Eclipse Project 3.5 Release Review

Eclipse Project PMC

Highlights

- 3.5 new features:
  - New platforms: Mac Cocoa 32 and 64-bit, Solaris x86, os/390
  - Declarative services, block selection in editors, improved compare editors, PDE target management, extensible execution environments, improved test and build infrastructure
- API quality:
  - High. 7 changes in porting guide.
  - Binary compatible for compliant plug-ins
  - New API: 168 types, 351 methods
  - Deprecated API: 160 types, 51 methods, 23 fields
  - 7 breaking changes: Platform (7), JDT (0)
- End-of-life issues:
  - Bundle org.eclipse.pde.p2.ui removed, content merged into org.eclipse.pde.ui. No API involved.
- IP Clearance and Licenses:
  - All licenses and about files are in place as per the Eclipse Development Process, the Due Diligence Process was followed for all contributions
- Community and Committer Diversity:
  - 107 committers, 63 active in past 9 months
  - Organizations: IBM (51), Individuals (6), Code 9 (2), Adobe (1),Wind River (1), BestSolution (1), Red Hat (1)
  - Geographies: Canada, USA, France, Switzerland, Poland, Germany, Austria, Japan, India
  - Commits: IBM (91%), Individuals (4%), Wind River, Red Hat, Code 9, Adobe, BestSolution (1% each)
  - Consumed by many other Eclipse projects

Themes and Plan Items

- Scalability
  - Performance testing infrastructure improvements
  - Performance focus
- Consumability
  - User assistance improvements
  - Extensible execution environments
  - PDE build and export enhancements
  - Declarative services tooling
  - Debug user interface enhancements
  - PDE performance and target management
  - Team shareable working sets
- Platforms
  - Support Cocoa on Mac OS X
  - BIDI Improvements
  - Accessibility standards compliance
- Robustness
  - API Tools test suite and infrastructure
  - Compare editor improvements
  - Build process improvements
  - API Tools enhancements
  - Provide API for missing features

http://www.eclipse.org/projects/project-plan.php?projectid=eclipse

Themes and Plan Items

- Deferred plan items:
  - Team shareable working sets
  - Provide API for missing/internal features
  - Accessibility standards compliance

http://www.eclipse.org/projects/project-plan.php?projectid=eclipse

New and Noteworthy - Platform

- Eclipse on Cocoa
- Eclipse Solaris x86
- Eclipse on s390 and s390x
- Customize Perspective dialog
- Improved switch editors and multi-page editors
- Open Resource dialog allows choice of editor
- Improved Network Connections page
- Project Explorer improvements
- JFace dialog button order on GTK+
- Multi-instance Properties view pinning
- New workspaces preference page
- Compare Word documents
- New Compare With Other Resource dialog
- Compare editor enhancements
- Switchable compare viewers
- Custom icons in help system
- Buttons to change font size in Welcome
- Help quick search
- Install/Update Changes
- New software updates menu items
- New install wizard
- Install wizard only loads chosen repositories
- Install wizard usability and workflow improvements
- Installation details in About dialog
- New installation information dialog
- Auto-completion of repository locations in install wizard
- New available software sites preference page
- Block selection mode in textual editors
- Double-click collapses the folding region
- History in Tag Resources dialog
- Apply patch in the Synchronize view
- Synchronize schedule dialog improvements
- Debug view breadcrumb
- Declarative services added to the platform
- New compare core plug-in
- Improved look for filtered tree
- Multiple fonts in one cell
- JFace support for Mac Sheet dialogs
- New "check state" provider
- Show welcome checkbox
- New help content filter extension point
- Jetty version 6.1
- New release of ICU4J
- Extensible about dialog
- A new intro theme called "Slate" has been added

New and Noteworthy - Platform

- Add support for Cairo on AIX
- New tool for generating the SWT JNI code
- Block selection in StyledText
- JavaScript-to-Java communication in Browser widget
- Improved Java-to-Javascript support in Browser widget
- New caret listener API on StyledText
- New SWT text editor demo
- Improved search field
- Improved copy and paste support between Eclipse and Nautilus
- StyledText now supports hyperlinked text
- Configurable StyledText margin spacing and color

