



# EMF Net4j 1.0.0 Ganymede Simultaneous Release

May 18th, 2008

# EMF Net4j Ganymede Release Talking Points



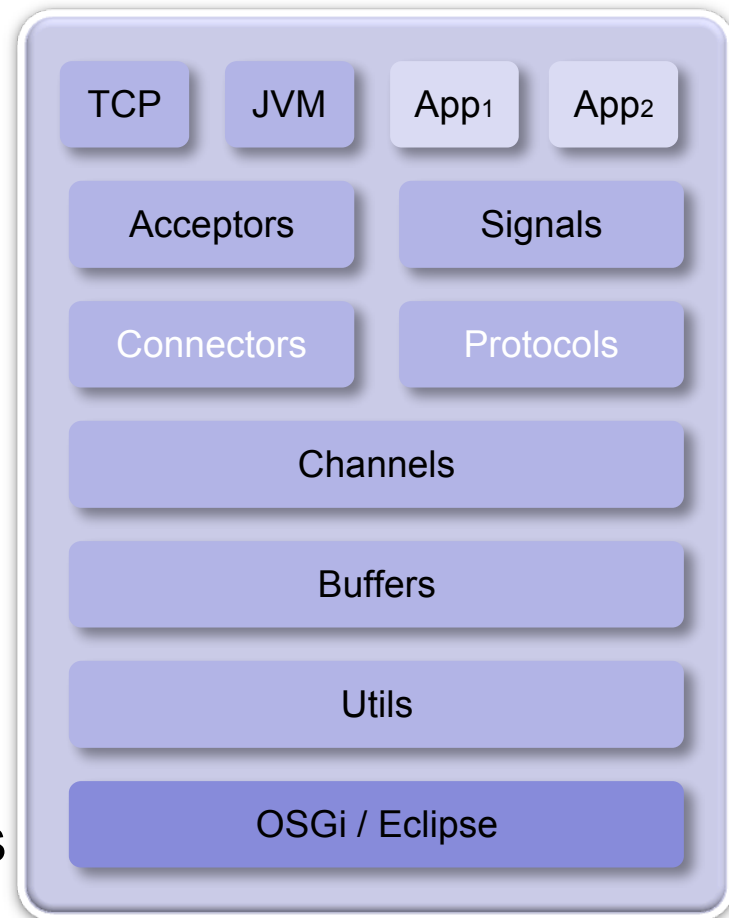
- Noteworthy New Features:
  - ♦ Challenge/response based negotiator ([205027](#))
- Quality of APIs
  - ♦ No "provisional" APIs, all non-API code in "internal" packages
- End of Life Issues:
  - ♦ None
- IP Issues:
  - ♦ None
- Committer Diversity
  - ♦ One committer
  - ♦ Working together with Scott Lewis (ECF) to integrate Net4j+ECF

# EMF Net4j

## Pluggable Client/Server Applications



- High performance
  - ◆ `java.nio.DirectByteBuffer`
  - ◆ zero copying, non-blocking
- Good scalability
  - ◆ `java.nio.channels.Selector`
  - ◆ single I/O thread possible
- Multiple transports
  - ◆ Shipped with TCP and JVM transports
- Pluggable protocols
  - ◆ Independent of chosen transport
- Server-initiated push services
  - ◆ Asynch and synch requests from the server
- OSGi™ and stand-alone modes





# Features & Non-Code Aspects

- Features
  - ◆ New features regularly announced via release notes
  - ◆ 23 enhancements provided since 0.7.0
- Non-Code Aspects
  - ◆ New examples
    - Buddy list
    - Collaboration + Chat
    - Fileshare
    - JMS™ provider implementation
  - ◆ Wiki with tutorials and design docs: <http://wiki.eclipse.org/Net4j>
    - (in progress)



# APIs

- No “provisional” APIs; all non-API code in “internal” packages
- Deprecated API
  - ◆ None
- No ‘discouraged access’ warnings (internal usage)
- 2 x 30 JUnit tests (JVM + TCP)

# Standards, Architectural Details, UI Usability



- Standards
  - ◆ None
- Architectural Details
  - ◆ Requires a 5.0 JVM or greater
  - ◆ Can run standalone (No OSGi)
  - ◆ Can run on OSGi and Eclipse
  - ◆ Delivers decoupling of transport and application logic
- UI Usability
  - ◆ As a pure runtime technology all UIs are only exemplary



## Project Plan – future features

- A draft development plan for Net4j 2.0.0 is not yet available
- Important planned features
  - ◆ HTTP transport layer implementation
  - ◆ Enhanced Failover Support
  - ◆ Asynchronous opening of IO channels



## Schedule

- Net4j is a “+1” component in the simultaneous release
- 1.0 Release Plan according to the Ganymede schedule
- Net4j uses a continuous build approach



# Bugzilla



- As of May 18<sup>th</sup>, 2008:

	NEW/ASSIGNED/REOPENED	RESOLVED/VERIFIED/CLOSED	Total
blocker			0
critical			0
major		1	1
normal		4	4
minor		5	5
trivial			0
enhancement	5	23	28
Total	5	33	38



