



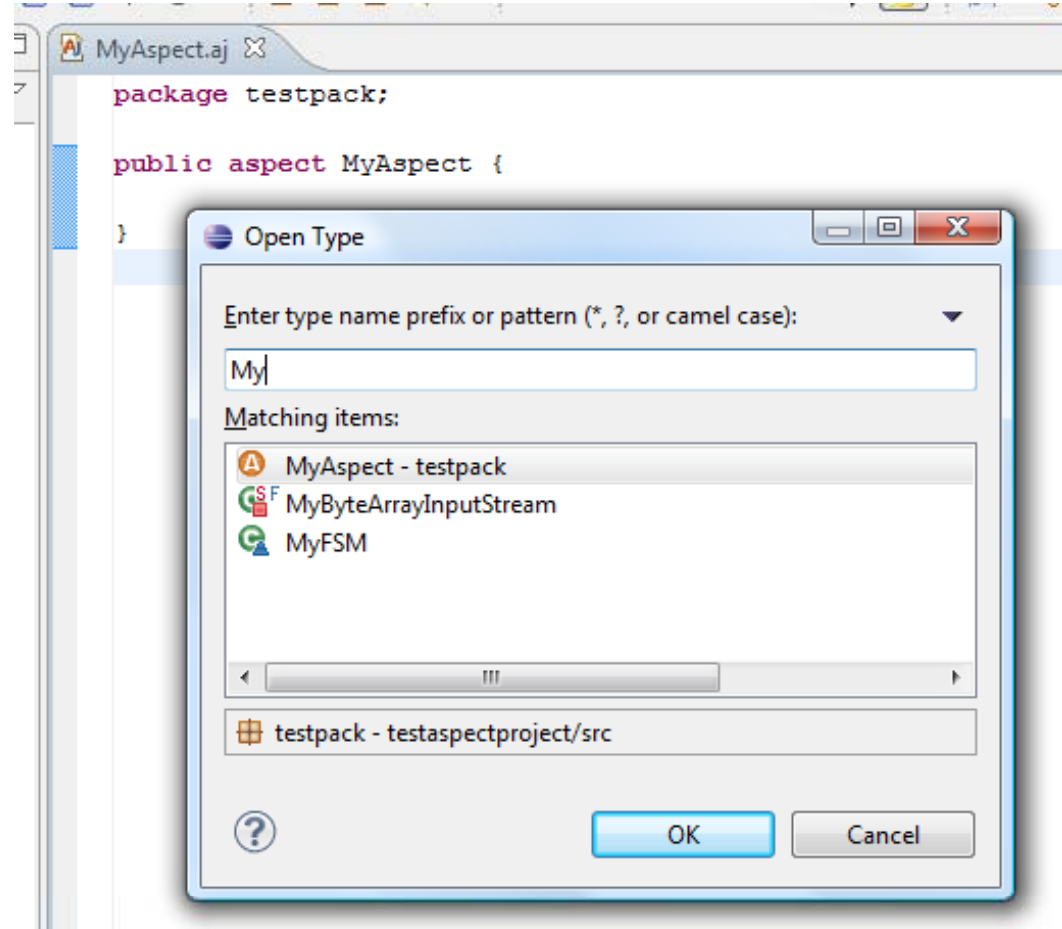
What's New in Equinox Aspects

Martin Lippert (akquinet it-agile GmbH)



Adoption

- AJDT now uses Equinox Aspects to weave into JDT
- Scala support interested in the same infrastructure



Short Talk by Andrew Eisenberg: **Aspects Everywhere: Using Equinox Aspects to Provide Language Developers with Deep Eclipse Integration** (unfortunately in parallel slot)



Weaving Dynamics

- “Be a good citizen of the OSGi community”
 - ◆ Weaving when resolved
- You can install and uninstall aspects at runtime
 - ◆ Refreshing other bundles
 - ◆ But: other bundles need to be dynamic-aware



Caching

- The goal: **Zero overhead for cached scenario**
 - ◆ And we are very close !!!
- Many tweaks already done
 - ◆ Fast cache read/write/lookup IO
 - ◆ Just one load op (not the default and then the cached)
 - ◆ No cache lookup for non-woven bundles
 - ◆ Awareness of bundle versions + updates
- More on the road:
 - ◆ Caching for generated classes (around closures)



Asynchronous Cache Writing

- **Benefits:**
 - ◆ Better performance for loading thread (doesn't have to wait for cache IO)
 - ◆ Less concurrency for IO itself (simpler implementation)
 - ◆ More robustness for the system in case of IO problems
- **Implementation:**
 - ◆ Concurrent bounded queue (even if cache writing hangs, memory usage is bounded)



Performance Improvements

- Improved AspectJ load-time weaving
- Many tweaks for caching
 - ◆ make the cached scenario really fast
- Lazy creation of weavers
 - ◆ create them only if they are really needed
- Less XML parsing
 - ◆ aop.xml files are parsed only once



P2 Installation

- P2 metadata added
- Installable via P2
 - ◆ Including automatic changes to config.ini
- AJDT already provides an update site
 - ◆ <http://download.eclipse.org/tools/ajdt/35/dev/update>
 - ◆ But weaving service is not started automatically
- Still some issues:
 - ◆ Non-writable locations
 - ◆ Different locations
 - ◆ Manual configuration cumbersome