- Ability to disable Javascript in the Browser widget
- Create and access cookies in Browser
- New Browser authentication listener
- Alternate button order on GTK
- New Shell modified state hint
- Support for drop-down style date/time widget

- SWT.SHEET style for Shell and Dialog on Mac OS X Cocoa
- Mozilla browser support on Solaris x86
- Improved DND feedback on Cocoa
- Cocoa drag and drop feedback
- Programmatically set paper orientation in PrintDialog

New and Noteworthy - JDT

- Code completion in constructors
- Open Implementation hyperlink
- Select comment with double click
- Quick Fix to start Rename refactoring
- Links in Javadoc headers
- Open in Properties File action in NLS Hover
- Formatter option to preserve user line breaks
- Coded formatted with never join lines
- The Java Compare editor now supports:
  - Move/copy/delete line
  - Formatting
  - Hyperlinking
  - Content assist
  - Javadoc hovers
  - Quick outline
  - Reconcile Java structure while typing
- Generate toString()
- Optionally use blocks when generating hashCode() and equals()
- Comparing identical values compiler warning
- Missing synchronized modifier compiler warning
- Problem hover with quick fix for missing synchronized modifier
- Missing hashCode() method detection
- New compiler dead code detection
- Classpath resolution honors the 'Class-Path' header of JAR manifest file
- Build path supports ".."
- Build path resolution tolerates duplicate entries
- Compiler compliance follows execution environment
- Emphasized matching characters in Open Type dialog
- Sort working sets in Package Explorer
- Delete working sets from the Package Explorer
- Paste patch into Package Explorer
- Call Hierarchy can expand with constructors
- Expand With Constructors
- Improved Javadoc view and hover
- Open *.jardesc files with JAR Export Wizard
- Open test result files in JUnit view
- Importing test run history
- Moved up JUnit version to 4.5

New and Noteworthy - PDE

- New declarative services editor
- DS tooling supports latest DS specification
- Jump to Console from stack trace
- New compiler options to validate version numbers on packages and bundle
- New properties editor for .options
- Bundle start levels and license info added to product editor
- You can now initialize product definitions from OSGi launch configurations.
- New NL fragment generation wizard
- Menu introspection added to Plug-in Spy
- Plug-in registry view now supports browsing OSGi services
- Eclipse launch configurations now support start levels and auto-start settings
- JUnit plug-in tests in non-UI thread
- The Plug-in Registry view now includes fragments.
- Plug-in export wizard now supports installing into the running platform
- Patched plug-ins will appear in the installed software list and can also be uninstalled.
- Export options have been enhanced:
  - Support added for generating source bundles
  - Export features, plug-ins, and products with binary cycles.
  - Export existing class files from the workspace.
- New category editor to support publishing feature categories
- JAR signing support expanded to include support for keypasses.
- Improved API tooling compatibility options
- @noextend restriction now supported on interfaces
- System library validation in API tools
- Warnings and quick fix for stale problem filters
- Ant tasks added to perform API analysis and generate simple HTML reports
- New Compare With > API Baseline in API tools
- PDE Build support added to sort plug-ins across feature boundaries
- Parallel compilation support added in PDE build
- Update integration between PDE/Build and p2
- Qualifier replacement support on .product files
- Added support to fetch artifacts for build using p2
- p2 repositories supported as a PDE/Build target
- New target platform and target definition page
- Support for multiple targets and target switching
- Target editor enhanced to support features in targets
- New Target Platform State view displays details about current target platform

3.5 Plug-in Changes from 3.4

Added Plug-ins (7)

- org.eclipse.compare.core
- org.eclipse.core.databinding.observable
- org.eclipse.core.databinding.property
- org.eclipse.pde.ds.core
- org.eclipse.pde.ds.ui
- org.eclipse.pde.ua.core
- org.eclipse.pde.ua.ui

Removed Plug-ins (1)

