



Eclipse Platform 3.1 Release Review

Eclipse Platform PMC



3.1 Highlights from 10,000 Meters

- J2SE 5 support
- Performance & scalability
- RCP is real
 - Tool support is improving
- Misc

3.1 Themes and Plan Items

- **Scaling Up**
 - Large-scale workspaces
 - Process improvements wrt performance
 - Scalability
 - Breakpoint improvements
 - Add critical performance instrumentation
 - Help search enhancements
 - RCP performance
- **Enterprise Ready**
 - Import/export of Ant build files
- **Design for Extensibility: Be a Better Platform**
 - Mapping logical views to physical files on disk
 - Improve action contributions
 - Generalized undo support
 - Content types
- **Rich Client Platform**
 - RCP performance
 - RCP infrastructure
 - OSGi
 - Dynamic plug-ins
 - JNLP Support
 - Support for launcher branding
 - Provide better text editor support for RCP
 - [PDE] RCP support
 - OSGi bundle manifest tooling



3.1 Themes and Plan Items

- Simple to Use
 - Overhaul preferences
 - Improve capabilities
 - Ant editor improvements
 - Initial user experience improvements
 - Help search enhancements
 - Pervasive context help pane
 - Debugger usability improvements
- Appealing to the Broader Community
 - Ant debugger
 - Add support for J2SE 5 features
 - Improved compiler checking
 - Improve support for externalized strings
 - Improve program manipulation infrastructure



3.1 New and Noteworthy Items Platform

- Folding in the Ant Editor
- Move to Ant 1.6.5
- Perspective button dragging
- Managed-only plug-ins
- View all available keyboard shortcuts directly from the UI
- Browser widget improvements
- CVS outgoing commit sets
- Downloading of updates in background
- Open properties dialog from active editor
- Open Untitled File and improved file buffers
- Ant 1.6.2 code completion
- Hyperlink navigation in Ant editor
- Errors now show in external Ant buildfiles
- Improved Ant navigation
- Ant manual at your fingertips
- File names containing colon and backslash characters
- JFace and Workbench support BIDI
- AbstractUIPlugin no longer uses runtime compatibility layer
- Field editor added for encoding support
- TableView supports SWT.VIRTUAL
- ITableColorProvider and ITableFontProvider support added to TableTreeViewer
- Improved error dialog for background operations
- Extra context for errors in background operations
- Browser.execute(String)
- GtkFileChooser
- Change encoding and convert line delimiters
- Support for CVSNT
- Update policy file allows suppression or redirection of discovery URLs
- Update site mirroring
- Custom install handlers can access UI classes
- Ant debugger
- Multi-line text search and replace
- New spinner widget
- GC.setLineCap, GC.setLineJoin
- Mouse wheel events
- Support for executable and archive attributes
- Improved layout support
- SWT for Solaris-GTK+
- Selection clipboard
- Importing multiple projects
- Property page categorization
- Preferences dialog filtering
- Import/Export to tar.gz format
- New RCP API
- CVS commit review
- CVS improved file type support
- CVS branch and merge
- CVS filtering tags
- RCP delta pack
- Shared editor preferences
- New hyperlink support
- Import/export update sites bookmarks
- Improved support for FTP update sites
- Streamlined feature installation
- Native tree with columns
- Reorderable table columns
- Windows XP look and feel for Buttons
- Starting SWT standalone applications inside Eclipse is now easy
- Advanced graphics
- New Link widget
- Don't delete config.ini
- Searching in the Properties dialog
- Deferred content providing
- Keywords in the preferences dialogs
- New Undo/Redo API
- WorkbenchAdvisor API refactored
- New wizard registry API
- New view registry API
- Import and Export specific sets of preferences
- Welcome launch bar
- XHTML support for authoring Welcome pages
- Controls in FormText

3.1 New and Noteworthy Items – continued

- New Help view
- CVS operations on Java packages are no longer deep
- Word completion
- Ant "objects" presentation in the UI
- Ant debugger call stack and properties
- Ant target per build kind
- Performance view
- Preference page links
- Content type preference page
- Editor lookup based on content type
- New capabilities preference page
- Dynamic Help now available for all the dialogs
- Workbench Browser support API now available
- Single JAR plug-ins
- Strike through and underline in StyledText
- Spinner supports decimal values
- Emulated table improvements
- Tree supports indices
- Preference for undo history size
- Mark Occurrences in the Ant editor
- More control over error reporting in the Ant editor
- Rename in File for the Ant editor
- Import existing project from archive
- MessageDialog has accessible listener
- Preferences and properties dialog highlights hits
- Improved error part
- File permissions propagated on import
- Improved handling of feature dependencies
- Undoing first change
- Dynamic content injection in Help topics
- Support for per-plug-in search index
- Help system topics in welcome
- HTML-based welcome on Linux and Mac OS
- [Ant] system-path variable
- Reconcile control in the Ant editor