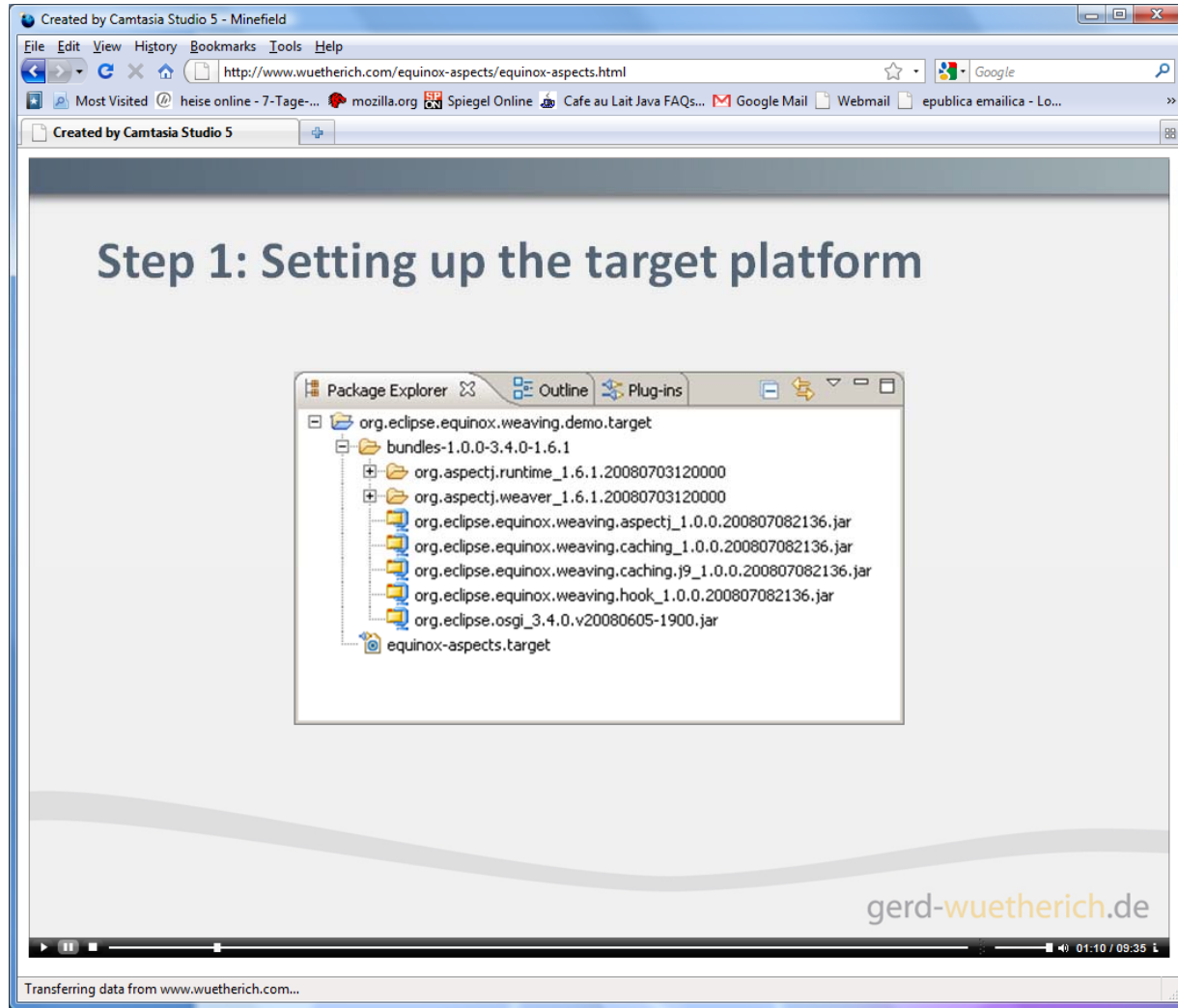
Other Weavers

- The base hook provides weaving API
 - ◆ And calls the weaver at classloading time
- The weaving itself is provided as OSGi service by other bundles
 - ◆ Pluggable weavers
 - ◆ Currently: AspectJ weaving
- No need to re-implement caching
- *This is not dynamic at the moment*



Screencast

- Introductory material
- Contributed by Gerd Wütherich (Thanks, Gerd!)





Aligned development + wiki-based planning

- http://wiki.eclipse.org/Equinox_Aspects_Plan

Current Milestone Plan

M6 - March 13, 2009 (stable build)

- General:
 - Alignment with the releng headless build [bug 256347](#) (Martin)
 - Prepare for graduation
- Weaving:
 - investigate weaving service implementation for ClassFileTransformer [bug 248047](#)
 - programmatically enabling/disabling weaving hook from within application [bug 262229](#)
 - using package versions on import-package in manifests [bug 258136](#)
 - enable/disable weaving service on a per-bundle basis [bug 255682](#)
- Caching:
 - caching service causes system to hang [bug 262299](#) (Martin)
- Website:
 - add screencast or step-by-step example for RCP apps using Equinox Aspects

Future Milestone Plans

M7 - May 1, 2009 (stable build)

RC1 - June 12, 2009

Previous Milestone Plans



What's Next?

- Ease of use
- Better tooling
- Management API
- More fine-grained control for aspect weaving
- Example for another weaver implementation

- Graduation
- First “official” release



Thank you for your attention!

- Questions and feedback welcome!

<http://www.eclipse.org/equinox/incubator/aspects/>

- Many thanks to Heiko Seeberger, Andrew Eisenberg and Andy Clement

Martin Lippert: lippert@acm.org