- org.eclipse.pde.p2.ui

Added 3rd Party Plug-ins

- org.apache.commons.codec
- org.apache.commons.httpclient
- org.eclipse.ecf.provider.filetransfer.httpclient
- org.eclipse.ecf.provider.filetransfer.httpclient.ssl
- org.eclipse.equinox.concurrent
- org.eclipse.equinox.ds
- org.eclipse.equinox.p2.publisher
- org.eclipse.equinox.p2.repository
- org.eclipse.equinox.p2.repository.tools
- org.eclipse.equinox.p2.ui.sdk.scheduler
- org.eclipse.equinox.util
- org.mortbay.jetty.server
- org.mortbay.jetty.util

Removed 3rd Party Plug-ins (1)

- org.mortbay.jetty

Note: 3rd party plug-ins are plug-ins consumed in the Eclipse SDK but not produced by the Eclipse Project

Non-Code Aspects

- The 3.5 release will contain updated User and ISV documentation
- Community is very active
  - Mailing lists and newsgroups have steady activity
    - new E4 mailing list: e4-dev@eclipse.org
  - Blogs dedicated to Eclipse are active e.g.
    - http://www.planeteclipse.org
  - Wiki content is growing
    - http://wiki.eclipse.org/index.php/Eclipse_Project
    - E4 wiki: http://wiki.eclipse.org/E4

Non-Code Aspects

- Internationalization
  - Latin1 and Latin2 locales are supported in all operating environments
  - DBCS locales are supported on all platforms
  - BIDI locales (with mirroring) supported on Windows and Linux GTK, BIDI text supported on Mac.
  - GB18030-1 Chinese codepage standard is supported on Windows, Linux GTK and Mac.
- Localization
  - Tested for Localization and participating in Babel Project
- Accessibility
  - Tested for accessibility
  - Open accessibility bugs: 8 major, 0 critical, 0 blocker

Non-Code Aspects

- Articles, examples, and tutorials
  - New and updated articles and tutorials on eclipse.org (3)
    - http://www.eclipse.org/articles/article.php?file=Article-Adapters/index.html
    - http://www.eclipse.org/articles/article.php?file=Article-DynamicCSH/index.html
    - http://www.eclipse.org/articles/article.php?file=Article-AutomatingDSLEmbeddings/index.html
  - Numerous Webinars and Podcasts
  - Some of the new/updated articles and tutorials were provided by the Eclipse community
  - Older articles need to be reviewed and updated for 3.5, if applicable

Platform Quality API

- API quality is a collaborative effort that involves the experience of the developers working on the Eclipse project, and feedback from consumers.
- API changes and proposed API additions are often broadcast to mailing lists to raise awareness of the changes and encourage discussion and feedback.
- API changes between 3.4 and 3.5 are checked automatically by API tooling integrated into integration build process.
- The 3.5 migration guide identifies 7 changes:
  - http://dev.eclipse.org/viewcvs/index.cgi/org.eclipse.platform.doc.isv/porting/3.5/incompatibilities.html?view=co
  - http://dev.eclipse.org/viewcvs/index.cgi/org.eclipse.jdt.doc.isv/porting/3.5/incompatibilities.html?view=co
  - For each, a description of the change, what code is affected, and the action that needs to be taken is described.
  - We are not aware of any API compliant plug-ins breaking as a result of these changes.
  - The 3.5 migration guide also describes changes required to adopt mechanisms and APIs that are new in 3.5.
- The PMC is comfortable supporting the API that is in the Eclipse project 3.5

3.5 API – Platform

New

- API to defer updates when changing command context
- Added API for Ant launch configuration attributes
- Added extension point for breakpoint import participants
- Added API to retrieve the debug context associated with a command's execution event
- Added API for debug preference settings
- API allows detail panes to specify a selection provider
- Added API for re-useable working directory selection UI component in launch tabs
- Added API to export/import breakpoints to/from a string buffer
- Added extension point to contribute editor/debug model specific breakpoint factories
- New API to allow a web server other than Jetty to be used as the help server.
- New API to allow preprocessing to be performed on html and xhtml pages returned from the help server.
- New API to support split button role for accessibility
- New API to support authentication in the Browser
- New API to support cookies in a Browser
- New API to support calling between Java and JavaScript
- New API to track the locations of the caret in a StyledText control
- New API to support block selection in a StyledText control
- New API to offset the source image when a drag is started
- Support for embedded links was added to StyledText
- Support for shared OpenGL contexts was added to GLCanvas
- Support for landscape printing was added to Printers
- Support for setting the Mac modified state was added to Shells