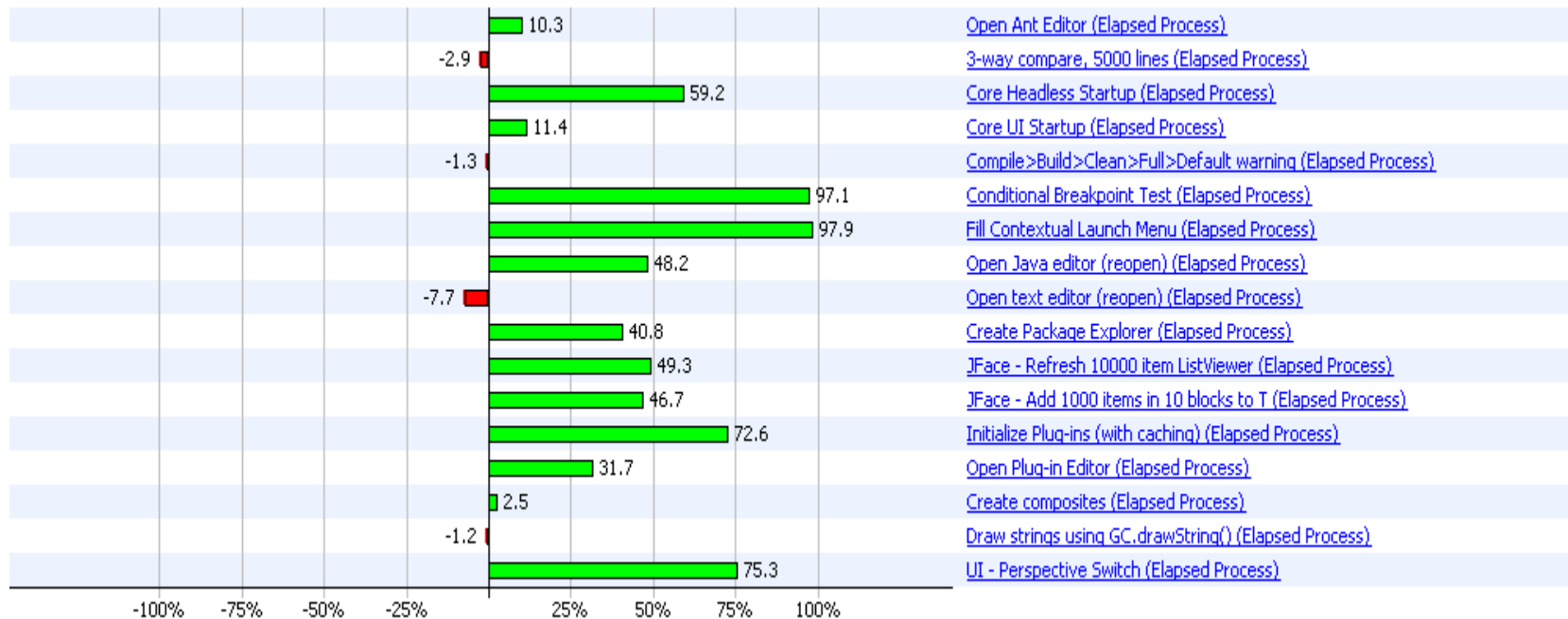
3.1 New and Noteworthy Items PDE

- Creating a new rich client application
- Quick launch of an Eclipse application
- Plug-in working set
- Validate plug-in set prior to launch
- PDE supports JAR'd plug-in format
- Test with your custom configuration file
- Improved plug-in manifest checking
- Sharing plug-in manifest compiler settings
- Improved plug-in dependencies view
- RCP application template
- No need to -clean
- New extension point schema attributes
- Improved manifest validation
- Consolidated target platform
- Build an Eclipse product with a single click
- New feature and update site manifest editors
- Manifest editor supports more OSGi bundle manifest headers
- Manifest.mf validation
- Create a plug-in from existing JAR
- Eclipse Product Export wizard
- Signed JARs
- PDE enforces code accessibility
- Cross-platform product export
- RCP template with an intro

