



Eclipse Monkey 1.0 Release Review

May 30, 2007

Eclipse Monkey PMC

One-Minute Overview

- **New Features**
 - Converted existing code to be scripting-language-neutral, allowing for new scripting languages such as Ruby and Groovy to be added.
 - Scripts may be embedded in plug-ins and run at startup.
 - Users can create web-browser-based views with two-way communication.
- **API Quality**
 - API is complete. Focus now is on getting additional projects to contribute DOM extensions.
- **No end-of-life issues (first release)**
- **All code is contributed under EPL, with one dependency on MPL code (Rhino).**
- **5 committers across three organizations, and a diverse set of users, from Eclipse committers to web developers.**

Introduction

- Major project milestones
 - Created December 2005
 - Aptana code contribution in December 2006
- Continuing to expand community
 - 5 Committers from 3 organizations
 - BOF at EclipseCon in 2006 and 2007

Eclipse Monkey 1.0

- Language-neutral mechanism for allowing end users to script against Eclipse APIs, extensible via extension points.
 - Although JavaScript is the default scripting language, APIs allow for additional languages to be added, such as Ruby or Goovy
- User scripts are added to menus allowing for easy accessibility
- Users may also assign key commands to scripts or trigger them to launch on Eclipse startup
- Easily scriptable Web-browser-based views allowing for two way Eclipse/browser communication.
- Users can use existing Eclipse Monkey “DOM” abstraction layer or easily create and share their own.

Non-code Aspects

- User documentation and tutorials, see <http://www.eclipse.org/dash/>
 - Getting Started Guide
- Additional resources at <http://www.apтана.com/monkey/>
 - EclipseCon 2007 Talk
 - Sample Scripts
 - Sample DOMs
 - User documentation

Non-code Aspects (cont.)

- Presentations (sample set)
 - EclipseCon 2006
 - EclipseCon2007: Simplify your Coding Life with Aptana and Eclipse Monkey
- Blogs
 - Strong interest from the community at the BOF sessions and later posted in blog notes
 - Numerous mentions on the web

Community Feedback

- Eclipse Monkey used by numerous organizations
- BOF session and 2007 tutorial generated many requests for future features as well as creative ways of using Eclipse Monkey in cases ranging from deployment to end user scripting for robots in an RCP application.

API: 1.0 Status

- Implementation
 - The Core Eclipse Monkey 1.0 API is 100% complete.
 - More work will be done to encourage additional projects to build Eclipse Monkey-friendly DOMs
 - Code has been refactored to be scripting-language neutral and allow for easy extensibility at many levels
- Documentation
 - The Eclipse Monkey 1.0 JavaDoc API will be published to:
http://www.eclipse.org/dash/monkey_javadoc/
- Test Cases
 - Some basic test cases have been written and will be automated in future releases
- Compatibility
 - No compatibility issues exist since this is the first release.

Tool Usability

- The novice user is able to code simple scripts with limited knowledge of Eclipse APIs.
- More advanced users have full control and flexibility to modify many things about the Eclipse environment in more “agile” ways than the typical code-compile-deploy cycle.

Architectural Issues

- Use of internal Eclipse APIs and deprecated code
 - The existing code uses some internal Eclipse APIs. They will be removed as much as possible in the next release.
- API
 - Additional DOMs need to be created to fully flesh out a core set of functionality for the end user across most common Eclipse tasks

End of Life

- N/A – there is no release to replace or EOL

Defect Statistics

- As of May 30, 2006:
 - 22 low severity defects deferred
- Release exit criteria
 - 0 high severity defects

Committers, Contributors and Process

- 5 committers from 3 organizations
- Emphasis on growing the project
 - 2 face-to-face meetings (BOFs)

Intellectual Property – Source Code

- All contributions to date have been made under EPL 1.0
- All plug-ins contain appropriate license files
- All committers have completed Eclipse Committer Agreements and have been approved by the PMC
- Major contributions have followed due diligence process

Future Plans

- Additional DOMs and sample scripts
- Sharing server allowing for easy sharing of scripts from within Eclipse to the broad Eclipse community
- Additional language support
- Eclipse Monkey is bundled into the next version of Aptana IDE, for further large distribution potential.

Feedback Wanted!

- Please provide us with your feedback, requirements, issues
 - Submit enhancement requests / defects in Bugzilla
https://bugs.eclipse.org/bugs/enter_bug.cgi?product=dash
 - Submit questions / issues in the Eclipse Monkey Mailing List: dash-dev@eclipse.org.