Review document:

  - http://wiki.eclipse.org/Development_Resources/HOWTO/Release_Reviews

- And the following template:

  - http://wiki.eclipse.org/images/4/4a/ReleaseReview20081211.zip