3.1 New and Noteworthy Items - JDT

- Watchpoints and method entry breakpoints
- Debugging 1.5 source code
- New Quick fixes for serial version IDs
- Enhanced hyperlink style navigation in Java editor
- Highlighting of deprecated class members in the Java editor
- Breakpoint groups
- Support for J2SE 5.0 in Java editor
- Operating on generic type parameters
- Compiler diagnosis for 'enum' identifier
- Convert to enhanced for loop
- Pull up/push down refactoring
- Rename refactoring
- New Quick Assists for conditional statements
- Sharing Java compiler setting
- Evaluation support for generics and enhanced for loops
- Early detection of references to internal classes
- New Javadoc compiler settings
- Searching enums, varargs and static imports
- Search in generics
- Varargs argument needing a cast
- Code assist uses static imports
- Move member type refactoring
- New refactoring: Augment Raw Container Clients
- Convert Anonymous Class to Nested refactoring
- Move refactoring
- Mark occurrences of inherited methods
- Search for implement occurrences
- Improved Java properties file editor
- Working with externalized strings
- Locks and deadlocks
- Navigating stack traces
- 'toString()' inline
- Improved breakpoint groups
- Console EOF
- Console encoding
- Annotations
- Autoboxing
- New quick fixes for J2SE 5.0
- Organizing workspace with many projects
- New Occurrences Quick Menu
- Spell checking in Properties File editor
- Navigate from property key in Properties File editor to its references
- Project-specific code style settings
- Rerun failed tests first
- User-defined logical structures
- Enhanced variable value modification
- Capturing program output
- Multiple console views
- Breakpoint organization
- Find variable
- Boxing/unboxing diagnosis
- Assignment with no effect diagnosis for postfix expression
- Searching generic methods and constructors
- Searching type parameters
- Source Actions
- Change Signature and other refactorings are J2SE5-aware
- Folding provides better captions
- Usage of annotation type as super interface diagnosis
- Type parameter declaration hiding another type diagnosis
- Missing @Override annotation diagnosis
- Incomplete enum switch statement diagnosis
- Missing @Deprecated annotation diagnosis
- Completion on annotation attribute
- Completion on enum constant
- Support for package-info.java
- Code select on enum constant
- Code select in Javadoc
- Access rules on libraries and projects
- New API to create code proposals like in the Java editor
- Navigate to project-specific settings
- Use Supertype Where Possible refactoring
- Infer Type Generic Arguments refactoring
- Filter search result matches can now be rendered differently
- New search result filters for reference search for parameterized types
- Externalize Strings wizards supports new message bundles
- New Eclipse default built-in formatter profile
- Run and Debug from editor
- Javadoc attachments
- New Open Type dialog
- Performance improvements
- New Type wizards support generics
- Improved folding icons
- Parameter guessing for type parameters
- Support for @SuppressWarnings annotation
- Support for capture conversion
- Improved completion on empty word
- Completion inside annotations
- Support for Javadoc inside package-info.java

Performance of 3.1RC1 relative to 3.0



3.1 Plugin LOC Growth From 3.0

- Deleted
 - org.eclipse.help.ide
- Small (< 1000 LOC)
 - org.eclipse.ant.core
 - org.eclipse.core.filebuffers
 - org.eclipse.jdt.junit.runtime
 - org.eclipse.jdt.launching
 - org.eclipse.ltk.core.refactoring
 - org.eclipse.pde.runtime
 - org.eclipse.pde.core
 - org.eclipse.team.core
 - org.eclipse.team.cvs.core
 - org.eclipse.text
 - org.eclipse.tomcat
 - org.eclipse.ui.browser [NEW]
 - org.eclipse.ui.cheatsheets
 - org.eclipse.ui.presentations.r21
 - org.eclipse.ui.views
 - org.eclipse.ui.workbench.texteditor
 - org.eclipse.update.configurator
 - org.eclipse.update.core
 - org.eclipse.update.ui
- Major (> 10000)
 - org.eclipse.jdt.core
 - org.eclipse.jdt.ui
- Significant (1000 – 10000 LOC)
 - org.eclipse.ant.ui
 - org.eclipse.core.commands [NEW]
 - org.eclipse.core.resources
 - org.eclipse.core.resources.compatibility [NEW]
 - org.eclipse.core.runtime
 - org.eclipse.debug.ui
 - org.eclipse.help.base
 - org.eclipse.help.ui
 - org.eclipse.jdt.debug
 - org.eclipse.jdt.debug.ui
 - org.eclipse.jface
 - org.eclipse.jface.text
 - org.eclipse.osgi
 - org.eclipse.pde
 - org.eclipse.pde.build
 - org.eclipse.pde.ui
 - org.eclipse.platform
 - org.eclipse.swt.win32.win32.x86
 - org.eclipse.team.cvs.ui
 - org.eclipse.team.ui
 - org.eclipse.ui.console
 - org.eclipse.ui.editors
 - org.eclipse.ui.forms
 - org.eclipse.ui.ide
 - org.eclipse.ui.intro
 - org.eclipse.ui.workbench



