



Mobile Tools for the Java Platform (MTJ) – 0.7 Release Review

Mika Hoikkala/Petri Virtanen



Review Topics

- Overview
- Community
- Schedule
- Features
- Extension Points
- Supported Standards
- IP Review
- Documentation status
- Quality status
- Future Plans
- Q&A

MTJ Project Overview



- The scope of the MTJ is to enhance Eclipse platform to support Java development for mobile devices
- The project was proposed autumn 2005 by Nokia and it was approved January 24 2006
- Is part of the Device Software Developer Platform top level project
- This version of the MTJ requires Eclipse 3.2 and EMF 2.2.x
- Project homepage: <http://www.eclipse.org/dsdp/mtj/>
- Project Wiki: <http://wiki.eclipse.org/index.php/DSDP/MTJ/>

Community



- Major Participants
 - Nokia (6 committers)
 - IBM (1 committer and 1 major contributor)
 - EclipseME (Major contributor: Craig Setera)
 - SonyEricsson (contributor)
- Other Participants
 - Sybase, Apogee Software, Sprint, Motorola, Four2B, Tira Wireless, RIM, Sysline Inc, Antenna, ShareME Technologies
- Bi-weekly status conf calls
 - Latest status provided and discussed
 - Possibility for developers to provide ideas and proposals
- Face-to-Face meetings arranged two times during the project
 - 1st : 8 Dec 2005 in Boca Raton
 - 2nd : 16-17 Aug 2006 San Francisco
- All meeting minutes can be found from the MTJ Wiki page
http://wiki.eclipse.org/index.php/DSDP/MTJ/Meeting_Minutes

Community, cont.



- Conferences

- EclipseCON 2006: long talk (<http://www.eclipsecon.org/2006/Sub.do?id=214>)
- Java ONE 2006: presented at Eclipse booth
- Eclipse Summit Europe 2006: talk
(<http://www.eclipse.org/dsdp/mtj/development/presentations/1MobileDevelopmentAndMTJ-v2.pdf>)

MTJ 0.7 Schedule



- Initial schedule (at creation review)
 - Official milestone and release candidate builds were not scheduled
 - 1st Release: 2Q 2006
- Actual 0.7 release schedule
 - M1 – 12 Sept
 - M2 – 2 Oct
 - RC1 – 25 Oct
 - RC 2 – 15 Nov
 - R0.7 – 22 Nov (proposed)

