

Kuali Rice 2.2.3- SNAPSHOT User Guide

Table of Contents

| | |
|--|----|
| 1. Global | 1 |
| The Kuali Community | 1 |
| Kualifoundation | 1 |
| Kualirice | 1 |
| Licensing and Intellectual Property | 1 |
| Software License | 1 |
| Contributors | 1 |
| Documentation License | 2 |
| How to Use This Guide | 2 |
| User Guide Structure | 2 |
| Typographic Conventions | 2 |
| Versioning | 2 |
| Electronic Navigation | 3 |
| Printed Documentation | 3 |
| Common Features and Functions | 3 |
| Standard Buttons on Lookup Screens | 3 |
| Useful Screens Are Where You Need Them | 4 |
| Common Functions | 5 |
| Lookup Wildcards | 6 |
| Result Set Limits | 7 |
| Frequently Used Tabs | 7 |
| Reporting an Issue with Kualirice | 10 |
| Step 1: Search for a Similar Issue | 10 |
| Step 2: Create an Issue in JIRA | 12 |
| Step 3: Fill Out Required Fields | 12 |
| Step 4: Save the Jira issue | 15 |
| 2. High-Level Features Guide | 16 |
| Nervous System (KNS) | 16 |
| Service Bus (KSB) | 16 |
| Workflow (KEW) | 16 |
| Important Features of KEW | 16 |
| Important Features of PeopleFlow | 17 |
| Notification (KEN) | 17 |
| Important Features of KEN | 17 |
| Identity Management (KIM) | 17 |
| Rules Management System (KRMS) | 17 |
| Important Features of KRMS | 17 |
| Rapid Application Development (KRAD) | 18 |
| Important Features of KRAD | 18 |
| 3. KEN | 20 |
| Channels | 20 |
| Notification Channel | 20 |
| Managing Channels | 20 |
| Notification Channel Subscriptions | 20 |
| Managing Subscribers | 21 |
| Notification Priority | 21 |
| Managing Priorities | 22 |
| 4. KEW | 23 |
| Kualienterpriseworkflow: Overview | 23 |
| What is KEW? | 23 |
| KEW Features | 23 |

| | |
|--|-----|
| Why Use KEW? | 25 |
| PeopleFlow - a new KEW feature | 25 |
| Creating a new PeopleFlow | 25 |
| Actions | 29 |
| Action List Guide | 29 |
| Action List Preferences | 33 |
| Action List Refresh and Filter | 38 |
| Action Requests and Actions Guide | 42 |
| Overview | 42 |
| Types of Action Requests | 43 |
| Action Request Recipients | 43 |
| Action Request Hierarchies | 44 |
| Action Request Activation and Deactivation | 45 |
| Actions and Action Takens | 46 |
| Documents | 51 |
| eDocLite Overview | 51 |
| eDocLite Lookup | 51 |
| eDocLite Inquiry | 54 |
| Create New eDocLite Document | 54 |
| Preferences | 56 |
| General Preferences | 57 |
| Fields Displayed in Action List Preferences | 58 |
| Document Route Status Colors for Action list Entries | 58 |
| Routing | 58 |
| Delegate Rules and Delegation Routing Rules | 58 |
| Delegation Rule Inquiry | 60 |
| Delegate Routing Rule Creation | 61 |
| Routing Rule Delegation Maintenance | 61 |
| Rule Lookup | 67 |
| Rule Inquiry | 68 |
| Routing Rule Creation | 69 |
| Routing Report | 69 |
| Rule Maintenance | 71 |
| Rule Template Lookup | 74 |
| Rule Template Inquiry | 75 |
| 5. KIM | 76 |
| KIM Overview | 76 |
| KIM Features | 76 |
| Person | 78 |
| Person Lookup | 78 |
| Person Maintenance | 80 |
| Displaying the Person Lookup Screen | 81 |
| Ad Hoc Recipients Tab | 87 |
| Route Log Tab | 87 |
| Group | 87 |
| Group Lookup Screen | 87 |
| Group Inquiry Screen | 88 |
| Group Maintenance Document | 89 |
| Role | 92 |
| Role Lookup Screen | 92 |
| Role Maintenance Document | 93 |
| KIM Type | 99 |
| KIM Type Lookup | 99 |
| KIM Type Inquiry | 101 |

| | |
|---|-----|
| Responsibility | 101 |
| Responsibility Lookup | 101 |
| Responsibility Inquiry | 103 |
| Permission | 104 |
| Permission Lookup | 104 |
| Permission Inquiry | 105 |
| Permission Template Inquiry | 106 |
| Delivered Permission Templates | 106 |
| 6. KNS | 109 |
| KNS Overview | 109 |
| What is KNS? | 109 |
| KNS Conceptual View | 109 |
| KNS Relational View | 110 |
| Parameter | 110 |
| Parameter Lookup | 110 |
| Parameter Inquiry | 112 |
| Parameter Maintenance | 112 |
| Process to create a new Parameter | 115 |
| Parameter Detail | 115 |
| Parameter Component Lookup | 115 |
| Parameter Component Inquiry | 117 |
| Parameter Namespace | 117 |
| Namespace Lookup | 117 |
| Namespace Inquiry | 118 |
| Parameter Type | 119 |
| Parameter Type Lookup | 119 |
| Parameter Type Inquiry | 120 |
| Parameter Type Maintenance | 120 |
| Pessimistic Lock | 122 |
| Pessimistic Lock Lookup | 122 |
| 7. KRAD | 123 |
| KRAD Architecture Overview | 123 |
| What is KRAD? | 123 |
| KRAD Conceptual View | 124 |
| KRAD Relational View | 125 |
| KRAD's User Interface Framework | 125 |
| UIF Overview | 125 |
| KRAD Design Components | 127 |
| Overview of Design Components built into KRAD | 127 |
| KRAD Layout Managers | 128 |
| KRAD Fields | 132 |
| Some key KRAD extensions beyond KNS concepts | 133 |
| 8. KRMS | 135 |
| Kuali Rule Management System: Overview | 135 |
| What is KRMS? | 135 |
| Example Rule | 135 |
| KRMS Terms and Concepts | 135 |
| When to use KRMS? | 136 |
| The KRMS User Interface | 136 |
| KRMS Agenda Editor | 136 |
| KRMS Rule Editor | 137 |
| 9. KSB | 140 |
| What is the Kuali Service Bus? | 140 |
| Features | 140 |

| | |
|---|-----|
| Bean-Based Services | 141 |
| Overview of Supported Service Protocols | 142 |
| Message Queue | 142 |
| Current Node Info | 143 |
| Message Filter and Fetch | 143 |
| Documents Currently in Route Queue | 145 |
| View | 145 |
| Thread Pool | 147 |
| Service Registry | 148 |
| Quartz | 149 |
| Security Management | 150 |
| 10. KEW Administration Guide | 152 |
| Kuali Enterprise Workflow: Overview | 152 |
| Document Operation | 152 |
| Document Actions | 153 |
| Document | 155 |
| Action Requests | 157 |
| Actions Taken | 159 |
| Action Items | 160 |
| Route Node Instances | 161 |
| Branch States | 162 |
| Annotation | 163 |
| Practical Uses of Document Operation | 163 |
| 11. Location | 164 |
| Location Overview | 164 |
| Campus | 164 |
| Postal Code | 171 |
| County | 174 |
| State | 177 |
| Country | 180 |
| Glossary | 183 |

List of Figures

- 1.1. Ad Hoc Recipients Tab 7
- 1.2. Ad Hoc Recipients Tab: Person Requests 7
- 1.3. Ad Hoc Recipients Tab: Group Requests Section 8
- 1.4. Document Overview Tab 8
- 1.5. Notes and Attachments Tab 8
- 1.6. Route Log Tab 9
- 1.7. Route Log Tab: ID Section 9
- 1.8. Route Log Tab: Future Action Requests Section 10
- 1.9. Jira Search: Query options 11
- 1.10. Create New Jira: Initial Screen 12
- 1.11. Create New Jira: Detail Section 14
- 3.1. Notification Channel: Channel Subscriptions 21
- 3.2. Chanel Subscriptions: Manage 21
- 4.1. PeopleFlow selection on the Main tab in the Kuali Portal 25
- 4.2. PeopleFlow Lookup 26
- 4.3. PeopleFlow - Create New 27
- 4.4. PeopleFlow - Create New (with additional type attributes) 28
- 4.5. PeopleFlow - Create New - with 2 stops added 29
- 4.6. Kuali Portal: Action List Button 30
- 4.7. Action List 32
- 4.8. Route Log 33
- 4.9. Action List: Preferences Button 34
- 4.10. Action List Preference Page 34
- 4.11. Action List Preferences: General section 35
- 4.12. Action List Preferences: Fields Displayed section 35
- 4.13. Action List Preferences: Document Route Status Colors Section 36
- 4.14. Action List Preferences: Document Route Status Colors Example 36
- 4.15. Action List Preferences: Document Route Status Colors Action List Example 37
- 4.16. Action List Preferences: Email Notifications Preferences 37
- 4.17. Action List Preferences: Document Type Notifications 38
- 4.18. Action List Preferences: Action Requested Email Notification 38
- 4.19. Action List Filter Page 39
- 4.20. Document Type Lookup 41
- 4.21. Date Widget 42
- 4.22. Action List: Clear Filter Button 42
- 4.23. Delegation Tree Example 44
- 4.24. Delegation Tree Example: Deactivation 46
- 4.25. New Superuser Tab Example 51
- 4.26. Workflow Channel: eDocLite Link 52
- 4.27. eDocLite Lookup 52
- 4.28. eDocLite Lookup: Search Results 53
- 4.29. eDocLite Inquiry 54
- 4.30. Workflow Channel: User Preferences Link 56
- 4.31. Workflow Preferences 57
- 4.32. Workflow Channel: Routing Rules Delegation Link 59
- 4.33. Delegation Lookup 59
- 4.34. Delegation Lookup: Results Example 60
- 4.35. Delegation Inquiry 60
- 4.36. Delegation Rule: Create New Screen 61
- 4.37. Delegate Routing Rule: Create New, Parent Selection 61
- 4.38. Routing Rule Delegation: Overview 62

| | |
|---|----|
| 4.39. Routing Rule Deletion: Details Section | 63 |
| 4.40. Routing Rules Delegation: Edit/Copy View | 63 |
| 4.41. Routing Rules Delegation: Delegate Rule Tab | 64 |
| 4.42. Routing Rule Delegation: Delegate Rule Tab, Edit/Copy View | 64 |
| 4.43. Routing Rules Delegation: Persons Tab | 65 |
| 4.44. Routing Rules Delegation: Persons tab, Copy/Edit View | 65 |
| 4.45. Routing Rules Delegation: Groups Tab | 66 |
| 4.46. Routing Rules Delegation: Groups Tab, Edit/Copy View | 66 |
| 4.47. Workflow Channel: Routing Rules Link | 67 |
| 4.48. Routing Rules Lookup | 67 |
| 4.49. Routing Rules Lookup: Results Example | 68 |
| 4.50. Routing Rules Inquiry | 68 |
| 4.51. Routing Rules Creation | 69 |
| 4.52. Workflow Channel: Routing report | 69 |
| 4.53. Routing Report: Template Selection | 70 |
| 4.54. Routing Report: Template Selection, Detail | 70 |
| 4.55. Routing Report: Routing Data Entry | 70 |
| 4.56. Routing Report: Route Log View | 70 |
| 4.57. Routing Report: Route Log View, Pending Action Requests | 71 |
| 4.58. Rule Maintenance Document Type Document | 71 |
| 4.59. Rule Maintenance Document Type Document: Rule Tab | 72 |
| 4.60. Rule Maintenance Document Type Document: Rule Tab, Edit/Copy View | 72 |
| 4.61. Rule Maintenance Document Type Document: Person Tab | 73 |
| 4.62. Rule Maintenance Document Type Document: Person Tab, Edit/Copy View | 73 |
| 4.63. Rule Maintenance Document Type Document: Groups Tab | 74 |
| 4.64. Rule Maintenance Document Type Document: Groups Tab, Edit/Copy View | 74 |
| 4.65. Rule Template Lookup | 75 |
| 4.66. Rule Template Lookup: Results Example | 75 |
| 4.67. Rule Template Inquiry | 75 |
| 5.1. KIM Architecture | 76 |
| 5.2. Detailed KIM Achitecture | 77 |
| 5.3. Person Lookup | 78 |
| 5.4. Person Lookup: Results | 79 |
| 5.5. Person Document | 79 |
| 5.6. Person Lookup: Create New Button | 80 |
| 5.7. Person Document | 81 |
| 5.8. Person Document: Overview Section | 82 |
| 5.9. Person Document: Overview Tab, Affiliations Section | 82 |
| 5.10. Person Document: Contact Tab | 83 |
| 5.11. Person Document: Contact Tab, Names Section | 83 |
| 5.12. Person Document: Contact Tab, Addresses Section | 84 |
| 5.13. Person Document: Contact Tab, Phone Numbers Section | 85 |
| 5.14. Person Document: Contact Tab, Email Addresses Section | 85 |
| 5.15. Person Document: Privacy Preferences Tab | 85 |
| 5.16. Person Document: Memberships Tab | 86 |
| 5.17. Identity Channel: Group Link | 87 |
| 5.18. Group Lookup | 88 |
| 5.19. Group Lookup: Results | 88 |
| 5.20. Group Inquiry | 88 |
| 5.21. KIM Type Lookup | 89 |
| 5.22. Group Maintenance Document | 90 |
| 5.23. Group Maintenance Document: Group Overview | 91 |
| 5.24. Group Maintenance Document: Assignees Tab | 91 |
| 5.25. Role Lookup | 92 |

| | |
|---|-----|
| 5.26. Role Maintenance Document | 93 |
| 5.27. Role Maintenance Document: Tabs | 94 |
| 5.28. Role Maintenance Document: Overview Tab | 94 |
| 5.29. KIM Type Lookup | 95 |
| 5.30. Role Maintenance Document: Permissions Tab | 95 |
| 5.31. Role Maintenance Document: Permissions Tab, Add Permissions | 96 |
| 5.32. Role Maintenance Document: Responsibilities Tab | 96 |
| 5.33. Role Maintenance Document: Responsibility, Added Responsibility | 97 |
| 5.34. Role Maintenance Document: Responsibility Tab, Action Section | 97 |
| 5.35. Role Maintenance Document: Assignees Tab | 98 |
| 5.36. Role Maintenance Document: Delegations Tab | 99 |
| 5.37. KIM Type Lookup | 100 |
| 5.38. KIM Type Lookup: Results Example | 100 |
| 5.39. KIM Type Inquiry | 101 |
| 5.40. Identity Channel: Responsibility Link | 102 |
| 5.41. Responsibility Lookup | 102 |
| 5.42. Responsibility Look: Results | 103 |
| 5.43. Responsibility Inquiry | 104 |
| 5.44. Permission Lookup | 104 |
| 5.45. Permission Lookup: Results Example | 105 |
| 5.46. Permission Inquiry | 106 |
| 5.47. Permission Template Inquiry | 106 |
| 6.1. KNS: Conceptual View | 109 |
| 6.2. KNS: Relational View | 110 |
| 6.3. Parameter Lookup Link | 111 |
| 6.4. Parameter Lookup | 111 |
| 6.5. Parameter Lookup: Results Set | 112 |
| 6.6. Parameter Inquiry | 112 |
| 6.7. Configuration Channel: Parameter Link | 113 |
| 6.8. Parameter Maintenance Document | 113 |
| 6.9. Parameter Maintenance Document: Document Overview Tab | 114 |
| 6.10. Parameter Maintenance Document: Edit Parameter Tab | 114 |
| 6.11. Parameter Maintenance Document: Routed Summary | 115 |
| 6.12. KNS Maintenance Documents: Parameter Component Lookup Link | 116 |
| 6.13. Parameter Component Lookup | 116 |
| 6.14. Parameter Component Lookup: Results Set | 116 |
| 6.15. Parameter Component Inquiry | 117 |
| 6.16. Namespace Lookup | 117 |
| 6.17. Namespace Lookup: Results Set | 118 |
| 6.18. Namespace Inquiry | 118 |
| 6.19. KNS Maintenance Documents Channel: Parameter Type Lookup Link | 119 |
| 6.20. Parameter Type Lookup | 119 |
| 6.21. Parameter Type Lookup: Results Set | 119 |
| 6.22. Parameter Type Inquiry | 120 |
| 6.23. Configuration Channel: Parameter Type Link | 120 |
| 6.24. Parameter Type Maintenance Document | 121 |
| 6.25. Parameter Type Maintenance Document: Edit Parameter Tab | 121 |
| 6.26. Pessimistic Lock Lookup | 122 |
| 7.1. KRAD Conceptual View | 124 |
| 7.2. KRAD: Relational View | 125 |
| 7.3. KRAD Conceptual Groupings | 127 |
| 7.4. Grid Layout | 129 |
| 7.5. Box Layout | 130 |
| 7.6. Table Layout | 131 |

| | |
|---|-----|
| 7.7. Stacked layout | 132 |
| 7.8. KRAD Input field - grouped | 133 |
| 8.1. KRMS Agenda Editor | 137 |
| 8.2. KRMS Rules Editor | 138 |
| 8.3. KRMS proposition and PeopleFlow Action | 139 |
| 9.1. Kuali Service Bus | 140 |
| 9.2. Supported Service Protocols | 142 |
| 9.3. Message Queue: Documents Currently In Route | 143 |
| 9.4. Message Filter Screen | 144 |
| 9.5. Execute Message Filter: Confirmation Screen | 144 |
| 9.6. Documents In Route Queue | 145 |
| 9.7. Requeue Documents: Confirmation Screen | 147 |
| 9.8. Thread Pool Administration Page | 148 |
| 9.9. Service Registry | 149 |
| 9.10. Service Reigstery Results | 149 |
| 9.11. Exception Routing Queue | 150 |
| 9.12. Create Keystore | 151 |
| 9.13. Create Keystore: File Section | 151 |
| 9.14. Create Keystore: Existing Keystore Section | 151 |
| 10.1. Initial Screen | 152 |
| 10.2. Document Actions | 153 |
| 10.3. Document | 155 |
| 10.4. Action Requests | 157 |
| 10.5. Actions Taken | 159 |
| 10.6. Action Items | 160 |
| 10.7. Route Node Instances | 161 |
| 10.8. Branch States | 162 |
| 10.9. Annotation | 163 |
| 11.1. Identity Channel: Campus Link | 164 |
| 11.2. Campus Lookup | 164 |
| 11.3. Campus Lookup: Results Example | 165 |
| 11.4. Campus Inquiry | 165 |
| 11.5. Campue Maintenance Document | 166 |
| 11.6. Campus Maintenane Document: Expanded | 167 |
| 11.7. Campus Maintenance Document: Edit Campus Tab | 167 |
| 11.8. Campus Type Lookup | 168 |
| 11.9. Campue Type Lookup: Results Example | 168 |
| 11.10. Campus Type Inquiry | 169 |
| 11.11. Identity Channel: Campus Type Link | 169 |
| 11.12. Campus Type Lookup | 170 |
| 11.13. Campus Maintenance Document | 170 |
| 11.14. Campus Maintenance Document: Overview Tab | 171 |
| 11.15. Campus Maintenance Document: Edit Campus Type Tab | 171 |
| 11.16. Identity Channel: Postal Code Link | 172 |
| 11.17. Postal Code Lookup | 172 |
| 11.18. Postal Code Lookup: Results Example | 172 |
| 11.19. Postal Code Inquiry | 173 |
| 11.20. Postal Code Manintenance Document | 173 |
| 11.21. Postal Code Manintenance Document: Edit Postal Codes Tab | 174 |
| 11.22. Postal Code Manintenance Document: Create Postal Codes | 174 |
| 11.23. Identity Channel: County Lin | 175 |
| 11.24. County Code Lookup | 175 |
| 11.25. County Code Lookup: Results Example | 175 |
| 11.26. County Inquiry | 176 |

| | |
|---|-----|
| 11.27. County Maintenance Document | 176 |
| 11.28. County Maintenance Document: Edit Counties Tab | 177 |
| 11.29. County Maintenance Document: Create County | 177 |
| 11.30. Identity Channel: State Link | 178 |
| 11.31. State Lookup | 178 |
| 11.32. State Lookup: Results Example | 178 |
| 11.33. State Inquiry | 179 |
| 11.34. State Maintenance Document | 179 |
| 11.35. State Maintenance Document: Add State | 180 |
| 11.36. State Maintenance Document: Edit States Tab | 180 |
| 11.37. Country Lookup | 180 |
| 11.38. Country Lookup: Results Example | 181 |
| 11.39. Country Inquiry | 181 |
| 11.40. Country Maintenance Document | 181 |
| 11.41. Country Maintenance Document: Edit Country Tab | 182 |

List of Tables

| | |
|--|-----|
| 1.1. Standard Rice Buttons | 3 |
| 1.2. Lookup Wildcards | 6 |
| 1.3. Account Number (String) Example of Wildcard Use | 6 |
| 1.4. Proposal Number (String) Example of Wildcard Use | 6 |
| 1.5. Create Date (Date) Example of Wildcard Use | 7 |
| 1.6. Ad Hoc Recipients: Person Requests attributes | 7 |
| 1.7. Ad Hoc Recipients: Group Requests attributes | 8 |
| 1.8. Document Overview Tab: Attributes | 8 |
| 1.9. Notes and Attachments Tab: Attributes | 9 |
| 1.10. Route Log Tab: ID Section Attributes | 9 |
| 3.1. Notification: Priority Attributes | 22 |
| 4.1. Types of Action Requests | 43 |
| 4.2. eDocLite Lookup Attributes | 52 |
| 4.3. Document Header Attributes | 54 |
| 4.4. Document Body Attributes | 55 |
| 4.5. Routing Action and Annotation, and Note Attributes | 55 |
| 4.6. User Preferences Attributes | 57 |
| 4.7. User Preferences: Fields Displayed Attributes | 58 |
| 4.8. Routing Rules Delegation Attributes | 63 |
| 4.9. Routing Rule Delegation: Delegate Rule Tab Attributes | 64 |
| 4.10. Routing Rules Delegation: Persons Tab Attributes | 65 |
| 4.11. Routing Rules Delegation: Groups Tab Attributes | 66 |
| 4.12. Routing Rule Creation Attributes | 69 |
| 4.13. Rule Maintenance Document Type Document: Rule Tab Attributes | 72 |
| 4.14. Rule Maintenance Document Type Document: Person Tab Attributes | 73 |
| 4.15. Rule Maintenance Document Type Document: Groups Tab Attributes | 74 |
| 4.16. Rule Template Lookup Attributes | 75 |
| 5.1. Person Document: Overview Attributes | 82 |
| 5.2. Person Document: Overview Attributes, Affiliations | 82 |
| 5.3. Person Document: Overview Attributes, Affiliations Continued | 83 |
| 5.4. Person Document: Contact Tab, Names Section Attributes | 84 |
| 5.5. Person Document: Contact Tab, Address Section Attributes | 84 |
| 5.6. Person Document: Contact Tab, Phone Numbers Attributes | 85 |
| 5.7. Person Document: Contact Tab, Email Address Attributes | 85 |
| 5.8. Person Document: Privacy Preferences Tab Attributes | 86 |
| 5.9. Person Document: Mememberships Tab, Groups Attributes | 86 |
| 5.10. Person Document: Mememberships Tab, Roles Attributes | 86 |
| 5.11. Group Inquiry: Assignees Attributes | 89 |
| 5.12. KIM Type Lookup Search Attributes | 89 |
| 5.13. Group Maintenance Document: Group Overview Attributes | 91 |
| 5.14. Group Maintenance Document: Assignees Tab Attributes | 91 |
| 5.15. Role Maintenance Document: Overview Attributes | 94 |
| 5.16. Role Maintenance Document: Permissions Attributes | 95 |
| 5.17. Role Maintenance Document: Permissions Tab, Add Attributes | 96 |
| 5.18. Role Maintenance Document: Responsibility Attributes | 96 |
| 5.19. Role Maintenance Document: Responsibility, Add Attributes | 97 |
| 5.20. Role Maintenance Document: Responsibility Tab, Action Section Attributes | 98 |
| 5.21. Role Maintenance Document: Assignees Tab Attributes | 98 |
| 5.22. Role Maintenance Document: Delegations Tab Attributes | 99 |
| 5.23. KIM Type Lookup Attributes | 100 |
| 5.24. KIM Type Inquiry Attributes | 101 |

| | |
|--|-----|
| 5.25. Responsibility Lookup Attributes | 102 |
| 5.26. Responsibility Lookup: Results Attributes | 103 |
| 5.27. Permission Lookup Attributes | 105 |
| 5.28. Permission Lookup: Results Attributes | 105 |
| 5.29. Delivered Permission Templates | 106 |
| 6.1. Parameter Maintenance Document: attributes | 114 |
| 6.2. Document Overview Tab: Attributes | 114 |
| 6.3. Edit Parameter Tab: Attributes | 114 |
| 6.4. Parameter Component Lookup: Results set Attributes | 116 |
| 6.5. Parameter Component Inquiry: Attributes | 117 |
| 6.6. Edit Parameter Type Tab: Attributes | 121 |
| 6.7. Pessimistic Lock Lookup: Search Attributes | 122 |
| 9.1. Message Filter Screen: Attributes | 144 |
| 9.2. Documents Currently in Route Queue: Attributes | 145 |
| 9.3. Message: Attributes | 146 |
| 9.4. Payload: Attributes | 146 |
| 9.5. Edit Screen: Attributes | 146 |
| 9.6. Edit Screen: Links | 147 |
| 9.7. Thread Pool: Attributes | 148 |
| 9.8. Exception Routing Queue: Attributes | 150 |
| 9.9. Existing Keystore Entries: Attributes | 151 |
| 11.1. Campus Maintenance Document: Edit Campus Attributes | 168 |
| 11.2. Campus Maintenance Document: Edit Campus Type Attributes | 171 |
| 11.3. Postal Code Maintenance Document: Create Postal Codes Attributes | 174 |
| 11.4. County Maintenance Document: Create County Attributes | 177 |
| 11.5. States Maintenance Document: Edit States Attributes | 180 |
| 11.6. Country Maintenance Document: Edit Country Attributes | 182 |

Chapter 1. Global

The Kuali Community

It is important that you have a general understanding of how Rice is a part of a larger Kuali community before you begin to use this as a reference.

Kualifoundation

The Kualifoundation employs staff to coordinate partner efforts and to manage and protect the Foundation's intellectual property. The Kualifoundation manages a growing portfolio of enterprise software applications for colleges and universities. A lightweight Foundation staff coordinates the activities of Foundation members for critical software development and coordination activities such as source code control, release engineering, packaging, documentation, project management, software testing and quality assurance, conference planning, and educating and assisting members of the Kualifoundation Partners program.

Kualifoundation Rice

Kualifoundation Rice is a higher-education, community-source software project that is merely a spoke in the wheel of the larger Kualifoundation Community. The Kualifoundation Community is the hub of a wheel of a growing number of communities, each of which are made up of people and functions. Together they form a comprehensive suite of open, modular and distributed administrative software systems that bring the proven functionality of legacy applications to the ease and universality of online services. The Kualifoundation serves them all, with specific responsibilities related to keeping the wheel in motion. Each individual community shares a similar organizational structure and some modular functionality while its components are able to stand alone to perform unique functions. While all are designed for seamless integration with each other, each project is made up of modules that offer a variety of implementation combinations to suit any Carnegie-class institution's unique business needs.

Should you want to learn more detail about how the Rice community is organized, structured and operates you should reference the Rice Charter which is available for review in the Foundation archives.

Licensing and Intellectual Property

Software License

Kualifoundation Rice software is licensed under the Educational Community License, Version 2.0. You may obtain a copy of the License [here](#). See the License for the specific language governing permissions and limitations.

If you have any questions about Kualifoundation Intellectual property, please contact the Kualifoundation Foundation at licensing@kuali.org.

Contributors

TODO: Link to Third Party Licenses Page

Documentation License

This work is licensed under a Creative Commons Attribution-ShareAlike 2.5 License. To view a copy of this license, visit <http://creativecommons.org/licenses/> or send a letter to Creative Commons, 543 Howard Street, 5th Floor, San Francisco, California, 94105, USA.

You are free:

- to copy, distribute, display, and perform the work
- to make derivative works
- to make commercial use of the work under the following conditions:
 - Attribution: You must attribute the work in the manner specified by the Quali Foundation, author and licensor.
 - For any reuse or distribution, you must make the licensing terms of this work clear to others.
 - Any of these conditions can be waived if you get permission from the copyright holder.

Your fair use and other rights are in no way affected by the aforementioned conditions.

How to Use This Guide

The table of contents leads you to headings and subtopic headings that provide information ranging from simple notes to a thorough review of what to do and why, with frequent examples. Sequential tasks are numbered, notes and action results are indented, and user interface element references are formatted to enhance readability. Mouse pointer icons and callouts are used frequently to show you what to click on at each process step.

User Guide Structure

Each chapter is comprised sections that correspond to modules within Rice. In order to reduce the required effort for documentation maintenance, the documentation is structured so as to not repeat the same material across various chapters.

Likewise, the chapters are designed to cross-reference appendices such as the Data Element Dictionary, Glossary of Terms, Report Catalog and Index, to reduce redundant information within the chapter bodies and give you one common location for information lookup.

Typographic Conventions

This document adheres to specific documentation standards and style conventions to optimize readability. The formatting of text used to name user interface elements is typically bold to enhance visual comprehension and improve usability.

Versioning

If you have received a printed copy of this user guide, please note that as Rice is improved and enhanced, corresponding updated releases of the documentation are available online. To ensure you have the latest

and most current user guide at any time, consult the links from the Documentation page on the [Rice web site](#) and compare the versioning.

Electronic Navigation

Cross-references, indexes, and the electronic table of contents are provided wherever possible to allow for efficient navigation to sections and subsections of instructional information (or external Web resources). Each underlined topic functions as a hyperlink that causes a new section to appear.

Printed Documentation

User documentation made available for online viewing, download and printing may require the use of Portable Document Format (PDF) reader software, such as Adobe Acrobat Reader. Refer to <http://www.adobe.com> for complete instructions regarding downloading and installing this free software.









Note














A plug-in may be available to allow you to view .pdf documents in Web browsers as either .pdf or HTML formats. Consult your Web browser's Help links to explore these viewing options. HTML pages can be printed using your computer's default printer.

Common Features and Functions

Standard Buttons on Lookup Screens

Table 1.1. Standard Rice Buttons

| Feature | Button | Description |
|----------------------------|--|---|
| Action List Button |  | Displays the Action List screen |
| Add Button |  | Use to add notes or attachments |
| Calendar Button |  | This button displays a pop-up calendar from which you can select a specific date. The date you select is then automatically entered in the field next to the Calendar button. |
| Cancel Button |  | Returns you to the main menu screen |
| Clear Button |  | Clears the content in all the fields on this screen |
| Clear Save Searches Button |  | Clears all saved searches. (If you give a search a name, you can save it permanently. This button clears all of those saved searches.) |
| Close Button |  | Closes a pop-up screen and returns you to the previous screen |
| Create New Button |  | Displays a screen where you can create a new document |
| Collapse All Button |  | Collapses an expanded list (in other words, it hides the detailed information below the main items in a list); use the Expand All button to display the entire list again. |
| Detailed Search Button |  | Opens the Detailed Search screen. This lets you do searches for documents using more specific information. |

| Feature | Button | Description |
|-------------------------|---|---|
| Doc Search Button |  | Displays the Document Search screen |
| Dropdown Arrow |  | Click the down-arrow on the right end of the box to display a list of options. Click an option in this list to automatically put that item in the field. |
| Error Mark |  | Symbol displayed by Rice when it finds an error on a required screen or field |
| Expand All Button |  | Expands a list (in other words, it shows the detailed information below each item in a list); use the Collapse All button to hide the detailed information again. |
| Field Lookup Button |  | Some fields have a magnifying glass button so you can search for information to put in that field. Click the magnifying glass icon to go to the Lookup screen where you can do a search for that field. |
| Help Button |  | When you click a Help button, a small window appears with information about the field that was next to the Help button. |
| Hide Button |  | Hides the contents of a tab, so you only see the small tab itself; use the Show button to display the contents again. |
| Inquiry Button |  | Find the Inquiry button next to some dropdown lists. Click it to display more information about the item in the dropdown field. (Rice displays the information in an Inquiry screen.) |
| Preferences Button |  | Allows you to customize the appearance and features of your screen |
| Search Button |  | Begins a Search using the information you entered in the search fields |
| Show Button |  | Shows the contents of a tab; use the Hide button to hide the contents again. |
| Submit Button |  | Click this button to send this screen's information to Rice for processing |
| Superuser Search Button |  | Click this button to search and then use Superuser functions with the search results. |

Useful Screens Are Where You Need Them

One of the beauties of Rice is that it shows you just the screens and fields that are useful for your Rice Role and the function you need. For example, if your Role in Rice lets you create new Rice documents, then you can see a create new button on appropriate screens. If you click this button, you go directly to the screen for creating new documents. People whose Role doesn't create new documents aren't bothered with this button on their screens.

Screens with information or functions that can be used in many places in Rice are very conveniently linked to those places, so useful screens are always at your fingertips. For example, screens that use Person information display a link to the Person Lookup screen. This lets you search for the exact Person you need and enter that Person's information automatically on those screens. (In Rice, a Person is a set of information about a real person or something that stands for real people, like a job title.)

Not only will you find links to a useful screen like Person Lookup on many other screens, but what you see on that useful screen may depend on how you get to it. For example, if you go to the Person Lookup screen from an Administration menu and search for a Person, the search results list has Actions in the left column, and the link for each Person in the Actions column takes you to the edit screen for that Person.

On the other hand, if you go to a Person Lookup screen from other locations, the left column of your search results list has a return value link. You can click this link on the row for the Person you need to close the

Person Lookup screen and automatically enter the information for that Person on the original screen (the screen where you clicked the magnifying glass icon).

Common Functions

Entering Search Information

- Dates should be entered as mm/dd/yyyy.
- Wildcards in searches: You may use the asterisk (*) and the percent symbol (%) as wildcards in searches. If you need more information about using wildcards in searches, do a search in your favorite Internet search engine to find one of many explanations of wildcards.
- Range operators allowed on numbers and dates in searches: >, <, >=, <=, or '..'. All operators except '..' should be before a date. Use operators to separate dates. If you need more information about using range operators in searches, do a search in your favorite Internet search engine to find one of many explanations of range operators in searches.

Search Results

- Each column in your list of search results has a link on the header (the title of the column) for sorting. Click the column title once to sort the search results list in ascending order by that column and click again to sort it in descending order.
- Some fields have links to the Inquiry screen for that field. If you click the link, the inquiry appears in a new window with information about that field.
- Click the return value link on a row to enter information from that row in your previous page. Select return with no value or click the cancel button if you wish to go back to your previous page without entering anything from your search list.

Maintenance Links

- The create new link in the upper left corner of the lookup screen displays a maintenance screen where you can create a new record for this lookup type. For example, the create new button on a Person Lookup screen takes you to the screen to create a new Person in Rice.
- On each row in your search results list, the Actions column displays edit and copy links:
 - The edit link takes you to a maintenance screen for editing that record.
 - The copy link takes you to a new document screen with information from that item already entered in the fields for the new document.

Exporting Your Data

Below the list of search results, there are links for exporting the search results list to a different format. Click:

- **CSV** to export the data as a comma-delimited file
- **Spreadsheet** to export the data as a spreadsheet

- **Xml** to export the data as xml

Lookup Wildcards

Rice allows you to use wildcards in the Lookup process. The available wildcards and their functions:

Table 1.2. Lookup Wildcards

| Operator | Name | Compatible Data Types | Precedence | Notes |
|----------|--|-----------------------|------------|---|
| | Or | All | Always | |
| && | And | All | Always | |
| ! | Not Equal to | String | 1 | If used repeatedly, e.g. !1031490!1031491, an && is assumed, leading to !1031490&&!1031491 |
| ?, * | Like | String | 7 | ? will match any one character and * will match any number of characters. These will still be used if ! has been used, but not if any of the range criteria below have been used. |
| > | Greater Than | String, Number, Date | 3 | |
| < | Less Than | String, Number, Date | 4 | |
| >= | Greater Than or Equal To | String, Number, Date | 5 | |
| <= | Less Than or Equal To | String, Number, Date | 6 | |
| .. | Greater Than or Equal to and Less Than or Equal to | String, Number, Date | 2 | |

Examples of how some of the wildcards operate:

Table 1.3. Account Number (String) Example of Wildcard Use

| Wildcard | Action |
|---------------------------|--|
| >=1031490 | Accounts with an account number greater than or equal to 1031490 |
| >1031490&&<1111500 | Accounts with an account number greater than 1031490 and less than 1111500 |
| >=1031490&&<=1111500 | Accounts with an account number greater or equal to 1031490 and less than or equal to 1111500 |
| 1031490..1111500 | Accounts with an account number greater than or equal to 1031490 and less than or equal to 1111500 |
| 103* | Accounts with an account number starting with 103 |
| 103?490 | Accounts with an account number 1031490, 1032490, 1033490... The ? will match any one character with 103 before it and 490 after it |
| 1031490..1111500&&1123400 | Accounts with an account number greater than or equal to 1031490 and less than or equal to 1111500 and accounts with account number like 1123400 |
| 103149* 105167* | Accounts with an account number that starts with 103149 or 105167 |
| 103149 105167 | Accounts with an account number that starts with 103149 or 105167 |
| *1111* | Accounts with an account number with 1111 somewhere in it |
| !1031490&&!1031491 | Accounts except those with account numbers 1031490 and 1031491 |
| !1031490!1031491 | Accounts except those with account numbers 1031490 and 1031491 |

Table 1.4. Proposal Number (String) Example of Wildcard Use

| Wildcard | Action |
|--------------------|---|
| >1031490&&<1111500 | Proposals with a proposal number between 1031490 and 111500 |

| | |
|----------|--|
| Wildcard | Action |
| 103* | Will not parse as a number, so it will not be included in criteria |

Table 1.5. Create Date (Date) Example of Wildcard Use

| | |
|------------------------|--|
| Wildcard | Action |
| 10/07/1976..10/07/1983 | Accounts with a create date greater than or equal to 10/07/1976 and less than or equal to 10/07/1983 |
| 10* | Will not parse as a date, so will not be included in criteria |

Result Set Limits

Some lookups for certain business entities in Rice return different numbers of results than the system default; as a result, system administrators may need to be able to limit this set of returned results on an entity-by-entity basis.

Frequently Used Tabs

To make it easier to do your work, Rice often provides access to the same information from several screens. Some of the most frequently used are described here:

Ad Hoc Recipients Tab

The Ad Hoc Recipients tab allows you to interrupt the normal workflow routing of a document and include other individuals or groups in its routing path. Ad Hoc Routing does not replace the normal workflow routing of the document; it adds people or groups to the normal routing.

The Ad Hoc Recipients tab has two sections: Person Requests and Ad Hoc Group Requests. Use one or both of the sections to route the document to an additional person, group, or both.