Non-Code Aspects

- The 3.1 release will contain updated User and ISV documentation.
- Members of the team authored several Eclipse books:
 - SWT by Steve Northover and Mike Wilson
 - RCP (in progress) by Jeff McAffer and Jean-Michel Lemieux
 - FAQ by John Arthorne and Chris Laffra
 - Community is writing many more (approx 20) including versions for non English speakers
 - <http://www.eclipse.org/community/index.html>
- Community around key Platform technologies like SWT and RCP is very active.
 - Many blogs dedicated to Eclipse <http://planet.eclipse.org>



Non-Code Aspects

- Internationalization
 - Latin-1 locales are supported by the Eclipse SDK in every operating environments;
 - DBCS locales are supported by the Eclipse SDK on the Windows, GTK, and Motif window systems;
 - BIDI locales including support for mirroring is supported by the Eclipse SDK on Windows
 - The Eclipse SDK supports GB 18030 (level 1), the Chinese code page standard, on Windows XP and 2000, and GTK
- Localization
 - The Eclipse platform is fully localized and externalized.
 - Updated message catalogs will be provided
- Accessibility
 - The Eclipse platform is accessible

Non-Code Aspects

- Articles, Examples and Tutorials
 - Eclipse.org website contains several new articles that describe new features in the Eclipse Platform project.
 - Several of the articles were provided by members of the Eclipse community.
 - We will review our existing articles for currency during the 3.2 development cycle.
 - www.google.com

Platform Quality API

- API quality is a collaborative effort that involves the experience of the developers working on the Platform, and feedback from consumers.
- API Proposals are announced publicly and documents made available for comment
- The 3.1 migration guide identifies 13 changes
 - In each case a description of the change, what code is affected and the steps that need to be taken is described.
 - API changes between 3.0 and 3.1 have been checked by the component teams and double checked by an automated API diff tool
 - We are not aware of any API compliant plugins breaking as a result of these changes
- The PMC is comfortable supporting the API that is in Eclipse Platform 3.1.



API Changes in 3.1

- JFace preferences has been migrated to use the preference support that the Core component implemented in 3.0
- IPath restrictions on valid file names has been relaxed
- API calls that manipulate plug-in registry objects (e.g. IExtensionPoint, IExtension, IConfigurationElement) now throw an unchecked exception to identify references to invalid objects.
- A new constant was added to DefaultCodeFormatterConstants.
- JFace no longer requires the runtime plug-in
- In 3.0 an experimental element was introduced into the `org.eclipse.ui.commands` extension point. It has been removed in 3.1
- Spec changes to fix a number bugs in the 3.0 specification. See migration guide for details.



Architectural Issues

- Wider internal adoption of features from 3.0
 - PreferenceStore implementation was adapted to use the core API
 - Improved support for dynamic extension points
 - SWT Virtual Tables used in Problems and Task View



End-of-Life

- When evolving API the Eclipse Platform will whenever possible, deprecate the affected API methods and continue to keep them operational.
- With the exception of the API changes identified in the migration guide there are no end-of-life issues in 3.1.

Bugzilla

- Between July 1, 2004 and June 1, 2005
 - More than 18000 reports were created
 - 9000 were fixed
 - 5000 were resolved without changing code (duplicates, invalid etc)

- Current state is:
 - 15 blocking and 70 critical defects
 - 6 P1 and 234 P2 defects

- 3.0 state was:
 - 5 Blocker, 47 Critical
 - 7 P1 , 249 P2



Standards

- J2SE
 - Our tools are built against J2SE 1.4
 - Clients can run 1.4 or 1.5
 - Compiler can generate 1.3, 1.4 and 5.0 code
- OSGi
 - Involved in specification process for OSGi R4 (expected later this year)

Schedule

- Milestones every 6 weeks
- Tracked schedule
 - Every milestone except M5 was delivered as promised
 - M5a was created to fix a change that broke EMF
- Development ended at M7 on May 13 and we are in the end game phase
 - http://dev.eclipse.org/viewcvs/index.cgi/%7Echeckout%7E/eclipse-project-home/plans/3_1/freeze_plan.html
- Anticipate completion at the end of June.

