Eclipse Project PMC

Project Update
Overview

- What’s new in Eclipse 3.3
- What’s coming in 3.4?
- What’s coming in the long term?
Overview

- What's new in Eclipse 3.3
- What's coming in 3.4?
- What's coming in the long term?
What we did

- **Eclipse 3.3 dates**
  - 3.3.0 - 25 / 06 / 2007
  - 3.3.1 - 24 / 09 / 2007
  - 3.3.2 – 18 / 02 / 2008

- **Themes for 3.3**
  - Components
  - Consumability
  - Java
  - Vista
  - UI Evolution
  - API
Theme: Components

- Enhance Eclipse's use of, and support for, software components
- All plug-ins are now signed
- Can develop and run with multiple versions of the same plug-in in the workspace or target platform
- IDE application moved to new plug-in
- Eclipse Orbit project to host third party libraries
- Multi-version support for build contributions
Target provisioning

- Plug-in locations now dynamically added to target platforms
- File system provisioner augments content of target platform with plug-ins from file system
- Update site provisioner downloads and installs plug-ins into target platforms with single click
- Additional provisioners declared via extension point
OSGi

- Equinox Framework implements OSGi R4.1 framework specification. Used as reference implementation for OSGi R4.1 and JSR291

- Eclipse now uses the OSGi MEG application admin service
  - More than one application running at a time
  - Applications have proper startup, suspend, shutdown lifecycle

- Support for launching Equinox in embedded in application servers

- New templates demonstrating the usage and implementation of programmatic OSGi services.
Analyze dependencies for JAR archives

When creating a plug-in from existing JAR archives (File > New > Project... > Plug-in Development > Plug-in from existing JAR archives), PDE now computes the dependencies required to run the code in the archive.
Theme: Consumability

- Make it easier for users to get Eclipse, install it on their systems, and configure it for their use.

- Enhance the error handling and reporting mechanisms to make it easier to service Eclipse in the field.

- Improve the scalability and performance of Eclipse, to provide a better experience for users working with many plug-ins and large data sets.
Improved workspace switching

- When you switch workspaces you can now transfer some of your current settings with you.

- The most recently used workspaces now appear on the file menu under the Switch Workspace item for quick access.

![Workspace Launcher](image)
Undo and redo are now available for most resource changes. When you manipulate projects, folders, or files, you now have the ability to undo and redo the changes you have made. This includes resource creation, deletion, move, copy, and rename.

Undo is now available for many task and bookmark operations. When you create, update, add, or delete a task or bookmark, you now have the ability to undo that change.
Custom splash screen

- New Eclipse launcher, starts faster, allows SWT usage
- You can use the new StartupMonitor API to update the splash screen as the platform is being started
- Three new templates for creating custom splash screens, complete with code, images and XML markup:
  1. Interactive: A simulated log-in session
  2. Browser: An embedded HTML browser
  3. Extensible: A dynamic set of image contributions
Performance: 3.3 vs. 3.2

Theme: Java

- Address the new capabilities of Java SE 6
- Continue to refine and extend the features JDT provides to maintain its leadership position
The Java Rename refactoring can now be performed directly in the editor, without showing a dialog.

Note: Refactorings no longer require all editors to be saved.
Java SE 6 support

- **Java SE 6 Debug**
  - display all references to an object
  - display all instances of a Java type
  - force an early return from a method

- **Pluggable annotation processing (JSR 269)**

- **Compiler API (JSR199)**
Clean up improvements

- Clean Up profiles can be defined, shared in a team through a version control system. It is also possible to export and import each profile.

- New clean ups:
  - Format source code
  - Remove trailing whitespace
  - Organize imports
  - Sort members

- It is now possible to automatically invoke code clean up whenever the Java editor is saved
Content assist (Ctrl+Space) improvements

- Content Assist can now propose completions for static members from a list of favorite imports that can be configured.
- Content Assist is more resilient:
  - Guess members for variable with unresolved type.
  - Guess the name of a declared local variable by detecting the presence of unresolved names in subsequent code.
- Content Assist inside a try-catch clause can now infer exception type names based on exceptions detected to be thrown in the corresponding try block.
- Code completion after an at sign "@" proposes all visible annotation types.
Access rules for execution environments

- Access rules are now assigned to each execution environment based on its profile.
- The rules are only applied to projects that are bound to execution environments. Projects bound to a specific JRE still have access to all of their libraries.
The win32 version of Eclipse now runs beautifully on Windows Vista. SWT now exploits native features of the Vista platform, such as native double-buffering to make painting in double-buffered canvases twice as fast.