New

- Support for Mac sheet windows was added
- IFileBufferManager now allows executing a runnable in the manager's synchronization context
- New method on FindReplaceDocumentAdapter to allow escaping a string for usage in a regular expression
- New APIs to support block selection in text views and editors
- New API in DefaultTextDoubleClickStrategy to allow subclasses to provide the double-click selection
- New editor action constructors for creation with default label
- New API in TemplateStore for restoring default values
- WhitespaceRule now allows setting a custom whitespace token
- New API in DefaultHyperlinkPresenter to support native hyperlink color
- New API to remove a search query
- New API for Properties, making it easier to set up bindings, and to implement custom observables
- Better support for binding to JFace viewers
- Support listening for disposal of data binding observables
- More general-purpose data binding observables
- Support opening multiple editors at once
- API to hide and show editors based on current perspective
- API for multi-instance properties views
- API to support an extensible About dialog
- Filling in missing API in org.eclipse.ui
- API to support check state and radio state for commands
- New API added for applying patches
- Better support for configuring and customizing compare editors

3.5 API – Platform

New

- Common Navigator enhancements:
  - capabilities support,
  - Go Into,
  - better drag&drop support for clients,
  - enable subclassing
- New API for better Open With support
- Support view folders that remain visible after their last child is closed
- API to configure new look for FilteredTree
- Allow programatic additions to the menu system via the extension point
- Improvements to error handling API
- New API to support toolbar configurability
- Enable setting style bits for JFace dialogs (for SWT.SHEET)
- Various new convenience API added to JFace:
- Helper dialog to make viewers with columns keyboard-accessible
- Common API for testing which platform is running
- Improve support for trees and tables with checkboxes
- Add font support for StyledCellLabelProvider
- Enable better performance for TableViewers with SWT.VIRTUAL
- New API to create content describers for XML files
- New API in IProxyService to handle URI

Deprecated

- Deprecated old Navigator view
- Deprecated org.eclipse.update component
- Deprecated org.eclipse.core.runtime.Preferences API
- Deprecated data binding BidirectionalMap and MappedSet in favor of BidiObservableMap
- Deprecated text editor SaveAction, replaced by ActionFactory.SAVE

- 51 assorted method deprecations
- 5 assorted constructor deprecations
- 23 assorted field deprecations

3.5 API – JDT

New

- Added API to create a wildcard type binding key from a generic type, a bound kind, a bound type and a rank.
- Added APIs for constructor invocation completion proposals.
- Added APIs that allow the cancellation of code completion.
- Added code assist options to define the prefixes and suffixes when completing a static final field name.
- New compiler option to report comparison of identical expressions.
- Added compiler option to report dead code.
- Added compiler option to report missing hashCode() method.
- Added compiler option to report a missing synchronized modifier on inherited method.
- Added naming convention API to get the base name from a variable name.
- Added APIs to allow a working copy owner to define a type or package that would be otherwise unknown to the type system.
- Added API to convert a char array to upper case.
- Added DOM/AST binding API to get the generic type of a wildcard type.
- Added DOM/AST binding API to obtain rank of a wildcard type.
- Added DOM/AST API to find the covered node and the covering node for a given range.
- Added code formatter options to join wrapped lines in code and to join lines in comments.
- Added search pattern API to get the regions in a name matching a pattern using a match rule.