Process

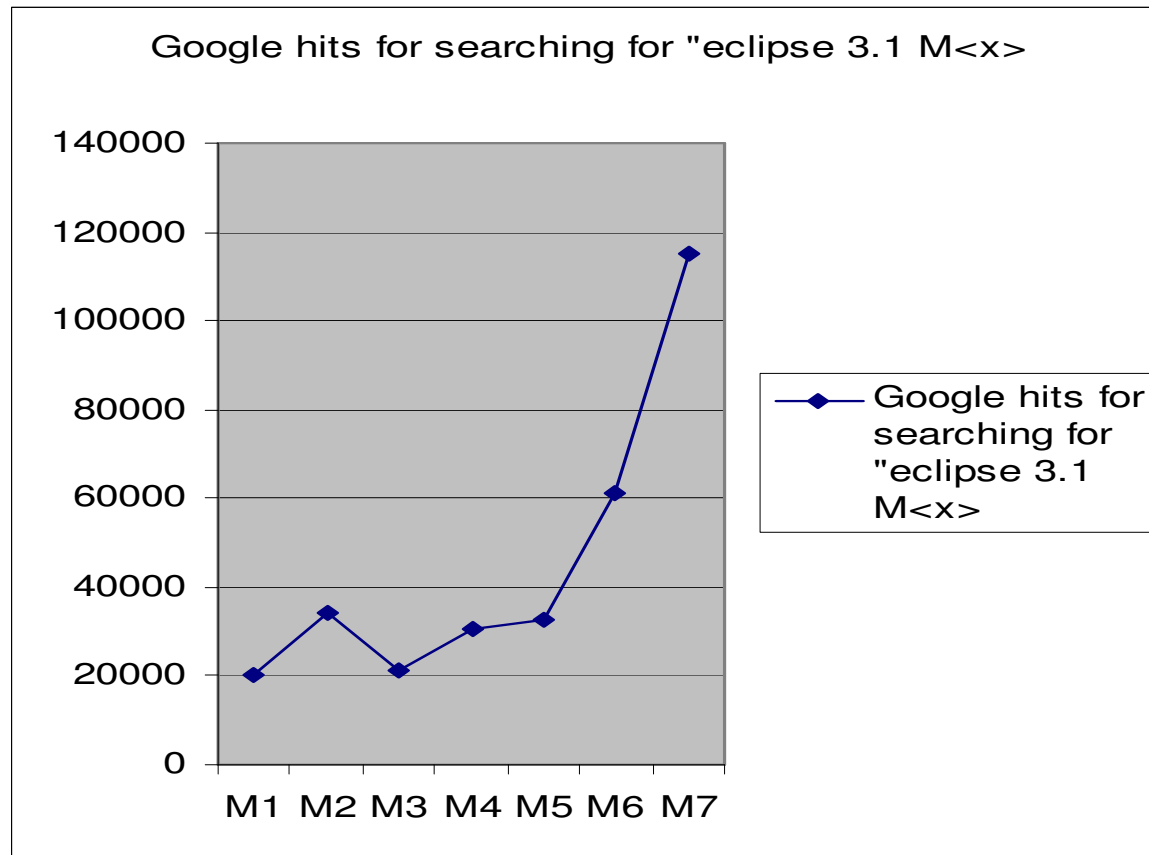
- The Eclipse platform is developed using an open, transparent and inclusive process
- Platform team relies on bugzilla, newsgroups and mailing lists.
 - New newsgroup were added to foster the growth of the RCP community
- Weekly planning calls conducted with the PMC and Component leads
 - Minutes posted to eclipse-dev mailing list



Community

- The Eclipse Platform team is active in bugzilla, mailing lists and newsgroups
- Several teams are using the eclipse-dev IRC channel
 - `irc.freenode.net#eclipse-dev`
- Several committers have started blogs and planetecclipse
 - <http://planetecclipse.org/>
- The Eclipse Platform team participates in codecamps, conference presentations and tutorials
 - Including JavaOne, WWDC, EclipseCon, OOPSLA, SDWest, JavaWorld, OSGi World,
- The Eclipse Platform team interacts with other open source projects, standards bodies as well as other projects at eclipse.org
 - Including: Apache Ant, OSGi, JLS

Community





IP Issues

- No outstanding IP issues



Project Plan

- Draft Eclipse 3.2 plan posted

http://www.eclipse.org/eclipse/development/eclipse_project_plan_3_2.html

- Key Themes
 - Scaling Up
 - Enterprise Ready
 - Be a Better Platform
 - Simple to Use