



# Orion Project Graduation and Release Review

# Orion Graduation Review



- Demonstrable Codebase
- Active Communities
- How Orion is an Open-Source advocate
- The Eclipse Way as it applies to Orion
- APIs and Consumability
- Standard Release review from 0.5 to 1.0

# Demonstrable Codebase



- Orion has been hosting stable releases at <http://orionhub.org>
- Up to our 1.0 release there have been over 14500 accounts created
- During 6 week milestones, roughly 1500 unique logins occur
- OrionHub has been a great outlet for the community to try Orion
- Many extensions are available externally at <http://mamacdon.github.com>
- The site is performant and stable, to our knowledge no data has been lost
- This site will be moved to a backup site ([backup.orionhub.org](http://backup.orionhub.org)) and [orionhub.org](http://orionhub.org) will start from a clean state post 1.0

# Active Communities



- Orion has committers from different companies including IBM, VMWare, Mozilla, Google and others
- The source for Orion is being used by different projects such as Firefox, the Scripted code editor by VMWare and a cloud based development environment at [cloudfier.com](http://cloudfier.com)
- There are various plugins developed by both the team and the community, many of which while originally hosted at our plugins site end up being part of our default distribution
- Contributions have been made and new committers granted based on the capabilities, stability and continued interest in adding to and extending Orion
- Orion maintains a newsgroup as well as developer mailing lists for questions and answers

# Open-Source advocate



- Orion maintains consumable licenses on both client and server portions of the platform through EPL and EDL
- Orion has provided release plans for the 4 month development cycles leading up to our 1.0 release. The plan is to continue these cycles and incrementing our version number with each iteration (1->2->3).
- There is significant documentation on how to acquire, consume, contribute and test the various components that make up an Orion release. All the results of our builds and testing are made available on our downloads page.
- We hold weekly calls on both the overall project where anyone is welcome and we hold a weekly User Experience call where comments and feedback are welcome on our UI
- All defects are addressed through Bugzilla and potential contributors can provide fixes and request reviews right from Orion itself

# The Eclipse Way



- Orion is following “The Eclipse Way” with the exception of having releases every 4 months
- We maintain a clear IP log given that Orion consumes other JavaScript components. This has and will continue to challenge the Eclipse Way due to the frequency JavaScript projects progress and the occasional lack of a Contributor Agreement on projects such as those maintained at GitHub
- The team is extremely active in contributing through presentations at EclipseCon, JavaScript groups and conferences, through blog posts, twitter, Google+ and others
- Orion differs greatly from a Standard Eclipse Java based project or plugin and the team is helping pave the way for new approaches to development outside the standard IDE but with the focus on providing the same solid capabilities that makes Eclipse the leading IDE

# APIs and Consumability



- In the JavaScript world, APIs to functionality from Orion surface themselves most importantly at our extension/plugin model
- In 1.0 this API has matured and should remain stable post 1.0 with the exception of adding functionality
- Consumability comes by decoupling our features such as the Editor so that they may be included in a product without requiring all of the Orion Framework. This is already happening both with Mozilla and VMWare
- Orion has online and embedded documentation for our APIs, how to use them, and examples to copy or learn from
- As JavaScript frameworks themselves evolve (require.js, esprima, node.js, etc.) Orion will remain agile in consuming the leading edge technologies in our 4 month releases



# Standard Release Review Material for 0.5 to 1.0



# New Features



- Navigation features:
  - Compare any two resources (directories or files)
  - Hierarchical Outline view with folding
  - Open “raw” file vs. delegation to Orion for action
- Overall UI features:
  - New search suggestions and crawling search for Filesystems
  - Extendible New Content section instead of hardcoded Content
  - Themes settings page compatible with EclipseThemes
- New Landing Page
  - Rebranded again, now with Mozilla Persona login support as well
  - New account creation with email confirmation (email required)
  - Server Announcements and log from announcements

# New Features



- Editor Improvements:
  - Line wrapping
  - Editor Themes (styling with support for EclipseThemes)
- Content Assist:
  - New Esprima based content assist by default
- Git Improvements
  - New Gitlog page and improved dynamic Git page loading
  - Git Review Requests directly from Orion
  - Git Merge/Squash support
  - Related Links additions for viewing commits at external site
  - Saved Git Credentials local storage only

# New Features



- Admin:
  - Paginated User List
- Plugins:
  - Plugins can now have options displayed in the settings page
  - Plugins can be disabled/enabled without requiring reinstallation

# Non-Code Aspects



- Orion User Guide and Orion Developer Guide included with release. Documentation developed using Mylyn WikiText
- Developer Guide includes server administration guide that describes how to setup and manage the server
- Public beta server hosted by Eclipse Foundation: [orionhub.org](http://orionhub.org)
- Blog for New and Noteworthy updates: [planetorion.org](http://planetorion.org)
- (See Graduation review above)

# API



- Orion has initial JavaScript API for client side services and components
- HTTP API for accessing server functionality
- No Java API on the server to allow for alternate server implementation languages and technologies
- Client API documented using jsdoc
- Since the Web community is agile with respect to changing components, Orion will aim to maintain plugin metadata compatibility however some of the APIs around the TextEditor and Microservices will likely change over time to adapt to Orion consumption usecases.