Figure 1.1. Ad Hoc Recipients Tab

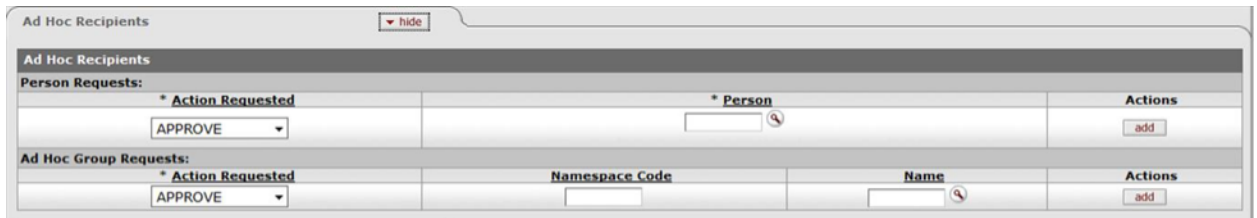


Figure 1.2. Ad Hoc Recipients Tab: Person Requests



Table 1.6. Ad Hoc Recipients: Person Requests attributes

| | |
|------------------|---|
| Field | |
| Action Requested | Required field. Select the desired action from the Action Requested list. The choices are APPROVE, ACKNOWLEDGE, and FYI. |
| Person | Required when you want to route the document to an individual. Enter a user ID or select it using the UserID lookup button. |

When you click the Add button, Rice adds this person to the routing path for this document.

Figure 1.3. Ad Hoc Recipients Tab: Group Requests Section

| Ad Hoc Group Requests: | | | |
|------------------------|----------------|------|---------|
| * Action Requested | Namespace Code | Name | Actions |
| APPROVE | | | add |

Follow the same guidelines as the Person Requests Section of the Ad Hoc Recipients Tab, using a group instead of a person.

Table 1.7. Ad Hoc Recipients: Group Requests attributes

| Field | Description |
|----------------|---|
| Namespace Code | Optional field. The Rice code for the Namespace being acted upon. |

Document Overview Tab

The Document Overview tab contains the short description for a document and two other optional fields. This tab is used for many documents.

Figure 1.4. Document Overview Tab

Table 1.8. Document Overview Tab: Attributes

| Field | Description |
|-------------|---|
| Description | Required field. Enter the short description for the document. The description appears in Inquiry screens, standard reports, Action Lists, and Document Searches as the primary identification for a document. |
| Org Doc # | Optional field. Enter the Org Doc # assigned by your institution. This number may provide departmental or organizational information for the document. This number is not the same as the Document Number, which is assigned by Rice. |
| Explanation | Optional field. Enter a more detailed explanation than the information supplied in the description field. |

Note and Attachments Tab

The Notes and Attachments tab has fields for user notes, attachments, and system-generated remarks about the document. The number of notes and/or attachments for the current document is displayed on the tab, in parentheses next to the tab title.

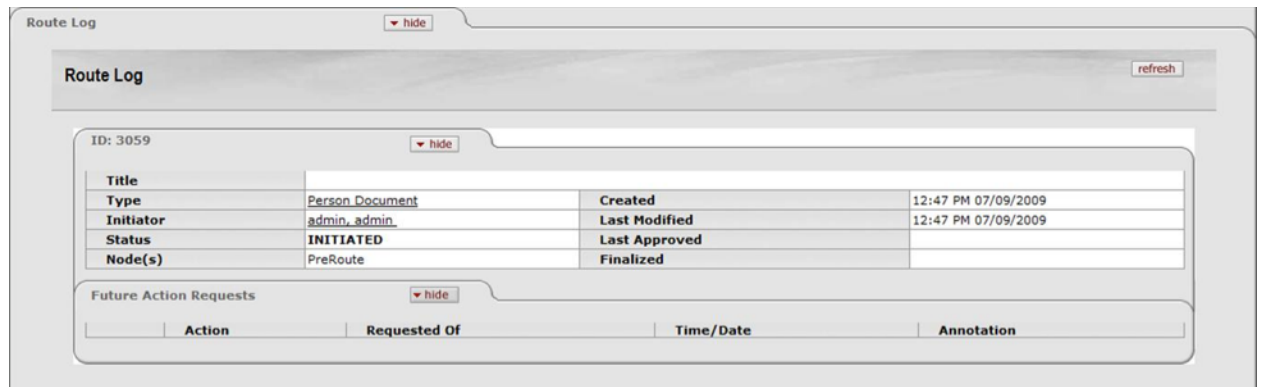
Figure 1.5. Notes and Attachments Tab

Table 1.9. Notes and Attachments Tab: Attributes

| Field | Description |
|------------------|---|
| Posted Timestamp | Display-only field. The time and date when the attachment or note was posted |
| Author | Display-only field. The full name of the user who added the notes or attachments |
| Note Text | Required field. Enter comments about the document. |
| Attached File | Optional field. Select the file to attach by clicking Browse and using Window's standard Choose File dialog box. Click CANCEL if you want to clear the file name that you select. |
| Actions | You have only one Action available if you are going to add a note or attachment, which is to Add it. Click the add button to add the note or attachment. |

Route Log Tab

The Route Log tab displays details of the workflow status.

Figure 1.6. Route Log Tab**Figure 1.7. Route Log Tab: ID Section**

The screenshot shows the 'Route Log' tab interface for ID: 2994. It includes a 'hide' button and a table of attributes:

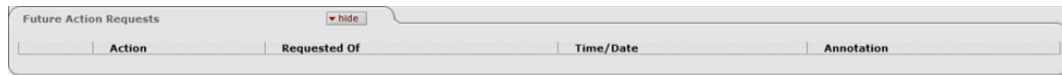
| Title | Type | Created | Last Modified |
|-------|-----------------|---------------------|---------------------|
| | Person Document | 07:45 AM 08/13/2009 | 07:45 AM 08/13/2009 |
| | admin, admin | | |
| | INITIATED | | |
| | PreRoute | | |

Table 1.10. Route Log Tab: ID Section Attributes

| Field | Description |
|-----------|---|
| Title | A combination of the Document Type, Description, and the Organization Document Number |
| Type | Type of transaction. The full name of the transaction, used to identify this Document Type in Workflow |
| Initiator | The User ID of the person who created the document |
| Status | Workflow document status |
| Node(s) | The steps that a document takes through the different levels of routing are also referred to as Route Nodes. This field shows the current Route Node of the document. |

| Field | Description |
|---------------|--|
| Created | Time and date that the document was created |
| Last Modified | Time and date that the document was last modified |
| Last Approved | Time and date that the last action was taken on this document |
| Finalized | Time and date that the document reaches Final, Canceled, or Disapproved status |

Figure 1.8. Route Log Tab: Future Action Requests Section



Once a document is Saved or ENROUTE, this section shows the action requests that Workflow will generate in the future, based on the information currently on the document.

Future requests appear in the order in which they are to occur.

Reporting an Issue with Kualu Rice

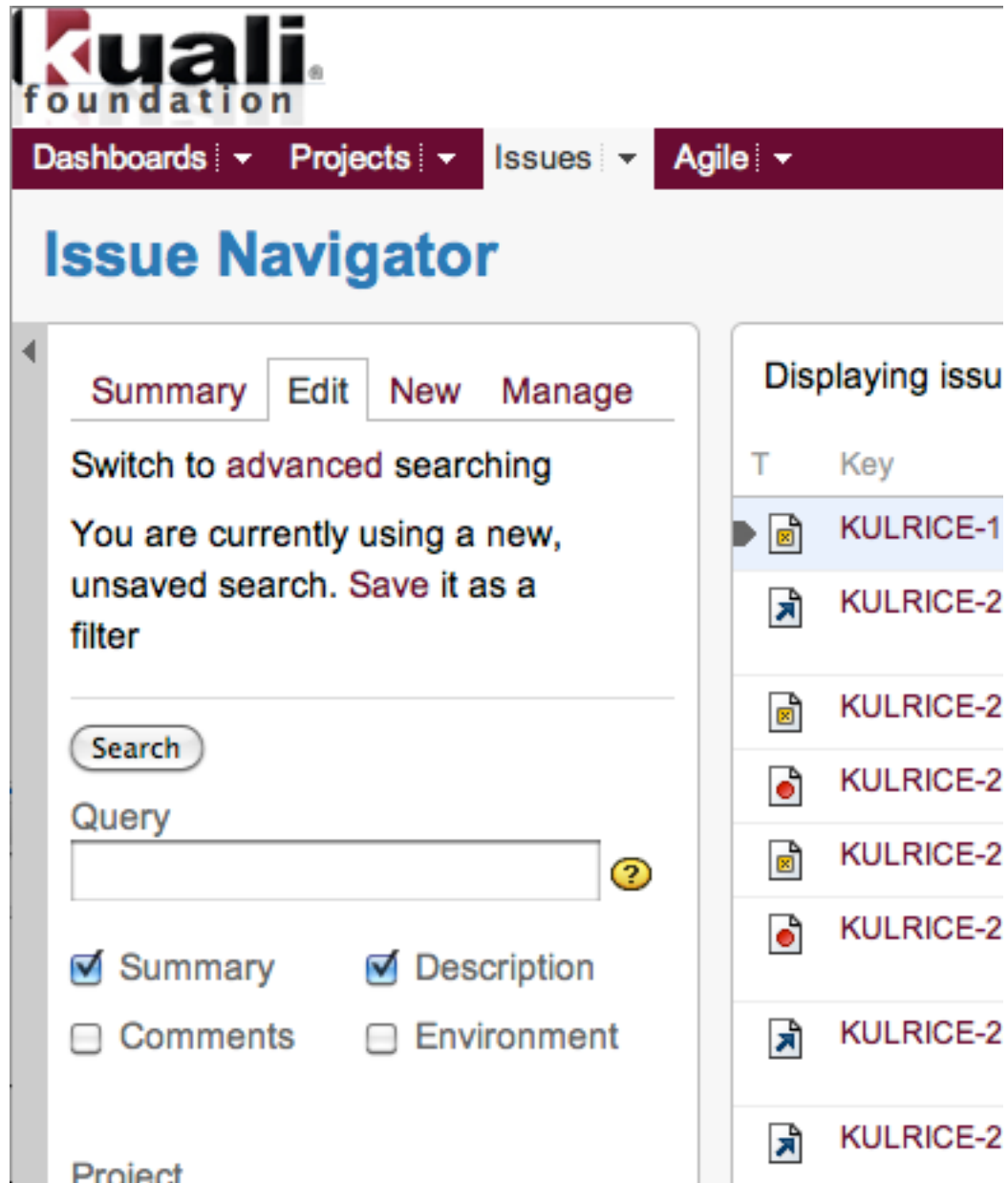
An issue is a problem or error that you see in Kualu Rice. When you report an issue, you help us improve Rice and make it work better for everyone.

Thank you for taking the time to report this issue!

Step 1: Search for a Similar Issue

Kualu Rice keeps an online list of issues (bugs and suggested improvements and new features) using an issue tracking application called Jira. Your first step is to find out if the issue you have has already been reported by someone.

1. Go to the Jira Issue Navigator page to search through the issues that have already been reported for Kualu Rice, looking for an issue like yours: <https://jira.kuali.org>.
2. Find the Text Search section at the top of the of the Issue Navigator page.

Figure 1.9. Jira Search: Query options

3. In the Query field of the Text Search section, enter some text that describes your issue. Try to use words that other people would also use to describe this issue.
4. After you enter some text in the Query field, press the Enter key to begin the search.
5. In a moment or two, Issue Navigator displays all the existing issues that contain your Query text. These are displayed in the main body of the page. Look through these to see if there is one that is similar to your issue.
6. If you don't find an issue similar to the one you want to report, try searching with different text in the query field.

7. If you do find an issue that is similar to the new one that you want to report, don't enter a new issue. Simply add a comment to that existing issue that describes any differences in your issue or adds information about the issue.
8. After trying a few searches, if you don't find an issue that is similar to the one you want to report, please proceed with the instructions below.

Step 2: Create an Issue in JIRA

If you didn't find a similar issue on the Issue Navigator page, then please create a new issue in JIRA. (If you are a developer, you also determined that this is an issue with Kuali Rice and not with your application.)

1. Go to Kuali JIRA (<https://jira.kuali.org/secure/CreateIssue!default.jspx>).
2. In the Project field, select Kuali Rice Development from the dropdown list.
3. Select the appropriate Issue Type from the dropdown list:
 - For bugs, select Bug Fix.
 - For enhancements, select either Improvement or, for something that's not yet developed: New Feature, as appropriate.
 - DO NOT select Task.
4. Click the Next>> button.

Figure 1.10. Create New Jira: Initial Screen

Step 3: Fill Out Required Fields

Note

JIRA only displays fields that are appropriate to the Issue Type that you selected on the previous page. You may not see all of the fields listed below, and the fields you see may be in a different order. Complete the fields from this list that you do see.

Only enter information in these fields:

- **Summary:** Enter a short, descriptive title for this issue.
- **Application Requirement:** Select the application for which this issue is a requirement.
- **Description:** Describe the issue in full. Provide enough information that the person who tries to fix this issue can find it easily and understand the bug or the improvement or feature you are suggesting.
- **Priority:** Select the correct priority:
 - **Blocker:** Blocks development and/or testing work, production cannot run.
 - **Critical:** Crashes, loss of data, severe memory leak.
 - **Major:** Major loss of function.
 - **Minor:** Minor loss of function or other problem where an easy workaround is available.
 - **Trivial:** Cosmetic problem like misspelled words or misaligned text.
- **Affects Version/s:** If this is a bug, select the version(s) of Rice in which it appears.
- **Environment:** Enter specific information about the operating system, software platform, and/or hardware as appropriate for this issue. Ignore all other fields. Do not change them. Other people use those fields. This screen print is from the Bug Fix screen. The screens for Improvement and for New Feature are slightly different. The fields in which you need to enter information are highlighted in this screen print.

Ignore all other fields. Do not change them. Other people use those fields.

This screen print is from the Bug Fix screen. The screens for Improvement and for New Feature are slightly different. The fields in which you need to enter information are highlighted in this screen print.

Figure 1.11. Create New Jira: Detail Section

kuali
foundation

Dashboards ▾ Projects ▾ Issues ▾ Agile ▾

Create Issue

Project **Kuali Rice Development**

Issue Type Bug Fix

Summary *

Component/s *
Start typing to get a list of possible matches or press down to select.

Security Level

Rice Module
Rice Core
KSB
KNS
KRAD
A select list of the Rice modules.

Application Requirement
KFS
KRA
KS
KRICE
Which application is requiring this?

Description

Start Date
Approximate Start Date

Priority *

Affects Version/s
Start typing to get a list of possible matches or press down to select.

Environment
For example operating system, software platform and/or hardware specifications (include as appropriate for t

Step 4: Save the Jira issue

After entering the information about the your issue, click the Create button at the bottom of the page. This creates your issue in JIRA.

Chapter 2. High-Level Features Guide

This document is planned as a high level overview of what the various modules of Rice have to offer. For a more in depth view please refer to the User's Guide or Technical Guide that we also provide. If you don't already have a Rice environment to use, please visit this site to download the latest version of Rice for yourself.

Nervous System (KNS)

The Kualu Nervous System (KNS) is a software development framework aimed at allowing developers to quickly build web-based business applications in an efficient and agile fashion. KNS is an abstracted layer of "glue" code that provides developers easy integration with the other Rice components. In this scope, KNS provides features to developers for dynamically generating user interfaces that allow end users to search, view details about records, interact electronically with business processes, and much more. KNS adds visual, functional, and architectural consistency to any system that is built with it, helping to ensure easier and more efficient maintainability of your software.

Service Bus (KSB)

Kualu Service Bus (KSB) is a simple service bus geared towards easy service integration in a SOA architecture. In a world of difficult to use service bus products KSB focuses on ease of use and integration.

Workflow (KEW)

Kualu Enterprise Workflow provides a common routing and approval engine that facilitates the automation of electronic processes across the enterprise. The workflow product was built by and for higher education, so it is particularly well suited to route mediated transactions across departmental boundaries. Workflow facilitates distribution of processes out into the organizations to eliminate paper processes and shadow feeder systems. In addition to facilitating routing and approval workflow can also automate process-to-process related flows. Each process instance is assigned a unique identifier that is global across the organization. Workflow keeps a permanent record of all processes and their participants for auditing purposes.

Important Features of KEW

- Flexiable Routing
 - Nodes
 - By Document
 - Actions
 - Delegation
- Action List - regardless of what application is using KEW, a single action list is used as a single point for a user to get access to any documents requiring their attention.
- Filters and Preferences - users level tools are provided to allow a user to quickly filter their action list or to setup presentation preferences for setting up a personalized feel to their list.

- eDocLite - eDocLite is a framework for developing and getting form based electronic documents, backed by a flexible workflow, out to users in a rapid fashion

Important Features of PeopleFlow

PeopleFlow is a new feature in Rice 2.0, that enables simple routing of actions and notifications based on business rules. PeopleFlow can be integrated with enterprise-wide workflow management in KEW, or can be used without KEW, with the Kuali Rules Management System (KRMS) engine that is also new in Rice 2.0. This offers a light-weight way to manage business rules and routing across applications.

Notification (KEN)

Kuali Enterprise Notification (KEN) acts as a broker for all university business related communications by allowing end-users and other systems to push informative messages to the campus community in a secure and consistent manner. All notifications are processed asynchronously and are delivered to a single list where other messages such as workflow related items (KEW action items) also reside. In addition, end-users can configure their profile to have certain types of messages delivered to other end points such as email, mobile phones, etc.

Important Features of KEN

- Channels - A channel is a stream used to organize notifications by topic or audience. These channels can be self-subscribed to by the user or pushed out automatically to users.
- Priority - each notification is given a priority of importance to determine how the notification is presented.

Identity Management (KIM)

Kuali Enterprise Notification (KEN) acts as a broker for all university business related communications by allowing end-users and other systems to push informative messages to the campus community in a secure and consistent manner. All notifications are processed asynchronously and are delivered to a single list where other messages such as workflow related items (KEW action items) also reside. In addition, end-users can configure their profile to have certain types of messages delivered to other end points such as email, mobile phones, etc.

Rules Management System (KRMS)

Kuali Rule Management System (KRMS) is a common rules engine for defining decision logic, commonly referred to as business rules. KRMS facilitates the creation and maintenance of rules outside of an application for rapid update and flexible implementation that can be shared across applications.

Important Features of KRMS

- Includes a repository for storing business rules in a database, accessible remotely via web services
- Provides user interface that sits on top of the rule repository and allows for authoring of business rules
- UI contains numerous plug points and ability to allow for custom attributes and data elements to be collected during rule authoring (i.e. associating organizational codes with business rules, etc.)

- An execution engine which can be embedded inside of the client application but which loads rules for execution from the remote rule repository
- The ability to plug in other sources for business rules into the KRMS execution model
- The business rule model supports the following concepts
 - Business Rule - decision logic that is used by an operational system, consists of a "condition" and an "action"
 - Condition - made of multiple propositions that evaluate to true or false, combined using Boolean algebra (such as AND, OR, NOT)
 - Proposition - a function that evaluates to true or false, potentially made up of term evaluations or custom functions
 - Term - defines a piece of business data that can be used in the construction of proposition (i.e. student GPA, account number, salary, etc.)
 - Action - is executed when a rule evaluates to true, can contain any arbitrary logic which (i.e. route an approval request, raise a validation message, send a notification, etc.)
 - Agenda - an execution plan for a set of rules
 - Context - contains multiple agendas and business rules, typically tied to some logical module or component of an application (i.e. a context containing agendas and business rules or Proposal Development in Quali Coeus)
- The execution engine is invoked by passing context and agenda selection criteria (which it will use to go to the remote rule repository and load the appropriate rules) as well as a set of "facts"
 - a fact is a value for an instance of a term (i.e. "student id" might be the term but "student id = 123456" is a fact)
- In general, the majority of KRMS is pluggable as well, there are features built in for what we are calling "term resolution" where certain terms can be derived based on values for other terms/facts that are supplied to the rule execution engine.
- Has built-in integration with KEW at the moment for doing routing rules these integrations though are built in using the standard "Action" capability that KRMS provides, so you can really integrate it with anything through the use of custom actions

Rapid Application Development (KRAD)

New for Rice 2.0, Quali Rapid Application Development (KRAD) is a framework that eases the development of enterprise web applications by providing reusable solutions and tooling that enables developers to build in a rapid and agile fashion. KRAD is a complete framework for web developers that provides infrastructure in all the major areas of an application (client, business, and data), and integrates with other modules of the Rice middleware project. In future releases, KNS will be absorbed into and replaced by KRAD.

Important Features of KRAD

- User Interface Framework (UIF) components
 - Built upon a rich JQuery library of standards that, among other items, includes

- Light-boxes
- Messages and Notifications
- Progressive Disclosure
- Client Side Validation
- Table Tools
- Themes
- AJAX Enabled Fields
- More UI Flexibility
 - No longer limited to the "vertical" tab layouts of KNS
- Improved Configuration and Tooling
- Inquiries
- Lookups
- Maintenance Documents
- Transactional
- Web MVC
- Rules
- Data Dictionary Enhancements
 - Min, Max, Valid Characters
 - Conditional Constraints
 - Custom Constraints
 - Lookup Constraints
- Business objects
- Improved Accessibility of Rice and its components

Chapter 3. KEN

Channels

Notification Channel

A notification channel is a communication stream used as a means to organize notifications by topic or audience. Users and groups can subscribe to a channel to receive the notifications that interest them.

You can think of a notification channel as being like a television channel. Producers put messages on a channel, like producers at television stations air programs. Subscribers receive the messages, like television viewers watch the programs. Unlike a television program, a message sent on a channel will wait for you to read it; you do not need to be actively reading the channel to avoid missing a message.

Each channel has the following attributes:

- **ID** – This numeric value is used in various tables to identify the channel. Each channel must have a unique ID, but there is no requirement for the IDs to be consecutive or to start with a particular value.
- **Name** – This is the text displayed to the user in the user interface. Each channel must have a unique name.
- **Description** – This is text which further describes the channel.
- **Subscription Flag** – This flag identifies whether users can adjust their subscription status on the channel. If “Yes”, users can see the channel in the “Channel Subscriptions” form and change their subscription status. If “No”, a user’s subscription to the channel is managed by other means (for example, group membership).
- **Producers** – This is a list of users who are authorized to send notifications on the channel.
- **Default Recipient List** – This is a list of users and groups who will receive notifications sent to the channel, regardless of channel subscriptions or notification-specific recipients.
- **Reviewers** – This is a list of users who are responsible for approving notifications sent to the channel.

Managing Channels

There is no user interface page to manage channels at this time.

Notification Channel Subscriptions

Subscribing to a notification channel causes all notifications sent on the channel to be delivered to the subscriber, regardless of the list of users or groups specified as recipients of the notification.

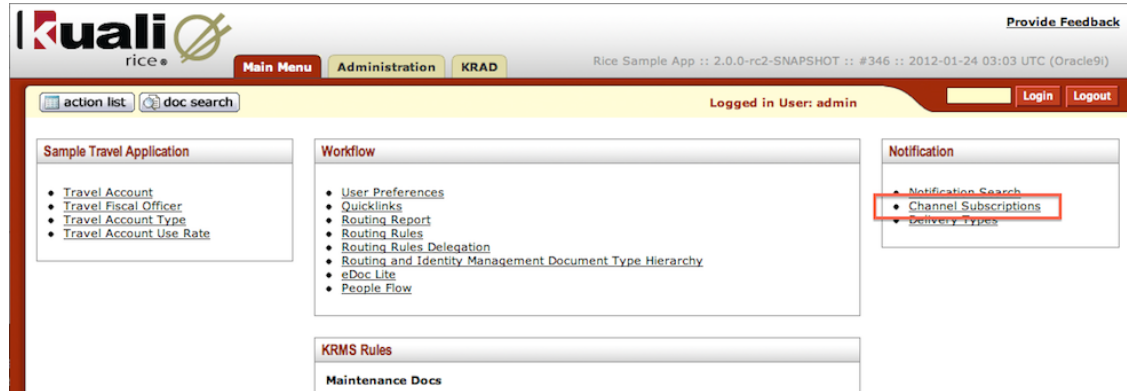
Subscriptions are divided into two categories:

- **Subscribers** – Users can subscribe to any channel where the Subscription Flag is set to “Yes”.
- **Default Recipients** – Each channel has a list of users and groups that receive all channel notifications.

Managing Subscribers

Users can manage their subscriptions to channels where the Subscription Flag is **Yes** using the Channel Subscriptions form. On the Main Menu, click the **Channel Subscriptions** link in the **Notification** box:

Figure 3.1. Notification Channel: Channel Subscriptions



This displays the list of channels that allow the user to manage their subscriptions:

Figure 3.2. Chanel Subscriptions: Manage



Click the **Subscribe** button for a channel to subscribe to a channel for which you are not already subscribed. Click the **Unsubscribe** button to cancel your subscription to the channel.

Notification Priority

The priority of a notification indicates its importance. It has no effect on how KEN processes the notification, but KEN can use it when delivering a message to determine how to present the notification.

Each priority has these attributes:

- **ID** – This numeric value defines the order that KEN displays the priorities in the user interface. The lowest ID is displayed at the top of the selection field and is the default value. The remaining priorities are listed in the selection field in ascending ID order. Each priority must have a unique ID, but there is no requirement for the IDs to be consecutive or to start with a particular value.
- **Name** – This is the text displayed to the user in the user interface. Each priority must have a unique name.
- **Description** – This text further describes the priority.
- **Order** – This numeric value determines the relative importance of the priority, with lower order numbers indicating a higher importance. Although not required, each priority should have a unique order value. There is no requirement for the order values to be consecutive or to start with a particular value.
- **Version** – This numeric value lets you perform optimistic locking on the database row. It is initialized to one when the row is created and should be incremented each time the row is updated.

By default, three priorities are defined in KEN:

Table 3.1. Notification: Priority Attributes

| ID | Name | Description | Order |
|----|--------|-----------------|-------|
| 1 | Normal | Normal priority | 2 |
| 2 | Low | A low priority | 3 |
| 3 | High | A high priority | 1 |

Managing Priorities

There is no user interface page to manage priorities, so changes to the list of priorities must be made by a programmer using SQL. See the KEN section of the Technical Reference Guide for further information on this topic.

Chapter 4. KEW

Kuali Enterprise Workflow: Overview

Kuali Enterprise Workflow, or KEW, is a module of Kuali Rice. KEW provides useful features for automatically sending documents to people for approval or action using workflows, managing the documents you receive through these workflows, searching for documents, and checking information about documents that have been in a workflow. KEW is built by and for higher education institutions and is especially designed for automatically routing work across departmental boundaries.

Using KEW you can:

- Automatically send (route) documents to individuals, groups, or roles (people who do a particular function) using workflows
- Create rules to automatically route documents based on the content of the document
- Change routing rules, and the documents affected by your changes are rerouted immediately
- Attach notes and other files to documents in workflows
- Build forms easily and use them in workflows
- Search for documents based on workflow information and some content in the document
- Check the history of documents and the people who took part in the workflow for each document
- Delegate items in a workflow to another person • Create and update groups for workflows
- Customize the email messages that are sent when someone needs to take an action for a workflow
- Choose whether or not you receive an email message when you have a new action item from a workflow
- Integrate KEW with the Kuali Service Bus (KSB) for routing documents

Your institution may use some or all of these KEW features. Your institution may also use some of these KEW features and combine them with other non-Kuali applications, such as an application that manages users and/or groups, or an application that manages notes or attachments.

Information Sources: <http://kew.kuali.org/>

What is KEW?

Kuali Enterprise Workflow provides a common routing and approval engine that facilitates the automation of electronic processes across the enterprise. The workflow product was built by and for higher education, so it is particularly well suited to route mediated transactions across departmental boundaries. Workflow facilitates distribution of processes out into the organizations to eliminate paper processes and shadow feeder systems. In addition to facilitating routing and approval, workflow can also automate process to process related flows. Each process instance is assigned a unique identifier that is global across the organization. Workflow keeps a permanent record of all processes and their participants.

KEW Features

- **Flexible Workflow Engine** - Support for sequential, parallel and dynamic routing paths. Extensible architecture allows for easy customization.

- **Content-Based Routing** - Routing decisions can be made based on XML document content. XPath and other XML tools can be used to determine routing without writing code.
- **Pluggable Components** - Components can be deployed to the system at runtime using Plugins. Hot deployable class loading space provides a robust enterprise ready deployment environment for workflow code.
- **People in the Routing Process** - Documents can be routed to individuals, groups or roles.
- **Action List** - Displays a list of each user's pending items which require his/her attention, such as documents which are awaiting approval. Users can configure whether they receive emails when the document enters their Action List.
- **Document Search and Route Log** - Allows users to search for documents and see an audit trail of what has happened to the document during its life cycle.
- **Document Search Customization** - Document based content can be associated with workflow data and searched on using our Document Search screens. Have a single place for all of your workflow document searches.
- **EDocLite** - EDocLite allows quick document building and integration with workflow using only XML.
- **Rules System** - Provides a mechanism for defining business rules which govern how a document routes. Rule screens give functional users ability to maintain the business rules for their applications. Documents affected by rule changes are re-routed real time.
- **Notes and Attachments** - Notes and Attachments can be attached to documents using KEW's notes and attachments services out of the box. Institution based attachment and note services can be used by overriding our default services.
- **Person Services** - Maintains users of the system. You can use the Out-of-the-Box service or override with your institution's user services.
- **Group Services** - Maintains groups of users. You can use the Out-of-the-Box service or override with your institution's group services.
- **Transactions** - All transactions and documents route in a JTA transaction.
- **Web Service API** - All system functions are available through Web Service APIs.
- **Security** - Web service calls can be authorized using digital signatures.
- **Scalability** - Can be clustered and run on multiple machines to allow for horizontal scalability.
- **Embeddable Engine** - Workflow engine can be embedded in an application as well as ran as a standalone service.
- **Embeddable Web Application** - Web portion can be embedded in an application as a Struts module. Run the Action List, Document Search, Route Log, Rules System and more from within your application with minimal effort.
- **Service Bus Integration** - Integration with the Kuali Service Bus (KSB). Override any service from within workflow by having workflow grab the service from the bus or use workflow's pluggable components to deploy bus enabled services.
- **JMX Support** - A set of management functions are exposed through JMX.
- **Spring Based Integration** - KEW is designed with Spring based integration in mind.

Why Use KEW?

- Provides a single action list for the constituents of the organization to see work that requires their attention
- Establishes a configurable way for service providers to define their processes, allowing them to alter those processes over time to reflect organizational change
- Promotes transparency of processes to the institution so that people can seamlessly see the status, actors, and the history of any institutional process which leverages KEW as its workflow engine

PeopleFlow - a new KEW feature

PeopleFlow is our Kualu Rice instantiation of the "maps" concept included in the original Coeus. For all intents and purposes it's a prioritized list of people to send requests to. Essentially, it's like a mini people-based workflow that doesn't require you to specify a KEW node in the document type for each individual who might need to approve or be notified. You can define "Stops" in a PeopleFlow, where everything in the same stop proceeds in parallel, but all must be done within the stop before proceeding to the next stop.

You can call/execute a PeopleFlow from within a KEW workflow node directly, or you can invoke the [Kuali Rules Management System \(KRMS\)](#) engine from an application and any PeopleFlows that get selected during rule execution, defined in a KRMS agenda, will be called. In this way, you can integrate business rules across applications and workflows.

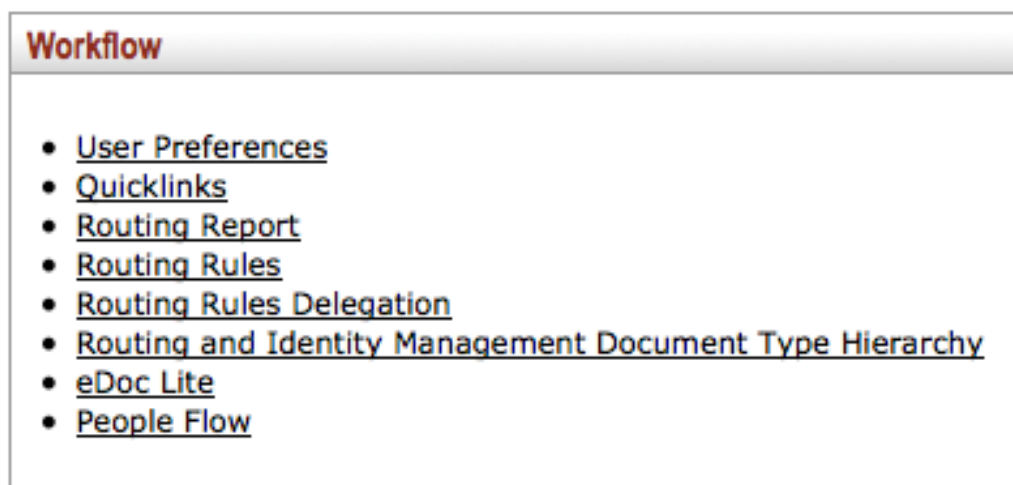
The same PeopleFlow that defines a routing order among a set of persons, groups or roles can be called by KRMS rules, with the KRMS rules defining which type of request to pass to the PeopleFlow (for example, an "approval" routing action or a "notification").

KRMS is also a new feature in Rice 2.0. See the KRMS Users' Guide for more information on KRMS.

Creating a new PeopleFlow

You can define a PeopleFlow (simple workflow) via a maintenance document. In the Kualu Rice portal on the main tab there is a Workflow category, and PeopleFlow is in that category. Select it:

Figure 4.1. PeopleFlow selection on the Main tab in the Kualu Portal



You can search for a PeopleFlow to copy from and modify (be sure to give it a new unique name) or can create a new one (see the top right of the lookup screen).

Figure 4.2. PeopleFlow Lookup

kuali rice® [Provide Feedback](#)

Main Menu Administration KRAD 05 11:24 EST (Oracle9i)

action list doc search Logged in User: admin Login Logout

Home » People Flow Lookup

People Flow Lookup

[Create New](#)

| | |
|------------------------|--|
| Id: | <input type="text"/> |
| Namespace Code: | <input type="text"/> |
| Name: | <input type="text"/> |
| Type: | <input type="text"/> |
| Active: | <input checked="" type="radio"/> Yes <input type="radio"/> No <input type="radio"/> Both |

Copyright 2005-2009 The Kuali Foundation. All rights reserved.
Portions of Kuali are copyrighted by other parties as described in the [Acknowledgments](#) screen.

Below is a view of the user interface to create a new PeopleFlow:

Figure 4.3. PeopleFlow - Create New

Kualo rice Provide Feedback
 Main Menu Administration KRAD Rice Sample App :: 2.0.0-b4 :: #11 :: 2011-12-21 14:23 EST (Oracle9i)
 action list doc search Logged in User: adr Login Logout
 Home > People Flow Maintenance
People Flow Maintenance Document Number: 3017 Document Status: INITIATED
Initiator Network Id: admin Creation Timestamp: 09:09 PM 12/23/2011
expand all collapse all
 * indicates required field
Document Overview
 * Description: Explanation:
 Organization Document Number:
PeopleFlow Summary
 * Namespace Code: * Description:
 * Name: Active:
 * Type:
PeopleFlow Members
 add
 * Stop: * Member Type Code:
 * Member: add
 Notes and Attachments (0)
 Ad Hoc Recipients
 Route Log
submit save blanket approve close cancel

The view below shows that some sections are collapsed, and also shows additional attributes that are required based on the user's selection of a type. This is an example of the Kualo Rapid Application Development (KRAD) framework's support of progressive disclosure, only showing these additional type attribute fields if/when they are required:

Figure 4.4. PeopleFlow - Create New (with additional type attributes)

Home » People Flow Maintenance

| | | |
|--------------------------------|------------------------------------|--|
| People Flow Maintenance | Document Number: 3018 | Document Status: INITIATED |
| | Initiator Network Id: admin | Creation Timestamp: 09:23 PM 12/23/2011 |

* indicates required field

▶ **Document Overview**

▼ **PeopleFlow Summary**

| | | | |
|--------------------------|----------------------|---------------------|-------------------------------------|
| * Namespace Code: | Kuali Rules Test | Description: | This is just an example |
| * Name: | Example PeopleFlow 1 | | |
| * Type: | KR-SAP - Sample Type | Active: | <input checked="" type="checkbox"/> |

| | | |
|----------------------------------|----------------------|----------------------------------|
| Travel Account Number: | <input type="text"/> | <input type="button" value="🔍"/> |
| Travel Fiscal Officer Id: | <input type="text"/> | <input type="button" value="🔍"/> |

▶ **PeopleFlow Members**

▶ **Notes and Attachments (0)**

▶ **Ad Hoc Recipients**

▶ **Route Log**

And below is a view of just the PeopleFlow Members section - expanded - after adding two stops:

Figure 4.5. PeopleFlow - Create New - with 2 stops added

PeopleFlow Members

add

* Stop: 1

* Member Type Code: Principal

* Member: 🔍

add

| * Stop | * Member Type Code | * Member | * All or First Action | Actions |
|-------------------------|--------------------|------------------------------------|------------------------|-----------------------|
| 2 | Role | KR-SYS : Technical Administrator 🔍 | FIRST | delete |
| Member Delegates | | | | |
| | * Member Type Code | * Member | * Delegation Type Code | * All or First Action |
| | Principal | <input type="text"/> 🔍 | Primary | add |
| 1 | Principal | eric Employee, Eric 🔍 | | delete |
| Member Delegates | | | | |
| | * Member Type Code | * Member | * Delegation Type Code | * All or First Action |
| | Principal | <input type="text"/> 🔍 | Primary | add |
| | Group | KR-WKFLW : MagazineManagers 🔍 | Primary | delete |

PeopleFlows that you create can be called by rules, with the rules determining whether the PeopleFlow will be called for an action (such as approvals) or for notifications. These PeopleFlows (similar to the Coeus concept of "maps") can be integrated with the new Kualu Rules Management System (KRMS), with KEW workflows, or with other custom-built applications.

Actions

Action List Guide

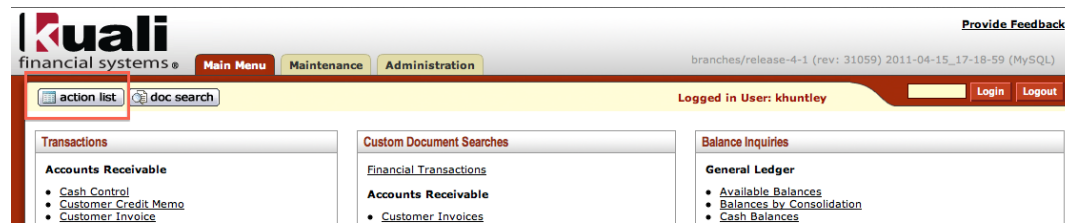
The **Action List** page includes your:

- Action List
- Outbox
- Action List Filter
- Preferences

In the Kualu Rice portal, the Action List button is available in the upper left hand corner of the screen on most pages.

Your institution may have KEW links inside other applications, such as Kualu Financial Systems (KFS). KFS may display an Action List button at the top left of the KFS page like this:

Figure 4.6. Kualu Portal: Action List Button



Clicking an Action List link or button displays your **Action List** page. This is a sample KEW Action List; your institution's may look somewhat different:

On your Action List page is the list of documents on which you need to take an action, such as Approve, Disapprove, or Acknowledge, and those which you have already completed. These are your action requests. Action requests that someone has delegated to you are also listed on your Action List page. Note: You can only see the Action List for the person who is logged into KEW or another Kualu application.

You can click the title at the top of a column to sort all of your action requests by the information in that column. For example, if you click the column title, Type, KEW sorts your action requests by their Type.

