Comparison of Bugzilla and JIRA for use in bug management (May 24th, 2011)

Color code	Major concern	Meets or exceeds expectation	Minor concerns and deficiencies

	Bugzilla	JIRA	Notes
Current Version	4.0.1	4.3.3	
Language implemented in	Perl	Java	The Problems of Perl & Bugzilla, http://avatraxiom.livejournal.com/58084.html
Required environment	Apache + CGI or mod_perl	Standalone or WAR/EAR	
Database	Oracle database unsupported. Support MySQL and PostgreSQL	MySQL, PosgreSQL, Oracle, SQL Server	Current Oracle driver is a "dangerous hack" Refer to http://bugzilla.org/cgi-bin/mj_wwwusr?user=jochen.wiedmann%40gmail.com&passw=&list=developers&brief=on&func=archive-get-part&extra=201008/8
Who do you call for support	Consultant, Community	Atlassian Support	
System Interfaces	BUGZILLA	JIRA	
• The bug system should be fully usable through a web browser	yes	yes	
• The bug system should be visible and accessible outside of the Oracle firewall	yes	yes	
• The bug system should have a web service API	XML-RPC	REST is preferred, SOAP and XML-RPC also available	http://confluence.atlassian.com/display/JIRA/JIRA+RPC+Services; http://www.bugzilla.org/docs/tip/en/html/api/Bugzilla/WebService/Server/XMLRPC.html
• The bug system should have a command line interface (CLI)	Contributed tool, not updated to the current version since 2.16	Contributed plugin, updated in April 2011	https://studio.plugins.atlassian.com/wiki/display/JCLI/JIRA+Command+Line+Interface ; http://www.bugzilla.org/docs/4.0/en/html/cmdline.html
• The bug system should have a rich client	3rd party client software available	3rd party client software available	
Access and permissions	BUGZILLA	JIRA	
• Ability to use an existing identity, an OpenJDK id or Oracle Single Sign-on	3rd party security plugins	JIRA Crowd	http://www.atlassian.com/software/crowd/
• Access rights should be applied at least at the bug level	Yes	Yes	
• Access right could be applied at the level of fields in a bug	Modifications to Bugzilla code	Field level data security plugin	
• Access permissions enforced in all interfaces, GUI, API, CLI etc.	Modifications to Bugzilla code	Field level data security plugin	
• Permissions system must be able to control visibility of sensitive information	Modifications to Bugzilla code	Field level data security plugin	
• It must be possible to search for issues without logging in and see public info on issues	Modifications to Bugzilla code	Feature	
• Clear Terms Of Use presented to end user before sign-up / use			
Query language and capabilities	BUGZILLA	JIRA	
• Strong query capabilities with a powerful query language	Perl	JIRA query language (JQL)	http://www.atlassian.com/software/jira/tour/reporting.jsp
• Query language like Monaco Expert Interface for Bugtraq2	Perl	JQL	
• System wide search across all fields including full text search	SQL, Perl scripts	JQL, SQL, Lucene	
• Search must cover all fields of an issue and must have a system to apply logic for combining search criteria	SQL, Perl scripts	JQL, SQL, Lucene	
• Must be able to support at least as rich of a query functionality needed to re-create current tools (Monaco, OKO, BugBoss, TrafficLight, etc)	None	JQL, SQL, Lucene	
• Bulk operations on multiple bugs on all fields through api or cli. All aspects of an issue should be editable in bulk form, including fields, state transitions, labels, comments etc.	SQL and Perl scripts	JQL, SQL	
• Search results are constrained by permission and privileges of the query creator	Modifications to Bugzilla code	JQL and permissions plugin	
• Direct SQL access to the underlying database	SQL	SQL	
Reports	BUGZILLA	JIRA	
• Generate bug reports in the system and provide bookmarks to the reports.	3rd party extension	Inbuilt	http://www.mediawiki.org/wiki/Extension:Bugzilla_Reports; http://www.atlassian.com/software/jira/tour/reporting.jsp
• Generate bug statistics (vote, watch, comments) with submitter info (ie. company name, customer name, email, etc)	3rd party extensions	Inbuilt	
• Historical bug charting on a per engineer and per project basis	Elementary system for generating	Inbuilt	http://www.bugzilla.org/docs/tip/en/html/reporting.html;
• Ability to easily share complex queries with other users.	Elementary system to share a query with a group. 3rd party plugins for more features.	Yes, Share JQL queries with arbitrary users and granular permissions	http://www.atlassian.com/software/jira/full-features.jsp#quick-search
• Query and reporting language to generate custom reports	SQL and Perl scripts	JQL	http://www.atlassian.com/software/jira/full-features.jsp#quick-search
• Ability to have bookmark-able and stable URLs to a bug	Extension to create bookmarkable URL's	Extension to create bookmarkable URL's	