New

- Previously internal CompilationUnitChange promoted to API
- Previously internal RefactoringSaveHelper promoted to API
- Many new APIs to support new clean up extension point
- New APIs to provide type parameter labels
- GenerateToStringAction introduced to invoke toString() wizard
- New API to allow inclusion of empty directories when creating a runnable JAR
- New API to configure a new Java project with a given compiler compliance
- New API in JUnitCore for finding tests for a given Java element
- New API to retrieve/specify compiler compliance options and system packages for execution environments

Deprecated

- Deprecated JDT “show in” actions, replaced by generic Navigate > Show In

3.5 API – PDE

New

- Added API for Eclipse Application launch configuration attributes
- Added API to support JUnit plug-in tests run in non-UI thread
- Added API to create a plug-in project creation wizard based on a specific template

Deprecated

- None

Tool Usability

- Eclipse is a superior IDE for Java tooling and plug-in development
- Many usability enhancements made in 3.5 to continue this tradition
  - Declarative services tooling
  - Execution environment validation in PDE and JDT
  - Powerful new PDE target definition support
  - Block selection in editors
  - Full editing capabilities in compare editors
  - Help quick search
  - More powerful perspective customization
  - Breadcrumb debug view
  - Content assist for constructors
  - Many new Java compiler diagnostics, including dead code analysis
  - And many more tooling improvements!

Architectural Issues

- Primary runtime is still a 1.4 JRE. Complementary functionalities on Java SE 5 (junit4, APT 5) and Java SE 6 (APT 6, compiler API)
- Minimum execution environment for RCP moved up from Foundation 1.0 to Foundation 1.1
- 20 new plug-ins, 2 removed plug-ins (org.mortbay.jetty and org.eclipse.pde.p2.ui)
  - 3 new plug-ins in platform
  - 4 new plug-ins in PDE
  - 13 new plug-ins due to external dependencies (Equinox, Orbit, ECF)

End of Life Issues

- When evolving API the Eclipse Platform will, whenever possible, deprecate the affected API methods and continue to keep them operational.
- Exceptions to this rule are in the 3.5 migration guide.
- All API and functionality provided by the org.eclipse.update component is deprecated in 3.5. This component is superseded by the Equinox p2 component.
- The platform moved to a new major version of the Jetty web server (from 5.1 to 6.1). There are breaking changes between these two major releases

Bugzilla

- Between June 25, 2008 and May 30, 2009 (RC3)
  - More than 10,500 reports were created
  - Over 9,100 were resolved
  - Over 3,800 were resolved without changing code
    - invalid, duplicate, worksforme, etc...
  - 395 were backported to 3.4.x maintenance
- Current state (RC3) is
  - 18 blockers, 41 critical
  - 0 P1, 49 P2 (3 planned for 3.5.1)
- 3.4 final state was
  - 18 blockers, 57 critical
  - 0 P1, 34 P2