On your Action List, you can see this information about the documents on which you need to take action:

- **Id:** A unique, system-generated ID for each document
 - When you click a Document Id, KEW displays the actions you can take on this document, such as Blanket Approve, Complete, Approve, Disapprove, and Cancel. The actions you can take are determined by the document's Type.
 - Based on your institution's setup, you may see a Show button in the Document Id column. Click the Show button to see a summary of the document.
- **Type:** The document's type determines the actions you can take on it.
- **Title:** The title that the initiator gives to the document. Your institution may have standards for document titles that give you specific information about that document.
- **Status:** The current location of the document in its routing path. The Status may be:
 - **Approved:** These documents have been approved by all required reviewers and are waiting for additional post-processing.
 - **Cancelled:** These documents have been stopped. The document's initiator can Cancel it before routing begins, or a reviewer of the document can cancel it after routing begins. When a document is cancelled, routing stops; it is not sent to another Action List.
 - **Disapproved:** These documents have been disapproved by at least one reviewer. Routing has stopped for these documents.
 - **Enroute:** Routing is in progress on these documents and an action request is waiting for someone to take action.
 - **Exception:** A routing exception has occurred on this document. Someone with the appropriate KIM Permission (or someone from the Exception Group for this Document Type, although this feature

is deprecated) must take action on this document, and it has been sent to the Action List of this workgroup.

- **Initiated:** A user or a process has created this document, but it has not yet been routed to anyone's Action List.
- **Processed:** These documents have been approved by all required users, and their post-processing is completed. They may be waiting for Acknowledgements. No further action is needed on these documents.
- **Saved:** These documents have been saved for later work. An author (initiator) can save a document before routing begins, or a reviewer can save a document before he or she takes action on it. When someone saves a document, the document goes on that person's Action List.
- **Final:** All required approvals and all acknowledgements have been received on these documents. No changes are allowed to a document that is in Final status.
- **Recalled:** The document has been recalled after being submitted and canceled. No changes are allowed to a document that is in Recalled status.
- **Action Requested:** The action that you have been asked to take on a document, such as to Approve it.
- **Initiator:** The person who created this document in KEW. Click the initiator's name to see information about the initiator, including (depending on the initiator's privacy preferences):
 - Principal Id and Name
 - Entity Id
 - Affiliations
 - Names
 - Phone Numbers
 - Email Addresses
 - Group and Role memberships
- **Delegator:** If someone else has given authority in KEW for you to take action on their action requests, their name appears in this column. KEW displays both primary and secondary delegates this way on your Action List. There are two types of delegate in KEW, you may be a Primary Delegate or a Secondary Delegate:
 - **Primary Delegate:** The delegator turns over his or her full authority to a primary delegate. The Action Requests for the delegator only appear in the Action List of the primary delegate.
 - **Secondary Delegate:** The secondary delegate acts as a temporary backup and has the same authority as the delegator when the delegator is not available to take action. Documents appear in the Action Lists of both the delegator and the delegate. When either acts on the document, it disappears from both Action Lists. People often have a secondary delegate who takes care of their action requests when they are out of the office.
- **Date Created:** The date and time the document for this action request was created.
- **Group Request:** When you receive an action request because you are part of a group, KEW displays the name of the group in this column. Only one member of a group is required to complete an action request. Click the group name on your Action List to see more details, including:

- Group Id
- Group Name
- Group Active Indicator
- Group Members and their user names
- **Actions:** For documents that you receive only for your information (they don't require that you take any further action), there is a dropdown menu in the Actions column. You can use this menu to agree that you've received the document for information purposes: Select FYI, then click the Take Actions button at the bottom of the page.

Figure 4.7. Action List

Action List preferences refresh filter help desk action list login

NONE ▾ apply default

Action List

16 items retrieved, displaying 1 to 10. [First/Prev] 1, 2 [Next/Last]

| <u>Id</u> | <u>Type</u> | <u>Title</u> | <u>Route Status</u> | <u>Action Requested</u> | <u>Initiator</u> | <u>Delegator</u> | <u>Date Created</u> | <u>Group Request</u> | <u>Actions</u> | <u>Log</u> |
|---------------------------|----------------------------------|---|-------------------------------------|---|-----------------------------------|----------------------------------|-------------------------------------|--------------------------------------|---|----------------------------|
| 3402 | Requisition | Requisition - | SAVED | COMPLETE | HUNTLEY, KEISHA Y | | 04:57 PM 07/11/2011 | | | |
| 3403 | Requisition | Requisition - | SAVED | COMPLETE | HUNTLEY, KEISHA Y | | 04:57 PM 07/11/2011 | | | |
| 3404 | Requisition | Requisition - | SAVED | COMPLETE | HUNTLEY, KEISHA Y | | 04:57 PM 07/11/2011 | | | |
| 3405 | Requisition | Requisition - PC req - DV - 2 | ENROUTE | APPROVE | HUNTLEY, KEISHA Y | | 05:01 PM 07/11/2011 | | | |
| 3461 | Disbursement Voucher Wire Charge | New WireCharge - Phil | EXCEPTION | APPROVE | HUNTLEY, KEISHA Y | | 10:49 PM 07/11/2011 | | | |
| 3809 | Vendor | New VendorDetail - New Vendor | FINAL | FYI | HUNTLEY, KEISHA Y | | 07:26 PM 11/28/2011 | | NONE ▾ | |
| 3819 | Disbursement Voucher | Disbursement Voucher - test check amount | SAVED | COMPLETE | HUNTLEY, KEISHA Y | | 08:29 PM 11/28/2011 | | | |
| 3822 | Vendor | New VendorDetail - New Vendor | FINAL | FYI | HUNTLEY, KEISHA Y | | 08:53 PM 11/28/2011 | | <div style="border: 1px solid red; padding: 2px;"> ✓ NONE FYI </div> | |
| 3842 | Requisition | Requisition - test of equipment requisition | SAVED | COMPLETE | HUNTLEY, KEISHA Y | | 09:48 PM 11/28/2011 | | | |
| 3961 | Disbursement Voucher | Disbursement Voucher - test delegation | SAVED | COMPLETE | HUNTLEY, KEISHA Y | | 02:58 PM 11/29/2011 | | | |

take actions

- **Log:** (also known as the Route Log) Click the paper/magnifying glass icon in the Log column to display the Route Log page. This page has details about the routing path and Actions Taken on this document. Learn more about this information in the Route Log Guide. A sample Route Log page:

Figure 4.8. Route Log

| Route Log | | | | refresh | |
|--|----------------------------|--|---------------------|---------------------|------------|
| ID: 3064 hide | | | | | |
| Title | | | | | |
| Type | Vacancy Notice | Created | 08:35 PM 12/02/2011 | | |
| Initiator | admin_admin | Last Modified | 08:40 PM 12/02/2011 | | |
| Route Status | PROCESSED | Last Approved | | | |
| Node(s) | InitiatorAcknowledgement | Finalized | | | |
| Actions Taken hide | | | | | |
| | Action | Taken By | For Delegator | Time/Date | Annotation |
| | COMPLETED | admin_admin | | 08:36 PM 12/02/2011 | |
| show | APPROVED | supervisor_supervisor | | 08:38 PM 12/02/2011 | |
| show | APPROVED | director_director | | 08:39 PM 12/02/2011 | |
| show | APPROVED | Two_Admin | | 08:39 PM 12/02/2011 | |
| show | APPROVED | One_Admin | | 08:40 PM 12/02/2011 | |
| Pending Action Requests hide | | | | | |
| | Action | Requested Of | Time/Date | Annotation | |
| show | IN ACTION LIST ACKNOWLEDGE | QAA_BI-BUS (Secondary Delegate) | 08:40 PM 12/02/2011 | | |
| show | IN ACTION LIST ACKNOWLEDGE | QAA_BI-COAS (Secondary Delegate) | 08:40 PM 12/02/2011 | | |
| show | IN ACTION LIST ACKNOWLEDGE | QAA_BI-DFAC (Secondary Delegate) | 08:40 PM 12/02/2011 | | |
| show | IN ACTION LIST ACKNOWLEDGE | QAA_BI-PSY (Secondary Delegate) | 08:40 PM 12/02/2011 | | |
| show | IN ACTION LIST ACKNOWLEDGE | admin_admin (Secondary Delegate) (Initiator) | 08:40 PM 12/02/2011 | | |

If enabled for a particular document, messages can also be logged through the route log tab (or page). This allows users to add a deferral message or other message that will be reflected in the routing without taking an action on the document.

This function is enabled by KIM permissions. These permissions have template 'Add Message to Route Log' which takes a document type name as a detail. The bootstrap dataset contains one permission with this template and document type detail 'KualiDocument', however this permission is not granted to any role. This means the functionality will be disabled until the permission is assigned or another permission record with this template and for another document type is created and assigned. When matching permissions for a particular document type the document type hierarchy is considered. Permission records for children document types will override any permission record for its parents.

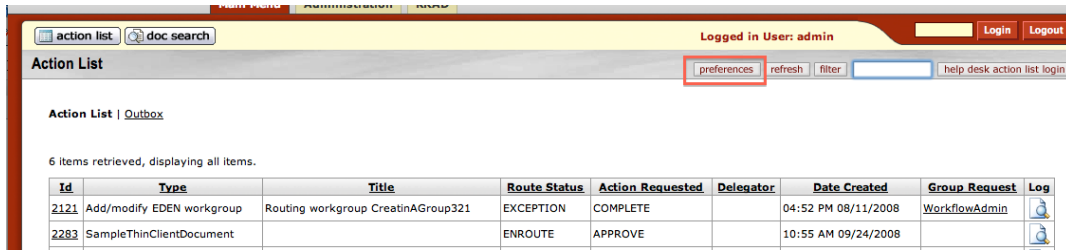
For example, if we wanted to enable the feature on all KualiDocuments except document type 'Foo', we would first grant the permission with document type detail 'KualiDocument' to a role. Then we would create an additional permission with template 'Add Message to Route Log' and document type detail 'Foo'. For the second permission we do not assign any roles.

By granting these permission(s) to one of the derived workflow roles we can enable this feature to users who have active routing requests. Examples of such roles are KR-WKFLW Approve Request Recipient and KR-WKFLW Acknowledge Request Recipient.

Action List Preferences

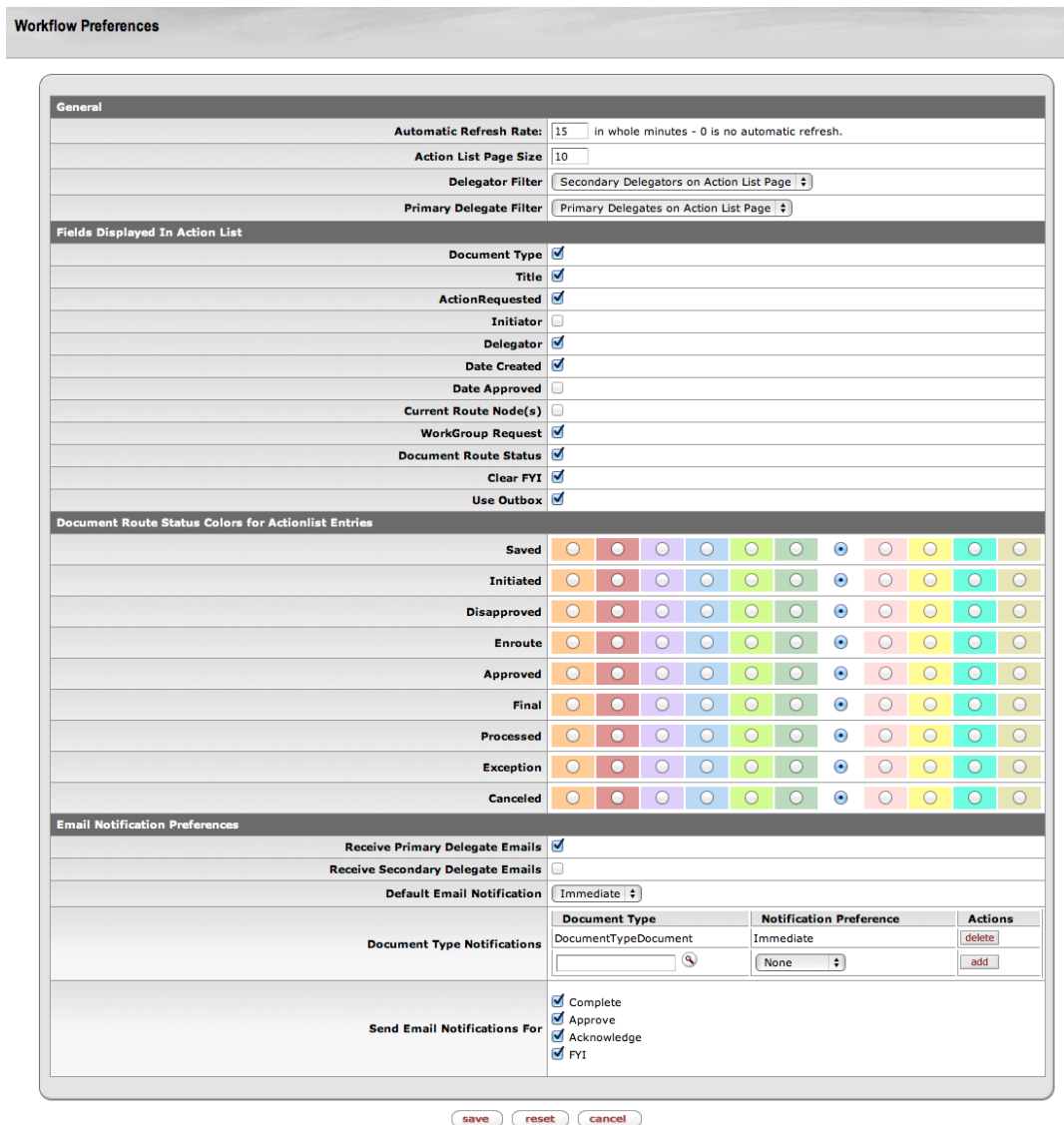
You can customize the appearance and features of your Action List using the Action List Preferences page on KEW. To display this page, click the Preferences button at the top of the Action List page or click the User Preferences link in the Workflow pane of the Main Menu tab. Action List page:

Figure 4.9. Action List: Preferences Button



This displays the Action List Preferences page, where you can make choices about how KEW displays and refreshes (updates) your Action List and about how and when you receive email reminders about your new action requests:

Figure 4.10. Action List Preference Page



The Action List Preferences page has four sections:

Action List Preferences: General Section

Figure 4.11. Action List Preferences: General section

| General | |
|---------------------------------|--|
| Automatic Refresh Rate: | 15 In whole minutes - 0 is no automatic refresh. |
| Action List Page Size | 10 |
| Delegator Filter | Secondary Delegates on Action List Page ↓ |
| Primary Delegate Filter | Primary Delegates on Action List Page ↓ |
| Fields Displayed In Action List | |

- In the General section of the Action List Preferences page, you can set the Automatic Refresh Rate for your Action List. Your action requests come in from the server and your Actions Taken are sent to the server each time KEW refreshes (updates) your Action List page. You only see changes to your Action List when KEW refreshes it.
- Depending on the network and computers at your workplace, your computer may or may not slow down when KEW refreshes your Action List. If a refresh slows down your computer, you may decide to set the Automatic Refresh Rate for a longer period, such as 10 or 15 minutes.
- The Action List Page Size tells KEW how many action requests to display at once on your Action List page.
- The Delegator Filter field lets you decide where to show action requests for which you are a secondary delegate. You can choose to have KEW display these action requests on your Action List page or not, using this dropdown menu. If you do not display action requests on which you are the Secondary Delegate on your Action List page, you can only see them on a Filter Page (see Filter, below).

Action List Preferences: Fields Displayed

Figure 4.12. Action List Preferences: Fields Displayed section

| Fields Displayed In Action List | |
|---------------------------------|-------------------------------------|
| Document Type | <input checked="" type="checkbox"/> |
| Title | <input checked="" type="checkbox"/> |
| ActionRequested | <input checked="" type="checkbox"/> |
| Initiator | <input type="checkbox"/> |
| Delegator | <input checked="" type="checkbox"/> |
| Date Created | <input checked="" type="checkbox"/> |
| Date Approved | <input type="checkbox"/> |
| Current Route Node(s) | <input type="checkbox"/> |
| WorkGroup Request | <input checked="" type="checkbox"/> |
| Document Route Status | <input checked="" type="checkbox"/> |
| Clear FVI | <input checked="" type="checkbox"/> |
| Use Outbox | <input checked="" type="checkbox"/> |

- You determine which fields are displayed on your Action List page using this section of the Action List Preferences page. Place a checkmark in the box next to fields that you want to display on your Action List page. (Click the box to checkmark it and click it again to un-checkmark it, and vice versa.)

Action List Preferences: Document Route Status Colors

Figure 4.13. Action List Preferences: Document Route Status Colors Section

| Document Route Status Colors for Actionlist Entries | | | | | | | | | | | |
|---|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|
| Saved | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Initiated | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Disapproved | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Enroute | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Approved | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Final | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Processed | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Exception | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Canceled | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |

- In this section of the Action List Preferences page, you can choose a color for each different Status of action requests. This can help you quickly find action requests on which you need to take action.

For example, if you set these colors for these routing status levels:

- Green for Saved status
- Red for Approved status
- Blue for Exception status

Your Action List Preferences page looks like this:

Figure 4.14. Action List Preferences: Document Route Status Colors Example

| Document Route Status Colors for Actionlist Entries | | | | | | | | | | | |
|---|----------------------------------|----------------------------------|-----------------------|-----------------------|-----------------------|-----------------------|----------------------------------|----------------------------------|-----------------------|----------------------------------|-----------------------|
| Saved | <input checked="" type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Initiated | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Disapproved | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Enroute | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Approved | <input type="radio"/> | <input checked="" type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Final | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input checked="" type="radio"/> | <input type="radio"/> |
| Processed | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input checked="" type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Exception | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input checked="" type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Canceled | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input checked="" type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |

Click the Save button at the bottom of the page to make these changes in your Action List.

Action Lists your color changes by Status:

Figure 4.15. Action List Preferences: Document Route Status Colors Action List Example

Action List preferences refresh filter help desk action list login

Action List | [Outbox](#)

6 items retrieved, displaying all items.

| ID | Type | Title | Route Status | Action Requested | Delegator | Date Created | Group Request | Log |
|-----------|---------------------------|------------------------------------|---------------------|-------------------------|------------------|---------------------|----------------------|------------|
| 2121 | Add/modify EDEN workgroup | Routing workgroup CreatinAGroup321 | EXCEPTION | COMPLETE | | 04:52 PM 08/11/2008 | WorkflowAdmin | |
| 2283 | SampleThinClientDocument | | ENROUTE | APPROVE | | 10:55 AM 09/24/2008 | | |
| 2601 | Travel Request | Travel Doc 2 - asdsadsadasd | SAVED | COMPLETE | | 01:33 PM 11/04/2008 | | |
| 2621 | Travel Request | Travel Doc 2 - asdsadsadasd | SAVED | COMPLETE | | 01:45 PM 11/04/2008 | | |
| 2622 | Travel Request | Travel Doc 2 - asdsadsadasd | SAVED | COMPLETE | | 01:49 PM 11/04/2008 | | |
| 2623 | Travel Request | Travel Doc 2 - asdsadsadasd | SAVED | COMPLETE | | 01:57 PM 11/04/2008 | | |

Action List Preferences: Email Notifications

Figure 4.16. Action List Preferences: Email Notifications Preferences

Email Notification Preferences

Receive Primary Delegate Emails

Receive Secondary Delegate Emails

Default Email Notification

| Document Type | Notification Preference | Actions |
|----------------------|-----------------------------------|---------------------------------------|
| DocumentTypeDocument | Immediate | <input type="button" value="delete"/> |
| <input type="text"/> | <input type="text" value="None"/> | <input type="button" value="add"/> |

Send Email Notifications For

Complete

Approve

Acknowledge

FYI

The fields used to maintain email preferences are located in a new “Email Notification Preferences” section at the bottom of the Workflow Preferences screen. The “Email Notification Preferences” section accommodates two new options and allows selection of how and when to receive reminder emails on items sent to the Action List.

In this section, one can select whether or not to receive emails as the Primary Delegate, as the Secondary Delegate, and also set the Default Email Notification preference (None, Daily, Weekly, or Immediate). If you are NOT a Primary or Secondary Delegate, you can leave those fields as unchecked.

You can choose to get email reminders on one of these Default Email Notification schedules:

- **None** (KEW won’t send you any email reminders for your Action List.)
- **Daily** (If you have any new action requests, KEW sends you an email reminder once a day.)
- **Weekly** (If you have any new action requests, KEW sends you an email reminder once a week.)
- **Immediately** (KEW sends you an email reminder as soon as you receive a new action request on a document. This is the default – the setting when you start to use KEW.)

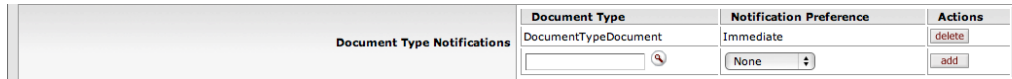
The first new option allows overriding of the ‘Default Email Notification’ preference and setting of different frequencies to receive notifications based on the document type of the action request (Timesheet, EPTO, Hire Employee eDoc, Create Additional Pay eDoc, Requisition, etc.).

The parameters set for the default email notification and document-type notification work together to determine when a given action item is included in the immediate, daily, or weekly email reminders.

If one sets a ‘Default Email Notification’ without any ‘Document Type Notifications,’ then emails are sent (or not sent), based on the default selection.

The second new option allows one to turn notifications on or off based on the request type (Complete, Approve, Acknowledge, or FYI) when “Immediate” is set as the ‘Default Email Notification.’

Figure 4.17. Action List Preferences: Document Type Notifications



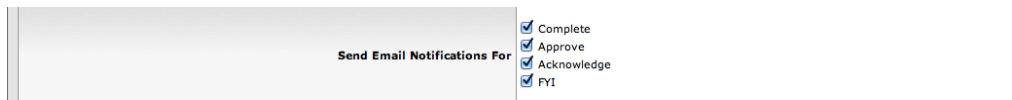
However, adding a ‘Document Type Notification’ that is different from the ‘Default Email Notification,’ forces the document-type notification to take precedence. This allows users to customize by document-type for email reminders. For example:

- If ‘Immediate’ is set as the ‘Default Email Notification,’ but a document-type rule is set to ‘None’ for the Maintain Time Assignment eDoc; then no email notifications are sent for that specific document.
- If ‘Daily’ is the ‘Default Email Notification,’ and a document-type rule is added to receive ‘Weekly’ notifications for the Hire Employee eDoc; then for that document, the only action request email received is a Weekly reminder.
- If ‘None’ is set as the ‘Default Email Notification,’ and a document-type rule is added to receive “Daily” notifications for EPIC Purchase Orders; then for that document, the only action request email received is a Daily reminder.

In addition, when the ‘Default Email Notification’ is set to “Immediate” one can use the Send Email Notifications For option to decide whether or not to receive notifications by request-type.

All four of the request-types are already “checked” by default in your Workflow Preferences screen.

Figure 4.18. Action List Preferences: Action Requested Email Notification



When an action request is received, and the preference is set to receive an immediate reminder, the system checks to ensure that the request-type checkbox is turned-on (checked) before an email is sent.

This allows these checkboxes to be used as a switch to ‘turn-on or turn-off’ the “immediate” email reminders for a given request type. For example:

- To stop the receipt of Acknowledgement emails when your preference is set to receive “Immediate” reminders, just uncheck the “Acknowledge” checkbox.

Remember, all of these preferences and settings are on your Action List Preferences page.

Action List Refresh and Filter

Refresh

As described above, you can use the Action List Preferences page to tell KEW to automatically update (refresh) the information on your Action List on a regular schedule, such as every 15 minutes. You can also click the Refresh button near the top of your Action List page to update the page at any time.

Filter

If you have a large Action List, the Filter function can help you by displaying only action requests of certain types. To use the Filter function, click the Filter button in the top right corner of your Action List page:

This displays the **Action List Filter** page:

Figure 4.19. Action List Filter Page

The screenshot shows the 'Action List Filter' page. At the top, there is a header 'Action List Filter'. Below it is a 'Parameters' table with the following rows:

| Parameters | |
|------------------------|--|
| Document Title | <input type="text"/> Exclude? <input type="checkbox"/> |
| Document Route Status | All <input type="button" value="v"/> Exclude? <input type="checkbox"/> |
| Action Requested | All <input type="button" value="v"/> Exclude? <input type="checkbox"/> |
| Action Requested Group | No Filtering <input type="button" value="v"/> Exclude? <input type="checkbox"/> |
| Document Type | <input type="text"/> Exclude? <input type="checkbox"/> |
| Date Created | from: <input type="text"/> <input type="button" value="v"/> to: <input type="text"/> <input type="button" value="v"/> Exclude? <input type="checkbox"/> |
| Date Last Assigned | from: <input type="text"/> <input type="button" value="v"/> to: <input type="text"/> <input type="button" value="v"/> Exclude? <input type="checkbox"/> |

At the bottom of the form, there are four buttons: filter, clear, reset, and cancel.

Keys to using the Action List Filter:

- Each of the fields on the Action List Filter page is called a criteria. You can choose to display only the action requests that are the same as one or more criteria, or you can choose to display all action requests except those that are the same as one or more criteria. You can also use some criteria to display action requests and some to not display action requests.
- If you want the filter to display action requests that meet one or more criteria, then just enter those criteria in the appropriate fields (field descriptions are below). Leave the Exclude? checkboxes blank for these criteria.
- If you want the filter to display only the action requests that don't match one or more criteria, then checkmark the Exclude? checkbox for each of these criteria. This tells KEW not to display action requests that match these criteria.
- When you've selected the criteria you want to use in your filter, click the Filter button at the bottom of the Action List Filter page. KEW displays your Action List filtered in the way you requested.
- To display all action requests on your Action List page again, click the Clear button at the bottom of the Action List Filter page or the Clear Filter button at the top of your Action List page.
- To undo your most recent filter criteria, click the Reset button at the bottom of the Action List Filter page. This returns you to the list of Action Requests that met your previous set of filter criteria.

Note

Note: The filter is case sensitive – use upper and lower case (capital letters and small letters) exactly the same as the item you're filtering. For example, if an actual document title is Travel Doc 2, then you must type this in the Document Title field with a capital letter at the beginning of

each word, exactly like the real document title. If you type it as travel doc 2, then the filter won't recognize this document because the capitalization is different.

Filter Criteria:

- **Document Title:** The title given to the document when it was created.
- **Document Route Status:** The current routing Status of the document. You may filter for documents with these routing statuses:
 - **All:** Selecting All causes KEW to display all documents that have been submitted for routing. If you select All and Exclude (for the Document Route Status criteria), then KEW displays only documents that are in Initiated status.

Note

Documents that are in Initiated Status have not yet been submitted for routing.

- **Approved**
- **Disapproved**
- **Enroute**
- **Exception**
- **Processed**
- **Saved**
- **Action Requested:** The action you need to take on a document. You may filter your action requests by these Action Requests:
 - **All:** matches all Action Requested types
 - **Acknowledge**
 - **Approve**
 - **Complete**
 - **FYI**
- **Action Requested Group:** You may filter your action requests by the group to which the action requests were sent. For example, if you are a member of two groups, you might use this filter to display only the action requests you've received as part of one of those groups.
- **Document Type:** Use this to filter based on Document Type.
 - The document's type tells KEW who needs to take what action on that document. This determines where KEW routes the document.
 - You can use the Type filter to display only documents of one type, or to display all of your documents except one type. Click the checkbox next to **Exclude?** if you do NOT want to display the selected Type.

- To choose a Type for your filter, click the magnifying glass icon on the Type row. This takes you to the **Document Type Lookup** page.

Enter information in one or more of these fields to find the Document Type you need for your filter, then click the Search button. This displays a list of document Types that match your search. Find the one you need, and click return value for that Type. This returns you to the previous page and automatically enters it in the Type field. For example, in the screenshot below you searched for a Type with maintenance in the Name. If you want to use the first one in the list, KIM Principal Maintenance Document, click return value in the row for KIM Principal Maintenance Document.

Figure 4.20. Document Type Lookup

Document Type Lookup create new
* required field

| | |
|--|--|
| Parent Name: | <input type="text"/> |
| Name: | <input type="text"/> |
| Label: | <input type="text"/> |
| Id: | <input type="text"/> |
| Active Indicator: | <input checked="" type="radio"/> Yes <input type="radio"/> No <input type="radio"/> Both |
| <input type="button" value="search"/> <input type="button" value="clear"/> <input type="button" value="cancel"/> | |

39 items retrieved, displaying all items.

| Actions | Id | Name | Parent Name | Label | Active Indicator | Application ID |
|---|------|----------------------|------------------------------|---|------------------|-------------------------|
| edit copy | 2011 | DocumentTypeDocument | RiceDocument | Workflow Maintenance Document Type Document | Yes | TRAVEL (System Default) |
| edit copy | 2012 | RoutingRuleDocument | RiceDocument | Rule Maintenance Document Type Document | Yes | TRAVEL (System Default) |
| edit copy | 2023 | KualiNotification | | Notification | Yes | TRAVEL (System Default) |

Note

If you do not find the Type that you need, you can either do another search or click the link for return with no value to return to the Action List Filter page without a Type.

- Date Created:** Use this filter criteria to select only action requests for documents that were (or were not) created between dates that you choose. You must enter each date in this format: MM/DD/YYYY, e.g., 12/25/2000 for December 25, 2000. Instead of typing the date in the field, you can click the small Calendar icon next to the date field and then click the date you need from the calendar. When you click a date on the calendar, the small calendar disappears and that date is automatically entered in the date field. Use the small arrowheads on either side of the month's name to display other months. A single arrowhead moves the calendar one month forward or back. A double arrowhead moves the calendar to the same month a year ahead or back.

For example, if you could click the single arrowhead to the left of "April" on this calendar, the month displayed would change to March 2009. If you could click the double arrowhead to the right of "April", the month would change to April 2010.

Figure 4.21. Date Widget

The screenshot shows a 'Parameters' form with several filter criteria, each with an 'Exclude?' checkbox. The criteria are: Document Title, Document Route Status (set to 'All'), Action Requested (set to 'All'), Action Requested Group (set to 'No Filtering'), Document Type, Date Created (with 'from:' and 'to:' fields), and Date Last Assigned (with 'from:' and 'to:' fields). A date selection calendar for December 2011 is overlaid on the 'Date Created' and 'Date Last Assigned' fields. The calendar shows the current date as the 2nd. Below the calendar are buttons for 'filter', 'clear', 'reset', and 'cancel'.

- **Date Last Assigned:** Use this filter criteria to select only action requests for documents that were (or were not) assigned to someone between dates that you choose. You must enter each date in this format: MM/DD/YYYY. For more detail on entering or selecting dates, see Date Created above.

Clear Filter

When you have used a filter on your Action List, you can see a Clear Filter button at the top of your page:

Figure 4.22. Action List: Clear Filter Button

Click this button to display all of your action requests again.

Note

If your Action List is not filtered, you cannot see the Clear Filter button.

Remember, you can click the Filter button at the top of your Action List page to return to the Filter page and change or Reset your filters.

Action Requests and Actions Guide

Overview

Action Requests are one of the core functions of the Quali Enterprise Workflow (KEW) system. An Action Request is a call from KEW to a person or group to take action on a document. Action Requests are created when the system or a person routes (sends) a document to another user and ask that user to do something with the document. You might ask someone to approve an expense, for example, or to acknowledge that he or she received a copy of an expense report.

Once an Action Request has been created in KEW, the associated document is sent to the appropriate users' action lists. A user can open his or her Action List and from there, view information about the document, open the document, and/or take the requested action. If a user takes an action, for example, approves or disapproves a document, that is called an Action Taken.

The path that the document takes from its starting point to final approval is called its route. Each point along that path where a person needs to do something with the document (approve, acknowledge, etc.) is called a node on the route.

Types of Action Requests

Each action request has one row on your Action List. There are five types of Action Requests:

Table 4.1. Types of Action Requests

| Action Requested | Description | Stops Document Until Action is Taken? |
|------------------|---|--|
| Complete | You need to Complete the document. You may cancel a document when you have a Complete action request on it. | Yes |
| Approve | You should verify the document and either Approve or Disapprove it. You may cancel a document when you have an Approve action request on it. | Yes |
| Acknowledge | You need to acknowledge (agree) that you have seen the document | No If there are no other requests, the document changes to Processed status even if there are outstanding Acknowledge requests. |
| FYI | You have been told about this document and you need to Clear the FYI on it | No If there are no other requests, the document changes to Final status even if there are outstanding FYI requests. |
| Print | You need to print this document and take some action with it, such as mailing it to a vendor. | |

- You can only Disapprove or Cancel a document if you have a Complete or Approve request.
- Complete and Approve are essentially the same: A Complete action satisfies an Approve request, and an Approve action satisfies a Complete request.
- When you Clear an FYI request, that action is not recorded on the Route Log (history of where the document has been and what has been done on it) for that document.

There are other actions that you can take in addition to Action Requests. We will look at those in the Action Taken section below.

Action Request Recipients

Every action request has a recipient. The recipient is the individual or group of individuals whose action is requested on the document. There are three types of recipients:

1. Person: An individual
2. Group: A group of users who are members of a KIM Group
3. Role: A group of users who are members of a KIM Role

Roles are groups of users or KIM Groups associated with a document.

The first user in a Group to take action on a request satisfies (completes) that request. Role requests can be set up to work the same way as Group requests or to require that everyone in that Role approves the document before the request is satisfied.

Action Request Hierarchies

In certain situations, you can use roles for action requests and you can delegate action requests. You can think of these as trees or as a parent action request that has children action requests.

Roles are groups of Principals or Groups that KEW creates based on rules your institution sets up. Since KEW can only create an action request for a single person or group but a role may have several people or groups in it, KEW creates an action request for the role (a parent request), and that action request creates child action requests that go to each person or group in that role.

The parent role request can have any number of children requests that are each either a user request or a group request.

Action Request Delegation

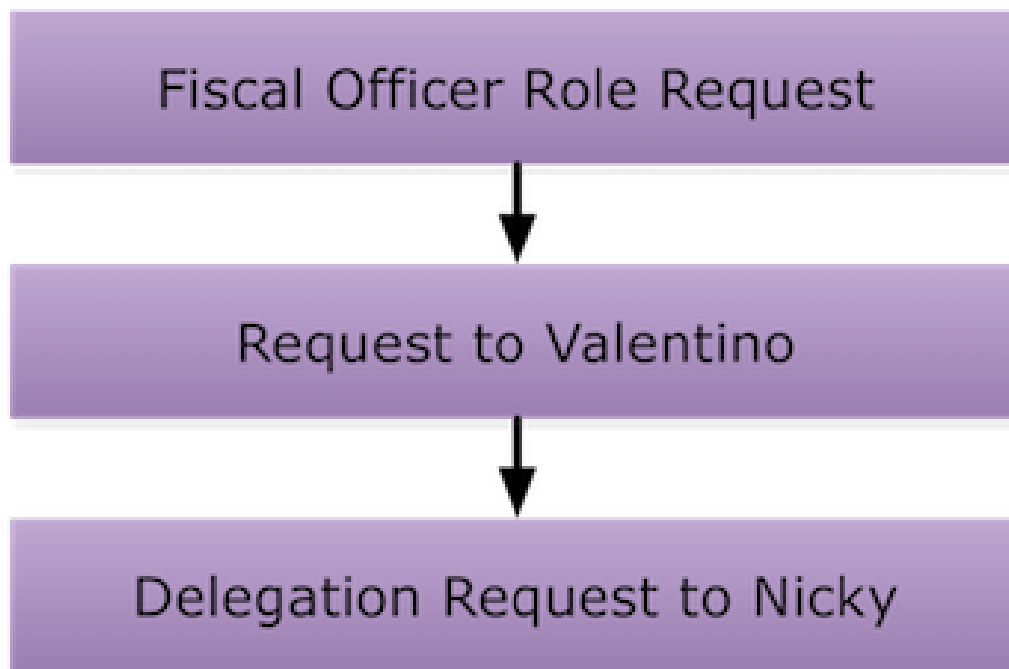
In KEW, you may delegate, or give, your request authority to other people. For example, if Joe is the fiscal officer for an account but he doesn't actually approve documents, he may delegate his approval authority to Jane. Both Joe and Jane can now take actions on Joe's action requests

If Jane approves a document as Joe's delegate, Joe's action request is satisfied. If Joe approves the document, Jane's action request is satisfied.

Combining Role Requests and Delegation

You can combine a role request with a delegate. In fact, in the previous example, the action request that is sent to Joe as a fiscal officer is really a role request. The action request tree for that scenario is:

Figure 4.23. Delegation Tree Example



This tree is three levels deep and shows a role delegating to a user. You can also have a role delegate to another role.

Action Request Activation and Deactivation

Activation

When action requests are created, they are in the Initialized state. They stay in this state until KEW determines that they need to be Activated. When a request becomes Activated, it appears in the user's Action List.

When action requests are created, they can be set to ignore previous actions or not.

- If *Ignore Previous* is true, the activation process will not consider previous actions that the user has taken.
- If *Ignore Previous* is false, the activation process will consider previous actions by the user and may even consider the action request satisfied by previous user actions. (This is sometimes referred to as a request being "auto-approved.")

During request activation, if the Ignore Previous flag is false and KEW determines that a previous action satisfies the current request, it will Deactivate the request instead of Activating it. Activation begins at a root request, but Deactivation begins at the request where KEW finds that a satisfying action has occurred if that request is set to not ignore previous actions.

Note

Action request structures can be hierarchical. Activation of requests always starts at the parent request and works down the tree, activating each level of the tree in turn.

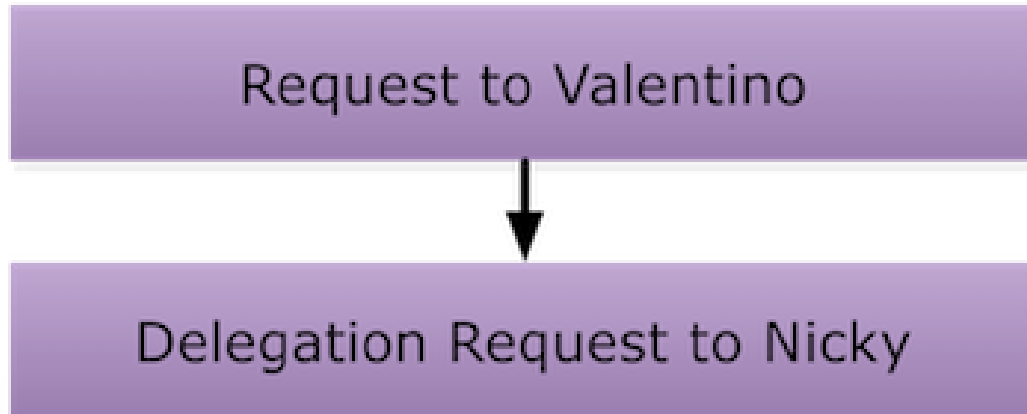
Roles can have an All Approve policy. All Approve means that all members of a role must approve the document before the entire request is deactivated. (See Deactivation below.)

Deactivation

When a user takes action on a document, that action may start the deactivation of other action requests on the document. For example, if Joe has a pending request and he takes an Approve action on it, his request is deactivated. Request deactivation changes the status of the request from Initialized or Activated to Deactivated.