## Communities

- Active newsgroup
  - ◆ [eclipse.technology.emft](mailto:eclipse.technology.emft)
- Conference Participation
  - ◆ Short Talk at EclipseCon 2008



# IP Issues

- The Net4j component leadership verifies that
  - ◆ The about files and use licenses are in place as per the Guidelines to Legal Documentation.
  - ◆ All contributions (code, documentation, images, etc) have been committed by individuals who are either Members of the Foundation, or have signed the appropriate Committer Agreement. In either case, these are individuals who have signed, and are abiding by, the Eclipse IP Policy.
  - ◆ All significant contributions have been reviewed by the Foundation's legal staff.
  - ◆ All non-Committer code contributions, including third-party libraries, have been documented in the release and reviewed by the Foundation's legal staff.



# IP Issues

- The Net4j component leadership verifies that (cont.)
  - ◆ All Contribution Questionnaires have been completed.
  - ◆ The "provider" field of each plug-in is set to "Eclipse.org".
  - ◆ The "copyright" field of each feature is set to the copyright owner (the Eclipse Foundation is rarely the copyright owner).
  - ◆ Any third-party logos or trademarks included in the distribution (icons, help file logos, etc) have been licensed under the EPL.
  - ◆ Any fonts or similar third-party images included in the distribution (e.g. in PDF or EPS files) have been licensed under the EPL.
  - ◆ IP Log at
    - <http://www.eclipse.org/modeling/emft/project-info/ipquery.php>
    - <http://www.eclipse.org/modeling/emf/project-info/ipquery.php>



## Graduation: Active Community

- Integration with both CDO and ECF
- Active Community currently consists mainly of CDO users
- Complete redesign and rewrite ~ 1 year ago
- Few bugzillas filed due to excellent stability and performance since redesign



## Graduation: Open and Transparent

- Very responsive support on newsgroup, open discussions and reactions on proposals from users
- All bugzillas solved immediately
- Project Materials are all available to the public
- Follow IP-policy (approval for use of jms jar file, for example)

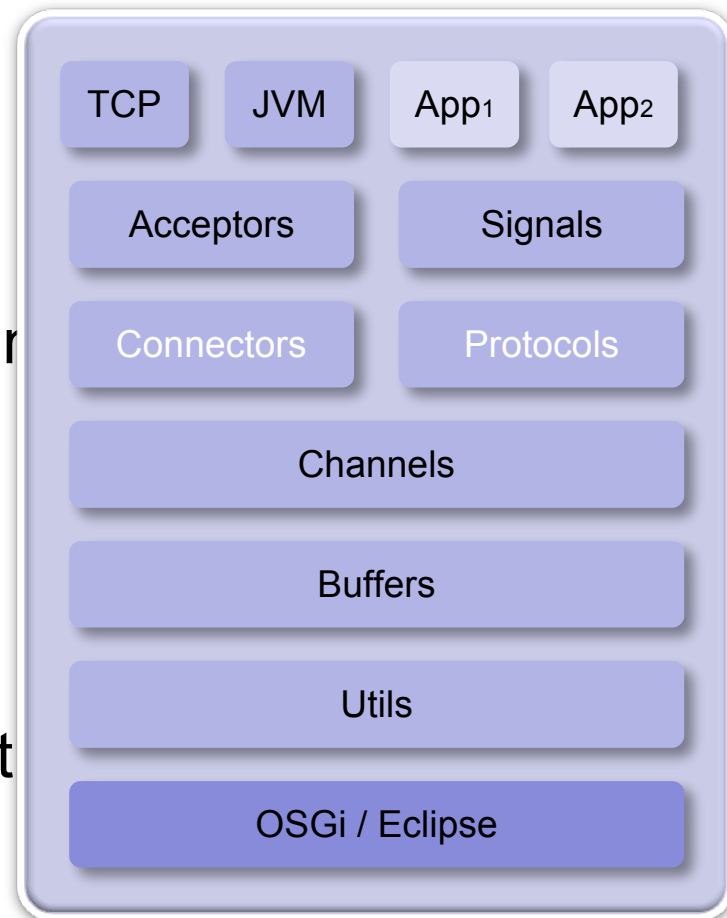


## Graduation Review: The Eclipse Way

- Short talk on EclipseCon 2008 (ESE 2008 planned)
- Integrates with other Eclipse projects: ECF, CDO
- Frequent/Continuous builds (every 1-2 weeks)
- Every build solves all outstanding bugs
- Builds are automatically tested in multiple scenarios, test cases run on different transport technologies (TCP, JVM)

# Graduation Review: Technical Architecture

- Buffer layer based on fast, asynchronous, non-blocking, fixed-size buffer exchange
- Channel layer supports multiplexing of different data streams through a single physical connection
- Protocol layer enables to write application protocols independent of the chosen transport



- **Each layer is exchangeable**





## Legal Notices

- OSGi is a trademark of the OSGi Alliance
- Java, JVM, JMS and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both
- Other company, product, or service names may be trademarks or service marks of others