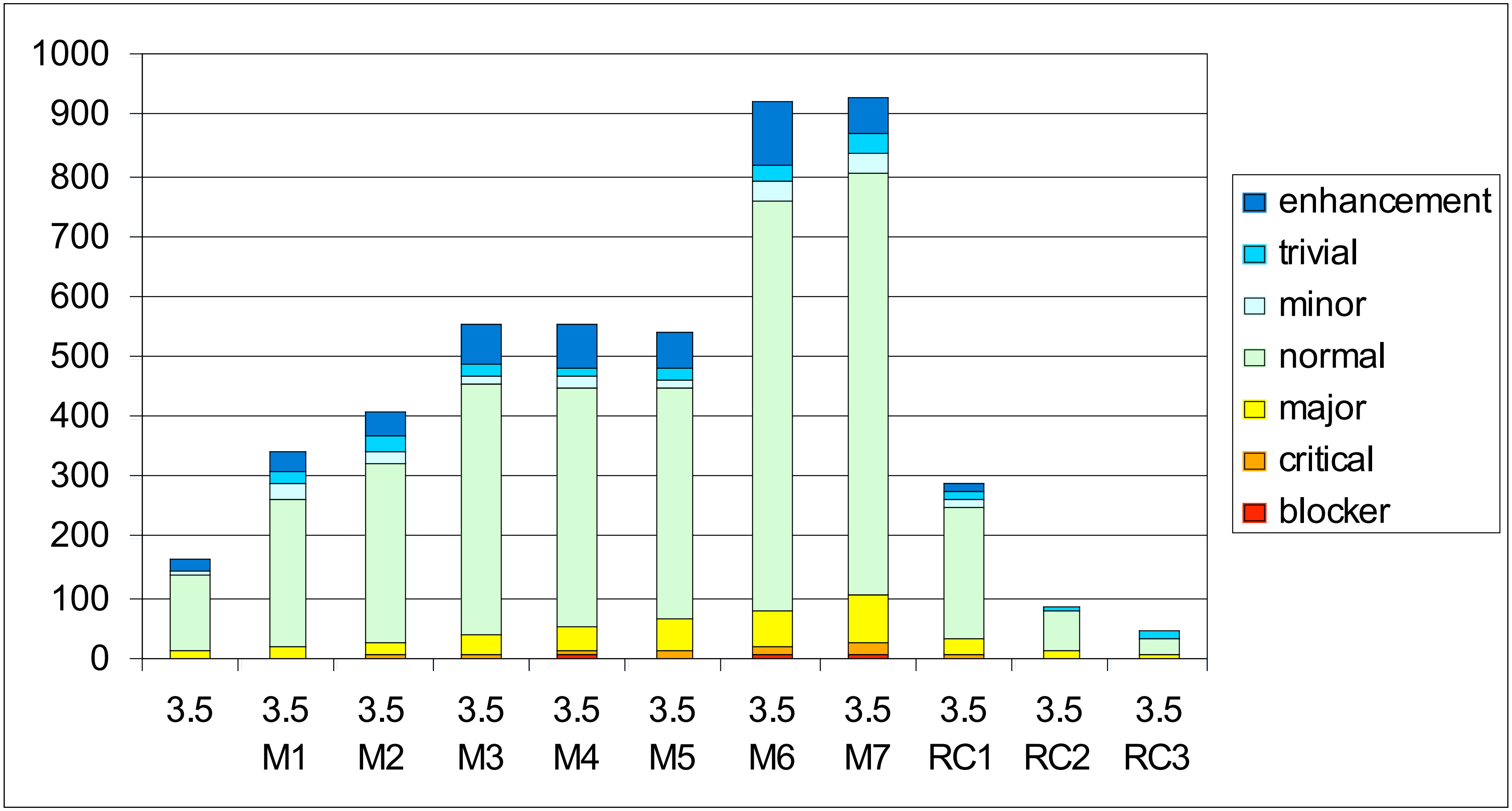
Bug resolution during 3.5

RESOLVED	M1	M2	M3	M4	M5	M6	M7	RC1	RC2	RC3	RC4	3.5	Total
blocker	1	3	1	6	2	10	6	2	3	1	0	0	**35**
critical	2	6	8	9	11	12	19	4	1	2	0	3	**77**
major	18	19	34	38	52	61	84	30	11	14	0	11	**372**
normal	242	295	416	395	386	681	697	217	68	43	4	130	**3574**
minor	26	24	14	21	15	27	31	15	1	5	1	5	**185**
trivial	20	21	15	15	16	29	38	14	3	13	1	2	**187**
enhancement	34	40	65	74	58	105	57	8	1	2	0	18	**462**
Total	343	408	553	558	540	925	932	290	88	80	6	169	4892