If Joe's action taken causes other action requests to be Deactivated, his action taken is associated with all of the requests that it Deactivates.

Unlike Activation, Deactivation always starts at the request that was satisfied by the action taken. For example, if Joe is delegating to Jane and Jane issues an Approve action on her request, Deactivation starts at Jane's request. However, if Joe issues the Approve action, Deactivation starts at the parent.

Figure 4.24. Delegation Tree Example: Deactivation

If there are no roles set to "All Approve" on an action request, one action can Deactivate the entire request tree. This is because KEW works up and down the tree, Deactivating requests as it goes.

However, when there is an "All Approve" role, requests for all parties within the role must be Deactivated before the parent role request is Deactivated. For example, if we have an "All Approve" role request with three children requests, when one of the users takes action, their request is Deactivated, but since there are two other requests that still need to be satisfied, the Deactivation cannot go back to the parent role request. When the last of the three users has taken action, the parent role request is Deactivated.

Actions and Action Takens

Actions are how a user interacts with KEW. Actions can be performed in response to an Action Request (such as an Approve action) or they can be user-initiated (such as the routing of a document). Most, but not all, actions create Action Takens. The Action Taken is recorded on the Route Log and associated with a satisfied action request, where appropriate.

The KEW Action library contains these Actions, explained below:

- Acknowledge
- Approve
- Blanket Approve
- Cancel
- Clear FYI
- Complete
- Create Document
- Disapprove
- Log Document
- Move Document
- Return to Previous Node
- Route Document

- Save
- Recall
- Superuser Actions
 - Superuser Approve Document
 - Superuser Approve Node
 - Superuser Approve Request
 - Superuser Cancel
 - Superuser Disapprove

Acknowledge Action

Use this action to satisfy an Acknowledge action request. When you take an Acknowledge action, it usually means that you have examined the document. Acknowledgement provides no real authority over the document; you can only take the Acknowledge action on this document.

When you take an Acknowledge action, KEW finds all pending acknowledge requests for that document that are routed to you and Deactivates them. It then records an Acknowledge Action Taken and associates it with the Deactivated requests. You may take an Acknowledge action even if you have no Acknowledge Action Requests.

Approve Action

Use this action to satisfy an Approve or Complete action request. When you use an Approve action, it means that you have looked at the document and approve that transaction.

When you use the Approve action, KEW Deactivates all pending approve or complete requests that have been routed to you for this document. You may take an Approve action even if you have no Approve or Complete action requests.

Blanket Approve Action

Use the Blanket Approve action to force a document to complete a set of action requests. This allows certain users to "push" a document through its routing. To take this action, you must have the appropriate KIM Permission (or be in the Blanket Approve Group which is set in the Document Type, but that is a deprecated feature). Unless you select a specific node (point in the document's routing path) when you take a Blanket Approve Action, KEW finds all pending Approve or Complete requests at the current level in the request tree and Deactivates them. It records a Blanket Approve Action Taken and associates it with each of the Deactivated requests.

Then, KEW sends Acknowledge action requests to each person whose action request was satisfied by the Blanket Approve. It does this for all levels in the request tree until it reaches the end point set in the Blanket Approve Action or it reaches the end of the request tree.

Cancel Action

Use this action when a document is no longer valid and you need to cancel it. You must have a pending Approve or Complete action request on the document before you can Cancel it. Cancelling a document is similar to Disapproving it except that the Cancel action does not send out notifications.

When you take a Cancel action, KEW finds all pending requests on the document and Deactivates them. It records a Cancel Action Taken and associates it with the Deactivated requests. Then it sets the document's status to CANCELLED and the document is effectively dead.

Clear FYI Action

Use this action to satisfy an FYI action request. You usually get an FYI just to notify you about a document. You don't need to take any action on an FYI request, but if you want to remove that FYI and document from your Action List, take the Clear FYI action.

When you take the Clear FYI action KEW locates all pending FYI requests on this document that are routed to you and Deactivates them. It then records an FYI Action Taken. You must have a pending FYI request on a document to use the Clear FYI action.

Complete Action

Use this action to satisfy an Approve or Complete action request. When you use a Complete, it means that you have looked at the document and completed any of the necessary information on the document so it can proceed. Complete action requests are typically created as the result of a Save action or in response to an Exception Routing request.

When you use the Complete action KEW finds all pending Approve or Complete requests that are routed to you and Deactivates them. If the document is in the Exception state, it changes it back to Enroute. It then records a Complete Action Taken. You may take a Complete action even if you have no Complete or Approve action requests.

Create Document Action

This action creates a new document in KEW of the chosen Document Type. This action does not record an Action Taken of any sort but simply begins a new document and assigns a document ID to it. The document begins in Initiated status. The Document type determines how the document is routed and what must be done for it. For example, an expense report might be routed (sent) to two or three different people for approval before it is paid.

Disapprove Action

When you have an Approve action request, you use a Disapprove action if the document does not meet the approval criteria. You must have a pending Approve or Complete request on the document before you can Disapprove it. Disapproving a document is similar to Cancelling it, except that Disapproval sends out notifications.

When you Disapprove a document, KEW finds all pending requests on the document and Deactivates them. Next, it creates Acknowledge notification requests to all users that had Approve or Complete action requests at the current node and all previous nodes. It also sends an Acknowledge notification request to the initiator of the document. Then the document's status is set to Disapproved and the document is effectively dead.

Log Document Action

This action simply enters a message on the document. It records an Action Taken but it is never associated with any action requests. This action displays a message on the document in the Route Log.

Move Document Action

This action moves a document either backward in the route path or forward. Moving the document backward works like the Return to Previous Node action and moving the document forward works like

a Blanket Approve action. The difference between these is that the Action Taken is recorded as a Move action.

Return to Previous Node Action

Use this action to move a document to a previous node in the route. You must have a pending Approve or Complete request on the document before you can use this action.

When you use this action, KEW Deactivates all pending requests on the document and sets the current node on the document to the previous node that you requested (this is called the target node). It records a Return to Previous Node Action Taken and associates it with the Deactivated requests. Any requests on nodes between the current node and the target node are removed, effectively erasing them. KEW then sends the document through again to recreate requests on the target node. This way a document is "returned" to a previous node, rolling it back to its previous state at that point in the route.

Route Document Action

Use the Route Document action to route an Initialized or Saved document to other users. Only the initiator of the document can take this action unless the user has an "Initiate Document" KIM permission (or initiator_must_route policy is set to false in the Document Type, although this feature is deprecated). The Action Taken for a Route Document action is Complete.

When you use the Route Document action, it Deactivates all pending requests for the initiator of the document and it associates a Complete Action Taken with the Deactivated requests. The document's state is then set to Enroute.

Save Document Action

Use this action to put a newly created document in the Action List of the document's initiator. You can only use this action on a document in an Initiated status. When you use this action, KEW creates and Activates a Complete action request for the document's initiator and then changes the document's status to Saved.

Recall

Use this action to recall a document that has been submitted to your action list. You can only use this action on a document if you have a specific role for the application and/or document you are trying to recall. When you use this action, you are given the option to recall the document to your action list for updating or you can recall and cancel the document with one action if it is no longer needed.

Superuser Actions

Superuser actions let you move a document past workflow nodes where it may be held up because another user is unavailable to take action on it or due to a system problem. Superuser actions are an administrative tool and safety net. Superusers are designated with a KIM Permission (or they are listed in the Document Type, but this feature is deprecated).

Using the Superuser Functions

To use the Superuser functions, go to Document Search (it is a link in the left menu) and click the Superuser Search link.

Now, do a normal document search for the document on which you need to perform a Superuser action. When you find the document, click the Document/Notification Id link. If you are authorized to perform

a Superuser action on this document, this takes you to the Superuser page. Otherwise, KEW displays a message that you are not authorized to take Superuser action on this document.

Superuser Approve Document Action

This action fully approves a document, Deactivating all remaining routing requests. KEW also records a Superuser Approve Action Taken and associates it with the Deactivated requests. The document's state changes to Approved and the document is scheduled for processing. The document status automatically changes to Final when it goes into the Workflow Engine.

Superuser Approve Node Action

This action approves the document through all nodes up to (but not including) a specified node. When the document gets to the specified node, requests are created as usual. This action is exactly the same as Blanket Approve except that no notification requests are created from the Superuser Approve Node action.

Superuser Approve Request Action

This action approves a single pending Approve or Complete action request. This works the same way as a standard Approve action except that the Superuser is acting in place of the responsible user.

Superuser Cancel Action

This action cancels the document. This action works exactly the same way as the standard Cancel action except that the Superuser does not need to have a pending request to Cancel a document.

Superuser Disapprove Action

This action disapproves the document. This action works the same way as the standard Disapprove action except that the Superuser does not need to have a pending request to perform the action. Also, on a Superuser disapprove, KEW does not send notification requests.

Superuser Return to Previous Node Action

This action returns the document to a previous node in the route path. This action works exactly the same way as the standard Return to Previous Node action except that the Superuser does not need to have a pending request to perform the action.

Superuser Quick Action's Tab

With the release of Rice 2.1 a new option is available for applications to enable. Superuser's can now access documents directly from the standard document search and quickly perform approval, acknowledgment, FYI, or disapproval actions from this new tab on the document. This streamlined version of the still available superuser form of document search will allow for common actions to be taken in a more user friendly fashion. Pending actions are displayed in a list that the super user can check off and take the actions necessary all at once (see example below). Before actions can be taken, an annotation must be provided.

By default, the new tab will not appear on a document.

```
<kul:superUserActions />
```

This needs to be included in the JSP of a document for the superuser tab to be shown. The logic for whether it should actually display anything is contained inside of this. Developers may also look at `KualiMaintenanceDocument.jsp` in the Rice code for an example.

Figure 4.25. New Superuser Tab Example

The screenshot shows a web interface titled "Super User Action" with a "hide" button. Below the title is a table with the following data:

| Super User Action | | | | |
|--------------------------|---------|--------------|------------------------|------------|
| <input type="checkbox"/> | Action | Requested Of | Time/Date | Annotation |
| <input type="checkbox"/> | APPROVE | Two, Admin | 03/28/2012 05:52 PM | |

Below the table is a text area labeled "Annotation*" and two buttons: "take selected actions" and "disapprove document".

Documents

eDocLite Overview

eDocLite is a framework designed for rapid development and implementation within an existing Quali Enterprise Workflow infrastructure. It allows for the development of simple web applications, their forms and routing configurations using XML. Users only have to enter data into the form and then submit it. Rules can be constructed so that the form is then routed to a specific user or KIM Group based on the data entered.

Web pages called eDoc's are generated and are associated with a specific document type definition that provides the overall definition for how the document can be routed. Document types can also exist in hierarchies which provide storage of common information at various levels.

The form uses an XSLT stylesheet to generate the html code. Certain workflow classes make helper data available to the stylesheet programmer and there are several features that can be 'plugged-in' to eDocLite to further enhance its usability in many situations.

Note

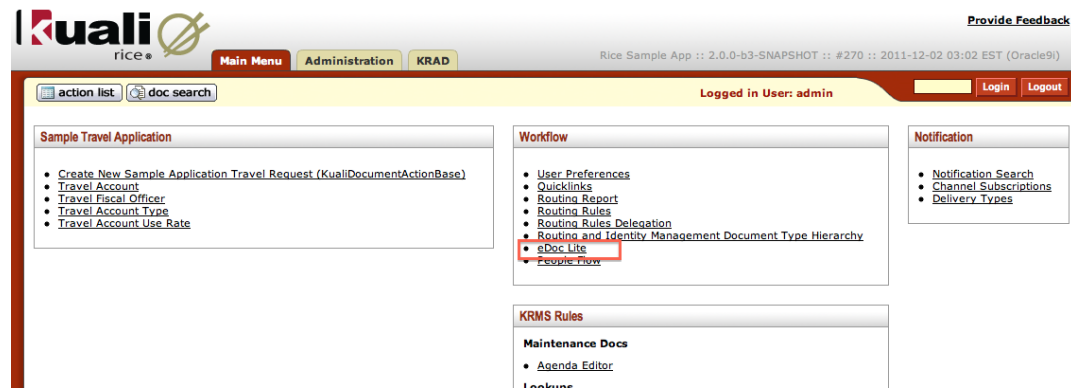
Key Ideas:

- Rapid implementation and development solution for simpler documents
- Easily re-configured
- Easily manageable
- Entirely web-based from design/development and user perspectives
- No java code required for developments; only XML with optional javascript for client side editing (workflow handles execution)
- Some validation javascript is automatically generated like regular expression editing and 'required field checking'.

eDocLite Lookup

Use the **eDocLite Lookup** screen to quickly find basic information about eDocLite documents and as the first step in creating a new eDocLite.

Figure 4.26. Workflow Channel: eDocLite Link



Finding the eDocLite Lookup Screen

You can go to the eDocLite Lookup by:

1. Click the Main Menu tab
2. Look in the Workflow section
3. Click eDoc Lite

eDocLite Lookup

Figure 4.27. eDocLite Lookup

EdocLite Lookup * required field

| | |
|--|--|
| Id: | <input type="text"/> |
| Document Type: | <input type="text"/> |
| Definition: | <input type="text"/> |
| Style: | <input type="text"/> |
| Active Indicator: | <input checked="" type="radio"/> Yes <input type="radio"/> No <input type="radio"/> Both |
| <input type="button" value="search"/> <input type="button" value="clear"/> <input type="button" value="cancel"/> | |

One item retrieved.

| Actions | Id | Document Type | Definition | Style | Active Indicator |
|---------------------------------|------|----------------------|--------------------|---------------------|------------------|
| Create Document | 2010 | eDoc.Example1Doctype | eDoc.Example1.Form | eDoc.Example1.Style | Yes |

Export options: [CSV](#) | [spreadsheet](#) | [XML](#)

On the eDocLite lookup page, users can search for an eDocLite document based on several criteria:

Table 4.2. eDocLite Lookup Attributes

| Field | Description |
|---------------|---|
| Id | The unique ID number assigned to each document. |
| Document Type | The name of the document type, which is specified in the Document Type attribute of an eDocLite. |
| Definition | The name of the EDL XML definition. |
| Style | The style specified in the EDL XML file is the style attribute of an eDocLite. Generally an EDL XML file has only one definition name which relates to one and only one style name. |

| Field | Description |
|------------------|---|
| Active Indicator | You have the choice of searching by the active status of an eDocLite. |

You can use the above criteria to limit your search results. A search will produce a list of one or more results that look similar to the following:

Figure 4.28. eDocLite Lookup: Search Results

| Actions | Id | Document Type | Definition | Style | Active Indicator |
|---------------------------------|----------------------|----------------------|----------------------|-----------------------|------------------|
| Create Document | 2010 | eDoc.Example1Doctype | eDoc.Example1.Form | eDoc.Example1.Style | Yes |
| Create Document | 2024 | InterviewRequest | InterviewRequestForm | InterviewRequestStyle | Yes |
| Create Document | 2027 | OfferRequest | OfferRequestForm | OfferRequestStyle | Yes |
| Create Document | 2030 | SearchStatus | SearchStatusForm | SearchStatusStyle | Yes |
| Create Document | 2033 | VacancyNotice | VacancyNoticeForm | VacancyNoticeStyle | Yes |
| Create Document | 2036 | WaiverRequest | WaiverRequestForm | WaiverRequest_xsl | Yes |

Clicking **Create Document** on any line takes you to the eDocLite document screen where new documents can be created. The line item you choose will result in that document being used as a template for the new one you are creating. More information on this follows in the section called **Create New eDocLite Document**.

Clicking any underlined Id will take you to the **eDocLite Inquiry** screen for that document.

Note

Exporting the output list to XML will give you the option of viewing the XML used to produce the list returned from the search.

Standard in KEW there is one eDocLite for electronic routing, and new eDocLites can be added based on business requirements. Some of the functions that eDocLites are used for in business include:

- Applicant Monitoring
- BLSC Request
- Course Change Request
- Grade Replacement Request
- Internship Contract
- Interview Request
- Mass Mailing Request
- Offer Request
- Program Plan Update
- REGR Access Request
- Removal Of Incomplete
- Revenue Producing Activity
- SUDS Request Document Type
- Search Status

- Special Credit Request
- Student Trip
- User Agreement
- Unit Change Request
- Vacancy Notice
- Vehicle Replacement
- Waiver Request
- New Course Request

eDocLite Inquiry

Figure 4.29. eDocLite Inquiry

The screenshot shows a window titled 'EdocLite' with a 'hide' button. Below the title bar is a table with the following data:

| | |
|--------------------------|----------------------|
| Id: | 2010 |
| Document Type: | eDoc.Example1DocType |
| Definition: | eDoc.Example1.Form |
| Style: | eDoc.Example1.Style |
| Active Indicator: | Yes |

At the bottom of the window are two buttons: 'export' and 'close'.

The Inquiry screen displays the same information as is found on a line of the Lookup output, but this screen provides you with the export option. Exporting from the Inquiry screen produces a XML file of the source for the eDocLite document.

Create New eDocLite Document

To create a new eDocLite document to be routed in KEW, click on **Create Document** in the row for eDocLite type wanted. It will take users to different forms of eDocLites depending on the document function, but they all have three general parts:

- Document header
- Document body
- Routing action and annotation, and note area

Document Header

The Document Header contains the following fields:

Table 4.3. Document Header Attributes

| Field | Description |
|-----------------|--|
| Document Type | The name defined by the document creator of this type of eDocLite. |
| Document Status | The status of this document based on its routing status. |
| Create Date | The date and time this document is created. |

| Field | Description |
|-------------|---|
| Document ID | The unique system generated ID for this document. |

Document body

This portion of the document is where the user identifies the routing and complete business function.

Table 4.4. Document Body Attributes

| Field | Description |
|-------|---|
| Title | specifies the actions users are taking on the EDocLite forms, including editing and reviewing . Other general information can be stored here such as contact information, important notes, etc. |
| Form | Renders form fields and input areas for the user to complete information required, depends upon the specific eDocLite requirements. |

Routing Action and Annotation, and Note area

This area is used to add annotation and choose action to be taken on this eDocLite document. Annotation is the comments associated with the routing process. You can add them to different nodes of the routing process and take actions on an eDocLite by adding annotations. The annotation appears in the route log of eDocLite as comments. Notes are the comments associated with this specific eDocLite form and appear with the form and not in the route log.

Table 4.5. Routing Action and Annotation, and Note Attributes

| Field | Description |
|----------------|---|
| Set annotation | The area to add annotation. |
| Action buttons | <ul style="list-style-type: none"> route: begins and continues routing the eDocLite document. save: saves the information currently on the eDocLite document. cancel: cancels the actions on this eDocLite document, and the information on the form is not saved. |
| create note | Area where users can add notes and attach documents to this eDocLite form. This part keeps track of Author, Date and time , and the Note added. Users have the right to add, edit and delete the notes they create. |

The following is one example of an eDocLite form:

Editing Document

** Questions with an asterisk are required.

| Indiana University EDL EDocLite Example | | eDocLite Example 1 Form | |
|---|--------------------------------------|---|-------------------------------------|
| User Information | | | |
| Full Name* | <input type="text"/> | | |
| Other Information | | | |
| Requested Date of Implementation:* | <input type="text"/> | | |
| Campus:* | IUB <input type="button" value="v"/> | | |
| Description of Request: | <input type="text"/> | | |
| (Check all that apply) | | | |
| <input type="checkbox"/> My research/sponsored program work is funded by NIH or NSF. | | | |
| <input type="checkbox"/> My research/sponsored program work involves human subjects. | | | |
| Supporting Materials | | | |
| Use the Create Note box below to attach supporting materials to your request. Notes may be added with or without attachments. Click the red 'save' button on the right. | | | |
| Create Note | | | |
| Author | Date | Note | Action |
| admin, admin | 12/02/2011 | <input type="text"/> | <input type="button" value="save"/> |
| Attachment: | | <input type="button" value="Choose File"/> no file selected | |
| <input type="button" value="submit"/> <input type="button" value="save"/> <input type="button" value="cancel"/> | | | |

Note

Notes that have been added to an eDocLite document can be edited or deleted.

Extendable functions

eDocLites are highly customizable. New eDocLites can be designed for new business and functional requirements. The document header and routing annotation and notes parts will be same or similar, the form will be different.

eDocLites can be designed to include following functions:

Restricted read/write rights

- Some fields in eDocLite can be set as **GlobalReadOnly**. With this setting they are disabled and can't be edited by any user other than the author.
- Some fields in eDocLite can be set as **ReadOnly**, but users with rights can still edit them. After the initiator writes them they are disabled and become locked to some of the users in the routing process. But for users with proper rights in certain nodes in the routing process, the fields will become editable again. These users can take actions on them, such as modify, add, and return to a former routing node.

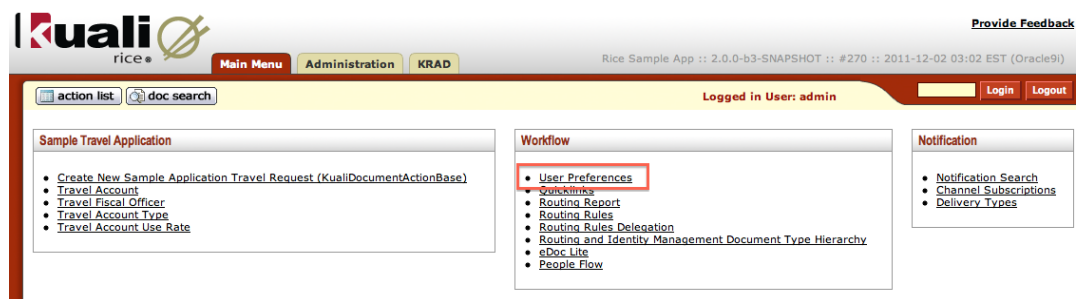
Hidden fields

To accommodate some business requirements, certain fields and notes can be hidden from certain nodes in the routing process. For instance, some administrative notes on a course waiver request will be hidden from students when s/he gets the eDocLite form in the final stage of the routing process.

Preferences

Workflow allows for users to individually configure certain aspects of the system. You should be able to access the User Preferences from the Main portal page:

Figure 4.30. Workflow Channel: User Preferences Link



After clicking the link you should be taken to Workflow Preferences screen:

Figure 4.31. Workflow Preferences

General

Automatic Refresh Rate: 15 in whole minutes - 0 is no automatic refresh.

Action List Page Size: 10

Delegator Filter: Secondary Delegates on Action List Page

Primary Delegate Filter: Primary Delegates on Action List Page

Fields Displayed In Action List

Document Type

Title

ActionRequested

Initiator

Delegator

Date Created

Date Approved

Current Route Node(s)

WorkGroup Request

Document Route Status

Clear FYI

Use Outbox

Document Route Status Colors for Actionlist Entries

| | | | | | | | | | | | | |
|-------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|
| Saved | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Initiated | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Disapproved | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Enroute | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Approved | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Final | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Processed | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Exception | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Canceled | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |

Email Notification Preferences

Receive Primary Delegate Emails

Receive Secondary Delegate Emails

Default Email Notification: Immediate

| Document Type | Notification Preference | Actions |
|----------------------|-------------------------|---------------------------------------|
| DocumentTypeDocument | Immediate | <input type="button" value="delete"/> |
| <input type="text"/> | None | <input type="button" value="add"/> |

Send Email Notifications For:

Complete

Approve

Acknowledge

FYI

save reset cancel

There are three configuration sections on this screen:

General Preferences

Note

Many of these preferences can have their system wide default values changed via configuration parameters. Look in the KEW technical documentation for information on how to override the default values.

Table 4.6. User Preferences Attributes

| Name | Default Value | Description |
|------------------------|---------------|---|
| Automatic Refresh Rate | 15 | How often your action list updates (minutes). |
| Action List Page Size | 10 | # of actions displayed on action list |

| Name | Default Value | Description |
|-------------------------------------|--|---|
| Email Notification | Immediate | When action items emails should be sent. Can be none, immediate, daily, weekly |
| Receive Primary Delegation Emails | True | User will receive primary delegate emails |
| Receive Secondary Delegation Emails | False | User will receive secondary delegate emails |
| Delegator Filter | Secondary Delegators on Action List Page | Determines what is displayed on the action list page. Options: <ul style="list-style-type: none"> • Secondary Delegators on Action List Page • Secondary Delegators only on Filter Page |
| Primary Delegate Filter | Primary Delegators on Action List Page | Determines what is displayed on the action list page. Options: <ul style="list-style-type: none"> • Primary Delegators on Action List Page • Primary Delegators only on Filter Page |

Fields Displayed in Action List Preferences

These are the columns that will be displayed on the user's action list page.

Table 4.7. User Preferences: Fields Displayed Attributes

| Name | Default Value |
|-----------------------------|---------------|
| Document Type | TRUE |
| Title | TRUE |
| ActionRequested | TRUE |
| Initiator | TRUE |
| Delegator | TRUE |
| Date Created | TRUE |
| Date Approved | FALSE |
| Current Route Node(s) | FALSE |
| Workgroup Request | TRUE |
| Document Route Status | TRUE |
| Application Document Status | FALSE |
| Clear FYI | TRUE |
| Use Outbox | TRUE |

Document Route Status Colors for Action list Entries

Each Document Route Status can be displayed with a certain color on the action list. This section allows you to configure which colors are used. By default, no colors are set.

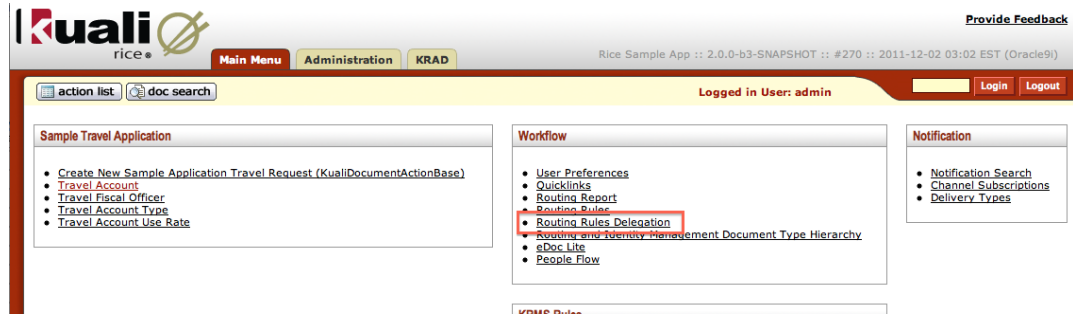
Routing

Please Note: The following sections on routing and rules are written as they relate the legacy/traditional KNS workflow. A new concept of workflows, called PeopleFlows, and their use and maintenance is covered in a separate section.

Delegate Rules and Delegation Routing Rules

Rice provides a quick and convenient way to obtain information on Delegation Rules through the **Delegate Rule Lookup**. This Lookup also lets you create new Delegation Rules using the **create new** function and edit existing Delegation Rules.

Figure 4.32. Workflow Channel: Routing Rules Delegation Link



To access the **Delegate Rule Lookup** screen, click **Routing Rules Delegation** on the Rice Main Menu.

Figure 4.33. Delegation Lookup

The screenshot shows the 'Delegate Rule Lookup' form. At the top left, there is a 'Delegate Rule Lookup' label with a help icon. At the top right, there is a 'create new' button and a note '* required field'. The form contains the following fields and options:

- Parent Rule Id:
- Responsibility Id:
- Document Type:
- Rule Template:
- Description:
- Group Reviewer Namespace:
- Group Reviewer Name:
- Person Reviewer:
- Person Reviewer Type: User Group Member Both
- Active: Yes No Both
- Rule Id:
- Delegation Type: Primary Secondary Both

At the bottom of the form, there are three buttons: 'search', 'clear', and 'cancel'.

Use the **Delegate Rule Lookup** screen to find information about a Delegation Rule. You can also create a new Rule by clicking the create new button in the upper right corner of the screen. The create new function is described in the Routing Rule Creation section later in this document.

When you do a search from the Delegate Rule Lookup screen, you can:

- View the Delegation Rule information returned by the search
- Edit a Delegation Rule
- Copy an existing Delegation Rule and create a new one based on that copy

Information about the Lookup function is in the next section of this document. The other two functions are described in the Routing Rule Delegation Maintenance section later in this document.

To find a Delegation Rule quickly, enter information in one or more of the fields on the Lookup screen, then click the **search** button. If you are not familiar with the search fields, you can leave all of the fields blank and click the **search** button. The **Delegate Rule Lookup** page then displays all of the Delegation Rule templates.

The fields on this screen are described in the Routing Rule Delegation Maintenance section of this document.

Note

Parent Rule Id is a unique identification number for a Rule Template; therefore, searching by the Parent Rule Id can be the fastest way to find a particular Rule.

After you click the search button, Delegate Rule Lookup displays a list of Rules that looks similar to this:

Figure 4.34. Delegation Lookup: Results Example

| Actions | Delegation Rule Id | Parent Rule Id | Name | Rule Template | Active | Document Type | Delegation Type |
|---|--------------------|----------------|--------------------------------------|------------------------------------|--------|---------------|-----------------|
| edit copy | 1641 | 1046 | A4CAF5AA-5352-5EC8-D2CA-977FD066FD50 | WorkflowDocumentDelegationTemplate | Yes | TravelRequest | Secondary |

Click **edit** on any row to go to the **Routing Rule Delegation** document for that Rule. You can use this document to do maintenance on that Rule. This is discussed later in this document, in the Routing Rule Delegation Maintenance section.

Click **copy** on any row to go to the **Routing Rule Delegation** document with the information from that Rule copied into the document. This lets you add a new Delegation Rule based on that information. This is discussed later in this document in the Routing Rule Delegation Maintenance section.

Click a **Delegation Rule Id** to go to the **Delegation Rule Inquiry** screen for that Rule. This screen displays specific information about how this Rule is set up in Rice. This screen is discussed later in this document in the **Delegation Rule Inquiry** section.

Click a Parent Rule Id or a Name to go to the **Rule Inquiry** screen, where you can see details about that Parent Rule ID. More information about this screen can be found in the Rule Inquiry section of this User Guide.

Click a Rule Template to go to the **Rule Template Inquiry** screen. More information about this screen can be found in the Rule Template Inquiry section of this User Guide.

The standard export options apply to this list of search results. An explanation of the export options can be found in the Common Features and Functions section of this User Guide.

Delegation Rule Inquiry

Figure 4.35. Delegation Inquiry

The screenshot shows the 'Delegation Rule Inquiry' screen. It features several sections with 'hide' buttons:

- Delegation Rule:** Contains fields for Delegation Rule Id (1641), Parent Rule Id (1046), Responsibility Id (2022), Document Type (TravelRequest), Rule Template (WorkflowDocumentDelegationTemplate), Description (User4 Destination Routing Secondary Delegation), From Date, To Date, Force Action (Yes), Active (Yes), and Delegation Type (Secondary).
- Person Responsibilities:** Shows Person (user2), Action Request, and Priority (0).
- Group Responsibilities:** (Empty section)
- Role Responsibilities:** (Empty section)

At the bottom of the screen are 'export' and 'close' buttons.

This screen provides specific information on how a **Rule** works. From this screen, you can either click the **close** button to return to the previous screen or you can export an XML version of this information.

The fields on this screen are described in the Rule Maintenance section of this document.

Delegate Routing Rule Creation

Figure 4.36. Delegation Rule: Create New Screen

To create a new Delegate Routing Rule, start by selecting a Parent Rule to associate with your new Delegate Rule:

1. Do a field lookup on the Select parent rule field.
2. This displays the **Rule Lookup** screen where you search in the Rules your institution has defined in Rice to find the appropriate Parent Rule for your new Delegate Routing Rule.
3. When you have found the appropriate Parent Rule, click its return value link to return to the Delegate Routing Rule Creation screen with this Parent Rule.

Figure 4.37. Delegate Routing Rule: Create New, Parent Selection

| Select | Reviewer | Type | Action Request Code |
|----------------------------------|----------|--------|---------------------|
| <input checked="" type="radio"/> | admin | PERSON | APPROVE |

4. Information from the Parent Rule you selected is now in the **Select Parent Responsibility to Delegate From** section of the **Delegate Routing Rule Creation** screen.
5. Click the **Select** button for this parent Rule.
6. Click the **continue** button.
7. Rice displays the **Routing Rule Delegation** screen. Use this screen to enter information for your new Rule. This is described in the **Routing Rule Delegation Maintenance** section (next).

Routing Rule Delegation Maintenance

You can use the **Routing Rule Delegation** document screen for both editing existing Delegation Rules and for adding new Delegation Rules to Rice.

Figure 4.38. Routing Rule Delegation: Overview

| Routing Rule Delegation | | Doc Nbr: 3007 | Status: INITIATED |
|-------------------------|--|------------------|------------------------------|
| | | Initiator: admin | Created: 06:59 PM 07/30/2009 |

- Document Overview
- Delegation Details
- Delegate Rule
- Persons
- Groups
- Notes and Attachments (0)
- Ad Hoc Recipients
- Route Log

submit save blanket approve close cancel

expand all collapse all
* required field

Note

When Rice first displays the **Routing Rule Delegation** screen, all of the tabs except the Notes and Attachments, Ad Hoc Recipients, and Route Log tabs, are expanded. They are all collapsed in the screen print above so you can see all of them.

Document Layout

The Routing Rule Delegation screen you see when creating a new Rule has eight tabs:

- Document Overview
- Delegation Details
- Delegate Rule
- Persons
- Groups
- Notes and Attachments
- Ad Hoc Recipients
- Route Log

The **Delegation Details**, **Delegate Rule**, **Persons**, and **Groups** tabs are discussed in the sections that follow. The remaining four tabs are commonly used throughout Rice and are described in the Common Features and Functions section of the User Guide.

Delegation Details Tab

The Delegation Details tab is where you associate a Parent Rule with a Delegate Rule. It is also where you assign the Reviewer role.

The Delegation Details screen is displayed differently, depending on whether you are creating a new Rule or editing/copying a Rule. The version you see when creating a new Rule looks similar to this:

Figure 4.39. Routing Rule Delegation: Details Section

The screenshot shows a 'Delegation Details' window with a 'hide' button. The form is titled 'New' and contains the following fields:

| | |
|-----------------------------------|--------------------------------------|
| Parent Rule | |
| Parent Rule Id: | 1034 |
| Description: | workflowdocumenttemplate description |
| Parent Rule Responsibility | |
| Reviewer: | admin |
| Type: | PERSON |
| Action Request: | APPROVE |

The information displayed is a copy of the information for the Parent Rule. Descriptions of these fields are below. Change this information as needed, then click the **submit** button to create this new Rule.

The version you see when editing or copying a Rule looks similar to this:

Figure 4.40. Routing Rules Delegation: Edit/Copy View

The screenshot shows a 'Delegation Details' window with a 'hide' button. The form is split into two columns: 'Old' and 'New'. Both columns contain the same information as Figure 4.39:

| Old | | New | |
|-----------------------------------|------------------|-----------------------------------|------------------|
| Parent Rule | | Parent Rule | |
| Parent Rule Id: | 1046 | Parent Rule Id: | 1046 |
| Description: | Destination Rule | Description: | Destination Rule |
| Parent Rule Responsibility | | Parent Rule Responsibility | |
| Reviewer: | user4 | Reviewer: | user4 |
| Type: | PERSON | Type: | PERSON |
| Action Request: | APPROVE | Action Request: | APPROVE |

When you first see this version, the information in both columns is the same. You can modify the information in the fields in the right column, but not the left column. After changing the information appropriately in the right column, click the submit button to save your changes to this Rule.

The fields are the same in both versions of the screen:

Table 4.8. Routing Rules Delegation Attributes

| Field | Description |
|----------------|---|
| Parent Rule Id | Rule Id of the Parent Rule for this Rule |
| Description | General description of this Rule |
| Reviewer | The person responsible for reviewing a document in the routing process for this Rule |
| Type | The Type of the Reviewer |
| Action Request | The type of Action this Reviewer can take. This may be: <ul style="list-style-type: none"> • Acknowledge • Complete • Approve • FYI |

Delegate Rule tab

The Delegate Rule tab is displayed differently, depending on whether you are creating a new Rule or editing/copying a Rule. The version you see when creating a new Rule looks similar to this:

Figure 4.41. Routing Rules Delegation: Delegate Rule Tab

Enter the appropriate information for your new Delegate Rule in these fields. There are descriptions of each of these fields below.

The version you see when editing or copying a Rule looks similar to this:

Figure 4.42. Routing Rule Delegation: Delegate Rule Tab, Edit/Copy View

When you first see this version, the information in both columns is the same. You can modify the information in the fields in the right column, but not the left column. After changing the information in the right column appropriately.

The fields are the same in both versions of the screen:

Table 4.9. Routing Rule Delegation: Delegate Rule Tab Attributes

| Field | Description |
|-----------------|---|
| Document Type | The Document Type defines the routing definition and other properties for a set of documents. Each document is an instance of a Document Type and conducts the same type of business transaction as other instances of that Document Type. |
| Rule Template | A Rule Template serves as a pattern or design for routing Rules. |
| Delegation Type | Required. Options for Delegation Type are: <ul style="list-style-type: none"> • Primary: This means that the delegator turns over full authority to the delegate. The Action Requests for this delegator only show up in the Action List of the primary delegate. A primary delegate must be registered in KEW to be in effect. • Secondary: This means that the secondary delegate acts as a temporary backup delegator who has the same authority as the primary approver, or the secondary delegate acts as the delegator if the delegator is not available. Documents appear in the Action Lists of both the delegator and the secondary delegate. When either person acts on the document, it disappears from both Action Lists. A secondary delegate must be registered in KEW as being in effect for a specific time period(s). • Both |
| Name | he unique Name of the routing delegate Rule you are creating or editing. |

| Field | Description |
|--------------|--|
| | Note: You are not required to enter a Name, but if you don't, Rice assigns a unique Name to the Rule. |
| Description | The general description of the Rule |
| From Date | The start date of the routing Rule |
| To Date | The expiration date of the routing Rule |
| Force Action | This is a Yes or No flag. A checkmark means Yes, blank means No. |
| Active | A True/False flag to indicate whether this Routing Delegate Rule is active in Rice. If it is active, you can use it in Rice. |

Persons tab

Rice displays the **Persons** tab differently, depending on whether you are creating a new Rule or editing/copying a Rule. This tab lets you add people to the Rule and set the type of Action each person needs to take under this Rule. The version of the Persons screen that you see when creating a new Delegate Rule looks similar to this:

Figure 4.43. Routing Rules Delegation: Persons Tab

Enter the appropriate information for the Persons section of your new Delegate Rule in these fields. There are descriptions of each of these fields below.

The version you see when editing or copying a Rule looks similar to this:

Figure 4.44. Routing Rules Delegation: Persons tab, Copy/Edit View

Use the top portion of this tab to add a new Person to this Rule and set the Action that Person needs to take and its Priority. Use the lower portion of this tab to change existing values for this Rule.

Note

When you create a new person, you must click the **add** button when you have entered information in the three required fields.

Table 4.10. Routing Rules Delegation: Persons Tab Attributes

| Field | Description |
|----------------|--|
| Person | Required. An individual user who receives the document for the Action Requested. |
| Action Request | Required. The type of Action this person needs to take. This Action may be one of these: |

| Field | Description |
|----------|---|
| | <ul style="list-style-type: none"> • Acknowledge • Complete • Approve • FYI |
| Priority | Required. The priority associated with the Person in this Rule. One is the highest Priority. |

Groups tab

The Groups tab differently, depending on whether you are creating a new Rule or editing/copying a Rule. The version you see when creating a new Rule looks similar to this:

Figure 4.45. Routing Rules Delegation: Groups Tab

Enter the appropriate information for the Groups section of your new Delegate Rule in these fields. These fields are described below.