• Ability to have bookmark-able URLs to bug queries. The URL should be able to link to a page that would display a given number of bugs.	Supported	Supported	
• Ability to provide bookmark-able URLs to reports	Reporting extensions available from 3rd parties. Limited bookmarking facilities available.	Sharable reports and extensive features to share reports	
Actions on issues / bugs / process	BUGZILLA	JIRA	
• Be able to very quickly create a new CR, or edit an existing one	Supported	Supported	
• Ability to create/use templates when creating bugs	Supported	Supported	
• CLI (maybe GUI, too) way to create a whole new CR by using all the information from an existing one; makes creating very similar CRs much faster.	Modification to a Perl plugin. http://www.bugzilla.org/docs/tip/en/html/cmdline.html	JIRA CLI minor modification	https://studio.plugins.atlassian.com/wiki/display/JCLI/JIRA+Command+Line+Interface#JIRACommandLineInterface-Examples
• Ability to set the rules regarding obligatory and optional fields when filing and processing particular types of bugs. Example: JCK's test challenges		Custom workflows can be triggered for custom fields	
• Human-readable bug identifiers, closer to 7 digits numbers rather than MD5 hashes	Supported	JAVA-123456789	
• Entire bugs can be marked as public (OpenJDK) or private(Oracle)	Permissions feature	Permissions feature	
• Add Attachments (for stack traces, screen shots etc...)	Feature	Feature	
• Add Tags/Labels/Keywords to bugs and also across different products or projects	Use of Custom Fields	Use of Custom Fields	
• Parts of bugs must be able to be made private (e.g. customer call records, info about system login info for machine that can reproduce the bug)	Modifications to Bugzilla code	Field level data security plugin	
• Flag bugs with appropriate keywords (security, confidential, licbug etc...)	Use of Custom Fields	Use of Custom Fields	
• Associate bugs with Companies / Licensees (similar to the way Service Requests work)	Use of Custom Fields	Use of Custom Fields	
• Bug categories (similar to current bug system product/category/subcategory)			
• Support different categories of issues (bug, RFE)	Single issue type, single workflow	Multiple issue types and workflows	
• Update bugs with additional information from Customers			
• Users need to be able to vote for issues so we can see which issues are important to users			
• Ability to create sub-issues to allow us to group our work into related units.	Create a dependency tree of bugs	Use sub-task feature which provides a child bug that inherits all properties of the parent and can have custom fields	
• Target a fix to a particular release (update/revision)	One targeted release supported. Custom fields for more targets	Custom fields for targeted / target releases.	
• Support of the multiple (fixed in) releases for the bug		Use sub-task fields or custom fields	
• Multiple version information, release and build (!) where the bug exists (SR in bugster) - mandatory (!) for submitter and where it is fixed (Targeted release in Bugster) - mandatory (!) for responsible engineer	Custom fields in bugzilla	Custom fields in JIRA. Use sub-task type to create issues with different	
• Bugs marked with a security flag must immediately be made invisible			
• Bugs can be marked as duplicates and grouped	Duplicates - Yes. Grouped - No	Linking issues (Duplicate is one link type). Linking is also used for grouping	
• Look up (old) bug ids from the existing bug system	Can force bug numbers from Bugtraq to be retained	Special field to display and link to old bugs. Possible to force bug numbers, but needs work. JIRA project name fix is not optional (JAVA-xxxxxx)	
• One-button click to mark a security or cte bug and populate into bug db	Extensions need to be written for interactions with internal systems	Plugins to interact with internal bug systems have to be written	
• Must be able to view complete bug history (detailed audit trail), with application of security priviliges	Modifications to Bugzilla code	Field level data security plugin	
• One-button click to populate a bug into bug db from new bug system and vice-versa	Extensions need to be written for interactions with internal systems	Plugins to interact with internal bug systems have to be written	
• Must be able to handle multiple call records			
Notifications	BUGZILLA	JIRA	
• Informative bug status change emails that highlight and identify what has changed	User customizable notifications	Extensive notifications system both by email and RSS	http://www.atlassian.com/software/jira/full-features.jsp#access-notifications