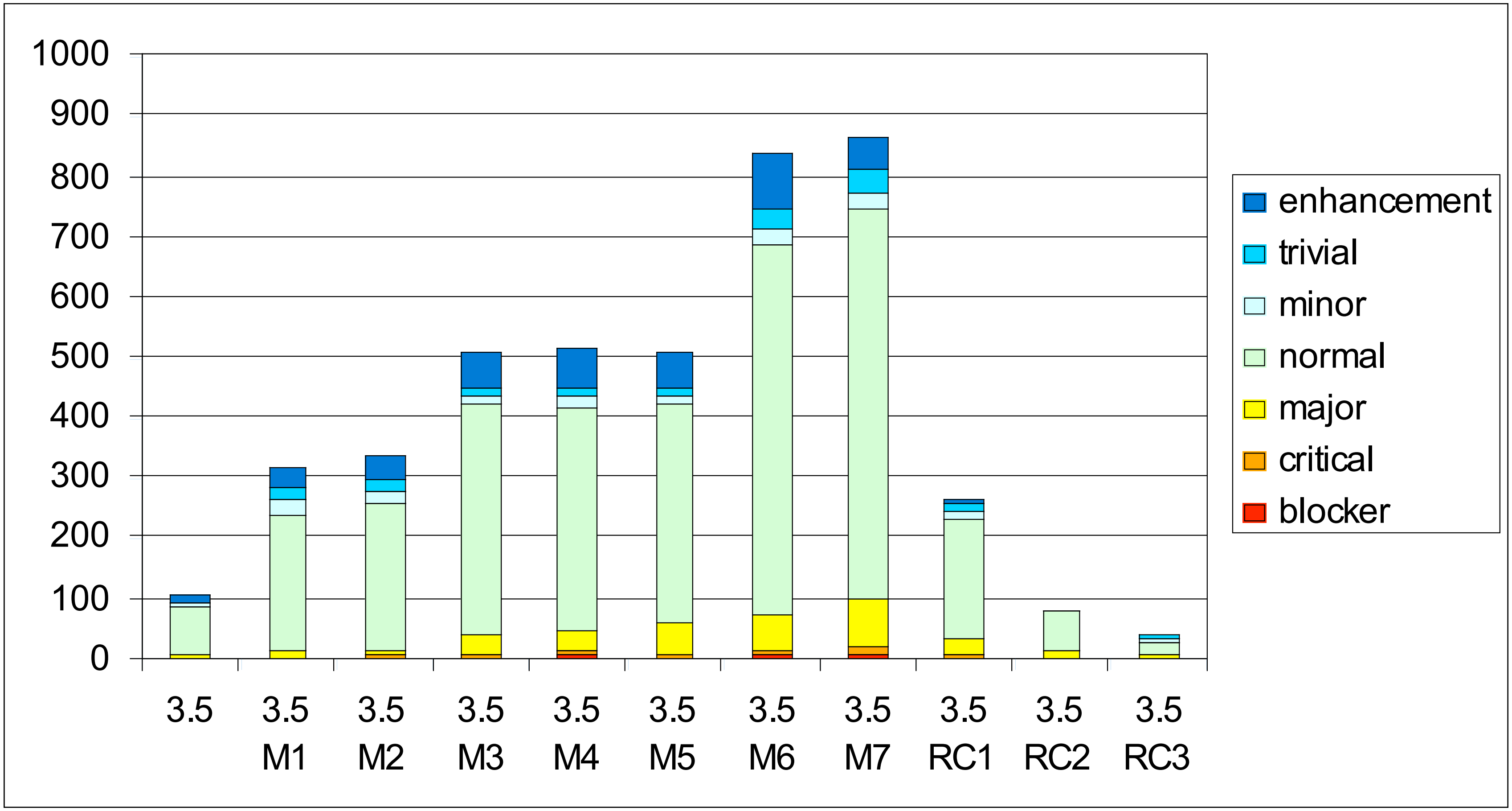
FIXED	M1	M2	M3	M4	M5	M6	M7	RC1	RC2	RC3	RC4	3.5	Total
blocker	0	2	1	6	1	8	5	2	3	0	0	0	**28**
critical	1	4	5	7	10	9	17	4	1	2	0	3	**63**
major	14	12	32	35	48	56	77	28	11	7	0	9	**335**
normal	226	240	385	372	364	618	647	197	63	19	4	79	**3236**
minor	26	23	13	19	13	26	28	14	1	4	1	3	**172**
trivial	20	20	14	14	16	29	37	14	3	9	1	0	**178**
enhancement	32	34	61	64	56	97	55	8	1	1	0	13	**423**
Total	319	335	511	517	508	843	866	267	84	72	6	107	4435

Resolved bugs

including fixed, invalid, …



Fixed bugs (only)