The version you see when editing or copying a Rule looks similar to this:

Figure 4.46. Routing Rules Delegation: Groups Tab, Edit/Copy View

Use the top portion of this tab to add a new Group. Use the lower portion to change existing information for a Group.

Note

When you create a new group, you must click the **add** button after you enter information in the required fields.

Table 4.11. Routing Rules Deletion: Groups Tab Attributes

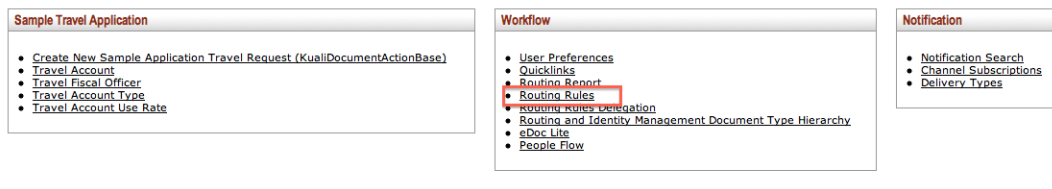
| Field | Description |
|-----------|---|
| Namespace | The Namespace to which this Group will belong. A Namespace is a way to set both Permissions and Entity Attributes. Each Namespace instance is one level of Permissions and Entity Attributes and is one record in Rice. |
| Name | The unique name of this Group |

| Field | Description |
|----------------|--|
| Action Request | Required. The type of Action this Group is expected to take. The Action may be one of these: <ul style="list-style-type: none"> Acknowledge Complete Approve FYI |
| Priority | Required. The priority associated with the Group in this Rule. One is the highest Priority. |

Rule Lookup

The Rice system provides a quick and convenient way to obtain information on Rules through the **Rule Lookup** feature. This feature also provides an entry point to creating a new Rule through the **create new** function.

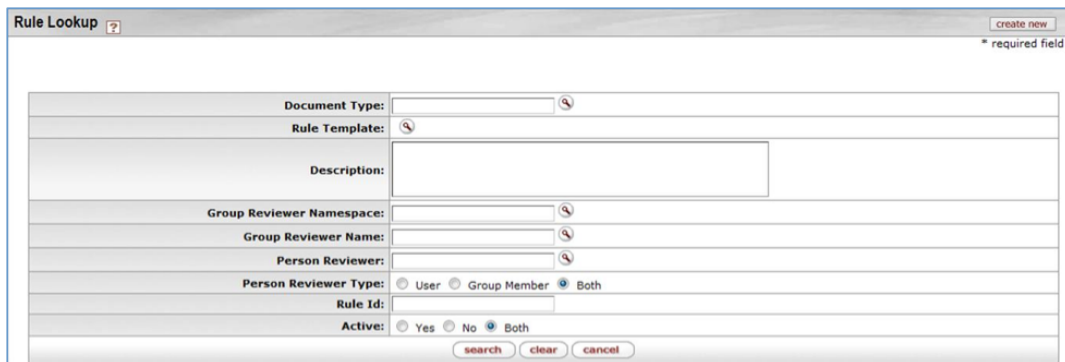
Figure 4.47. Workflow Channel: Routing Rules Link



To access the **Rule Lookup** screen click on **Routing Rules** on the Rice **Main Menu**.

The **Rule Lookup** screen is used to view, edit or copy information about routing rules defined in Rice. It is also the starting point for creating a new rule using the **Routing Rule Creation** screen.

Figure 4.48. Routing Rules Lookup



Click on the **create new** button to go to the Routing Rule Creation screen. The process for creating a new Routing Rule is discussed in the **Routing Rule Creation** section of this document.

To limit your search results you can select one or more of the available search criteria.

The fields on this screen are described in the **Rule Maintenance** section of this document.

Note

Rule Id is a unique identification number for a particular rule template; therefore, search by rule Id can produce the most accurate search result.

If you are not familiar with the search criteria provided above you can simply click on the **search** button and the Rule Lookup page will automatically provide a list of rule templates.

Rule Lookups return a list that looks similar to the following:

Figure 4.49. Routing Rules Lookup: Results Example

| Actions | Name | Rule Id | Rule Template | Active | Document Type | Delegate Rule |
|---|------------------------------------|---------|----------------------------------|--------|---------------------------------|---------------|
| edit copy | SendNotificationRequest.Reviewers | 1044 | ReviewersRouting | Yes | SendNotificationRequest | No |
| edit copy | TravelRequest.Destination.LasVegas | 1046 | TravelRequest-DestinationRouting | Yes | TravelRequest | No |
| edit copy | TravelRequest.Supervisor | 1049 | TravelRequest-TravelerRouting | Yes | TravelRequest | No |
| edit copy | TravelRequest.DeanDirector | 1050 | TravelRequest-SupervisorRouting | Yes | TravelRequest | No |
| edit copy | TravelRequest.FiscalOfficer | 1051 | TravelRequest-SupervisorRouting | Yes | TravelRequest | No |
| edit copy | 659D86718DD514C7E0404F8189D877C3 | 1052 | TravelRequest-AccountRouting | Yes | TravelRequest | No |
| edit copy | eDoc.Example1Doctype.IUB | 1103 | eDoc.Example1.Node1 | Yes | eDoc.Example1Doctype | No |
| edit copy | eDoc.Example1Doctype.IUPUI | 1106 | eDoc.Example1.Node1 | Yes | eDoc.Example1Doctype | No |
| | RecipeMastersGroupApproval | 1642 | | Yes | RecipeParentMaintenanceDocument | No |
| | ChickenRecipeMastersGroupApproval | 1643 | | Yes | RecipeMaintenanceDocument | No |
| | MagazineGroupApproval | 1644 | | Yes | RecipeMaintenanceDocument | No |

Clicking **edit** on any line item takes you to the **Rule Maintenance Document Type Document** where you can perform maintenance on that rule. This is discussed later in this document in the Rule Maintenance section.

Clicking **copy** on any line item takes you to the **Rule Maintenance Document Type Document** where you can add a new rule based upon the rule you are copying. This is discussed later in this document in the Rule Maintenance section.

Clicking a **Rule Id** will take you to the Rule Inquiry screen for that specific rule. This is discussed later in this document in the **Rule Inquiry** section.

Clicking a **Document Type** will take you to the **Document Type Inquiry** screen for that specific rule. This is discussed in the **Document Type Inquiry** section of the User Guide.

The standard export options apply to this list. Explanation of the export options can be found in the **Common Features and Functions** section of the User Guide.

Rule Inquiry

Figure 4.50. Routing Rules Inquiry

This screen provides specific information on how the **Rule** is set up in Rice. The fields on this screen are described in the **Rule Maintenance** section of this document.

Routing Rule Creation

Figure 4.51. Routing Rules Creation

When creating a new Routing Rule you start by entering the **Document Type** and the **Rule Template** you want associated with your new Rule, then click the continue button. This takes you to the **Rule Maintenance Document Type Document** described in the **Rule Maintenance** section that follows.

Table 4.12. Routing Rule Creation Attributes

| Field | Description |
|---------------|--|
| Document Type | You can select a new document type by clicking on the field lookup button located at the right side of the Doc Type field. |
| Rule Template | The name of the Rule Template with which this rule is associated. A Rule Template serves as a pattern or design for the routing rules and is a mechanism for providing configuration information to rules. You can select a new template by clicking on the field lookup button located at right side of the Rule Template field. |

Routing Report

The routing report is a tool to check workflow routing. You select a rule template and enter any routing data required by the rule template. You can then view the workflow routing that would be assigned to such a document.

Figure 4.52. Workflow Channel: Routing report

Using the Report

In the default implementation of Rice you can find the link to the Routing Report on the "Main" tab of the portal inside the "Workflow" pane. If for some reason the link is not visible on the "Main" tab you can still directly navigate to the Routing Report. To access the tool directly, replace the last part of the Rice Portal URL (**portal.jsp**) with **kew/RoutingReport.do** (for example, **http://my.kuali.host/kuali/portal.jsp** would become **http://my.kuali.host/kuali/kew/RoutingReport.do**). The following page is displayed:

Figure 4.53. Routing Report: Template Selection

The screenshot shows a web form titled "Routing Report". Below the title is a label "Select A Rule Template" followed by a dropdown menu. The dropdown menu is currently empty and displays the text "Please select a template".

Select a Rule Template

The first thing to do is select the rule template to use for the test from the drop-down list. When you select a template the page will update to show the routing data that you need to enter to run the test:

Figure 4.54. Routing Report: Template Selection, Detail

The screenshot shows the "Routing Report" form with the following details:

- Select A Rule Template:** eDoc.Example1.Node1
- Enter Routing Data:**
 - Effective Date:** 12/02/2011
 - Time:** 5 :00 PM
 - Document Type:** (empty)
 - Campus:** (empty)
- view results** button

Enter Routing Data

The data entry fields for the routing data use the standard controls described in the Standard Buttons on Lookup Screens document. Here is the example with the routing data:

Figure 4.55. Routing Report: Routing Data Entry

The screenshot shows the "Routing Report" form with the following details:

- Select A Rule Template:** eDoc.Example1.Node1
- Enter Routing Data:**
 - Effective Date:** 12/02/2011
 - Time:** 5 :00 PM
 - Document Type:** eDoc.Example1.Doctype
 - Campus:** IUB
- view results** button

The specific information you need to enter is specific to the rule template you selected. Once you have entered the information, click the button to display the report:

Figure 4.56. Routing Report: Route Log View

The screenshot shows the "Route Log" view with the following data:

| ID: hide | | | |
|-----------------------|------------------------------------|----------------------|--|
| Title | Routing Report | | |
| Type | eDoc.Example1.Request.DocumentType | Created | |
| Initiator | admin_admin | Last Modified | |
| Route Status | INITIATED | Last Approved | |
| Node(s) | Routing Report | Finalized | |

| Pending Action Requests hide | | | | |
|---|-----------------|--|---------------------|------------|
| | Action | Requested Of | Time/Date | Annotation |
| show | PENDING APPROVE | eDoc.Example1.IUB.Workgroup (Secondary Delegate) | 09:19 PM 12/02/2011 | |

Report Contents

The upper section of the report (labeled “ID: 0”) contains the basic document information fields as if this was an actual document; several fields are blank since there is no actual document.

The lower section of the report (labeled “Pending Action Requests”) contains the routing that the document would be given. Clicking the button on any action displays more detail of the action:

Figure 4.57. Routing Report: Route Log View, Pending Action Requests

| Node(s) | | Routing Report | Finalized |
|---|--|---------------------|------------|
| Pending Action Requests hide | | | |
| Action | Requested Of | Time/Date | Annotation |
| hide PENDING APPROVE | eDoc.Example1.IUB.Workaroup (Secondary Delegate) | 09:19 PM 12/02/2011 | |
| Node | eDoc.Example1.Node1 | | |
| Priority | 1 | | |
| Approval Policy | Single Approval Required | | |
| Forced Action | false | | |
| Rule | 1103 | | |

Rule Maintenance

The **Rule Maintenance Document Type Document** screen is used for both editing existing Rules and for adding new Rules to Rice.

Figure 4.58. Rule Maintenance Document Type Document

| | | |
|---|-------------------|------------------------------|
| Rule Maintenance Document Type Document ? | Doc Nbr: 3069 | Status: INITIATED |
| | Initiator: admin | Created: 09:25 PM 12/02/2011 |
| expand all collapse all * required field | | |
| Document Overview | show | |
| Rule | show | |
| Rule Attributes | show | |
| Persons | show | |
| Groups | show | |
| Notes and Attachments (0) | show | |
| Ad Hoc Recipients | show | |
| Route Log | show | |
| submit save blanket approve close cancel | | |

Note

When Rice presents the **Rule Maintenance Document Type Document** screen all of the tabs except the Notes and Attachments, Ad Hoc Recipients and Route Log are expanded.

Document Layout

The Rule Maintenance Document Type Document screen you see when creating a new rule is composed of seven tabs:

- Document Overview
- Rule
- Persons
- Groups

- Notes and Attachments
- Ad Hoc Recipients
- Route Log

The screen you see when you are editing an existing Rule includes an additional tab called **Roles**.

The Rule, Persons, Groups and Roles tabs are unique and are discussed in the sections that follow. The remaining four tabs are commonly used throughout Rice and are described in the Common Features and Functions section of the User Guide.

Rule tab

The Rule tab is presented differently depending on whether you are creating a new Rule, or editing/copying a Rule. The version you see when creating a new Rule looks similar to this:

Figure 4.59. Rule Maintenance Document Type Document: Rule Tab

The version you see when editing or copying looks similar to this:

Figure 4.60. Rule Maintenance Document Type Document: Rule Tab, Edit/Copy View

Table 4.13. Rule Maintenance Document Type Document: Rule Tab Attributes

| Field | Description |
|---------------|--|
| Document Type | The Document Type defines the routing definition and other properties for a set of documents. Each document is an instance of a Document Type and conducts the same type of business transaction as other instances of that Document Type. |
| Rule Template | The name of the Rule Template with which this rule is associated. A Rule Template serves as a pattern or design for the routing rules and is a mechanism for providing configuration information to rules. |
| Name | The unique Name of the routing Name you are creating or editing. Note: It is not required that you enter a Name, but if you don't the system will assign a unique value. |

| Field | Description |
|--------------|---|
| Description | The general description for the rule selected. |
| From Date | The start date of the routing rule. |
| To Date | The expiration date of the routing rule. |
| Force Action | Force Action is used to control workflow routing by making the rule actions mandatory. This is a yes or no flag. A check means yes, blank means no. |
| Active | A true/false flag to indicate if Routing Rule is active in Rice can be applied or not. |

Persons tab

The Persons tab is presented differently depending on whether you are creating a new Rule, or editing/copying a Rule. The version you see when creating a new Rule looks similar to this:

Figure 4.61. Rule Maintenance Document Type Document: Person Tab

The version you see when editing or copying looks similar to this:

Figure 4.62. Rule Maintenance Document Type Document: Person Tab, Edit/Copy View

Table 4.14. Rule Maintenance Document Type Document: Person Tab Attributes

| Field | Description |
|----------------|--|
| Person | An individual user who receives the document for the Action Requested. |
| Action Request | The type of action this person can take. <ul style="list-style-type: none"> Acknowledge Complete Approve FYI |
| Priority | The priority associated with the person in this rule. Lower values mean higher priorities, with 1 being the highest priority. |

Groups tab

The Groups tab is presented differently depending on whether you are creating a new Rule, or editing/copying a Rule. The version you see when creating a new Rule looks similar to this:

Figure 4.63. Rule Maintenance Document Type Document: Groups Tab

The version you see when editing or copying looks similar to this:

Figure 4.64. Rule Maintenance Document Type Document: Groups Tab, Edit/Copy View

The top portion of this tab is used to add a new Group while the lower portion is used to change existing values.

Table 4.15. Rule Maintenance Document Type Document: Groups Tab Attributes

| Field | Description |
|----------------|---|
| Namespace | The Namespace to which this Group will belong. A Namespace is a way to scope both Permissions and Entity Attributes. Each Namespace instance is one level of scoping and is one record in the system. |
| Name | The unique name of the Group. |
| Action Request | The type of action this Group is expected to take. <ul style="list-style-type: none"> • Acknowledge • Complete • Approve • FYI |
| Priority | The priority associated with the Group in this rule. One being the highest. |

Rule Template Lookup

The **Rules Template Lookup** function is available from any screen with the field lookup button next to the Rule Template field. Simply click on the button and you will go a screen that looks like this.

Figure 4.65. Rule Template Lookup

The **Rule Template Lookup** page will automatically provide a complete list of the rule templates. You can reduce the number of items returned by your search by entering values in the fields, or if you are not sure or familiar with the key word(s) for the search criteria, simply click on the search button.

Figure 4.66. Rule Template Lookup: Results Example

| Return Value | Id | Name | Description | Delegation Rule Template |
|------------------------------|------|------------------------------------|------------------------------------|--------------------------|
| return_value | 1015 | WorkflowDocumentDelegationTemplate | WorkflowDocumentDelegationTemplate | |
| return_value | 1016 | WorkflowDocumentTemplate | Workflow Document Template | ... |
| return_value | 1017 | Ack1Template | Acknowledgement 1 Template | |
| return_value | 1018 | Ack2Template | Acknowledgement 2 Template | |
| return_value | 1019 | RemoveReplaceWorkgroupTemplate | Remove/Replace Workgroup Routing | |
| return_value | 1021 | RemoveReplaceRuleTemplate | Remove/Replace Rule Routing | |
| return_value | 1023 | ReviewersRouting | Routes to channel reviewers | |

Clicking on the **Id** field will take you to the **Rule Template Inquiry** screen for that particular Rule Template.

Table 4.16. Rule Template Lookup Attributes

| Field | Description |
|--------------------------|--|
| Return Value | This is a link which will copy the Rule Template information back to the screen which you originated from. |
| Id | A unique Id number for a specific rule template. |
| Name | The name of the rule template. |
| Description | A general description of the rule template. |
| Delegation Rule Template | The delegate rule template selected. |

Rule Template Inquiry

The **Rule Template Inquiry** screen provides specific information about one particular **Rule Template**.

Figure 4.67. Rule Template Inquiry

Clicking the export button will export a copy of the complete Rule Template in XML format.

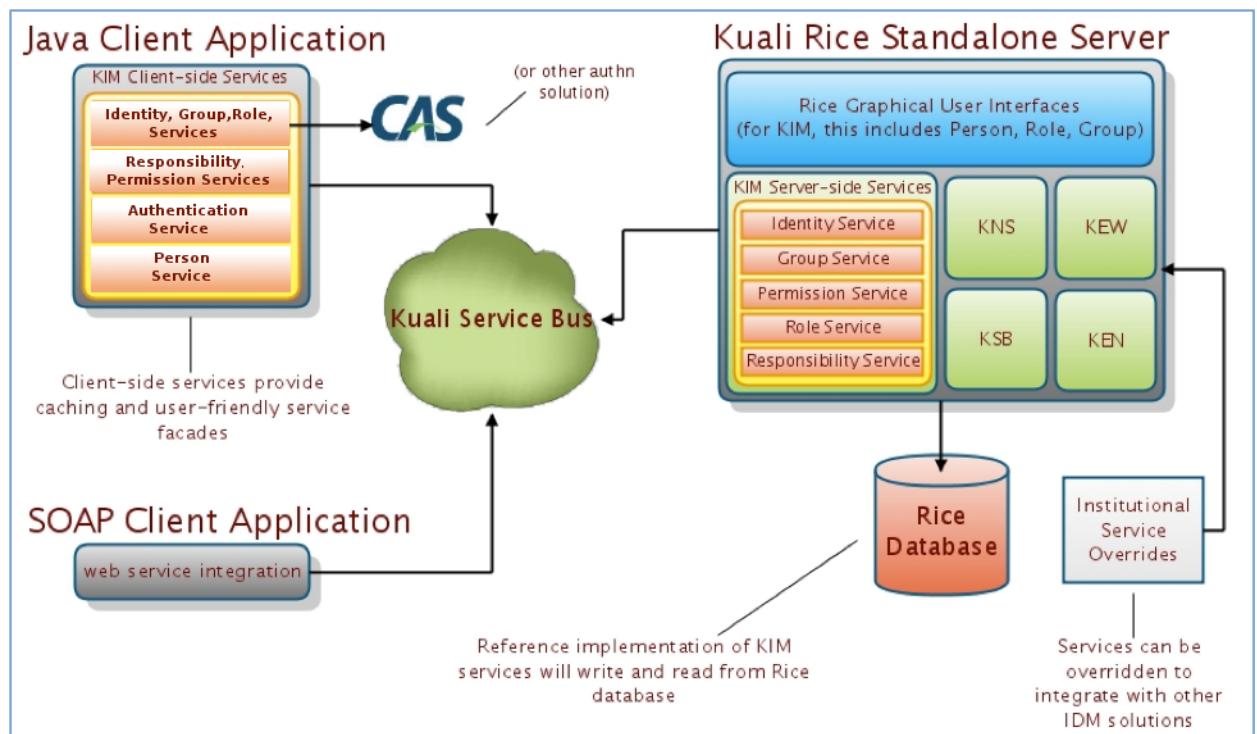
Chapter 5. KIM

KIM Overview

Kuali Identity Management (KIM) provides identity and access management services to Rice and other applications. All KIM services are available on the service bus with both SOAP endpoints. KIM provides a service layer and a set of GUIs that you can use to maintain the identity information.

KIM is designed to be used with both Kuali and non-Kuali applications. The permissions and responsibilities it provides are defined by each user's Role or Roles in the system. Roles can be customized to handle permissions and responsibilities in a variety of ways based on your particular needs.

Figure 5.1. KIM Architecture



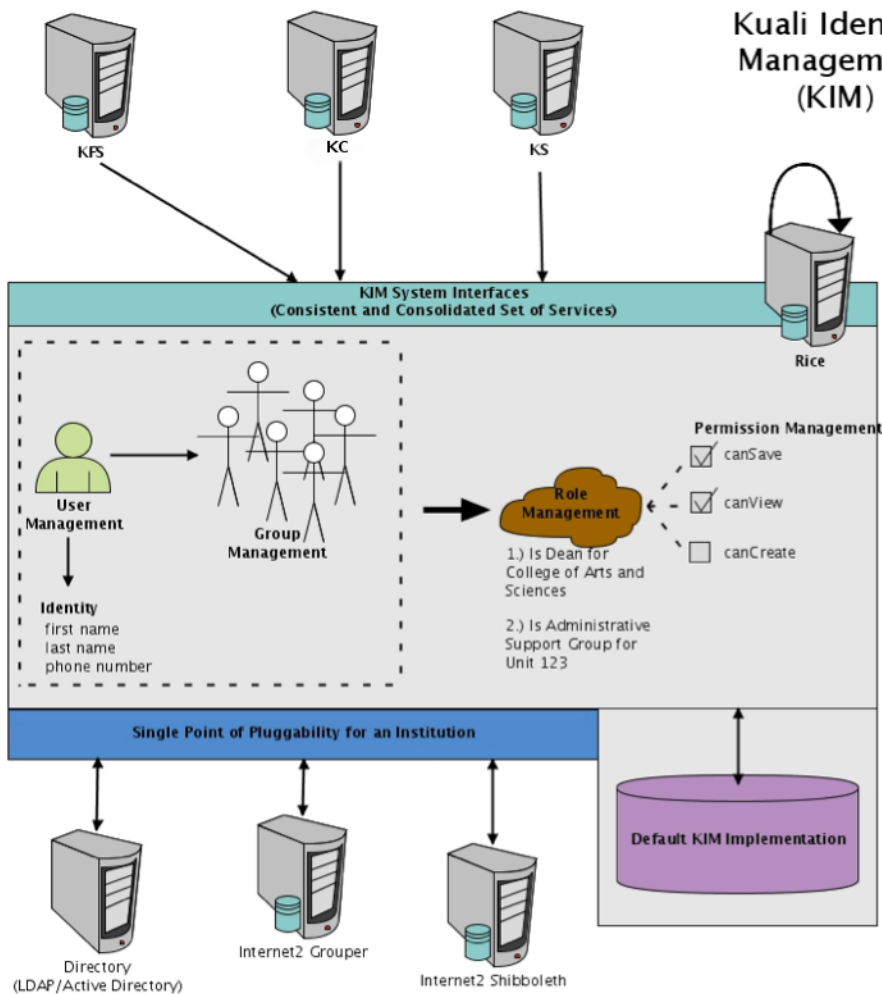
KIM Features

- KIM provides a reference implementation of the services but allows for customization and/or replacement to facilitate integration with institutional services or other third party identity management solutions.
- KIM allows you to override its core services.
- KIM consists of these services, which encompass its API:
 - AuthenticationService
 - GroupService

- IdentityService
- PermissionService
- PersonService
- ResponsibilityService
- RoleService
- KimTypeInfoService
- KIM evaluates permissions through its permission service. KIM provides plug points for implementing custom logic for permission checking, such as permission checks based on hierarchical data.

A more detailed picture of the KIM architecture:

Figure 5.2. Detailed KIM Achitecture



Person

Person Lookup

Use the **Person Lookup** screen to quickly find basic information about Persons in Quali Rice (Rice). (In Quali, a Person is a set of information about a real person or something that stands for real people, like a job title.) From the Person Lookup screen, you can also link to screens where you can create new Persons and edit a Person's information if you have permission to do so.

Finding the Person Lookup Screen

You can go to the **Person Lookup** screen from many Quali screens by clicking the field search button next to a Name-related field. How you get to the Person Lookup screen and your **Role** in Quali determine what you see and what you can do there.

If you have permission to use the Administration screens in Rice, you can go directly to the **Person Lookup** screen:

1. Click the **Administration** tab
2. Look under **Identity**
3. Click **Person**

Person Lookup Options

From the Field Search Button

If you clicked the field search button on another Quali page to get to the Person Lookup screen, your screen looks like this:

Figure 5.3. Person Lookup

The screenshot shows the 'Person Lookup' interface. At the top left is the title 'Person Lookup' with a help icon. At the top right is a note '* required field'. The main area contains a list of search criteria, each with a text input field and a search icon (magnifying glass):

- Principal Name
- Principal ID
- Entity ID
- First Name
- Middle Name
- Last Name
- Email Address
- Phone Number
- Employee ID
- Campus Code
- Primary Department Code
- Employee Status Code
- Employee Type Code
- Active Indicator: Yes No Both

At the bottom of the form are three buttons: 'search', 'clear', and 'cancel'.

To find a Person, enter some information about that person in the appropriate fields and click the **search** button at the bottom of the screen.

- You can display a list of all Persons in KIM if you leave the search fields empty and set the **Active Indicator** to **Both**, then click the search button.
- Rice displays a list of Persons who match your search information. The list is just below the search fields on the Person Lookup page:

Figure 5.4. Person Lookup: Results

One item retrieved.

[return with no value](#)

| <u>Return Value</u> | <u>Principal ID</u> | <u>Principal Name</u> | <u>Name</u> | <u>Entity ID</u> | <u>Campus Code</u> | <u>Primary Department Code</u> | <u>Employee ID</u> |
|------------------------------|---------------------|-----------------------|---------------|------------------|--------------------|--------------------------------|--------------------|
| return value | 1000000007 | ismith | SMITH, IGGY G | 4054 | BL | UA-PUR | 1000000007 |

Export options: [CSV](#) | [spreadsheet](#) | [XML](#)

Click one of the underlined column titles on the search results list to sort the list by that column. Click it again to sort the list in the opposite order.

You can also export the search results list in CSV, spreadsheet, or XML format. Click the appropriate link at the bottom of the search results list to do this.

If you need to see more information about one of the Persons in the search results list, click one of the information fields for that Person. Kuali then displays the **Person Document** screen for that Person.

Figure 5.5. Person Document

The **Return Value** column has a **return value** link for each person. Click this return value link for the Person whose information you need, and Kuali goes back to the screen you were on before and automatically enters the information about that person on the screen for you.

From the Administration Tab

If you come to the **Person Lookup** screen after clicking **Person** (under **Identity**) on the **Administration** tab, you may see a **create new** button in the upper right corner of your screen if the user logged in has permissions to create a Person, like this:

Figure 5.6. Person Lookup: Create New Button



You can click this create new button to go to the **Person Document** screen, where you can add a new Person as a Quali user. For more information on adding a Person, see the **Person Maintenance** function section (next) in this User Guide.

If you need to edit information for a Person in KIM, you first need to display that Person's information. Search for the Person (see search instructions above) from the **Administration** tab's link for **Person**. When Quali displays the list of Persons that match your search, you'll notice one difference in the columns because you did your search from the Administration tab – there is an **Actions** column on the left.

The **Actions** column has an **edit** link for each person. Click this edit link to go to the **Person Document** screen where you can edit this Person's information. For more information on working with Person information, see the **Person Maintenance** section (next) in this User Guide.

Person Maintenance

The Person Document allows you to identify and maintain each user in KIM. Each Person Document includes data about that user's relationship with your institution as well as the roles and groups to which the person belongs.

In KIM, a Person is a unique combination of an *Entity ID* and a *Principal ID*. The Entity ID represents a Person with a unique number, and the Person Document associates the Entity ID with the user's Principal ID number and Principal Name (often referred to as a user name or user ID). When searching for or working with users in KIM, you usually use either the Principal ID or the Principal Name. A single Entity ID can have multiple Principals associated with it, but the base implementation of KIM assumes that each Entity ID has only a single Principal.

Note

Person and HR Systems: Many institutions choose to override parts of the Person Document (especially the affiliation and contact information) with data from an HR system.

Business Rules

- Each Person must have at least one Affiliation.
- Each faculty or staff affiliation must have at least one Employment Information record associated with it.
- If a Person has any faculty or staff affiliations, then one Employment Information record must be marked as Primary.
- Each Person must have a default Name record in the Contacts section.
- Each affiliation must be associated with a Campus.

Routing

Only people with the appropriate role can initiate routing for Person documents.

Displaying the Person Lookup Screen

On the **Administration** tab, click **Person** to display the **Person Lookup** screen. Do a lookup to see a list of Persons. Click any Person on the search results list to display his or her **Person Document** screen.

Document Layout

The Person Document includes **Document Overview**, **Overview**, **Contact**, **Privacy Preferences**, **Membership**, **Ad Hoc Recipients**, and **Route Log** tabs.

Figure 5.7. Person Document

Document Overview Tab

The **Document Overview** tab combined with the Overview tab identifies the **Person** as a unique combination of Entity and Principal ID.

Additional information on the **Document Overview Tab** can be found in the Common Features and Functions section of this User Guide.

Overview Tab

The Overview tab identifies the **Person** as a unique combination of **Entity ID** and **Principal ID**. It also contains information about how this Person is affiliated with your institution. Two types of affiliations—**Staff** and **Faculty**—contain additional data fields to further define a person's relationship with your institution.

The instructions below assume that you are manually entering this information. Many institutions either have this data fed from an existing Person database or simply override this information with existing Person data.

Figure 5.8. Person Document: Overview Section

Overview Section

Table 5.1. Person Document: Overview Attributes

| Field Name | Description |
|---------------------------|---|
| Entity Id | Display-only. The unique ID number identifying this Person in your database. An individual may have multiple Principal IDs but only one Entity ID. The base implementation assumes that each user has only one Entity ID and one Principal ID. KIM automatically assigns an Entity Id to a new Person when you save or submit a Person Document. |
| Principal Id | Display-only. The unique ID number identifying this Principal. Whereas Entity Id represents a unique Person, Principal Id represents a set of login information for that Person. When selecting a Person, you ordinarily reference his or her Principal Id. KIM automatically assigns a Principal Id to a new Person when you save or submit the Person Document. |
| Principal Name | Required. The user name for this Principal |
| Tax Identification Number | Required. Enter the Individual Tax Identification Number (ITIN) for this Principal ID |
| Principal Password | Optional. Enter the password for this Principal ID |
| Active | Check the box to indicate that this Principal ID is Active. Uncheck the box to indicate that this Principal ID is Inactive. |
| Actions | Click the Add button under Actions to save the information you enter in this section. |

Affiliations Section

Use the **Affiliations** section of the **Overview** tab to add Affiliations for this Principal ID. Depending on the Affiliation type you select, you may need to complete additional fields.

Table 5.2. Person Document: Overview Attributes, Affiliations

| Field Name | Description |
|------------------|---|
| Affiliation Type | Optional. Select the Type of Affiliation from the list. Options: <ul style="list-style-type: none"> • Affiliate: Users in your system who are neither employees nor students • Faculty: A faculty employee • Staff: A non-faculty employee • Student: A non-employee identified as a student of your institution Affiliation types of Faculty and Staff require additional information (see below). |
| Campus Code | Required. Select the Campus Code associated with this Affiliation |
| Default | Check the box to indicate that this Affiliation is this Principal’s default association with your institution. Each Principal must have at least one default Affiliation. |
| Actions | Click the Add button to save the affiliation information. |

If you select an Affiliation of Faculty or Staff, KIM displays additional fields to collect employment information:

Figure 5.9. Person Document: Overview Tab, Affiliations Section

Table 5.3. Person Document: Overview Attributes, Affiliations Continued

| | |
|-------------------------|--|
| Employment ID | Optional. Enter the Employment ID number associated with this Faculty or Staff affiliation. Ordinarily this entry is the ID number identifying this Principal in your HR system. |
| Primary | Check this box to indicate that this Faculty or Staff affiliation represents the Principal's primary job with your institution. Each Principal with a Faculty or Staff affiliation must have exactly one affiliation marked as Primary. |
| Employee Status | Required. Select the current status of this Faculty or Staff affiliation. Options: <ul style="list-style-type: none"> • Active • Deceased • On Non-Pay Leave • Status Not Yet Processed • Processing • Retired • Terminated |
| Employee Type | Required. Select the type of employment for this Affiliation. Options: <ul style="list-style-type: none"> • Non-Professional • Other • Professional |
| Base Salary Amount | Required. Enter the base annual salary for this Faculty or Staff Affiliation. |
| Primary Department Code | Optional. Enter the code for the Department associated with this Faculty or Staff affiliation. |
| Add | Click the Add button to save this row of employment information. |

Contact Tab

The **Contact** tab records the names, addresses, phone numbers, and email addresses associated with this Person. Any Person record can store multiple contact information records of each type (name, address, phone number, and email address). You must select one value of each type as the default for that Person record.

Figure 5.10. Person Document: Contact Tab

The screenshot shows the 'Contact' tab interface. It contains four main sections, each with a table of fields and an 'Add' button:

- Names:** Fields include Name Type, Title, First Name, Last Name, Suffix, Default, Active, and Actions.
- Addresses:** Fields include Address Type, Line 1, Line 2, Line 3, City, State, Postal Code, Country, Default, Active, and Actions.
- Phone Numbers:** Fields include Phone Type, Phone Number, Extension, Country, Default, Active, and Actions.
- Email Addresses:** Fields include Email, Type, Default, Active, and Actions.

Names Section

Figure 5.11. Person Document: Contact Tab, Names Section

This is a close-up of the 'Names' section from the previous screenshot. It shows a table with the following columns: Name Type, Title, First Name, Last Name, Suffix, Default, Active, and Actions. There is an 'Add' button at the end of the row.

Table 5.4. Person Document: Contact Tab, Names Section Attributes

| Field Name | Description |
|------------|---|
| Name Type | Optional. Select the type of name to be added in this row. Options: <ul style="list-style-type: none"> • Other • Preferred • Primary |
| Title | Optional. Select the appropriate title for the name being added in this row. Options: <ul style="list-style-type: none"> • Ms • Mrs • Mr • Dr |
| First Name | Optional. Enter the first name for this record. |
| Last Name | Optional. Enter the last name for this record. |
| Suffix | Optional. Select a suffix for this name record. Options: <ul style="list-style-type: none"> • Jr • Sr • Mr • MD |
| Default | Check this box to indicate that this Name record is the default for this person. Each Person record must have exactly one Name record identified as the default. |
| Active | Check the box to indicate that this Name record is Active. Uncheck the box to indicate that this record should be considered Inactive. |
| Actions | Click the Add button to save this Name record. |

Addresses Section

Figure 5.12. Person Document: Contact Tab, Addresses Section

The screenshot shows a web form for adding addresses. It features a table with the following columns: Address Type (with a dropdown menu), Line 1, Line 2, Line 3, City, State (with a dropdown menu), Postal Code, Country (with a dropdown menu), Default (checkbox), Active (checkbox), and Actions (with an 'Add' button). Below the table is a section for 'Phone Numbers'.

Table 5.5. Person Document: Contact Tab, Address Section Attributes

| Field Name | Description |
|--------------|--|
| Address Type | Optional. Select the type of address being added on this row. Options: <ul style="list-style-type: none"> • Home • Other • Work |
| Line 1 to 3 | Optional. Use lines 1, 2, and 3 to enter the street address for this row. |
| City | Optional. Enter the city associated with this address. |
| State | Optional. Select the state associated with this address from the list. |
| Postal Code | Optional. Enter the postal code (zip code in the U.S.) associated with this address. |
| Country | Optional. Select the country associated with this address. |
| Default | Check this box to indicate this address record should be used as the Default. A Person record can have only one Default Address record. |
| Active | Check this box to indicate that this Address record is Active. Uncheck the box to indicate that this record is Inactive. |
| Actions | Click the Add button to save this Address record. |

Phone Numbers Section

Figure 5.13. Person Document: Contact Tab, Phone Numbers Section



Table 5.6. Person Document: Contact Tab, Phone Numbers Attributes

| Field Name | Description |
|--------------|---|
| Phone Type | Optional. Select the type of phone number being added on this row. Options: <ul style="list-style-type: none"> • Home • Mobile • Other • Work |
| Phone Number | Optional. Enter the area code and phone number. |
| Extension | Optional. Enter the appropriate extension. |
| Country | Optional. Select the country associated with this Phone Number record. |
| Default | Check this box to indicate that this Phone Number record should be used as the Default. A Person record can have only one default Phone Number record. |
| Active | Check this box to indicate that this Phone Number record is Active. Uncheck the box to indicate that this record is inactive. |
| Actions | Click the Add button to save this Phone Number record. |

Email Addresses Section

Figure 5.14. Person Document: Contact Tab, Email Addresses Section

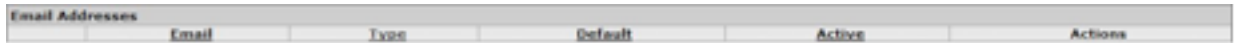


Table 5.7. Person Document: Contact Tab, Email Address Attributes

| Field Name | Description |
|------------|--|
| Email | Optional. Enter the email address for this record. |
| Type | Optional. Select the type of email address on this row. Options: <ul style="list-style-type: none"> • Home • Other • Work |
| Default | Check this box to indicate that this Email Address record should be used as the Default. A Person record can have only one default Email Address record. |
| Active | Check this box to indicate that this Email Address record is Active. Uncheck the box to indicate that this record is Inactive. |
| Actions | Click the Add button to save this Email Address record. |

Privacy Preferences Tab

The **Privacy Preferences** tab allows you to suppress the display of (hide) fields on the **Contact** Tab.

Figure 5.15. Person Document: Privacy Preferences Tab

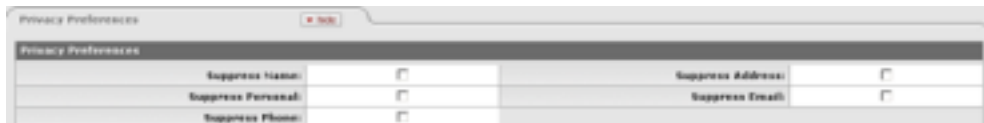


Table 5.8. Person Document: Privacy Preferences Tab Attributes

| Field Name | Description |
|-------------------|--|
| Suppress Name | Optional. Check this box to specify NOT to display this person's names. |
| Suppress Personal | Optional. Check this box to specify NOT to display any of this person's personal data. |
| Suppress Phone | Optional. Check this box to specify NOT to display this person's phone numbers. |
| Suppress Address | Optional. Check this box to specify NOT to display this person's addresses. |
| Suppress Email | Optional. Check this box to specify NOT to display this person's email addresses. |

Membership Tab

The **Membership Tab** allows you to associate a person with groups and roles and, by extension, with KIM permissions and responsibilities. Assigning a person to a role is the most direct way to give him or her KIM permissions and responsibilities.