• Interest lists - people should be able to subscribe to product/subcategory/category to get email notifications upon the bug change	User customizable notifications	Extensive notifications system both by email and RSS	http://www.atlassian.com/software/jira/full-features.jsp#access-notifications
• Owner email notification for new bugs & and bug state changes	User customizable	User customizable	
• Arbitrary users/mail aliases should also be able to sign up for email notification for any bug	Yes	Yes	
Usergroups			
• External OpenJDK committers can create and edit bugs	Yes	Yes	
• File Bugs on behalf of customers	Yes	Yes	
Workflows & system customization	BUGZILLA	JIRA	
• Ability to customize system to efficiently represent our workflow.	Supports one workflow	Multiple workflows	http://www.atlassian.com/software/jira/tour/workflow.jsp
• Ability to create custom dashboards showing various queries, reports and graphs.	3rd party tool to do dashboards customizations	Inbuilt	http://server.bugsdashboard.com:81/wp/product-info/overview
• Ability to create multiple workflows and have multiple workflows active in the same system	No	Multiple workflows	http://www.atlassian.com/software/jira/tour/workflow.jsp
Integration to other systems	BUGZILLA	JIRA	
• See the source code associated with the fix for a bug or issue	-NA-		
• Access the provided test case for an issue (attachments)	Attachments	Attachments management system	http://confluence.atlassian.com/display/JIRA/Configuring+File+Attachments
• See what test case is being used to verify a fix by SQE	Attachments	Very powerful attachments management system.	http://confluence.atlassian.com/display/CONFEVAL/Using+JIRA+for+Test+Case+Management
• Integration with code review system. See the code review and approval status, without having to visit some other tool/UI	patchviewer	Crucible	
• Integration with Mercurial. Connection between code reviews and changesets and any build and test job ids. E.g. a push into a repository puts the bug in the right state for the right release and includes a URL to the changeset	-NA-	Fisheye/Crucible	https://studio.plugins.atlassian.com/wiki/display/FISH/JIRA+FishEye+Plugin
• Ability to run other workflows on top of the database, such as ccc, core team bug approval, etc.	One active workflow	Yes. JIRA supports multiple customizable workflows	http://www.atlassian.com/software/jira/tour/workflow.jsp
• Ability to create custom life-cycle policy per project to customize workflow (e.g. bug states). Example: would be nice to have "Fix is in PIT".	-NA-	Yes. JIRA supports multiple customizable workflows	http://www.atlassian.com/software/jira/tour/workflow.jsp
• Synchronization between bugdb and OpenJDK bug system needs to be automatic/ automated	Can be done with API's. Need work on authentication and access issues.	Can be done with API's. Need work on authentication and access issues.	
Administration	BUGZILLA	JIRA	
• Auto account creation with the lowest access-level to reduce workload in user administrations.	Yes	Yes	
• OpenJDK users need to be able to register accounts without any human interaction.	Yes	Yes	
Migration	BUGZILLA	JIRA	
• All historical bug information, including attachments, brought into new system	Yes	Yes	
• Historical bugs copies into new system (note: old private bugs needs to stay private)	Yes	Yes	
• Ability to look up old bugs by historical bug ids (e.g. 4958831)	Modifications to Bugzilla code	Old bug id can be stored	http://www.atlassian.com/software/jira/tour/bugzilla-importer.jsp
System features and properties	BUGZILLA	JIRA	
• Available 24/7 reliability with 99.9% uptime	With an appropriately provisioned resources this is achievable	With an appropriately provisioned resources this is achievable	
• Performance atleast at the level of the current (internal Bugtraq2) bugsystem used by the Java Group.	With an appropriately provisioned resources this is achievable	With an appropriately provisioned resources this is achievable	
• Scale in capacity to store many times more issues than and be able to dynamically provision to scale for higher traffic and bulk transactions. System needs to handle 500,000 issues.	Systems with this size in production. Cases of performance problems reported for larger systems.	Systems with this size in production. Cases of performance issues reported for larger systems.	In both cases, a dedicated db admin is required to monitor and flush out performance issues both in the application and the database layer.
• Redundancy features for a guaranteed SLA.	Hw/Db redundancy features needed	Clustering solutions and Hw/Db redundancy features needed.	
• Meets any 508 requirements as per Oracle Accessibility policies			
• Must be able to display/input non-Latin languages, such as Chinese/Japanese/Korean/ Arabic/Hebrew/Thai/Hindi.	Limited language support	19 languages	