Eclipse is now available on Windows Presentation Foundation (WPF), a new window system that comes pre-installed on Microsoft Windows Vista. This port is in early access form for 3.3.
Theme: UI Evolution

- Provide a better basis for building modern, rich user-interfaces, by addressing areas where more flexibility and more consistency are needed.
- Use these capabilities to refine the user-interface of the Eclipse IDE
- Implement some of the most requested missing IDE productivity features
New Eclipse presentation

- New Minimize/Maximize behavior
  - Move the view stack to the nearest trim area, showing a 'Restore' button along with the icons for the views in the stack.

- New workbench tab treatments
  - New color scheme based on system title background color, and unselected tabs now also have rounded corners to match the appearance of selected tabs.
Mozilla everywhere

- Mozilla can now be used as the underlying browser control on Windows and OS X, providing that you have XULRunner installed and registered.

- To use this just create your Browser with the SWT.MOZILLA style.
Working set improvements

- You can now add and remove selected elements to and from working sets in your workspace.

- You can also create a new working set directly from the Add to working set toolbar dropdown.

- The Project Explorer can now group and filter elements by working set.

- The export wizard can now persist active working sets as part of a Team Project set.
Compare improvements

- Compare editors now kept in sync with editors open on the same file
- New high performance text differencing algorithm
- Background compare editor initialization improves UI responsiveness
- Typical editor commands now available in compare editors
  - find/replace, show whitespace, show line numbers, Show In sub-menu
- Highlighting of individual changes within a block of changes.
- Compare Structure in Outline view
Help improvements

- Help hosted infocenter servers
  - Initial download significantly smaller

- On-the-fly Help generation

- Link arbitrary commands in context-sensitive help
  - link to a related cheat sheet

- Highlighting search terms in help documents
**Theme: API**

- Encourage adoption of the more recent additions to Eclipse API
- Ensure that needed functionality is made available as API, when it reaches appropriate levels of maturity
  - Data binding supports automatic synchronization between UI state and model state and drastically reduces the amount of listener code required to implement common UIs.
  - The debug platform's launch API now supports mixed mode launching. For example, an ISV could contribute a launcher to profile and debug an application at the same time.
  - Teams are no longer restricted to the specific XML formats defined by the Help system, such as table of contents XML, keyword index XML, etc. A new set of API has been added to enable Java code to generate the content at runtime.
  - Many more…
Overview

- What’s new in Eclipse 3.3
- What’s coming in the short term?
- What’s coming in the long term?
The Eclipse project today

- Seven releases in a row
- Innovating and establishing a stable platform

- No longer just the Eclipse project
  - 70+ open source projects and proposals at eclipse.org
  - 800+ third party plug-ins at eclipse.org
  - 1000+ Eclipse related projects on SourceForge
  - 1700+ Eclipse-based solutions, 350+ from by IBM
  - More than 150 member companies

- Simultaneous releases
  - Callisto (06/2006) : 10 participating projects
  - Europa (06/2007) : 22 participating projects
Stability vs. Innovation

- Is the platform moving too fast?
  - Make it easier for products to port to new major releases (3.3-3.4)
  - Minor releases should be free (3.3.1 – 3.3.2)
  - Internals can change in service releases
  - Challenging to keep current as Platform evolves

- Is the platform scaling well enough?

- Is the platform performing well enough?

- Is the platform adopting a new OS platform in a reasonable timeframe?

  ➔ Maybe it is time to do things a bit differently
Splitting streams

- Eclipse 3.x for consolidation
  - One major release every year (June)
  - Two service releases every year (Fall, Winter)
  - Focus on stability and selected innovation

- Eclipse 4.0 for innovation
  - 2 or 3 years from now
  - Could be break existing plug-ins
  - EclipseCon 2007 Blue Sky BOF
  - Brainstorming in progress
  - Help wanted

- Incubators
Short term focus

- Eclipse 3.4 dates
  - 3.4M3 in progress
  - 3.4.0 - 06 / 2008
  - 3.4.1 - 09 / 2008
  - 3.4.2 – 02 / 2009

- Themes for 3.4
  - Platforms
  - Consumability
  - Reliability
  - The Future
Theme: Platforms

- Ensuring that Eclipse takes full advantage of the capabilities of the underlying technologies that it is based on, be they operating system, window system, Java or other.
  - Port SWT win32 to 64-bit
  - Complete SWT WPF port
  - Support BIDI on Linux GTK
  - Provide full internationalization on Mac OS X
  - Implement accessibility on Mac OS X
  - Exploit the capabilities of modern JREs
SWT for 64-bit Windows

- Windows x86_64/AMD64 has joined the lineup of supported configurations for which SWT and Eclipse are routinely built.