Figure 5.16. Person Document: Memberships Tab

The Membership tab is divided into two sections, one for managing assignments to **Groups** and another for **Roles**.

Groups Section

Table 5.9. Person Document: Memberships Tab, Groups Attributes

| Field Name | Description |
|------------------|--|
| Group | Optional. Enter the name of the KIM group you want to assign this person to. You can also use the Group lookup to search for and select a valid Group. |
| Namespace Code | Display-only. After you select a group to add this person to, KIM displays the namespace code associated with the group you selected. |
| Name | Display-only. After you select a group to add this person to, KIM displays the name of that group. |
| Type | Display-only. After you select a group to add this person to, KIM displays the type of that group. |
| Active From Date | Optional. If this user's assignment to this group is to be effective as of a certain date, enter that date here. |
| Active To Date | Optional. If this user's assignment to this group is to terminate as of a certain date, enter that date here. |
| Actions | Click the Add button to add this group assignment. |

Note

There is no way to delete a person's assignment to a group. To remove a person from a group, use the Active To Date field to specify a date in the past.

Roles Section

Table 5.10. Person Document: Memberships Tab, Roles Attributes

| Field Name | Description |
|----------------|---|
| Role | Optional. Use the Name lookup to search for and select the role you want to assign this person to. |
| Namespace Code | Display-only. After you select a role to assign to this Person record, KIM displays the namespace code associated with that role. |

| Field Name | Description |
|------------------|--|
| Name | Display-only. After you select a role to assign to this Person record, KIM displays the name associated with that role. |
| Type | Display-only. After you select a role to assign to this Person record, KIM displays the role type for the selected role. |
| Active From Date | Optional. If this user's assignment to this role is to be effective as of a certain date, enter that date here. |
| Active To Date | Optional. If this user's assignment to this role is to terminate as of a certain date, enter that date here. |
| Actions | Click the Add button to save this role data. |

Note

There is no way to delete a person's assignment to a role. To remove a person from a role, use the Active To Date field to specify a date in the past.

When assigning some roles, you may need to supply additional qualifying information that further defines this person's assignment. For more information on Roles see the Role Maintenance section in this User Guide.

Ad Hoc Recipients Tab

Ad Hoc Recipients tab information can be found in the Common Features and Functions section of this User Guide.

Route Log Tab

Route Log tab information can be found in the Common Features and Functions section of this User Guide.

Group

Group Lookup Screen

The Group Lookup function provides a quick reference to key Group information.

You get to this screen by clicking the **Group** option in the **Identity** section of the **Administration** screen.

Figure 5.17. Identity Channel: Group Link

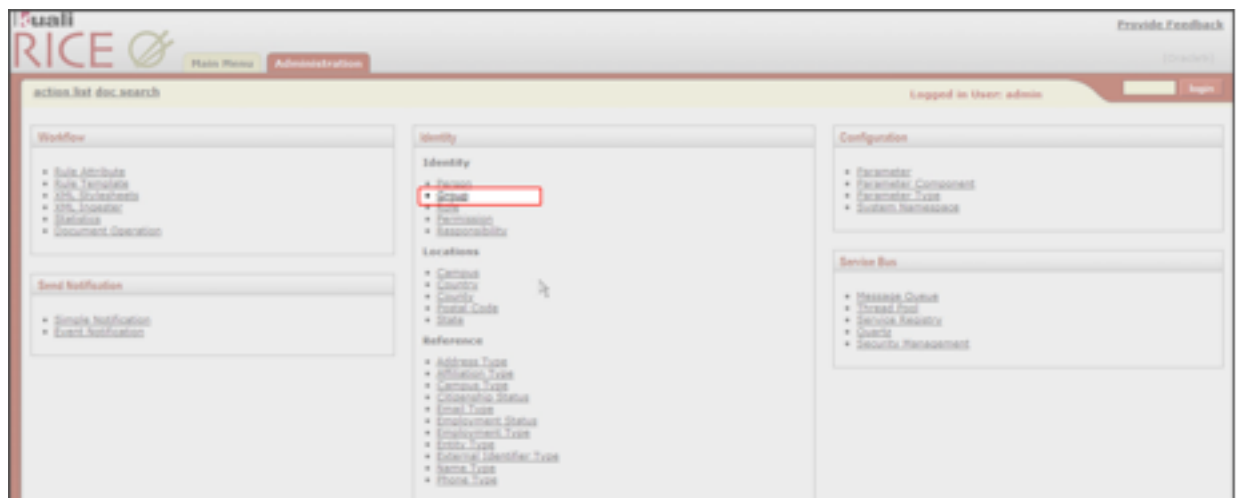


Figure 5.18. Group Lookup

On the **Group Lookup** screen, click the **create new** button to go to the **Group Maintenance** screen, where you can create a new Group in Rice.

Conducting a search from the Group Lookup screen returns a list similar to this:

Figure 5.19. Group Lookup: Results

| Actions | Group Type Name | Group Namespace | Group Name |
|---------|-----------------|-----------------|----------------------|
| edit | Default | KR-WKFLW | WorkflowAdmin |
| edit | Default | KR-WKFLW | RecipeMasters |
| edit | Default | KR-WKFLW | ChickenRecipeMasters |
| edit | Default | KR-WKFLW | Messins@Messagers |
| edit | Default | KR-WKFLW | NotificationAdmin |

For information on any of the fields above, please refer to the **Group Maintenance** section below.

Group Inquiry Screen

This screen provides high-level information about a Group and shows who is assigned to it.

Click a **Group Request** value on the **Action List** screen to display the **Group Inquiry** page for that Group:

Figure 5.20. Group Inquiry

Overview Tab

For information on these fields please refer to the **Group Maintenance** section that follows.

Assignees Tab

Table 5.11. Group Inquiry: Assignees Attributes

| Field | Description |
|-------------------|--|
| Type Code | The code for the type of this Group member |
| Member Identifier | The user ID for the Member |
| Name | The user name for the Member |
| Active From Dt | The effective date of membership in this Group for this member; if it is blank, membership is valid as soon as the member is assigned This field is useful when you want to add a Person, Role, or Group to a Group before he/she/they should be active in the Group. This might happen, for example, when you add a new employee before their start date. |
| Active To Dt | The termination date of membership for this member; if blank, the membership does not terminate |

Group Maintenance Document

The Group document allows you to associate Persons, Roles, or other Groups with each other so you can assign the same role to all Group members.

Groups have no inherent permissions or responsibilities of their own. Only by associating a Group with a role do the members of that Group become associated with permissions and responsibilities.

Creating New Groups

If you are creating a new Group and have clicked the **Create New** button on the Group Lookup screen, you will be redirected to the Group creation page. One of the first things to do, if wanting to create a group with a non-default Group Type, is to use the kim type lookup to select a different group type. Once clicked you will be redirected to a screen like the following:

Figure 5.21. KIM Type Lookup

Table 5.12. KIM Type Lookup Search Attributes

| Field | Description |
|----------------|---|
| Namespace Code | Optional. Select the code for the application and module to which this KIM Type pertains. |

| Field | Description |
|------------------|--|
| Type Name | Optional. Enter the name for this KIM Type. |
| Type Identifier | Optional. Enter the unique system-assigned Identifier for this KIM Type. |
| Active Indicator | Required (defaults to Yes). Change the default selection to view KIM types that are inactive or see all Types (active and inactive). |

The search results list includes the same fields as the Lookup screen. The search results are displayed below the search fields on the Lookup screen.

To select the Type you want to use for your new Group, click the return value link for that Type. KIM will copy the Type information to use in creating your new Group.

Group Maintenance Document: Layout

The Group document includes these tabs:

- Document Overview
- Overview
- Assignees
- Ad Hoc Recipients
- Route Log

The Overview and Assignees tabs are described below. Information on the others can be found in the Common Features and Functions section of this User Guide.

Figure 5.22. Group Maintenance Document

The screenshot displays the 'Group Maintenance Document' interface. At the top, there is a header bar with the following information: 'Doc.Nbr: 2761', 'Status: INITIATED', 'Initiator: admin', and 'Created: 01:31 PM 07/17/2009'. Below this, there are two tabs: 'Document Overview' (selected) and 'Overview'. The 'Document Overview' tab contains a form with fields for '* Description:', 'Org. Doc. #:', and 'Explanation:'. The 'Overview' tab contains a form with fields for 'Group Id: 1000000', 'Group Namespace: KR-WOPLW', 'Active?: ', 'Type Name: Default', and 'Group Name: RecipeMasters'. At the bottom of the interface, there are buttons for 'submit', 'save', 'blanket approve', 'close', and 'cancel'.

Overview Tab

This tab identifies the Group with a unique system-assigned ID number, a namespace, and a name. Each Group also has a Type that specifies any qualifiers that this Group might require.

Figure 5.23. Group Maintenance Document: Group Overview
Table 5.13. Group Maintenance Document: Group Overview Attributes

| Field | Description |
|-----------------|--|
| Group ID | Display-only. The unique system-assigned ID number that identifies this Group; Rice completes this field when you submit the document. |
| Type Name | Required. The type of attributes that are associated with this Group. Some Group types, such as the Default Type, do not require attributes. Note: When creating a new Group, you must select the Type before Rice can generate the document. See the Creating New Groups section of this document |
| Group Namespace | Required. An indicator that associates the Group with a particular application and module |
| Group Name | Required. The common descriptive name by which this Group is known |
| Active | Check this box to indicate that this Group is active and is a valid choice for assigning to roles. Uncheck the box to indicate that this Group is inactive (it is no longer valid when making role assignments). |

Assignees Tab

This tab displays the members who belong to this Group. You can also use this tab to add new members or to edit the fields for existing members.

Note that for members not added to the group on this maintenance document, the “delete” button to remove that member is inactive. If a member’s association with a group is past, users should set the “Active To Date” field to the end of the membership; the member will not be an active member of the group after that date.

Figure 5.24. Group Maintenance Document: Assignees Tab
Table 5.14. Group Maintenance Document: Assignees Tab Attributes

| Field | Description |
|-------------------|---|
| Member Type Code | Required. Select the Type of the member you are adding to this Group. Group members can be Principals (as defined on the Person document), Roles, or other Groups. |
| Member Identifier | Required. Enter the ID that identifies the member you are adding, or use the lookup to search for and select a valid Member ID. The lookup directs you to the Principal , Group , or Role lookup, based on your Member Type Code selection. |
| Active From Date | Optional. To specify the earliest date on which this member is to be considered a valid member of this Group, enter a date in this field. |
| Active To Date | Optional. To specify a date on which this member is no longer to be considered a valid member of this Group, enter a date in this field. As of this date, the Member will no longer be considered a member of this Group. |

| | |
|---------|---|
| | Note: You cannot delete or inactivate Group members. To remove a member from a Group, enter an Active To Date. |
| Actions | Click the Add button to add this member to the Group. |

Role

Role Lookup Screen

This screen provides detailed information relating to the Permissions assigned to a **Role** in KIM.

The **Role Lookup** screen is accessible directly off the Rice **Administration** menu by clicking **Role** in the Identity section.

Click the **Granted to Roles** field on the **Document Configuration** screen to display the **Role Lookup** screen. You can also access the Role Lookup screen by clicking a **Granted to Roles** value in any list where it appears.

Figure 5.25. Role Lookup

The screenshot shows the 'Role Lookup' screen with a 'create new' button and a '* required field' indicator. The form contains the following fields:

- Role: [text input]
- Role Name: [text input]
- Type: [dropdown menu]
- Namespace: [dropdown menu]
- Principal Name: [text input]
- Group Namespace: [dropdown menu]
- Group Name: [text input]
- Permission Namespace: [dropdown menu]
- Permission Name: [text input]
- Permission Template Namespace: [dropdown menu]
- Permission Template Name: [text input]
- Responsibility Namespace: [dropdown menu]
- Responsibility Name: [text input]
- Responsibility Template Namespace: [dropdown menu]
- Responsibility Template Name: [text input]
- Active?: Yes No Both

At the bottom of the form are three buttons: 'search', 'clear', and 'cancel'.

Document Layout

This screen may have three or four tabs of information, depending on the Role. The first three tabs are **Overview**, **Permissions**, and **Responsibilities**. KIM only displays the fourth tab, **Assignees**, when Assignee information is associated with this **Role**.

Figure 5.26. Role Maintenance Document

The screenshot displays the 'Role Maintenance Document' interface. It features three main tabs: Overview, Permissions, and Responsibilities. The Overview tab is active, showing details for Role 1, including its Namespace (KURL), Active status (Yes), Type Name (Derived Role: Principal), and Role Name (User). The Permissions tab is also visible, containing a table with 13 rows of permission data. The Responsibilities tab is partially visible at the bottom.

| Permission Namespace | Permission Identifier | Permission Name | Permission Detail Values | Active Indicator |
|----------------------|-----------------------|-----------------|---------------------------|--|
| 1 | KB-SYS | 146 | Ad Hoc Review Document | Document Type Name : RiceDocument Yes |
| 2 | KB-SYS | 149 | Initiate Document | Document Type Name : RiceDocument Yes |
| 3 | KB-SYS | 156 | Copy Document | Document Type Name : RiceDocument Yes |
| 4 | KB-SYS | 161 | Inquire Into Records | Namespace Code : KR* Yes |
| 5 | KB-SYS | 162 | Look Up Records | Namespace Code : KR* Yes |
| 6 | KB-SYS | 165 | Open Document | Document Type Name : RiceDocument Yes |
| 7 | KURL | 174 | Log On | Yes |
| 8 | KB-SYS | 161 | Inquire Into Records | Namespace Code : KR* Yes |
| 9 | KB-SYS | 162 | Look Up Records | Namespace Code : KR* Yes |
| 10 | KB-SYS | 163 | Maintain System Parameter | Namespace Code : KR* Yes |
| 11 | KB-SYS | 166 | User Screen | Namespace Code : KR* Yes |

Information on the fields in all of these tabs can be found in the **Role Maintenance** section of this user guide.

Role Maintenance Document

The Role Maintenance Document allows you to create a new KIM Role and edit existing Roles. Each Role collects a specific set of Permissions and Responsibilities and allows you to assign members to it.

The purpose of each Role is defined by its associated Permissions and Responsibilities. Roles are classified by **Types** that generally indicate with which Permissions and Responsibilities they are associated.

Note

The process of creating a new Type for Roles requires technical assistance.

To access the **Role** screen:

1. Do a **Role Lookup**.
2. Find the Role you want to change in the list of search results.
3. Click **Edit** for the row that has the Role that you need to change.

Role Maintenance Document

The Role document includes eight tabs:

- Document Overview
- Overview
- Permissions
- Responsibilities

- Assignees
- Delegations
- Ad Hoc Recipients
- Route Log

Those contains unique information are addressed below. For more information about the Document Overview, Ad Hoc Recipients and Route Log tabs, please see the **Common Features and Functions** document.

Figure 5.27. Role Maintenance Document: Tabs



Overview Tab

This tab gives the Role a unique system-assigned ID number, a Namespace, and a Name. Each Role also has a Type that usually matches the kinds of Permissions and Responsibilities associated with it.

Figure 5.28. Role Maintenance Document: Overview Tab



Table 5.15. Role Maintenance Document: Overview Attributes

| Field | Description |
|-----------------|--|
| Role Identifier | Display-only. The unique, system-assigned ID number that identifies this Role |
| Type Name | Display-only. The Type Name usually identifies the general kinds of Permissions and Responsibilities associated with this Role. Note: When creating a new Role, you must select its Type before you can display a new Role document. See Creating New Roles below. |
| Role Namespace | Required. An indicator that associates the Role with a particular application and module |
| Role Name | Required. The common descriptive name by which this Role is known |
| Active | Check this box to indicate that this Role is Active and is, therefore, to be included by KIM when it evaluates Permissions and Responsibilities. Uncheck the Active box to indicate that this Role is inactive. |

Creating New Roles

You must search for and select an existing Type for KIM to generate a new Role document. When you click the Create New button, KIM displays the **KIM Type Lookup** screen:

Figure 5.29. KIM Type Lookup

Note

While you use the KIM Type Lookup screen both for creating new Groups and new Roles, not all KIM Types are valid for both Groups and Roles. When using this Lookup, you may see different results depending on the KIM Types that are appropriate for your task.

Permissions Tab

This tab identifies the Permissions associated with this Role. Permissions authorize specific actions in the system with which they are associated. A Role can have any number of Permissions (including no Permissions) associated with it.

Figure 5.30. Role Maintenance Document: Permissions Tab

Table 5.16. Role Maintenance Document: Permissions Attributes

| Field Name | Description |
|-------------------|--|
| Add Permission ID | To add a Permission to this Role, enter the appropriate Permission ID, or search for and select a value using the Permission lookup |
| Add | Click the Add button to add the selected Permission to this Role document. |

After you add a Permission to the document, KIM displays additional information about the Permission:

Note

You *cannot* edit Permissions from the **Role Maintenance Document**.

Figure 5.31. Role Maintenance Document: Permissions Tab, Add Permissions



Table 5.17. Role Maintenance Document: Permissions Tab, Add Attributes

| Field | Description |
|--------------------------|--|
| Permission Namespace | Display-only. The Namespace identifies the application and module associated with this Permission. |
| Permission Identifier | Display-only. The unique system-assigned ID number for this Permission |
| Permission Name | Display-only. The descriptive name of this Permission. This often tells you, in general terms, what the Permission authorizes. |
| Permission Detail Values | Display-only. The document Types, tabs, and/or fields that this Permission authorizes the holder to see or use. Not all Permissions have Detail Values. |
| Active Indicator | Display-only. Indicator showing whether this Permission is active in KIM |
| Actions | Click the Delete button to remove this Permission from this Role. Warning: You may delete a Permission from a Role only if it has not yet been saved. (In other words, you can delete a Permission from a Role only if you added it to this Role, but have not yet submitted the document.) |

Responsibilities Tab

This tab identifies the Responsibilities associated with this Role. Responsibilities define the workflow actions that will be requested of the Role. A Role can have any number of Responsibilities (including none) associated with it.

Figure 5.32. Role Maintenance Document: Responsibilities Tab



Table 5.18. Role Maintenance Document: Responsibility Attributes

| Field Name | Description |
|-----------------------|---|
| Add Responsibility ID | To add a Responsibility to this Role, enter the Responsibility ID, or search for and select a value using the Responsibility lookup . |
| Add button | Click the Add button to add the selected Responsibility to this Role document. |

After you add a Responsibility to the document, KIM displays additional information about this Responsibility.

Note

In general, you cannot edit Responsibilities using the Role document, but some Responsibilities have associated attributes that you must define at the Role level.

Figure 5.33. Role Maintenance Document: Responsibility, Added Responsibility



Table 5.19. Role Maintenance Document: Responsibility, Add Attributes

| Field Name | Description |
|------------------------------|---|
| Responsibility Namespace | Display-only. The Namespace tells you the application and module associated with this Responsibility. |
| Responsibility Identifier | Display-only. The unique system-assigned ID number for this Responsibility |
| Responsibility Name | Display-only. The descriptive name of this Responsibility. For most Responsibilities, the name is Review . |
| Responsibility Detail Values | <p>Display-only. This gives you more specific information about the Responsibility. Responsibility Detail Values are formatted in a standard way with these attributes, separated by commas:</p> <ul style="list-style-type: none"> • Route Node: The location where this Responsibility is invoked in Workflow. • Document Type: The Document Type for which this Responsibility generates workflow requests. • Action Details at Role Member Level: When this field is True, action details are kept for each Member of this Role. When this field is False, action details are only kept at the Responsibility level for this Role. (see Assigning Action Detail Values, below). • Required: Indicates if the routing represented by this Responsibility should be required. If this is set to True and the Responsibility fails to generate an Action Request (perhaps because no one is assigned to the associated Role), then the document will go into Exception status. If this routing is optional, this will be False and the document will simply skip this Responsibility if no requests are generated. |
| Active Indicator | Display-only. Indicator showing whether this Responsibility is active within the system |
| Actions | <p>Click the Delete button to remove this Responsibility from this Role.</p> <p>Warning: You can delete a Responsibility only if it has not yet been saved to the database (in other words, you can only delete it if you have added it to this Role but have not yet submitted the document).</p> |

Assigning Action Detail Values

When you add a Responsibility with an **Action Detail Values at Role Member Level** value of **False**, you must complete additional fields in a Responsibility Action sub-section. KIM automatically displays this sub-section immediately beneath the Responsibility you’ve just added.

The fields in this sub-section define the type of Action Requests generated for, and the general workflow behavior associated with, this Responsibility. Entries in these fields cause KIM to generate the same Type of Action Requests for all members of this Role and handle actions by all members of this Role in the same way.

Figure 5.34. Role Maintenance Document: Responsibility Tab, Action Section



Table 5.20. Role Maintenance Document: Responsibility Tab, Action Section Attributes

| Field Name | Description |
|--------------------|---|
| Name | Display-only. The namespace and name of the Responsibility associated with these action details |
| Action Type Code | Required. The Type of Action Request that KIM generates for this Responsibility. Action Type Codes are: Approve, FYI, and Acknowledge. |
| Priority Number | Optional. If multiple requests are generated at the route node specified on this Responsibility, this determines the order in which KIM generates these requests. KIM processes requests with lower priority numbers before processing requests with higher numbers. Requests with no number are treated as a priority 1. |
| Action Policy Code | Required. This determines what happens if multiple members of this Role receive the same Action Request and one of them takes the action. For example, if a Role has a Group with three members assigned, all of these members receive the Action Request defined here; this code then determines what KIM does when one of them takes action on the document. A value of <i>FIRST</i> indicates that the first Group member to take action on the document causes KIM to clear all the requests for this Responsibility that may be in other Group member's action lists. A value of <i>ALL</i> indicates that each Group member must take individual action to clear his or her own requests. |
| Force Action | Check this box to indicate that each user must take action for this request, even if the user has previously taken action on this document. Leaving the box unchecked allows a request to be immediately fulfilled if the Role member has previously taken action on this specific document. |

Assignees Tab

This tab displays all members who belong to this Role. You may also use the tab to add new members and to edit the values associated with existing members.

Just as with group, role members not added by the specific maintenance document will have their “delete” buttons inactivated. To end that member’s active association with the group, set the “Active To Date” field to the end of the membership.

Figure 5.35. Role Maintenance Document: Assignees Tab
Table 5.21. Role Maintenance Document: Assignees Tab Attributes

| Field Name | Description |
|-------------------|--|
| Type Code | Required. Role members can be Principals (as defined on the Person document), Groups, or other Roles. Select the Type of member you want to add to this Role. |
| Member Identifier | Required. Enter the ID of the member you want to add, or use the lookup to search for and select a Member. The lookup icon displays the Principal, Group, or Role Lookup screen based on the Type Code you selected for this Role member . |
| Namespace Cd | Display-only. Identifies the namespace code associated with this Role member. Note that only Groups and Roles display a namespace code. |
| Name | Display-only. Identifies the name of the member you are assigning to this Role |
| Active From Dt | Optional. Enter a Active From date to define the earliest date on which this member is a valid member of this Role. |
| Active To Dt | Optional. Allows you to deactivate a member’s association with a Role on a specific date. Enter the date the user is no longer a member of this Role. Note: You cannot delete or inactivate Role members. To remove a member from a Role, enter an Active To Dt . |
| Actions | Click the Add button to add this member to the Role. |

Delegations Tab

This tab defines and identifies Delegates associated with this Role. Delegates are users that a member of this Role has authorized to have the same Permissions and take the same actions as that member is authorized to take. For example, a manager may make someone a Delegate for his or her actions in a particular Role.

The Assignees Tab dealing with Delegates is slightly different, as shown in the following table. Note that if the members of a Role require qualifying values, the delegation requires these values as well. In most cases, Delegates must have the same qualifiers as the Role member with which they are associated.

Just as with Group and Role Associations, delegations added outside of the current maintenance document have inactive “delete” buttons and must have the “Active To Date” field set to end the delegation.

Figure 5.36. Role Maintenance Document: Delegations Tab

Table 5.22. Role Maintenance Document: Delegations Tab Attributes

| Field Name | Description |
|-----------------------|--|
| Role Member | Required. Use the lookup to search for, and return, the Member of this Role for whom you wish to create a Delegate. |
| Member Type Code | Required. Select the Type of Delegate you want to add to this Role. Delegates may be Principals (as defined on the Person document), Groups, or other Roles. |
| Member Identifier | Required. Enter the ID that identifies the Delegate you want to add, or use the lookup to search for and select a Delegate. Note that the lookup takes you to the Principal, Group, or Role Lookup screen, depending on the Member Type Code you selected. |
| Member Namespace Code | Display-only. The namespace of this Delegate. Note that only delegations to Groups or Roles display a Member Namespace Code. |
| Member Name | Display-only. The name of the selected Delegate |
| Active From Date | Optional. You may choose to qualify this Delegate’s association with this Role by date. Entering an Active From Date is the earliest date on which this Delegate is valid for this Role. |
| Active To Date | Optional. Allows you to deactivate a Delegate’s association with a Role on a specific date. The date you enter is the date on which this user is no longer a Delegate for this Role. Note: You <i>cannot</i> delete or deactivate Delegates. To remove a Delegate from a Role, enter an Active To Date . |
| Delegation Type Code | Required. Select <i>Secondary</i> or <i>Primary</i> . Note that this selection only applies to Responsibilities associated with the Role. This indicates whether the Delegate automatically receives documents directly in their Action List (<i>Primary</i>) or if the Delegate may only choose to view documents in their Action List using the Secondary Delegate dropdown (<i>Secondary</i>). |
| Actions | Click the Add button to add this Delegate to this Role. |

KIM Type

KIM Type Lookup

Use this lookup when you create a new group in KIM to find the KIM Type for the new group.

Figure 5.37. KIM Type Lookup

You may enter information in one or more of the search fields to find the correct Type for your new group:

Table 5.23. KIM Type Lookup Attributes

| Field Name | Description |
|------------------|---|
| Namespace Code | Optional. Select the code of the application and module for this KIM Type. |
| Type Name | Optional. Enter the name for this KIM Type. |
| Type Identifier | Optional. Enter the unique system-assigned Identifier for this KIM Type. |
| Active Indicator | Required. Defaults to Yes (Yes means display only Active Types). Change the default selection to view KIM Types that are Inactive or Types that are both Active and Inactive. |

Figure 5.38. KIM Type Lookup: Results Example

* required field

| Return Value | Namespace Code | Type Name | Type Identifier | Active Indicator |
|------------------------------|----------------|-----------|-----------------|------------------|
| return value | KUALI | Default | <u>1</u> | Yes |

The search results list for this screen has the same fields as the **Lookup** screen. To select the Type you want to use for your new group, click the **return value** link for it in the search results list.

KIM Type Inquiry

The KIM Type Inquiry screen provides details about a KIM Type data element.

To access this screen, from the **Role Lookup** screen, click a **Role Type Name** in your search results list.

Figure 5.39. KIM Type Inquiry

The screenshot shows a window titled "Kim Type" with "expand all" and "collapse all" buttons in the top right. Inside the window, there is a sub-section titled "Kim Type" with a "hide" button. Below this, a table displays the following details:

| | |
|-------------------|--------------------------------|
| Type Identifier: | 2 |
| Type Name: | Derived Role: Principal |
| Service Name: | activePrincipalRoleTypeService |
| Active Indicator: | Yes |
| Namespace Code: | KR-IDM - Identity Management |

A "close" button is located at the bottom center of the window.

All fields are defined above in *KIM Type Lookup* except these:

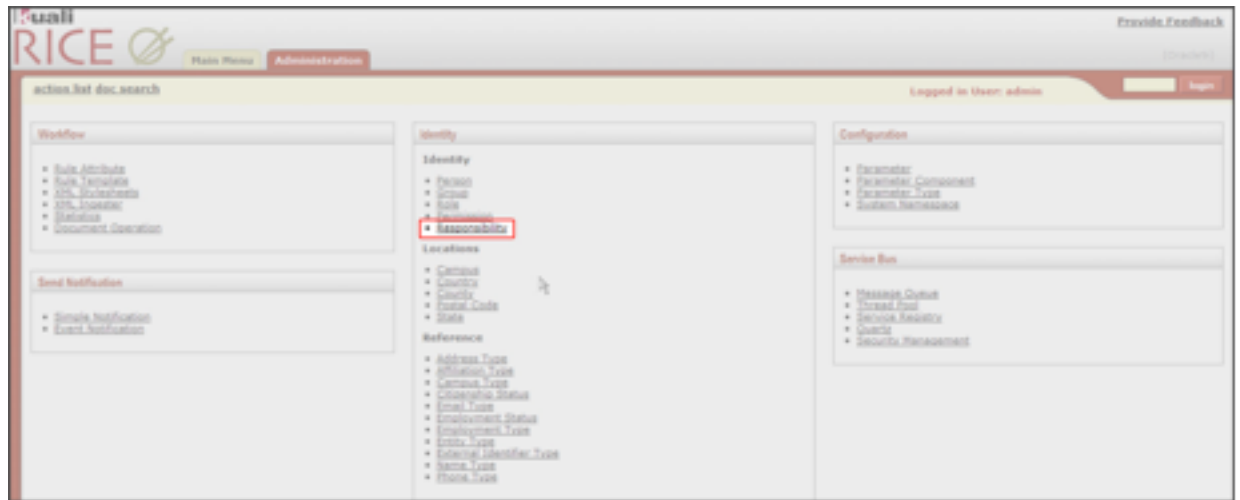
Table 5.24. KIM Type Inquiry Attributes

| Field Name | Description |
|----------------|--|
| Service Name | Display-only. The name of the service associated with this KIM Type |
| Namespace Code | Display-only. The namespace code associated with the selected KIM Type |

Responsibility

Responsibility Lookup

Similar to the Permission Lookup, you can use the Responsibility Lookup to search for and view existing responsibilities in KIM. You can view summarized information about the responsibility detail values as well as the roles with which the responsibility is currently associated.

Figure 5.40. Identity Channel: Responsibility Link

To display this screen, from the **Administration** menu, click **Responsibility** in the Identify section of the menu.

Note

This table is display-only. Technical assistance is required to modify responsibilities.

Figure 5.41. Responsibility Lookup

 The screenshot shows the 'Responsibility Lookup' form. It has a title bar with the text 'Responsibility Lookup' and a search icon. Below the title bar, there's a row of fields: 'Template Namespace' (dropdown), 'Template Name' (text), 'Responsibility Namespace' (dropdown), 'Responsibility Name' (text), 'Role Namespace' (dropdown), 'Role Name' (text), 'Principal Name' (text), 'Group Namespace' (dropdown), 'Group Name' (text), and 'Attribute Value' (text). Each field has a search icon to its right. At the bottom of the form, there are three buttons: 'SEARCH', 'CLEAR', and 'CANCEL'. A small asterisk and the text '* required field' are in the top right corner.

To find information about a Responsibility, enter information in one or more of the fields on the Lookup page and then click the Search button.

Table 5.25. Responsibility Lookup Attributes

| Field Name | Description |
|--------------------------|---|
| Template Namespace | Optional. The code identifying the application and module the template pertains to. Because responsibilities pertain to workflow, most responsibility templates are associated with the KR-WKFLW (Kuali Rice-Workflow) namespace. |
| Template Name | Optional. The template the responsibility is based on. A template usually defines, in a broad sense, what the responsibility is. Since responsibilities normally are associated with action requests for user review, most responsibilities have a template name of "Review." |
| Responsibility Namespace | Optional. The code designating the application and module this responsibility is associated with. This code usually corresponds to the namespace of the document type for which the responsibility generates action requests. |

| Field Name | Description |
|---------------------|--|
| Responsibility Name | Optional. The name of this responsibility. In most cases, the responsibility name is the same as the associated template name (“Review”). Like permission names, responsibility names are not unique. |
| Role Namespace | An indicator that associates the Role with a particular application and module. To search for a responsibility based on the namespace of the role to which it is assigned, enter the name of that namespace. |
| Role Name | Optional. The name by which a Role is known in the system. To search for a responsibility based on the Role to which it is assigned, enter that Role name. |
| Principal Name | Optional. One of the principals that currently have this responsibility through their association with a role |
| Group Namespace | Optional. The namespace of groups that have this responsibility through the group’s association with a role |
| Group Name | Optional. The name of a group that has this responsibility through its association with a role |
| Attribute Value | Optional. A specific responsibility detail value associated with a responsibility |

Figure 5.42. Responsibility Look: Results

| Template Namespace | Template Name | Responsibility Namespace | Responsibility Name | Responsibility Detail Values | Granted to Roles |
|--------------------|---------------|--------------------------|---------------------|---|--------------------------------|
| HR-WOFLW | Review | KFS-PP | Review | Route Node Name : Campus, Document Type Name : DV, Action Details At R... | KFS-PP Disbursement Manager |
| HR-WOFLW | Review | KFS-PP | Review | Route Node Name : Purchasing, Document Type Name : DV, Action Details ... | KFS-PUR&P Purchasing Processor |

Table 5.26. Responsibility Lookup: Results Attributes

| Field Name | Description |
|------------------------------|--|
| Responsibility Detail Values | <p>Display-only. Detailed information that defines:</p> <ul style="list-style-type: none"> • What document this responsibility generates action requests for • When the requests are generated • How the requests are handled by workflow <p>Unlike permissions, which sometimes have different detail values, responsibility detail values generally contain these elements:</p> <ul style="list-style-type: none"> • routeNodeName: The point in a document’s workflow routing at which this responsibility generates requests. • documentTypeName: The name of the document type for which this responsibility generates action requests. This value may also be a parent document type, which indicates that this responsibility applies to all child documents. • actionDetailsAtRoleMemberLevel: A True or False indicator that defines where the system collects details of this workflow action request. If the value is True, the system collects action details when members are assigned to the role. If the value is False, the system collects action details when this responsibility is assigned to a role. • Required: A True or False value that indicates whether the system is required to generate an action request for this document type. If the value is True and the document generates no requests associated with this responsibility, then the document will go into exception status. If the value is False and the responsibility generates no action requests, then the document continues to route as normal. |
| Granted to Roles | Display-only. Lists the namespace and name of roles that have this responsibility. Click a linked name to view the Role Inquiry for that role name. |

Responsibility Inquiry

To view the **Responsibility Inquiry** screen for a responsibility, click its **Responsibility Name** in the search results list displayed when you do a Responsibility Lookup. The Responsibility Inquiry screen contains the same information displayed in the search results, but in a slightly different format:

Figure 5.43. Responsibility Inquiry

The screenshot shows the 'Responsibility Inquiry' interface. It is divided into three main sections:

- Responsibility Attributes:** Contains fields for 'Template Namespace' (value: k11-WKFLW - Workflow), 'Template Name' (value: Review), 'Responsibility Namespace' (value: KFS-PP - Financial Processing), and 'Responsibility Name' (value: Review).
- Responsibility Detail Values:** A list of detail values with expandable sections:
 - Detail Values(routeNodeName-Travel):** Attribute Name: routeNodeName, Attribute Value: Travel
 - Detail Values(documentTypeName-DV):** Attribute Name: documentTypeName, Attribute Value: DV
 - Detail Values(actionDetailsAtRoleMemberLevel-false):** Attribute Name: actionDetailsAtRoleMemberLevel, Attribute Value: false
 - Detail Values(required-false):** Attribute Name: required, Attribute Value: false
- Responsibility Assigned Roles:** Shows 'Assigned Roles(KFS-PP - Financial Processing-Travel Manager-Default)' with fields for 'Namespace' (KFS-PP - Financial Processing), 'Role Name' (Travel Manager), and 'Role Type Name' (Default).

The fields on this screen are documented in the Responsibility Lookup section above.

Permission

Permission Lookup

The Permission Lookup screen allows you to search for and view existing permissions. It displays summarized information about the permission detail values as well as the roles that are currently associated with this permission.

Note

This table is display-only. Technical assistance is required to modify permissions.

You get to this screen by clicking **Permissions** in the **Identity** section of the **Administration** menu.

Figure 5.44. Permission Lookup

The screenshot shows the 'Permission Lookup' search interface. It features a series of input fields for search criteria, each with a search icon (magnifying glass) to its right:

- Template Namespace: []
- Template Name: []
- Permission Namespace: []
- Permission Name: []
- Role Namespace: []
- Role Name: []
- Principal Name: []
- Group Namespace: []
- Group Name: []
- Permission Detail Value: []

At the bottom of the form are three buttons: 'search', 'clear', and 'cancel'. A small asterisk icon in the top right corner indicates that some fields are required.

Enter information in one or more fields on the **Permission Lookup** screen and then click the **search** button to display permissions that match your information.

Table 5.27. Permission Lookup Attributes

| Field Name | Description |
|--------------------------|---|
| Template Namespace | Optional. The code identifying the application and module the template pertains to. Because templates tend to be general categories, they are often associated with system-level namespaces. |
| Template Name | Optional. The template the permission is based on. A template usually defines, in a broad sense, what the permission controls. Similar types of permissions use the same template. |
| Permission Namespace | Optional. The code designating the application and module this permission is associated with. |
| Permission Name | Optional. The name of this permission. In most cases, the permission name is the same as its associated template name. |
| Role Namespace | Optional. An indicator that associates the role with a particular application and module. |
| Role Name | Optional. The common descriptive name by which this role is known. |
| Principal Name | Optional. The principals that currently have this permission through their association with a role |
| Group Namespace | Optional. The namespace of groups that have this permission through the groups' association with a role |
| Group Name | Optional. The name of a group that has this permission through its association with a role |
| Permission Detail Values | Optional. Detailed information that, in combination with the permission name, defines the permission's function. For example, if the permission name is Initiate Document , the Permission Detail Values field indicates the specific type of document the initiate permission pertains to. Permission detail values can include many different types of data. Some common permission details: <ul style="list-style-type: none"> • documentTypeName: The name of the document Type associated with this permission. • routeNodeName: The point in a document's workflow routing at which this permission becomes relevant. • routeStatusCode: The routing status that a document must be in for this permission to apply. • propertyName: A field or document element that the permission pertains to. |

When you click the **search** button for a Permission Lookup, KIM displays your search results in a table like this:

Figure 5.45. Permission Lookup: Results Example

| Template Namespace | Template Name | Permission Namespace | Permission Name | Permission Detail Values | Granted to Roles |
|--------------------|---------------|----------------------|-----------------|---|--|
| KR-NS | Use Screen | KR-WKFLW | | Namespace Code : KR-WKFLW, Action Class : org.kuali.nice.kew.documents... | KR-SYS Technical Administrator, KR-SYS Technical Administrator |
| KR-NS | Use Screen | KR-BUS | | Namespace Code : KR-BUS, Action Class : org.kuali.nice.kab.security.ad... | KR-SYS Technical Administrator, KR-SYS Technical Administrator |
| KR-NS | Use Screen | KR-BUS | | Namespace Code : KR-BUS, Action Class : org.kuali.nice.kab.messaging.w... | KR-SYS Technical Administrator, KR-SYS Technical Administrator |
| KR-NS | Use Screen | KR-BUS | | Namespace Code : KR-BUS, Action Class : org.kuali.nice.kab.messaging.w... | KR-SYS Technical Administrator, KR-SYS Technical Administrator |

The information in the search results table is display-only and is defined above. New field:

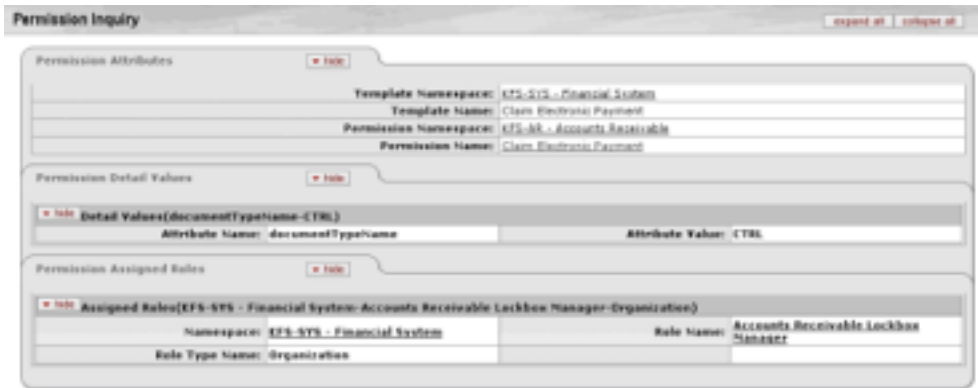
Table 5.28. Permission Lookup: Results Attributes

| Field Name | Description |
|------------------|--|
| Granted to Roles | Lists the namespace and name of roles that have this permission. Click on a linked name to view its Role Inquiry screen. |

Permission Inquiry

To view the **Permission Inquiry** screen for a Permission, click the **Permission Name** in the search results from a Permission Lookup. The Permission Inquiry screen contains the same information as the Permission Lookup search results, but in a slightly different format:

Figure 5.46. Permission Inquiry



The fields on this screen are documented above in the **Permission Lookup** section.

