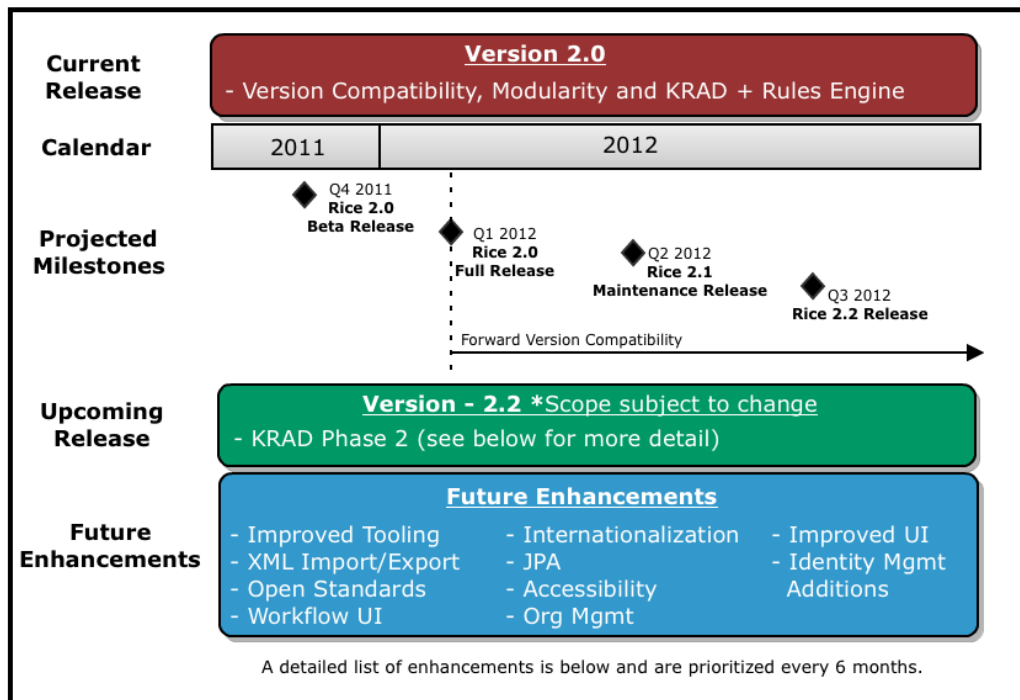


Kuali Rice Roadmap 2011



Timelines

The team is currently developing Rice 2.0 and is scoping Rice 2.2, which will be released approximately 6 months after the 2.0 Full Release. View a more detailed timeline and breakdown of Rice 2.0 features at [Rice 2.0 Milestones](#).

Maintenance patch releases and Milestone code releases are made available with greater frequency.

See details on our release cycles, i.e. frequency of major, minor, and patch releases at [Rice Roadmap - Release Cycles](#).

Features

Version 2.0 (Release Dates and Scope Firm)

key	summary	assignee	fixVersion	status
-----	---------	----------	------------	--------

com.atlassian.sal.api.net.ResponseException: javax.net.ssl.SSLHandshakeException: Received fatal alert: handshake_failure
[View these issues in JIRA](#)

Future Versions

Below is the prioritized list of features for future releases of Rice. Items for 2.2 will be pulled from the top of this list and number of items completed will be based on available resources.

1. [KRRM-135](#) Phase 2 essential enhancements for KS 2.0, ~ 4000 developer hours
2. [KRRM-50](#) A consistent documentation framework for all Rice components, ~ 200 developer hours
3. [KRRM-56](#) Expand support for XML import and export of data, ~ 500 developer hours
4. [KRRM-12](#) Implement Escalation Functionality, ~ 200 developer hours
5. [KRRM-14](#) Improve Capabilities of Exception Routing, ~ 50 developer hours
6. [KRRM-119](#) Implement ability to define custom "flow" for exception routing processes, ~ 50 developer hours

7. KRRM-70 Standardize Configuration Parameter Names, ~ 200 developer hours
8. KRRM-36 Research into various open standard workflow process definition languages (such as BPEL, BPMN, YAWL, XPDL, etc.), ~ 500 developer hours
9. KRRM-127 Research the approach and requirements for a GUI for KEW, ~ 200 developer hours
10. KRRM-102 Implement declarative loop support within the Kualu workflow engine, ~ 50 developer hours
11. KRRM-132 Implement integration of open standard workflow process definition languages, ~ 500 developer hours
12. KRRM-124 Rice tooling: administrative tools for all Rice components (KEW, KIM, BRMS etc). Generalizes KRRM-29, ~ 2500 developer hours
13. KRRM-129 Create a tool that can be used to generate various KRAD-required artifacts and accelerate general development velocity
14. KRRM-126 Implement a GUI for business activity monitoring
15. KRRM-128 Create improved and comprehensive documentation on how to use the KRAD development framework
16. KRRM-125 Improvements to the KSB Service Registry UI
17. KRRM-130 Create a "drag-and-drop" graphical tool that can be used to construct KRAD user interfaces
18. KRRM-6 Batch File Scheduler and Monitor
19. KRRM-1 Replace OJB with Java Persistence API (JPA)
20. KRRM-41 Research how Kualu Rice can better support Federated Identity Management Services
21. KRRM-22 Implement "update" operations for the various KIM services
22. KRRM-131 Implement Accessibility Standards
23. KRRM-114 Provide a remotable service API for import/export
24. KRRM-115 Allow for XML import operations to be decentralized
25. KRRM-101 Rice Framework Compatibility
26. KRRM-133 Implement Support for Federated Identity Management Services
27. KRRM-24 Add support for "versionable" documents
28. KRRM-16 Improved Error Handling and User Messages
29. KRRM-8 Document Search Improvements
30. KRRM-13 Implement simple Document-Type-based Delegation Features
31. KRRM-90 Add support in Rice for integration of a DMS system with Kualu applications
32. KRRM-44 Security Metadata - Data Dictionary enhancements
33. KRRM-25 Design and implement Kualu Organization Management (KOM)
34. KRRM-59 Architecture and functionality review of KEN
35. KRRM-74 Implement Referential Integrity on KEW database tables
36. KRRM-7 KIM Management Screens
37. KRRM-9 Action List Improvements
38. KRRM-27 Support for different DBMS platforms
39. KRRM-64 Work on a strategy for User Interface that can leverage a common dictionary service model
40. KRRM-31 RIA (Rich Internet Application) UI Frameworks
41. KRRM-82 Internationalization and Localization Features
42. KRRM-75 Consolidate Notes and Attachments between KNS and KEW and KS
43. KRRM-58 Convert KEN user interfaces to use the KNS
44. KRRM-77 Implement "Remove/Replace Principal" functionality
45. KRRM-61 Convert the KSB User Interfaces to use the KNS
46. KRRM-15 Remaining improvements to handling and consistency of dates and timestamps
47. KRRM-68 Implement proper handling of FERPA and FIPPA regulations
48. KRRM-19 Navigational Improvements in the KNS
49. KRRM-23 Implement a user interface for making changes to various aspects of the Data Dictionary
50. KRRM-94 Expand KIM functionality to incorporate full Identity and Access Management capabilities
51. KRRM-86 Investigate whether the "embedded" workflow engine model should be changed to not require direct database access from client applications
52. KRRM-21 Finish up remaining conversion of KEW screens to use the KNS framework
53. KRRM-88 Improve configuration of the KNS and Rice
54. KRRM-76 Rewrite eDocLite to use the KNS
55. KRRM-32 Mobile Application Support
56. KRRM-4 Replace KSB with alternative ESB