

m2e 1.0 Release/Graduation Review



Planned Review Date: [Date]

Communication Channel: m2e-dev@eclipse.org

Igor Fedorenko (ifedorenko@sonatype.com)

Jason van Zyl (jason@sonatype.com)

Introduction

- m2e provides both a framework for building Maven-centric Eclipse tools and a set of tools and user interface elements directly consumable by Eclipse Users.
- As a framework, m2e provides facilities to import and configure Maven projects in Eclipse workspace as well as APIs that allow m2e extensions to access Maven project metadata and participate in Maven project configuration and workspace build.

Introduction cont'ed

- For the end user, m2e provides
 - wizards to import existing and to create new Maven projects
 - rich form-based and text-based pom.xml file editor
 - launch configuration types to launch Maven build directly from Eclipse IDE
 - access Maven repository index to browse repository contents and in various code-assist
- JDT integration, for directly useful to endusers and extensible

Project history

- m2e started as “m2eclipse” at codehaus in 2005
- Codebase moved to Sonatype public SVN repository in 2008 and to a public Github repository in 2010
- Proposed as Eclipse project in 2008 but actual move was delayed due to m2e dependency on prereleased version of Maven 3.0 (released on 2010-10-08)
- 5 releases since January 2010
- m2eclipse was #2 most downloaded on Eclipse Marketplace in Jan-May 2011

Features

- The following features have been added or significantly reworked in m2e 1.0 release
 - Reworked mapping of Maven project configuration and build lifecycle to Eclipse workspace
 - Automatic discovery and installation of m2e extensions
 - Major pom.xml editor overhaul
 - Introduction of public extensions API
 - Overall stability and performance improvements
-

Non-Code Aspects

- Localization, Internationalization, Accessibility
 - All m2e strings are externalized, but only English strings are provided.
 - m2e development team did not do any localization and/or accessibility testing due to resource constraints.
- Active user and development mailing lists as well as #m2clipse IRC channel at codehaus.
- User and Developer Documentation and Wiki are outdated and largely not applicable to m2e 1.0
- m2e team maintains large set of opensource extensions at Github, which are advertised as a learning tool.
- We plan to improve documentation situation in the future

APIs, workspace compatibility

- This is the first m2e release from Eclipse and as such we had to break compatibility with all existing m2e 0.12 and earlier extensions because of required use of new "org.eclipse.m2e" namespace.
- At the same time we took this opportunity to significantly reduce m2e API surface and are comfortable evolving public API in backwards compatible manner in the future releases.
- Another unfortunate side-effect of changing m2e namespace is incompatibility of m2e 1.0 with workspaces created with earlier m2e versions. Due to limited resources m2e development team was not able to provide migration logic and the users will have to recreate their workspaces when migrating to m2e 1.0.

▪

Architectural Issues

- All integration with specific Maven plugins and/or Eclipse tools (JDT, WTP, PDE and so on) have been moved outside of the core framework and reimplemented using m2e extensions APIs.
- Every aspect of m2e/JDT integration can be customized using provided APIs and extension points.
- Still outstanding
 - Separation of Maven Indexer and Maven Archetyper support from m2e core, but these are clearly marked as `*.internal.*` and will be dealt with in a future release.
 - Split of monolith `o.e.m2e.maven.runtime` bundle in a number of smaller bundles, which is necessary to facilitate some reuse and coexisting scenarios. This work requires changes to Maven core and could not be contained in 1.0.

Tool Usability

- Generic tools for working with Maven projects inside Eclipse workspace
 - Feature-rich pom.xml editor, error markers and quick-fixes for common problems and best practices
 - Maven launch configuration types
 - Automatic project dependency resolution
- Support for “plain” Java projects out-of-the box via JDT integration
 - JDT project configuration based on pom.xml
 - Automatic compile classpath management
 - Maven specific classpath in RunAs JavaApplication and JUnitTest
-

Tool Usability improvements in 1.0

- Numerous usability improvements to m2e GUI and backend have been implemented during 1.0. Some of specific changes
 - Automatic discovery and installation of required m2e extensions based on pom.xml
 - Ctrl-LeftClick navigation to pom.xml property and referenced artifacts, plugins, etc. More and better Ctrl-Space codeassist suggestions
 - Many error markers are now placed on the most appropriate pom.xml element. Error markers and corresponding quick-fixes for common Maven best-practices.
 - Some long-running operations like project configuration update are significantly faster and better interact with workspace build jobs.
 - Removed many redundant/stale GUI elements

Bugzilla

- From 2010-07-01 until 2011-05-18
 - 350 bug records were opened
 - 245 bug records were resolved, out of which 179 were closed as “FIXED”
 - 130 bug records were opened by non-committers
- As of 2011-05-15, there are 115 outstanding bug records
 - critical 1; major 2; normal 78; minor 9; trivial 1; enhancement 24

Standards

- m2e 1.0 is fully compatible with pom.xml format used by Maven 3.0.2, 2.1.x and 2.0.x
- m2e 1.0 is fully compatible with Maven “default” repository layout and Maven repository Index format
- m2e 1.0 provides limited support for Maven Archetyper descriptor format

UI Usability

- m2e team did not evaluate m2e conformance to the User_Interface_Guidelines
- The code is expected to support multiple languages but this has not been tested by m2e team

Project Health

- Active Code Base
 - 748 commits from 2011-01-01 to 2011-05-24
- Separate git repository at Github for unit and integration tests due to significant effort involved in moving test code and its dependencies to Eclipse
- Continues build on Windows and Linux
 - Public access to m2e build on <https://grid.sonatype.org/ci> has been disabled middle of May, 2011.
 - We plan to move CI build to Eclipse server for the next release

Schedule

Communities

- Active user community
 - 60-100 m2e-users mailing list messages/month in first half of 2011
- Active developer and adopter communities
 - 40-100 m2e-dev mailing lists messages/month in first half of 2011
 - Several thridparty extensions in active development and have been updated for m2e 1.0 already. For example, m2e/wtp and m2e/adt integration
 - 8 non-committer contributions in m2e 1.0
- All project decisions are discussed on m2e-dev mailing list
- Active [irc://irc.codehaus.org/m2eclipse](http://irc.codehaus.org/m2eclipse) IRC channel
- Sonatype presented m2e 1.0 work during Eclispecon 2011
- m2e team maintains a number of opensource extensions at Github that are both useful to endusers and as working examples for adopters
- Close collaboration with Apache Maven; collaboration with m2e/wtp, JBossTools and other opensource projects

Committer Diversity

- At this point only Sonatype is involved in m2e core development

IP Log

- All third party dependencies have been reviewed and approved by Eclipse legal.
- Draft IP Log is available at

http://www.eclipse.org/projects/ip_log.php?projectid=technology.m2e

IP Issues

Notes

Credits and Kudos