Permission Template Inquiry

This screen provides detailed information about a **Template Namespace**. To display it, click a **Template Name** on the **Document Configuration** screen or the **Permission Inquiry** screen.

Figure 5.47. Permission Template Inquiry



Information related to the fields on this screen can be found above, in the **Permission Lookup** section of this document.

Delivered Permission Templates

Rice is delivered with a number of permission templates. Below is a listing of each along with a brief description of their use.

Table 5.29. Delivered Permission Templates

| Template ID | Template Namespace | Template Name | Permission Description | Template | Permission Details |
|-------------|--------------------|---------------------------------|--|----------|--------------------|
| 1 | KUALI | Default | | | |
| 2 | KR-NS | Copy Document | | | documentTypeName= |
| 3 | KR-WKFLW | Administer Routing for Document | | | documentTypeName= |
| 4 | KR-WKFLW | Blanket Approve Document | Authorizes user to bypass specified approval route nodes | | documentTypeName= |

KIM

| Template ID | Template Namespace | Template Name | Permission Description | Template | Permission Details |
|-------------|--------------------|--|--|----------|---|
| 5 | KR-WKFLW | Route Document | Authorizes user to route a document from their action list | | documentTypeName= |
| 8 | KR-NS | Take Requested Action | Authorizes user to take applicable actions (approve, disapprove, acknowledge, etc.) on documents in their action list. | | actionRequestCd= |
| 9 | KR-WKFLW | Ad Hoc Review Document | Authorizes user to review a document that has been sent to them via an Ad Hoc Request | | documentTypeName= |
| 10 | KR-SYS | Initiate Document | Authorizes user to create a new document | | documentTypeName= |
| 14 | KR-WKFLW | Cancel Document | Authorizes users to cancel a document prior to it being submitted for routing. | | documentTypeName= routeNodeName= |
| 15 | KR-WKFLW | Save Document | Authorizes user to save a document that has been initiated or routed to them for action back to their action list | | documentTypeName= |
| 16 | KR-NS | Edit Document | | | documentTypeName= routeStatusCode= routeNodeName= |
| 23 | KR-NS | Look Up Records | | | namespaceCode= componentName= |
| 24 | KR-NS | Inquire Into Records | | | namespaceCode= componentName= |
| 25 | KR-NS | View Inquiry or Maintenance Document Field | | | |
| 26 | KR-NS | Modify Maintenance Document Field | | | |
| 27 | KR-NS | Full Unmask Field | | | propertyName= componentName= |
| 28 | KR-NS | Partial Unmask Field | | | |
| 29 | KR-NS | Use Screen | | | actionClass= namespaceCode= |
| 30 | KR-NS | Perform Custom Maintenance Document Function | | | |
| 31 | KR-NS | Use Transactional Document | | | |
| 32 | KR-NS | Modify Batch Job | | | namespaceCode= |
| 33 | KR-NS | Upload Batch Input File(s) | | | |
| 34 | KR-NS | Maintain System Parameter | | | namespaceCode= |
| 35 | KR-IDM | Assign Role | | | namespaceCode= |
| 36 | KR-IDM | Grant Permission | | | namespaceCode= |
| 37 | KR-IDM | Grant Responsibility | | | namespaceCode= |
| 38 | KR-IDM | Populate Group | | | namespaceCode= |
| 40 | KR-NS | Open Document | | | documentTypeName= |
| 42 | KR-NS | Create / Maintain Record(s) | | | documentTypeName= |

KIM

| Template ID | Template Namespace | Template Name | Permission Description | Template | Permission Details |
|-------------|--------------------|--|---|----------|--|
| 43 | KR-NS | View Inquiry or Maintenance Document Section | | | |
| 44 | KR-NS | Modify Maintenance Document Section | | | |
| 45 | KR-NS | Add Note / Attachment | Authorizes a user to add a note / attachment to a document | | documentTypeName= |
| 46 | KR-NS | View Note / Attachment | Authorizes a user to view notes and download attachments that have been added to a document | | documentTypeName= |
| 47 | KR-NS | Delete Note / Attachment | Authorizes a user to remove notes and attachments that have been added to a document | | documentTypeName= createdBySelfOnly |
| 49 | KR-NS | Send Ad Hoc Request | Authorizes a user to create an action request to user(s) outside the normal route chain | | actionRequestCd= documentTypeName= |
| 51 | KR-WKFLW | Add Message to Route Log | Authorizes user to add annotations to a document that appear in the Route Log of a document | | documentTypeName= |
| 52 | KR-RULE | KRMS Agenda Permission | Authorizes user to view and edit KRMS agendas | | namespaceCode= |
| 53 | KR-KRAD | Open View | | | |
| 54 | KR-KRAD | Edit View | | | |
| 55 | KR-KRAD | Use View | | | |
| 56 | KR-KRAD | View Field | | | |
| 57 | KR-KRAD | Edit Field | | | |
| 58 | KR-KRAD | View Group | | | |
| 59 | KR-KRAD | Edit Group | | | |
| 60 | KR-KRAD | View Widget | | | |
| 61 | KR-KRAD | Edit Widget | | | |
| 62 | KR-KRAD | Perform Action | | | |
| 63 | KR-KRAD | View Line | Authorizes a user to view the lines in a collection | | |
| 64 | KR-KRAD | Edit Line | Authorizes a user to edit the lines in a collection | | |
| 65 | KR-KRAD | View Line Field | Authorizes a user to view the individual fields in the lines of a collection | | |
| 66 | KR-KRAD | Edit Line Field | Authorizes a user to edit the individual fields in the lines of a collection | | |
| 67 | KR-KRAD | Perform Line Action | Authorizes a user to take actions (add, delete, edit, etc.) to the lines of a collection | | |
| 68 | KR-WKFLW | Recall Document | Authorizes a user to recall a submitted document to their action list for edit and resubmission or cancellation | | documentTypeName= routeNodeName= routeStatusCode= appDocStatus= |

Chapter 6. KNS

KNS Overview

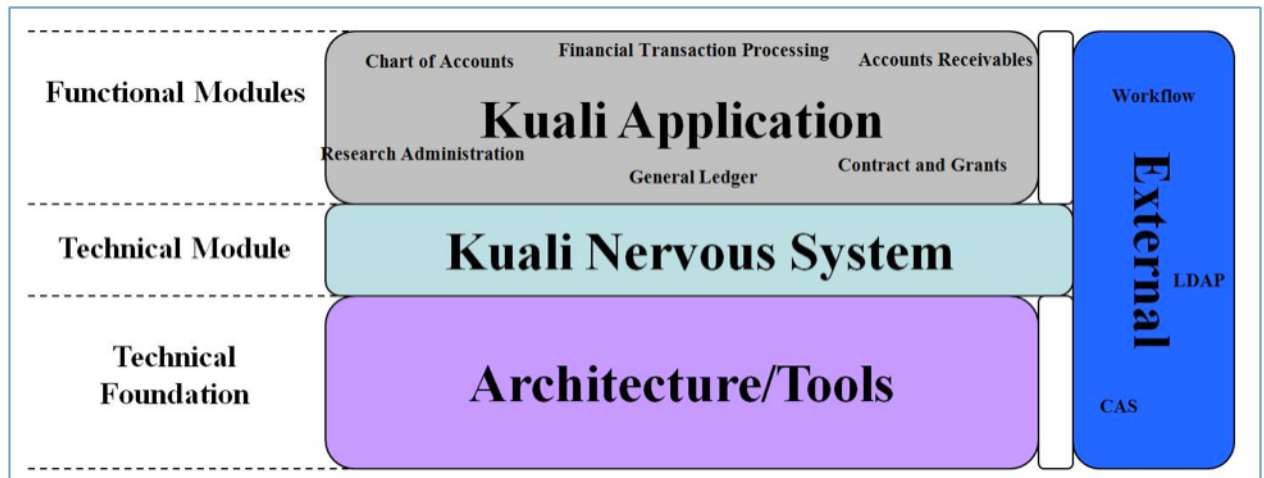
The Kuali Nervous System (KNS) is the core of the Kuali Rice system. It embraces a document- (business process) centric model that uses workflow as a core concept.

What is KNS?

- It's a framework to enforce consistency.
- It adheres to development standards and architectural principles.
- It's a stable core for efficient development.
- It reduces the amount of code you need to write and supports code reuse.

KNS Conceptual View

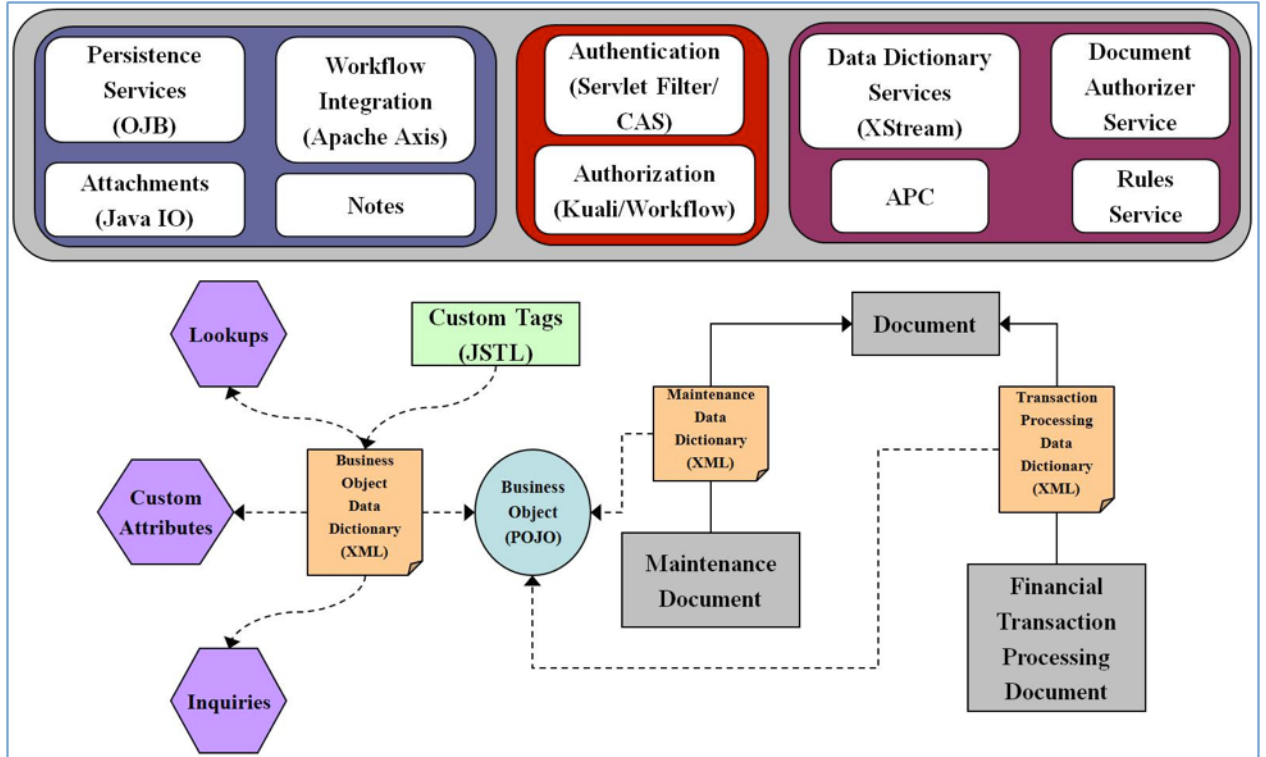
Figure 6.1. KNS: Conceptual View



KNS is the core technical module in Rice. It has reusable code components to provide functionality.

KNS Relational View

Figure 6.2. KNS: Relational View



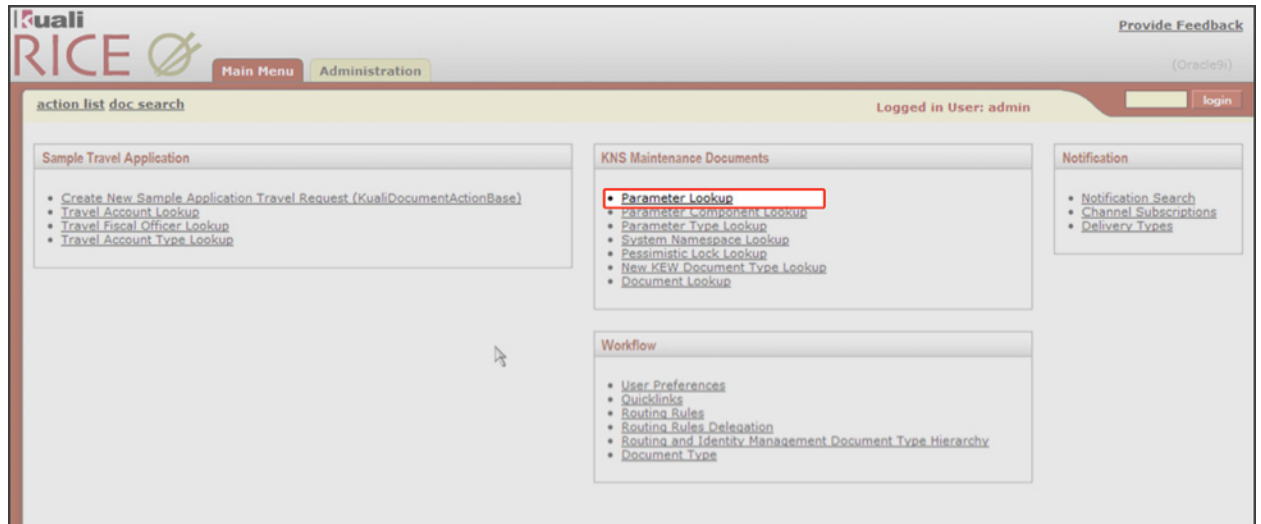
Essentially, KNS is a common set of service methods you can use to take actions on a document. It's very similar to a business object service, for example, one that persists documents and integrates with a workflow. One of the ways the KNS subsystem accomplishes this is by using Spring's transaction management features:

- `documentService.save(document);`
- `documentService.approve(document);`

Parameter

Parameter Lookup

KNS parameters are used throughout Rice. They reside in special system tables that are used for configuring the system during runtime. Parameters control whether certain features are turned on or off. They also establish default values within the Rice system or by a process such as validation.

Figure 6.3. Parameter Lookup Link

To access the Parameter Lookup screen:

1. Go to the Rice **Main Menu** screen
2. Locate the KNS Maintenance Documents section of the page
3. Click **Parameter Lookup**

Figure 6.4. Parameter Lookup

 The screenshot shows the 'Parameter Lookup' search form. It features five input fields: 'Namespace Code', 'Component', 'Parameter Name', 'Parameter Value', and 'Parameter Type Code'. Each field has a search icon and a help icon. The 'Parameter Value' field is a text area. At the bottom of the form, there are three buttons: 'search', 'clear', and 'cancel'. A legend in the top right corner indicates that an asterisk (*) denotes a required field.

There are two ways to search for information on specific parameters:

- Enter information about the parameter in one or more fields on the **Parameter Lookup** screen, then click the **Search** button. This displays a list of parameters that match the information you entered, similar to the list in the screen print below.
- Leave the fields on the **Parameter Lookup** screen blank and click the **Search** button. This displays a list of all available KNS parameters, similar to this:

Figure 6.5. Parameter Lookup: Results Set

| Namespace Code | Component | Parameter Name | Parameter Description | Parameter Value | Parameter Constraint Code |
|----------------|-------------|---------------------------------------|--|---|---------------------------|
| KR-WKFLW | Action List | ACTION_LIST_DOCUMENT_POPUP_IND | Flag to specify if clicking on a Document... | Y | A |
| KR-WKFLW | Action List | ACTION_LIST_ROUTE_LOG_POPUP_IND | Flag to specify if clicking on a Route L... | N | A |
| KR-NS | Batch | ACTIVE_FILE_TYPES | Batch file types that are active options... | collectorInputFileType;procurementCardIn... | A |
| KR-WKFLW | | APPLICATION_CONTEXT | Web application context name of the appl... | en-dev | A |
| KR-WKFLW | Rule | CACHING_IND | Indicator to determine if rule caching i... | Y | A |
| KR-NS | All | CHECK_ENCRYPTION_SERVICE_OVERRIDE_IND | Flag for enabling/disabling (Y/N) the de... | Y | A |

You can sort the list of parameters in ascending or descending order by one column by clicking the appropriate column heading. If you want to save the list, you can export the entire list in CSV, spreadsheet, or XML format.

Click a Namespace Code in the list of parameters to display more information about that code.

Parameter Inquiry

You arrive at this screen by clicking a **Parameter Name** on the **Parameter Lookup** list.

Figure 6.6. Parameter Inquiry

The screenshot shows a 'Parameter Inquiry' window with a title bar containing 'expand all' and 'collapse all' buttons. The main content area is titled 'Parameter Information' and includes a 'hide' button. The information is displayed in a table-like format:

| | |
|-----------------------------------|---|
| Namespace: | KR-WKFLW - Workflow |
| Component: | Action List |
| Parameter Name: | ACTION_LIST_DOCUMENT_POPUP_IND |
| Parameter Value: | Y |
| Parameter Constraint Code: | Allowed |
| Parameter Description: | Flag to specify if clicking on a Document ID from the Action List will load the Document in a new window. |
| Parameter Type: | Config |

A 'close' button is located at the bottom center of the window.

Parameter Maintenance

In Rice, Parameters control key aspects of how the system operates.

Finding the Parameter Maintenance Screen

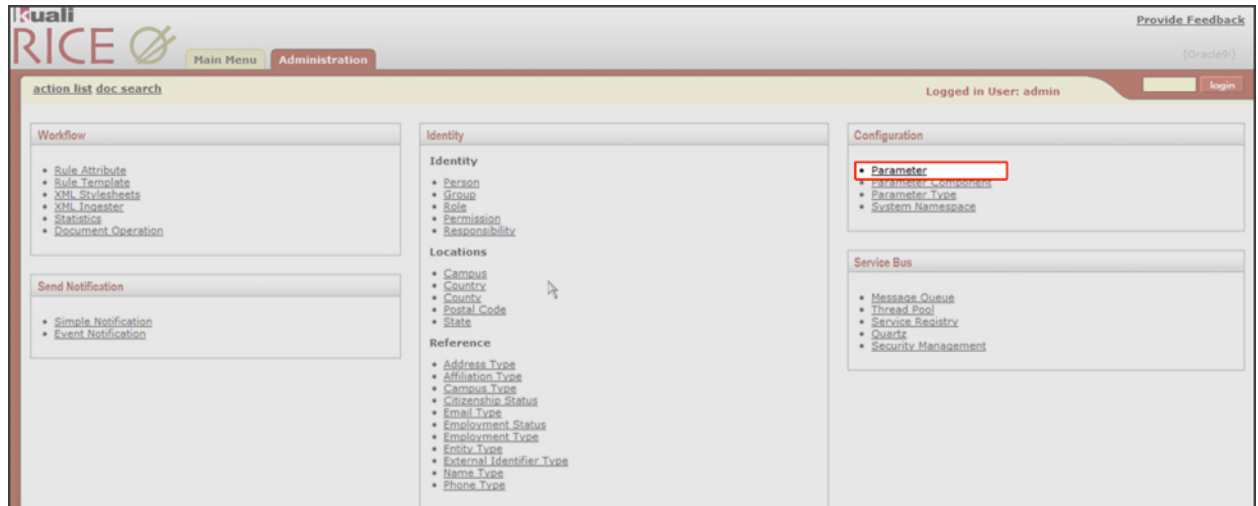
Getting to the **Parameter Maintenance** screen requires that you have permission to use the Administration screens in KRice. You can go directly to the **Parameter Maintenance** screen by:

1. Click the **Administration** tab
2. Look under **Configuration**
3. Click **Parameter**

Parameter Maintenance Screen

KNS Parameters can be created and edited by selecting the Parameter option on the Rice Administration screen.

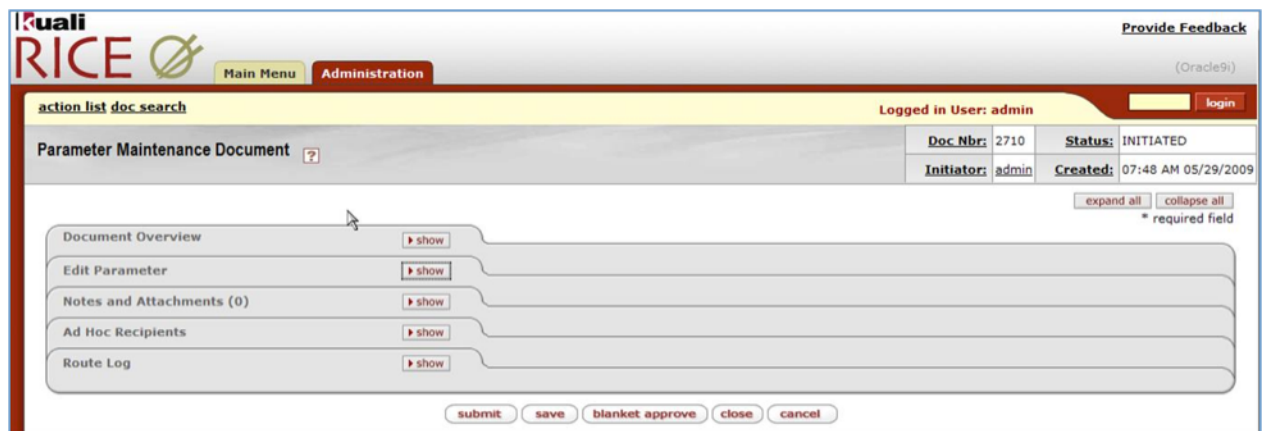
Figure 6.7. Configuration Channel: Parameter Link



After selection you will be presented with the **Parameter Maintenance Document** screen. The picture below is the complete screen with all tabs hidden. By default the Document Overview and Edit Parameter tabs are shown (i.e. open) when this screen is first accessed.

The system will automatically assign a new document number every time this screen is accessed. The Initiator, Status and Created information will also automatically be filled in.

Figure 6.8. Parameter Maintenance Document



The Notes and Attachments tab, the Ad Hoc Recipients tab, the Route Log tab, and the buttons at the bottom of this screen perform standard functionality as described in the Common Features and Functionality document found in the Global section of the User Guide.

The following descriptions apply to the fields in the upper right hand corner of the screen.

Table 6.1. Parameter Maintenance Document: attributes

| Field Name | Description |
|------------|---|
| Doc Nbr | The document id is generated by the workflow environment and is unique to each installation of Kuali. It is system generated. |
| Initiator | The user name of the person who created the document. |
| Status | This represents the status of this document. (Initiated, Enroute, Final) |
| Created | The date and time that this document was created. It is system generated. |

Document Overview Tab

Figure 6.9. Parameter Maintenance Document: Document Overview Tab
Table 6.2. Document Overview Tab: Attributes

| Field Name | Description |
|-------------|--|
| Description | Required. A description of the request itself, not the parameter. This text should briefly describe why the request is being made. |
| Org Doc # | Also sometimes simply referred to as document number. A unique system assigned number used to identify each document. |
| Explanation | Free form text which provides a detailed description of the request. |

Edit Parameter Tab

Figure 6.10. Parameter Maintenance Document: Edit Parameter Tab
Table 6.3. Edite Parameter Tab: Attributes

| Field Name | Description |
|---------------------|---|
| Namespace Code | Required. This value is used to categorize parameters by namespace. |
| Parameter Component | Required. Code identifying the parameter Component. |
| Parameter Name | Required. This will be used as the identifier for the parameter. Parameter values will be accessed using this field and the namespace as the key. Note – spaces are not allowed in this field |
| Parameter Value | Required. This field houses the actual value associated with the parameter. This is what's returned by the KualiConfigurationService. |

| Field Name | Description |
|---------------------------|--|
| Parameter Description | Required. This field houses the purpose of this parameter. |
| Parameter Type code | Required. Code identifying the parameter type. Parameter Type Code is the primary key for its table. |
| Parameter Constraint Code | Required. Code identify the constraints that apply. |

Process to create a new Parameter

There are eight mandatory fields on the Document Overview and Edit Parameter tabs that must be filled in before pressing the submit button. If the system finds any errors it will return the screen with the fields in error identified, otherwise the system will respond with the Parameter Maintenance Document screen showing that the:

- document was successfully submitted
- new document number for the document
- status is now ENROUTE
- actual creation date and time

Figure 6.11. Parameter Maintenance Document: Routed Summary

Parameter Maintenance Document Doc Nbr: 2712 Status: ENROUTE
Initiator: admin Created: 03:10 PM 06/10/2009

Document was successfully submitted. expand all collapse all
* required field

Document Overview hide

| * Description: | Explanation: |
|----------------|--------------|
| test1 | |
| Org. Doc. #: | |

Edit Parameter hide

New

| | |
|----------------------------|------------------------------|
| Namespace Code: | KR-NS - Kualu Nervous System |
| Parameter Component: | All |
| Parameter Name: | testparameter |
| Parameter Value: | test value |
| Parameter Description: | test parameter |
| Parameter Type Code: | Document Validation |
| Parameter Constraint Code: | Allowed |

Notes and Attachments (0) show

Ad Hoc Recipients show

Route Log show

send ad hoc request reload close

You'll also notice that the screen has changed and now has different buttons at the bottom. Click the close button to close the current screen and return to your starting point.

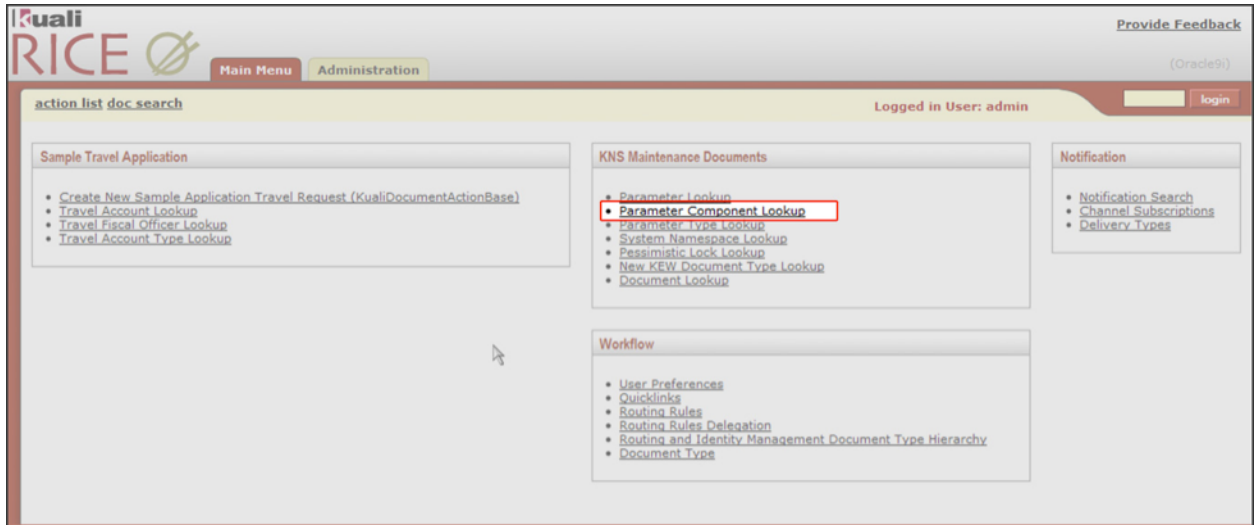
Parameter Detail

Parameter Component Lookup

Use the Parameter Component Lookup screen to get information on what *Namespace* a *Parameter Component* is associated with, and whether it is active or not.

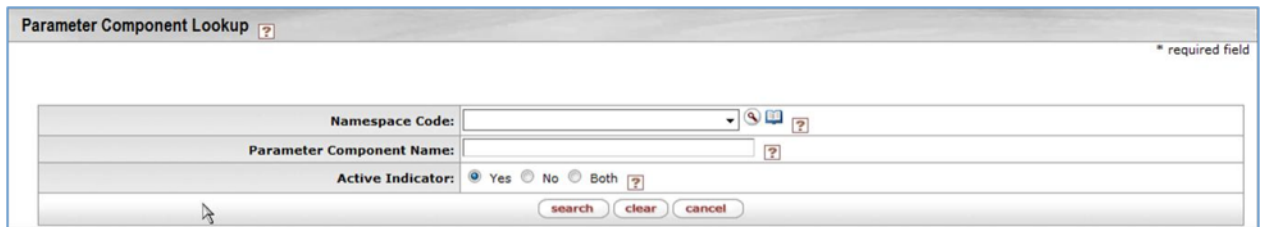
To get to the **Parameter Component Lookup** screen click that option in the **KNS Maintenance Documents** section on the **Main Menu**.

Figure 6.12. KNS Maintenance Documents: Parameter Component Lookup Link



This leads you to the **Parameter Component Lookup** screen.

Figure 6.13. Parameter Component Lookup



Performing a search from this screen returns a list of one or more Parameter Components which looks similar to the following example.

Figure 6.14. Parameter Component Lookup: Results Set

| Parameter Component Name | Namespace | Active Indicator |
|--------------------------|------------------------------|------------------|
| Action List | KR-WKFLW - Workflow | Yes |
| All | KR-IDM - Identity Management | Yes |
| All | KR-NS - Kuali Nervous System | Yes |
| Backdoor | KR-WKFLW - Workflow | Yes |
| Batch | KR-IDM - Identity Management | Yes |
| Batch | KR-NS - Kuali Nervous System | Yes |
| Document | KR-IDM - Identity Management | Yes |
| Document | KR-NS - Kuali Nervous System | Yes |
| Document Search | KR-WKFLW - Workflow | Yes |
| eDocLite | KR-WKFLW - Workflow | Yes |
| Feature | KR-WKFLW - Workflow | Yes |
| Global Reviewer | KR-WKFLW - Workflow | Yes |

To limit the number of items returned you simply provide criteria which is then used to limit the search.

Table 6.4. Parameter Component Lookup: Results set Attributes

| Field Name | Description |
|--------------------------|--|
| Parameter Component Name | The name by which the Parameter Component is commonly known. |
| Namespace | The Namespace Code and Namespace Name associated with the Parameter Component. |
| Namespace Code | The code associated with the namespace. |

| Field Name | Description |
|------------------|---|
| Active Indicator | Indicator for whether the Parameter Component is active in Rice or not. |

Parameter Component Inquiry

Figure 6.15. Parameter Component Inquiry

You arrive at this screen by clicking on a **Parameter Component** on the **Parameter Lookup** list.

Table 6.5. Parameter Component Inquiry: Attributes

| Field Name | Description |
|---------------------|--|
| Parameter Component | A code identifying the component utilizing the parameter |

Parameter Namespace

Namespace Lookup

Kuali uses Namespaces in all Kuali applications. A Namespace is a set of information about the system and groups of information in the system. Everything is tied to a Namespace in some way in Rice. A Namespace provides a way to set boundaries around both Permissions and Entity Attributes. Each Namespace instance is one level of scoping and is one record in that Kuali module.

Use the **System Namespace Lookup** function to quickly find basic information about a Namespace in Rice.

When you click **System Namespace Lookup** in the **KNS Maintenance Documents** section on the **Main Menu**, Rice displays a screen titled **Namespace Lookup**, like this:

Figure 6.16. Namespace Lookup

There are two ways to search for information on a specific Namespace:

- Enter information about the Namespace you want to find in one or more fields on the **Namespace Lookup** screen, then click the **Search** button. This displays a list of Namespaces that match the information you entered, similar to the list in the screen print below.
- Leave the fields on the **Namespace Lookup** screen blank and click the **Search** button. This displays a list of all available Namespaces, similar to this:

Figure 6.17. Namespace Lookup: Results Set

| Namespace Code | Namespace Name | Application Namespace Code | Active Indicator |
|----------------|---------------------------|----------------------------|------------------|
| KR-SYS | Enterprise Infrastructure | | Yes |
| KR-IDM | Identity Management | | Yes |
| KR-NS | Kuali Nervous System | | Yes |
| KUALI | Kuali Systems | | Yes |
| KR-NTFCN | Notification | | Yes |
| KR-BUS | Service Bus | | Yes |
| KR-WKFLW | Workflow | | Yes |

You can sort the search results in ascending or descending order by one column by clicking that column's heading. If you want to save the search results, you can export the entire list in CSV, spreadsheet, or XML format.

Click a Namespace Code in the search results list to display more information about that code.

Namespace Inquiry

Use the **Namespace Inquiry** screen to find key information about Namespaces.

You can go to the **Namespace Inquiry** screen from several places in Rice:

- On the Rice Main Menu, click **System Namespace Lookup**.
- On the **Parameter Lookup** screen, click the book icon after selecting a valid **Namespace Code** from the dropdown list.
- On the **Parameter Component Lookup** screen, select a valid **Namespace Code** from the dropdown list, then click the book icon.
- When you do a **Parameter Lookup**, click the **Namespace Code** for one of your search results.
- On the **Permission Inquiry** screen, click the **Template Namespace** or the **Permission Namespace** date field.

The Namespace Inquiry screen and information you see will be similar to this:

Figure 6.18. Namespace Inquiry

Namespace Inquiry

Namespace Information ▼ hide

| | |
|------------------------------------|-------------|
| Namespace Code: | KR-BUS |
| Namespace Name: | Service Bus |
| Application Namespace Code: | |
| Active Indicator: | Yes |

close

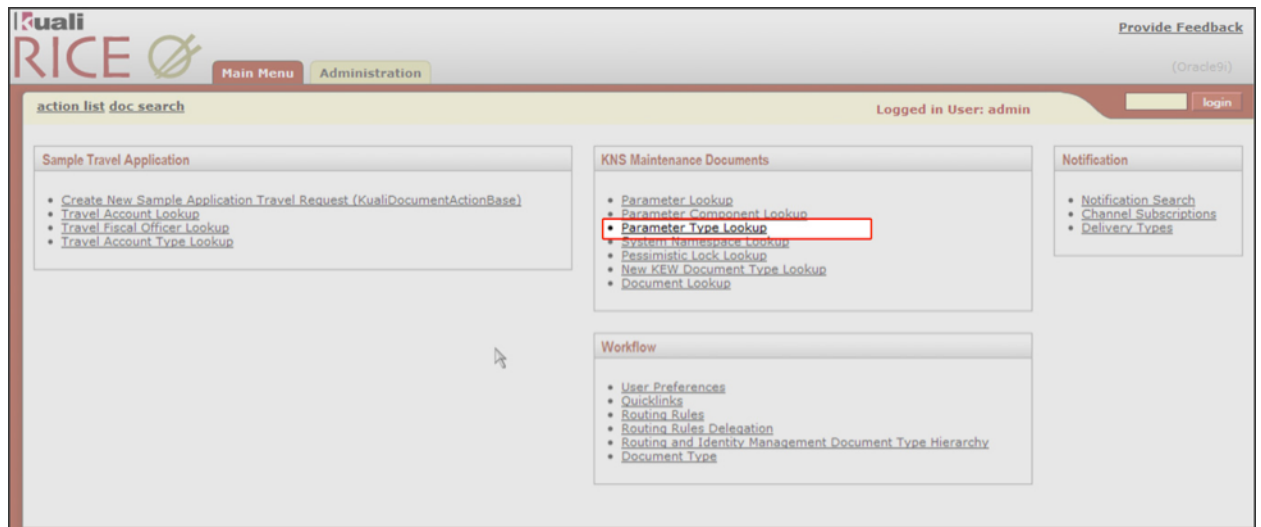
Parameter Type

Parameter Type Lookup

The **Parameter Type Lookup** function provides a convenient way to obtain key information about the Parameter Types defined in your Rice system.

To get to this screen, click **Parameter Type Lookup** on the **Main Menu**.

Figure 6.19. KNS Maintenance Documents Channel: Parameter Type Lookup Link



You can also get to this screen by clicking the field lookup button on any screen where it appears next to the **Parameter Type Code** field.

Figure 6.20. Parameter Type Lookup

The screenshot shows the 'Parameter Type Lookup' search form. It includes the following fields and controls:

- Parameter Type Code:** A text input field with a help icon (?) and a required field asterisk (*).
- Parameter Type Name:** A text input field with a help icon (?) and a required field asterisk (*).
- Active Indicator:** Radio buttons for 'Yes' (selected), 'No', and 'Both', with a help icon (?) and a required field asterisk (*).
- Buttons:** 'search', 'clear', and 'cancel' buttons.

Performing a search from the **Parameter Type Lookup** screen displays a table of Parameter Types that looks similar to this:

Figure 6.21. Parameter Type Lookup: Results Set

| Parameter Type Code | Parameter Type Name | Active Indicator |
|---------------------|---------------------|------------------|
| AUTH | Authorization | Yes |
| CONFG | Config | Yes |
| VALID | Document Validation | Yes |
| HELP | Help | Yes |

The fields on this screen are described in the **Parameter Type Maintenance** section of this document.

You can sort the search results in ascending or descending order by clicking the appropriate column title. In addition to the online list of search results, you can also export the entire list in CSV, spreadsheet, or XML format. If you get too many Parameter Types from your search, please enter more information in the search fields and click the **Search** button again.

Parameter Type Inquiry

The **Parameter Type Inquiry** screen gives you information about a parameter type. On the **Parameter Lookup** screen, select a valid **Parameter Type Code** from the dropdown list. Then, click the inquiry button to display the Parameter Type Inquiry screen with the information for that parameter:

Figure 6.22. Parameter Type Inquiry

The screenshot shows a window titled "Parameter Type Inquiry". Inside, there is a section labeled "Parameter Type Information" with a "hide" button. Below this, a table displays the following information:

| | |
|-----------------------------|---------------|
| Parameter Type Code: | AUTH |
| Parameter Type Name: | Authorization |
| Active Indicator: | Yes |

At the bottom of the window, there is a "close" button.

The fields on this screen are described in the **Parameter Type Maintenance** section of this document.

Parameter Type Maintenance

The Parameter Type Maintenance function allows you to define new Parameter Types or edit information related to **Parameter Types**.