MTJ 0.7 Features

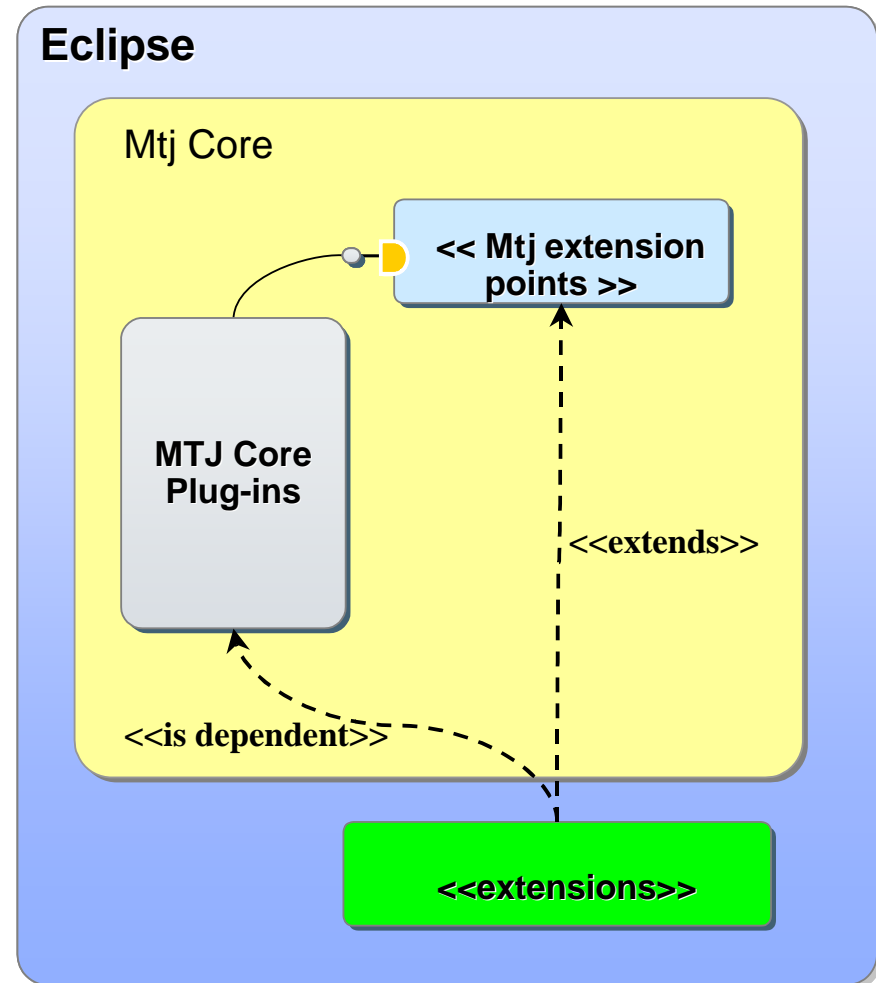


- Tools to develop mobile Java applications.
 - Create a project (MIDP specific and general Java ME) wizard
 - Simple template for MIDP application (Hello World)
 - Create a code
 - Mobile target SDK management
 - Build tool (compile and create application package)
 - Run and Debug in emulators
 - Management for keys & certificates and signing tool for applications
 - Transfer to real Device (only Nokia, delivered from Nokia site)
- User and developer documentation

Extension points



- Runtime management framework
 - adding device adapter to manage emulators + real devices
- Build framework, customized and extensible build process
 - Packaging (CDC, CLDC, Java in Palm devices, Java in Nokia devices,...)
 - Signing (differences between devices)
- Deployment framework
- Developer documentation and API javadocs





Supported Standards

- Mobile Information Device Profile (MIDP) 1.0/2.0
- Connected Limited Device Configuration (CLDC) 1.0/1.1
- Unified Emulator Interface (UEI) 1.0
- JDWP (Java Debugging Wire Protocol) (as supported by JDT)

IP Review



- All code under EPL?: Yes (except two 3rd party components, see project IP Log)
- About files and license files are complete and correct?: Yes
- All significant and third-party contributions have been reviewed by Eclipse legal?: Yes (see project IP log)
- All committers have completed Eclipse Committer Agreements?: Yes
- Project IP log complete and approved by eclipse legal. Available <http://www.eclipse.org/dsdp/MTJ/development/mtj-log.csv>

Documentation Status



- User Documentation
 - Near comprehensive user documentation available in Eclipse Help.

- Developer Documentation
 - Initial developer documentation available in Eclipse Help. Covers basic architecture, preliminary extension points, and gives brief examples
 - Javadocs from extension points to provide API references (included in Eclipse Help)
 - More Architecture and Design documents available in CVS.

Quality Status



- JUnit tests integrated with the build process (org.eclipse.mtj/tests). Unit testing mostly done during the development. All tests passed.
- System tests (manual) done for every released build during the whole development cycle.
- Bug counts
 - 8 Open (6 Bugs, 2 enhancements)
 - 79 Fixed (72 Bugs, 7 enhancements)

Future Plans



- MTJ version 1.0, Initial Schedule
 - High level content freeze 11/2006
 - M1 01/2007.... M5 05/2007
 - RC 1 06/2007
 - RC 2 07/2007
 - Release 1.0 Q3/2007
- MTJ version 1.0, Targets
 - With needed quality
 - With needed features (to have "full" development environment)
 - With needed documentation
 - Schedule driven approach in development
- Content & Tasks
 - Fixes and enhancements to R0.7
 - Finalize the Extension APIs
 - Enhance documentation
 - Project based preferences
 - Support for non UEI SDKs (user provides the information)
 - Enhance build mechanism to manage resources
 - Bug fixes
 - Visual Designers
 - LCDUI
 - eSWT (co-operation with eRC, which now part of the DSDP)
 - Fragmentation
 - Build time solutions e.g. pre-processing
 - Device Information database access
 - Obfuscation
 - External obfuscators are possible to plug-in
 - Possible one default integrated
 - New Profiles and configurations
 - CDC
 - Foundation Profile, Personal Profile (no AWT visual editor), Personal Basis Profile
 - MIDP 2.1 support, MIDP 3.0 support (not included, but may come to the picture)
 - Mobile JUnit
 - Localization (low)
 - Manage resources
 - Visual designer support
- Further grow the community



Questions and feedback

- Please send us feedback, requirements and issues at
 - Newsgroup: `eclipse.dsdp.mtj`
 - Mailinglist: `dsdp-mtj-dev@eclipse.org`
 - Bugzilla: `https://bugs.eclipse.org/bugs/enter_bug.cgi?product=MTJ`
- Please visit the MTJ website

<http://www.eclipse.org/dsdp/mtj/>

Thanks!