- To use it you must be running on a 64-bit edition of Windows with a 64-bit JRE.

- Note that the 32-bit versions of SWT and Eclipse can still be used on these configurations with a 32-bit JRE.)
SWT right-to-left support for Linux GTK

- SWT now provides support for mirroring on Linux GTK platforms.
Theme: Consumability

- Make it easier for users to get Eclipse, install it on their systems, and configure it for their use.

- Improve the scalability and performance of Eclipse, to provide a better experience for users working with many plug-ins and large data sets.

- Enhance the error handling and reporting mechanisms to make it easier to service Eclipse in the field.
Provisioning

- p2 = Provisioning Platform? Provisioning 2.0?
- Equinox Incubator effort destined Eclipse 3.4 (Ganymede)

Goals for p2:
- Functionally equivalent replacement for UM
- Solution must scale down and up
- Lightweight, dynamic provisioning choices and selections
- Standards based, standards influencing or de facto standard
- Easy API
- Easy User Experience
- Powerful underpinnings
As the basis for the entire Eclipse eco-system, the Eclipse SDK must be robust, flexible and secure.

- Providing API for missing/internal features
- API Tooling
- Provide commonly requested Java security features
- Focus on architectural integrity
- Invest in PDE Build and Release Engineering
API Tooling

- Components are defined by their API so tooling is very important
- Backward compatibility
  - Is version X binary compatible with version X-n
  - Detect problems ASAP
- Global references
  - Who’s using my code?
  - What am I using?
- Version management
  - Ensure code changes are properly reflected in version numbers
- API refinement
  - Public/private, API/internal is not enough
  - SPI, client implementable, client callable, …

- Participate in the PDE API Tooling Incubator
  - http://wiki.eclipse.org/PDE_UI_Incubator_ApiTools#Usecases
Plug-in Spy

Ever wondered where the source is for something you're looking at in Eclipse?

Simply select what you're interested in and invoke Plug-in Spy by pressing Alt+Shift+F1. Plug-in Spy currently supports inspection of:

- Selections
- Editors
- Views
- Dialogs
- Preference Pages
- Wizards
CLDC 1.1 class file compatibility

- The Java compiler can now create CLDC 1.1 compatible class files
- You can set this on the **Java > Compiler** preference page
Overview

- What’s new in Eclipse 3.3
- What’s coming in the short term?
- What’s coming in the long term?
Theme: The future

- Push the limits of our technology in almost every dimension
  - new, multi-year focus on innovation,
  - Eclipse SDK continues to be a vibrant, powerful, dynamic basis for our community's use

- Create the Eclipse 4.0 plan

- OSGi standards participation

- Model the IDE

- Investigate new user-interface directions

- Investigate the next generation of JDT capabilities
EclipseCon 2007 Blue Sky BOF

- Java7
- Web OS
- Think big – thousands of bundles, action contributions
- Provisioning
- Dynamic code
- More concurrency
- Get rid of old APIs – separate compatibility layers
- User interface, move away from toolbar/menus -> ribbon
- More customization possibilities
- Automation
- ...

...
Eclipse 4.0

- **Underlying forces**
  - Make it easier for developers (API) and end-users (UI)
  - Address new targets / changing technology

- **Multi-year effort**
  - **This year:**
    - scoping, modeling existing function, prototyping, working on base technology
    - focus still on shipping 3.4
  - **Next year:**
    - significant changes across multiple components
    - most developers working on 4.0 stream
Eclipse 4.0

- Areas currently under investigation
  - Provisioning
    - "p2" work already well underway
    - "zero pain" delivery of function
  - User-interface / SWT
    - Model the capabilities of the workbench
    - Need to make it easier for developers to work with
    - Investigating: skinning, scripting, web based presentations
  - SWT
    - Investigating new capabilities of existing platforms animation, resolution independence, vector graphics
    - Investigating the potential for web-based targets
  - Java
    - New language features (Java7)
    - Multi-processor and distributed compilation
Conclusion

- Eclipse 3.3.1 is available
- Eclipse 3.4 is in progress
  - Focus on consolidation
  - Selective innovation
    - API Tooling, Provisioning, …
- Eclipse 4.0 plan is incubating
- Now is a great time to get involved!
Questions
Not so fine print

- Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.
- Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.
- Other company, product, or service names may be trademarks or service marks of others.
- Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.
- OSGi is a registered trademark of the OSGi Alliance in the United States, other countries, or both.