# Non-Code Aspects



- **Internationalization**
  - I18N localization supported within 1.0 ([Blog post](#))
  - New architecture for allowing off site language packs
  - Plugin/Page to assist with generating message bundles
  - Editor is capable of displaying non-latin characters and supporting bidirectional text
  - Externalized message bundles for Orion pages
  - Working with Babel project for translations
- **Accessibility**
  - Remains a moving target due to high rate of code change
  - Team focused on adding ARIA tags as applicable through 1.0 and beyond

# Architectural Issues



- Strong architecture and modularity of Orion client
- CommonJS module format used to facilitate client side reuse
- JavaScript plugin architecture allows for integration via linking rather than a monolithic application: architecture designed for the web
- Editor and “text view” components factored out with no external dependencies to enable reuse by other applications
- Exploring how to make seamless tooling workflows across multiple web sites
  - Enabling commands on links inbound to Orion ("Clone this repository", "Install this Plugin")
  - Allowing Orion plugins to define outbound links to related content on other sites
- Targeting modern HTML5-compliant browsers: Chrome, Firefox, Internet Explorer, Safari

# Tool Usability



- Landing page now easier to use, branded and direct registration
- High priority was unification of pages for style and selection models
- Unified settings page for plugins/profile/other settings
- Esprima parser based content assist now the default
- More search options including REGEX
- Addressing performance concerns with in-depth analysis to page and plugin load times
- First time user presented with options to populate Navigator/Workspace
- Theming supported for Orion UI and Editor



# End of Life Issues



- None.

# Bugzilla



- Between Jun 22, 2012 and Oct 5, 2012 (RC1)
  - 424 reports were created
  - 321 were resolved/closed
- Current state (RC1) is:
  - 0 blockers, 0 critical
  - 0 P1, 5 P2

# Bugs fixed during 1.0



		Target Milestone				
		1.0	1.0 M1	1.0 M2	1.0 RC1	Total
Severity	critical	.	<u>2</u>	<u>2</u>	<u>1</u>	<u>5</u>
	major	.	<u>7</u>	<u>3</u>	.	<u>10</u>
	normal	<u>14</u>	<u>51</u>	<u>117</u>	<u>70</u>	<u>252</u>
	minor	.	<u>1</u>	<u>1</u>	<u>2</u>	<u>4</u>
	trivial	.	.	<u>1</u>	.	<u>1</u>
	enhancement	.	<u>2</u>	<u>9</u>	<u>4</u>	<u>15</u>
	Total	<u>14</u>	<u>63</u>	<u>133</u>	<u>77</u>	<u>287</u>

# Standards



- Client components conforming to CommonJS Asynchronous Module Definition (AMD) format
- HTML5 standard source code
- URI Templates
  - <http://tools.ietf.org/html/draft-gregorio-uritemplate-08>
- Participating in specification of CrossFire browser debug protocol
  - <http://getfirebug.com/doc/crossfire/crossfire-splash-2011.pdf>
- Mozilla Persona – New Identity system for login validation
  - <http://www.mozilla.org/en-US/persona/>

# Community



- Orion team members are active in Bugzilla, newsgroups, mailing lists, Eclipse forums and Twitter
- <http://wiki.eclipse.org/Orion/Buzz> - Continuous press
- Two Google Summer of Code Projects
  - Improving Git forking from GitHub into multiple Orion destinations
  - Google Chrome DevTools integration with Orion components
- Blogs started by Orion committers are active:
  - <http://planetorion.org>
  - <http://planetecclipse.org>
  - Team is using the eclipse-orion IRC channel
  - [irc.freenode.net/#eclipse-orion](http://irc.freenode.net/#eclipse-orion)
  - also see: <http://wiki.eclipse.org/index.php/IRC>
- The Orion team participates in code camps, conference presentations, and tutorials, including
  - EclipseCon, What's Next Conference, jsconf, Eclipse Demo Camps, OttawaJS
- The Orion team interacts with other open source projects, standards bodies, and other projects on eclipse.org, including
  - Mozilla, Dojo Foundation (Dojo framework, Maqetta), JGit, Mylyn, jsdoc, Firebug, CodeMirror, Esprima

# Schedule



- Utilizing a web-scale release cycle
  - Four month release cycle
  - 13 weeks development, 3 weeks end-game
- Daily build and deployment to self-hosting server
- Tracked schedule
  - All milestones delivered as planned

# Process



- Orion is developed using an open, transparent, and inclusive process
- Team relies on Bugzilla, mailing lists, forums, social media for input
- Weekly planning calls conducted with all contributors welcome
- Self-hosted development on <http://orion.eclipse.org>
  - Meeting minutes posted to the orion-dev mailing list
- Publicly available plan: <http://wiki.eclipse.org/Orion/Plan>

# 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.
- Client code (JavaScript) is dual-licensed with EPL and EDL
- Waiting for the 1.0 release log to be approved there was a server failure in submitting it



# Future Plans



- Continue with 4 month release cycle to keep up with pace of development in the web community. For example with new browsers shipping every 3 months we can't ship Orion once a year and expect to be current
- Aiming for a 2.0 release in February 2013
- Focus on APIs, Accessibility, Cross-site workflows, User Interface, continued improvements to language support, Git Workflows, Documentation, Performance, Security and deployment of Orion developed applications to hosted providers