Standards

- Annotation Processing APIs
  - com.sun.mirror 1.5
  - javax.annotation.processing 1.6
- Java compiler API
  - javax.tools 1.6
- User Assistance consumes (parses) a small subset of RSS 1.0 to get news from eclipse.org
- JUnit 3.8.2 and JUnit 4.3.1
- J2SE
  - Tools are built against J2SE 1.4
  - Compiler can generate 1.3, 1.4, 1.5, and 1.6 code
  - Clients can run 1.4, 1.5 or 1.6.
- SWT
  - Win32, GDI, GDI+, OLE, IE, Carbon, Cocoa, Core Graphics, Quick Draw, Safari, ATSUI, X Windows, X/t, Motif, GTK, GDK, Pango, cairo, ATK, Mozilla, Uniscribe, WPF, OpenGL

UI Usability

- Strings are externalized to support translation into other languages.
- Extensive use of mnemonics and shortcut keys in the user interface enhances usability.
- Full Bidirectional support (mirroring) on Windows and Linux GTK, bidirectional text on Mac OS X
- Accessibility support for Windows, Linux GTK and Mac OS X
- Open accessibility bugs:
  - 8 major, 0 critical, 0 blocker

Schedule

- Milestones every 6 weeks, 6 cycle duration
  - API frozen on March 13 (M6), Feature freeze May 1 (M7)
  - Adjusted M5/M6 duration (resp. 7 weeks and 5 weeks) for EclipseCon
  - http://www.eclipse.org/projects/project-plan.php?projectid=eclipse#release_milestones
- Tracked schedule
  - All milestones except M5 delivered as promised
    - M5 three days late due to last minute Eclipse Foundation certificate change
- End game (release candidate) milestones for 4 cycles
  - Duration reduced from 2-week to 1-week cycles at RC2 milestone
  - No new features or API allowed without proper approvals
  - Development to end on June 12, 2009
  - Increasingly stringent approval, checking, and change notification requirements in this stage
  - http://www.eclipse.org/eclipse/development/freeze_plan_3.5.php

Process

- The Eclipse project is developed using an open, transparent, and inclusive process
- Teams rely on Bugzilla, mailing lists and newsgroups for input
- Weekly planning calls conducted with the PMC and component leads
  - Meeting minutes posted to the eclipse-dev mailing list
- Component teams have publicly available milestone plans
  - Use project's web space on eclipse.org to broadcast component milestone plan items and provide status on each item, per milestone

Community

- Eclipse team members are active in Bugzilla, newsgroups, and mailing lists
- Blogs started by Eclipse committers are evolving
  - Use blogging infrastructure at Eclipse.org
  - http://www.planeteclipse.org
- Some teams are using the eclipse-dev IRC channel
  - irc.freenode.net#eclipse-dev
  - irc://irc.freenode.net/#eclipse-e4
  - also see: http://wiki.eclipse.org/index.php/IRC
- The Eclipse team participates in code camps, conference presentations, and tutorials, including
  - EclipseCon, JavaOne, JavaWorld, JAOO, Eclipse Summit Europe, Eclipse Forum Europe, JAX, JAX Asia
- The Eclipse team interacts with other open source projects, standards bodies, and other projects on eclipse.org, including
  - OSGi, Apache Ant, JLS, WTP, Apache Harmony, GCJ, GTK

IP Issues

- All significant and third party contributions have been reviewed and approved by Eclipse legal.
- About files and license files are complete and correct.
- Draft project logs:
  - http://www.eclipse.org/projects/ip_log.php?projectid=eclipse.platform
  - http://www.eclipse.org/projects/ip_log.php?projectid=eclipse.pde
  - http://www.eclipse.org/projects/ip_log.php?projectid=eclipse.jdt

Project Plan for Eclipse 3.6/4.0

- Still in planning stage
- Areas in exploration in e4 currently include:
  - Declarative definition of user interfaces
  - Model-based workbench
  - Skinnable UIs
  - Support for multiple languages, in particular Javascript and Clojure
  - A cleaner, simpler, and more powerful programming model
  - Compatibility layer for running full 3.x Eclipse platform on e4