You get to the **Parameter Type Maintenance** screen by clicking **Parameter Type** in the **Configuration** section on the Rice **Administration** menu:

Figure 6.23. Configuration Channel: Parameter Type Link

The screenshot shows the Rice Administration interface. The top navigation bar includes "Main Menu" and "Administration". The "Administration" section is active, and the user is logged in as "admin". The main content area is divided into several sections:

- Workflow:**
 - Rule_Attribute
 - Rule_Template
 - XML_Stylesheets
 - XML_Ingestor
 - Statistics
 - Document_Operation
- Send Notification:**
 - Simple_Notification
 - Event_Notification
- Identity:**
 - Person
 - Group
 - Role
 - Permission
 - Responsibility
- Locations:**
 - Campus
 - Country
 - Postal_Code
 - State
- Reference:**
 - Address_Type
 - Affiliation_Type
 - Campus_Type
 - Citizenship_Status
 - Email_Type
 - Employment_Status
 - Employment_Type
 - Entity_Type
 - External_Identifier_Type
 - Name_Type
 - Phone_Type
- Configuration:**
 - Parameter
 - Parameter_Component
 - Parameter_Type** (highlighted with a red box)
 - System_Namespaces
- Service Bus:**
 - Message_Queue
 - Thread_Pool
 - Service_Registry
 - Quartz
 - Security_Management

This takes you to the **Parameter Type Maintenance Document** screen:

Figure 6.24. Parameter Type Maintenance Document

Document Layout

The Parameter Type Maintenance screen has five main tabs:

- Document Overview
- Edit Parameter Type
- Notes and Attachments
- Ad Hoc Recipients
- Route Log

All of these except the Edit Parameter Type tab are described in the Common Features and Functions section of this User Guide.

Edit Parameter Type tab

Figure 6.25. Parameter Type Maintenance Document: Edit Parameter Tab

Table 6.6. Edit Parameter Type Tab: Attributes

| Field | Description |
|---------------------|--|
| Parameter Type Code | Required. Code identifying the parameter type. |

| Field | Description |
|---------------------|---|
| Parameter Type Name | Required. The name by which the Parameter Type Code is commonly known. |
| Active Indicator | An indicator specifying whether an entity in the system is active or not. |

Pessimistic Lock

Pessimistic Lock Lookup

The **Pessimistic Lock Lookup** screen lets you review existing locks and manually release them.

You get to this screen by clicking the **Pessimistic Lock Lookup** option on the **Main Menu**.

Figure 6.26. Pessimistic Lock Lookup

Table 6.7. Pessimistic Lock Lookup: Search Attributes

| Field Name | Description |
|---------------------------|---|
| Lock Owner Principle Name | The name of the person who placed the lock on the document |
| Lock Descriptor | A specification for the type of lock placed on the document |
| Generated Time From | The date and time the lock was put in place |
| Generated Time To | The date and time when the lock expires |
| Document Number | The document ID for the locked document |

Chapter 7. KRAD

KRAD Architecture Overview

The Kualu Rapid Application Development (KRAD) framework and tools are provided to enable universities to more quickly build "rich-UI" web-based and client applications. KRAD expands the set of application features beyond what was formerly available through KNS alone.

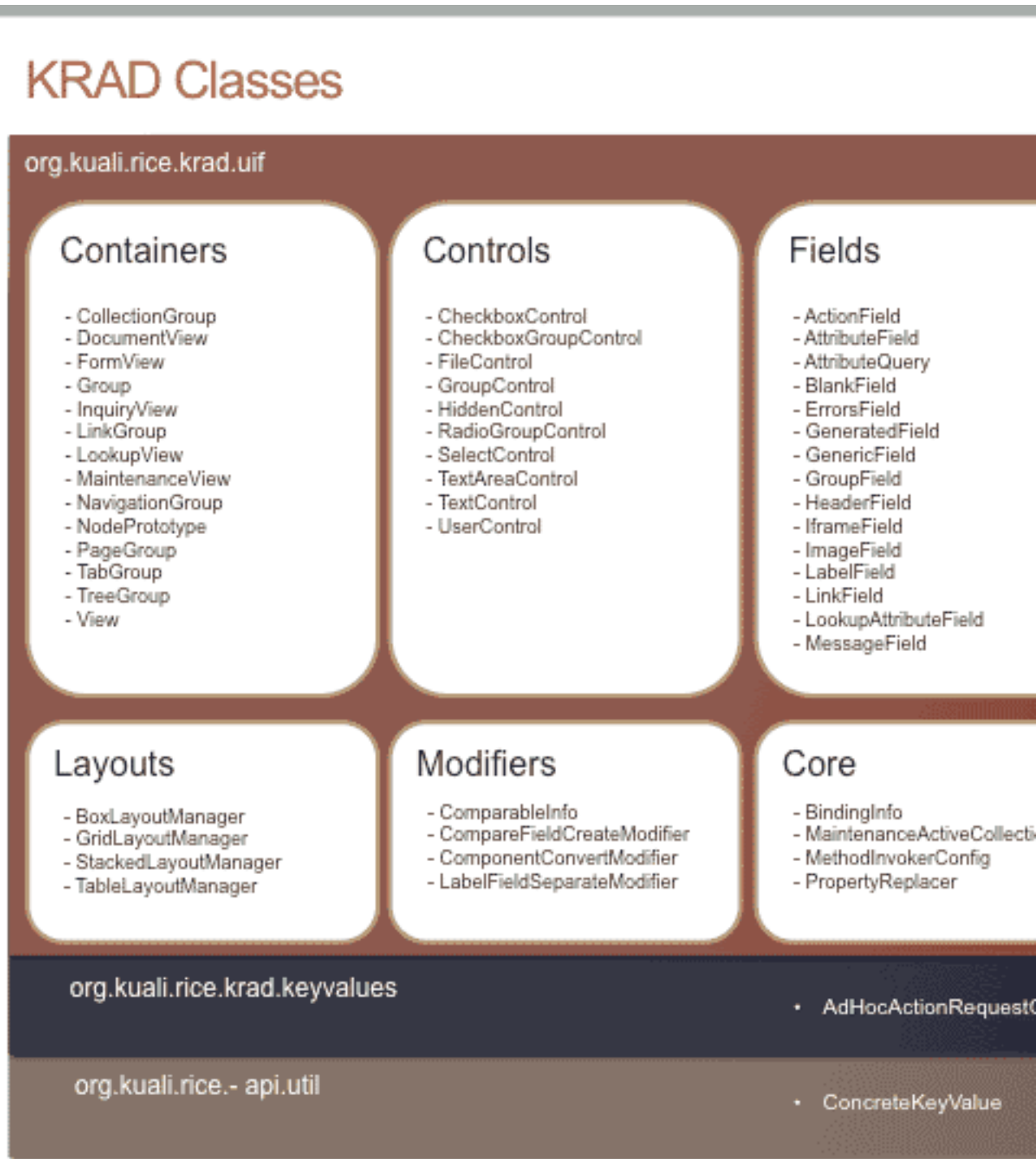
What is KRAD?

- As with KNS, KRAD is a developer framework to enable consistency.
- It adheres to development standards and architectural principles.
- It's a stable core for efficient development.
- It reduces the amount of code you need to write and supports code reuse.

KRAD supports the KNS document types - Lookups, Inquiries, and Maintenance pages - while it also provides more flexibility in user interface layouts, for example, beyond the "vertical" tab section and collection layouts that are supported by KNS.

KRAD Conceptual View

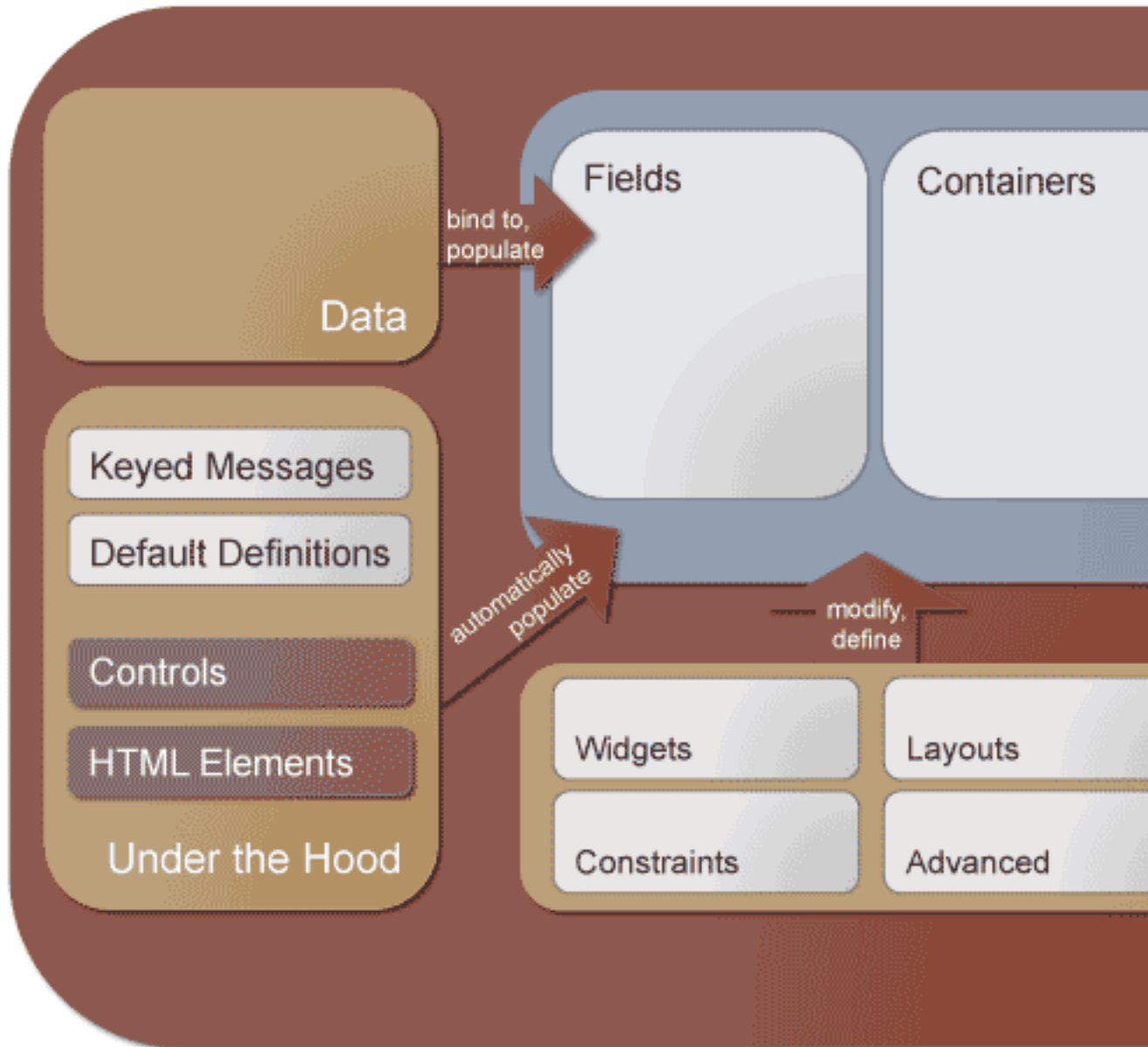
Figure 7.1. KRAD Conceptual View



(more in progress here)

KRAD Relational View

Figure 7.2. KRAD: Relational View



(more in progress here)

KRAD's User Interface Framework

UIF Overview

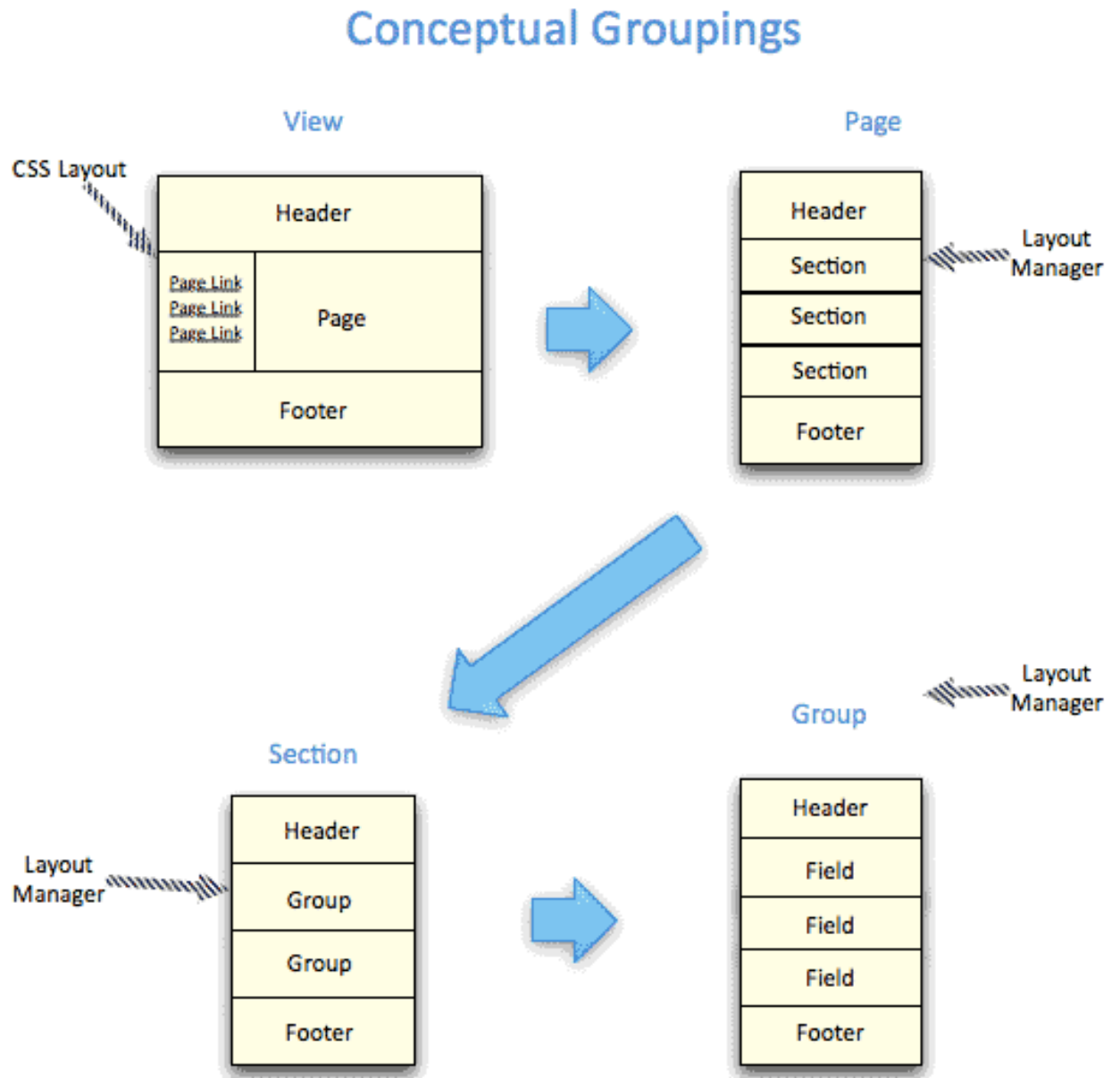
KRAD introduces the concept of a "View" and a hierarchy of pieces which can be added to a view. A view can represent a whole page or even, in some cases, multiple document pages. Inside the view, a number of different groups can be arranged - some as tabs, perhaps, others as field sets within other groups or

even collections of field sets. Some of these groups are made up of fields, controls, and widgets (controls providing an extra level of interactive functionality by incorporating rich user interface techniques).

All of these - views, groups, fields, controls, and widgets - all are components of a KRAD renderer, and therefore all implement the `org.kuali.rice.kns.uif` component interface.

In KRAD, a container is simply a component with a main job of containing other components to render. A view is a container, and several containers have the ability to contain other containers - views can contain groups, for instance, and groups, in turn, can contain any other kinds of components - including other groups. By default, a container such as a view has three areas: 1) a header, encapsulated in a `HeaderField`; 2) a footer, and then 3) any other encapsulated items which make up the body. The `renderHeader` and `renderFooter` properties of the container can be set to `false` to turn off presentation of the header and footer rendering. Body rendering cannot be turned off because the point of the container is to render some kind of contained body elements - the content. However, the body elements are defined in the container (the semantics) but they are laid out on a page by a `LayoutManager` - in this way, you can keep the page layout "rendering" separate from the container's content - the body rendering attributes which define the semantics - enabling you to architect applications so they can more easily be ready for different view windows and devices.

Figure 7.3. KRAD Conceptual Groupings



KRAD Design Components

Overview of Design Components built into KRAD

KRAD Layout Managers

Overview of Layout Managers

Controls, Fields, and Widgets - and even other Groups - can be encapsulated in a Container. The question then becomes how KRAD will lay out those controls.

The designers of KRAD decided to apply a venerable idea present in the first version of the Java Advanced Window Toolkit - the concept of Layout Managers. A LayoutManager is simply a bean which will build the HTML to display Components with a certain algorithm. In this first version of KRAD, there are 4 layout managers for collections (and developers can create additional ones): Grid, Box, Table, and Stack. See material below for additional details.

Grid Layout Manager

The GridLayoutManager places sub-Components into a set of side-by-side columns. For those familiar with KNS, when a maintenance document copies a business object, it has shown four columns side by side: two columns for the original business object's labels and controls and two columns for the copied object's labels and controls. In KRAD, GridLayoutManager can be used to provide this type of layout.

Figure 7.4. Grid Layout

Grid Layout 3 Columns x 2 Rows

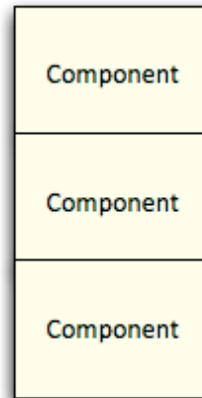
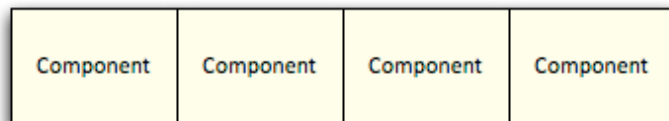
| | | |
|-----------|-----------|------|
| Component | Component | Comp |
| Component | Component | Comp |

Grid Layout N Columns x M Rows

| | | | |
|-----------|-----------|-----|---|
| Component | Component | ... | C |
| ... | ... | | |
| Component | Component | ... | C |

Box Layout Manager

There's also the `BoxLayoutManager`, which works just like the `BoxLayoutManager` in the Java AWT. Nesting Groups with different `BoxLayoutManager` aspects provides a simple yet flexible way to arrange sub-Components. The `BoxLayoutManager` also provides the ability to set the padding in between sub-Components.

Figure 7.5. Box Layout**Box - Vertical****Box - Horizontal****Table Layout Manager**

When laying out the rendering of collections, `TableLayoutManager` provides a lot of power for rendering that data in a tabular fashion. The `TableLayoutManager` holds a `List` of `Fields` and a `List` of `Labels`. For each row in a collection, `TableLayoutManager` will generate a rendered row of the specified `Fields`.

By default, `TableLayoutManager` provides a sequence field for each row of `Fields`, at the very left of each line. This typically shows a number, but through the `sequenceFieldPrototype` property, a different rendering for the sequence can be shown. The sequence fields can be turned off by setting the `renderSequenceField` property to `false`. There's also the ability to specify an `actionFieldPrototype`, which will show up in the rightmost cell of a line of `Fields`, which contains actions that can be performed on that specific row of the collection.

Figure 7.6. Table Layout

Table Layout Manager

ROWS

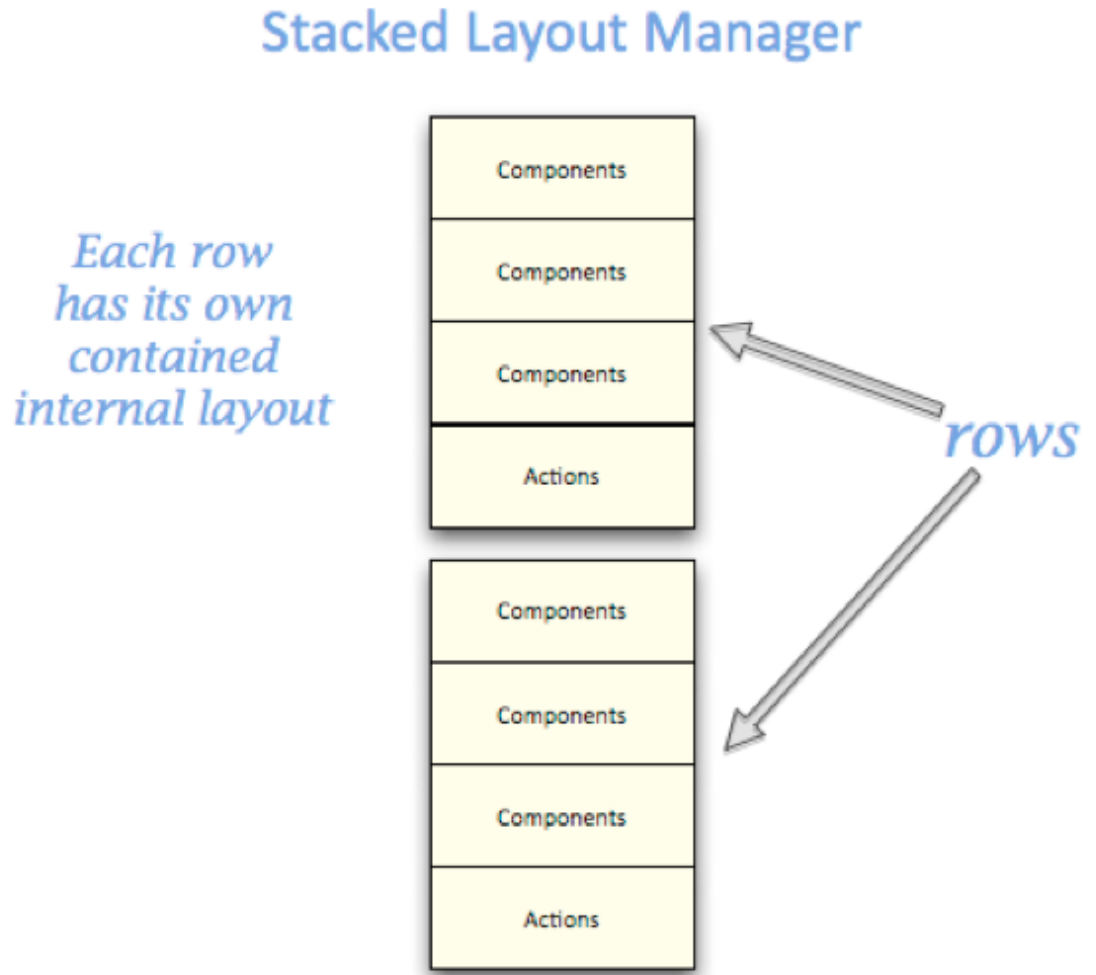
| | Header Label | Header Label | Header Label | |
|------|--------------|--------------|--------------|---------|
| add: | Component | Component | Component | Actions |
| 1 | Component | Component | Component | Actions |
| 2 | Component | Component | Component | Actions |
| 3 | Component | Component | Component | Actions |

Stack Layout Manager

When rendering a Collection, maybe a TableLayoutManager is too much of a straight jacket for each row. Perhaps, for each row in a Collection, a whole different Group needs to be specified which can be customized to show the rows in a very different way than the line of Fields TableLayoutManager provides. A good example comes from the Maintenance Document framework, where collection rows aren't tabular but rather boxed. In cases like this, there's the StackedLayoutManager which takes a List of Groups and renders each row of a collection within that List of Groups.

The StackedLayoutManager provides a way to add a summaryTitle and summaryFields for each collection row: special information that will be featured prominently in rendering so each row line can be quickly recognized for what data it holds. There's also a lineGroupPrototype property which can be used to override the Groups that each line will render as.

Figure 7.7. Stacked layout



KRAD Fields

Developers need ways to interact with the users, to put up text and images, to create interactive and informative web pages in their applications. To accomplish this, there are three families of Components - Fields, Controls, and Widgets - that provide all of the elements that are contained in the Containers. Fields are fundamental - indeed, a large amount of functionality can be created through specifying views, groups, and fields alone and letting fields figure out what text and which controls are needed on a page. Controls and widgets are grouped within fields, which makes fields terribly diverse.

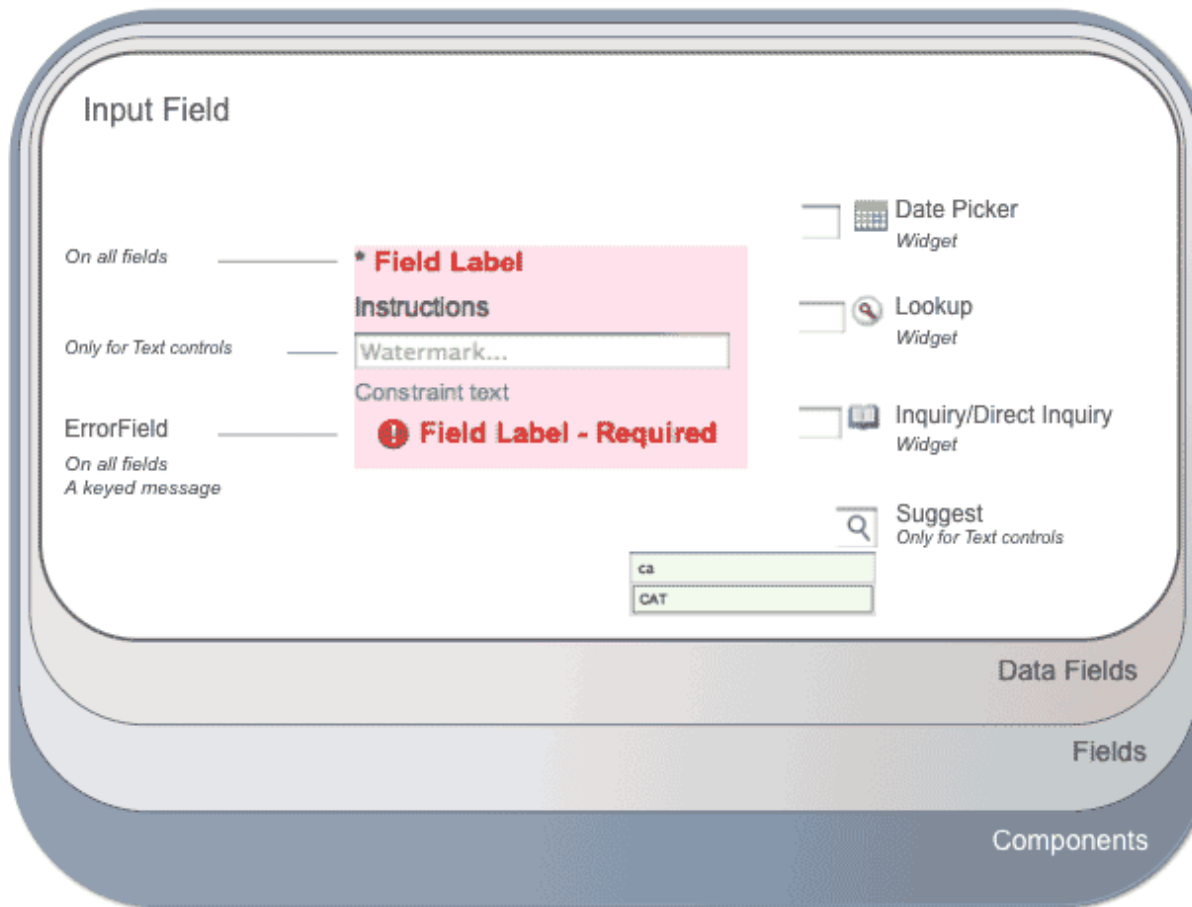
Think of all the different kinds of visual elements on a page: images, buttons, messages, even blank space. Each of these elements can be encapsulated in different kinds of fields.

KRAD Input Fields

An Input field enables user input. This means that this "grouped" field control will display an entry field for user input, and can optionally include instructions, watermarks, constraint text, a lookup widget, inquiry widget, or help widget, and includes a place for error messages associated with the field to appear. This

could arguably be considered the most complex of all fields, and additional information on this field can be found in the Developers' Guide.

Figure 7.8. KRAD Input field - grouped



Other KRAD Fields

There are many other field types provided by KRAD, but none are as complex and rich as the input field. For a full list, see the "KRAD Conceptual View - KRAD classes" in the Architecture Overview section above.

Some key KRAD extensions beyond KNS concepts

Progressive Disclosure

Multi-value Lookups

(other...)

Chapter 8. KRMS

Kuali Rule Management System: Overview

What is KRMS?

Kuali's Rule Management System (KRMS) supports the creation, maintenance, storage and retrieval of business rules and agendas (ordered sets of business rules) within business contexts (e.g., for a particular department or for a particular university-wide process). It allows applications to externalize business logic that commonly needs customization, and empowers business analysts to modify them to reflect changes in policy.

KRMS enables you to define a set of rules within a particular business unit or for a particular set of applications. These business rules define logical conditions and the set of actions that result when those conditions are met. KRMS enables you to call and use this logic from any application, without having to re-write and manage all the rules' maintenance logic within the application.

Example Rule

Previously if a newly hired employee met a set of criteria, they would need to have an eVerify check ran in addition to the standard I-9 process. The initial requirement stated that if the original hire date of an employee is greater than August 1, 2006, the grant is of a Government type, the grant amount is greater than \$5,000, the grant is longer than 2 months, and the employee has not previously completed an eVerify check then their hire document needs to be routed to the special HR eVerify workgroup.

...or put another way

```
IF hr.hire_date > "08/01/2006) and cg.grant_type = "GOV" and
cg.grant_amount > "$5,000" and cg.grant_duration > 60 and
hr.eVerify_comp = "N" THEN ...
```

KRMS Terms and Concepts

- **Agenda** - a collection of rules in a defined plan.
- **Rule** - the logical expression in an Agenda. It consists of two parts, a proposition that returns a true or false value and an action or set of actions. The entire example presented above is a rule.
- **Proposition** - the logic that makes up a rule. A proposition is a single expression of logic that returns true or false. Propositions can be compounded, created using AND, OR, or both to create more complex logic. The entire IF statement in the example above comprises a compound proposition.
- **Action** - the steps to perform in the event that the rule, after being evaluated against the propositions, returns TRUE. In the example stated above this would involve routing the document to the HR eVerify workgroup and presenting a warning on the document. Other examples of actions include:
 - Route to a [PeopleFlow](#)
 - Present a questionnaire
 - Display a validation error

- **Term** - the definition of data that is evaluated in a proposition. `hr.hire_date`, `cg.grant_type`, etc. are examples of Terms in the working example.
- **Term Specification** -
- **Fact** - the actual data for the term being evaluated in a proposition, the data being passed in for evaluation. In the example above, if the grant in question had an amount of \$10,000, then \$10,000 would be the fact.
- **Context** - a collection of agendas, rules, terms, term specifications. In our example a context of "HReVerify" would be established for easy identification of the items related to these business rules.

When to use KRMS?

A big question is, when should you use KRMS rules versus logic directly coded in an application or document?

The KRMS User Interface

KRMS Agenda Editor

Rules in KRMS are placed into ordered sets called Agendas. The order of the Rules in an Agenda determines the sequencing: which rule gets evaluated first, second and so on. The Agenda also enables you to include conditional branching logic between Rules.

In turn, Agendas are created in Contexts, which may represent any categories that are relevant within your institution. For example, they could correspond to document types or business processes or any other categories. In some university environments, the following might be relevant contexts: Awards, Proposals, IRB reviews, Course co-requisites, Course pre-requisites, Student plan evaluations, and so on.

Each Context contains its own agendas, and each Agenda contains its own rules. Rules aren't shared across agendas (though you can copy/paste, they become unique Rule instances), and Agendas aren't shared across Contexts. There is no Context hierarchy, that is, Agendas and Rules can't be inherited across contexts within any sort of hierarchy.

See below for a view of the Agenda Editor in KRMS.

Figure 8.1. KRMS Agenda Editor

Home » Agenda Editor

Agenda Editor

| | |
|-----------------------|-------|
| Document Number: | 3127 |
| Initiator Network Id: | admin |

▼ Agenda

| | |
|--------------|--|
| * Namespace: | Sample App ▼ |
| * Name: | Special Routing <small>size 100</small> |
| * Context: | Travel Account 🔍 |
| Type: | Campus Agenda ▼ |
| Active: | <input checked="" type="checkbox"/> |

Type Attributes

| | |
|--------------------------|---|
| * Campus: | BL 🔍 |
| Optional Test Attribute: | this is an optional attribute for testing |

Rules

add rule
edit rule
Move: [
← left
→ right
↑ up
↓ down
]
✂ cut
📄 paste
✖

[+] expand all
[-] collapse all

- ▼ initiated by admin?
- ▼ **When FALSE**
 - EAT Special Routing [Route to PeopleFlow: EAT special approvers]
 - CAT special routing [Route to PeopleFlow: CAT Special Approvers]

submit
save
blanket approve
close
cancel

Rules in the Agenda can be selected by clicking on them. The order and conditional logic within the agenda can be manipulated by selecting rules and clicking the buttons on the toolbar above the Rules. Clicking on the "Add Rule" button on the Agenda Editor screen will take you to the Rule Editor with a new blank Rule, and selecting a Rule and clicking on the "Edit Rule" button will open that existing Rule in the Rule Editor.

KRMS Rule Editor

See below for views of editing a Rule from an Agenda in KRMS.

Figure 8.2. KRMS Rules Editor

Home » Agenda Editor

Agenda Editor Document Number: 3
Initiator Network Id: a

Agenda: Special Routing

▼ Rule

| | |
|--------------|--|
| Namespace: | KR-SAP |
| Type: | ▼ |
| * Name: | EAT Special Routing <small>size 100</small> |
| Description: | Enter a Rule Description <small>size 4000</small> |
| Active: | <input checked="" type="checkbox"/> |

Propositions

add add parent edit Move: [left right up down] cut paste

Summary:
(Account Type Code = EAT AND Subsidized Percent > 50.0)

[+] expand all [-] collapse all

↓ invoke special routing?

- EAT Acct?
- Account Type Code = EAT
- AND ▼
- subsidy over threshold
- Subsidized Percent > 50.0

And below is the same Rule but scrolled down further on the page to show the Action configuration as well. In this example, when the logical proposition for this rule is satisfied (when it is true), the rule will call a [PeopleFlow](#) to route a request to it.

Figure 8.3. KRMS proposition and PeopleFlow Action

Propositions

add
 add parent
 edit
Move: [left right up down] cut paste

Summary:
(Account Type Code = EAT AND Subsidized Percent > 50.0)

[+] expand all [-] collapse all

↓ invoke special routing?

EAT Acct?
Account Type Code = EAT

AND ▾

subsidy over threshold
Subsidized Percent > 50.0

▼ Action

| | |
|---------------------------|---|
| Type: | <input type="text" value="Route to PeopleFlow"/> |
| * PeopleFlow ID: | <input type="text" value="10000"/> <small>size 40</small> |
| * PeopleFlow Name: | <input type="text" value="EAT special approvers"/> <small>size 40</small> |
| * Name: | <input type="text" value="EAT special approvers"/> <small>size 40</small> |
| Description: | <input type="text" value="instructional text"/> <small>size 4000</small> |

You can add Simple Propositions via "add", or Compound Propositions via "add parent". Simple Propositions are configured with a Term and a user-supplied value to compare it with.

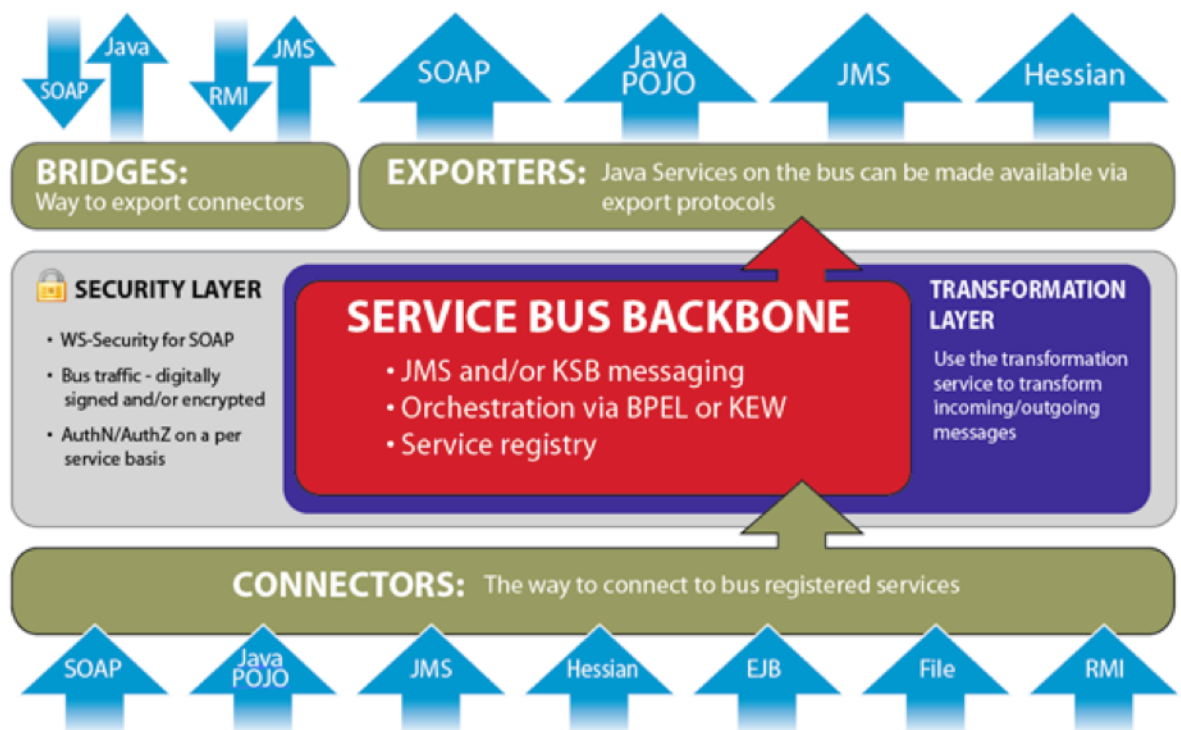
For additional information on KRMS see the KRMS section in the TRG.

Chapter 9. KSB

What is the Kuali Service Bus?

The Kuali Service Bus (KSB) is a lightweight service bus designed to allow developers to quickly develop and deploy services for remote and local consumption. At the heart of the KSB is a service registry. This registry is a listing of all services available for consumption on the bus. The registry provides the bus with the information necessary to achieve load balancing, failover, and more.

Figure 9.1. Kuali Service Bus



You can deploy services to the bus using Spring or programmatically. Services must be named when they are deployed to the bus. Services are acquired from the bus using their name.

Features

- **Transactional Asynchronous Messaging** - You can call services asynchronously to support a 'fire and forget' model of calling services. Messaging participates in existing JTA transactions, so that messages are not sent until the currently running transaction is committed and are not sent if the transaction is rolled back. You can increase the performance of service calling code because you are not waiting for a response.
- **Synchronous Messaging** - Call any service on the bus using a request response paradigm.
- **Queue Style Messaging** - Supports executing Java services using message queues. When a message is sent to a queue, only one of the services listening for messages on the queue is given the message.

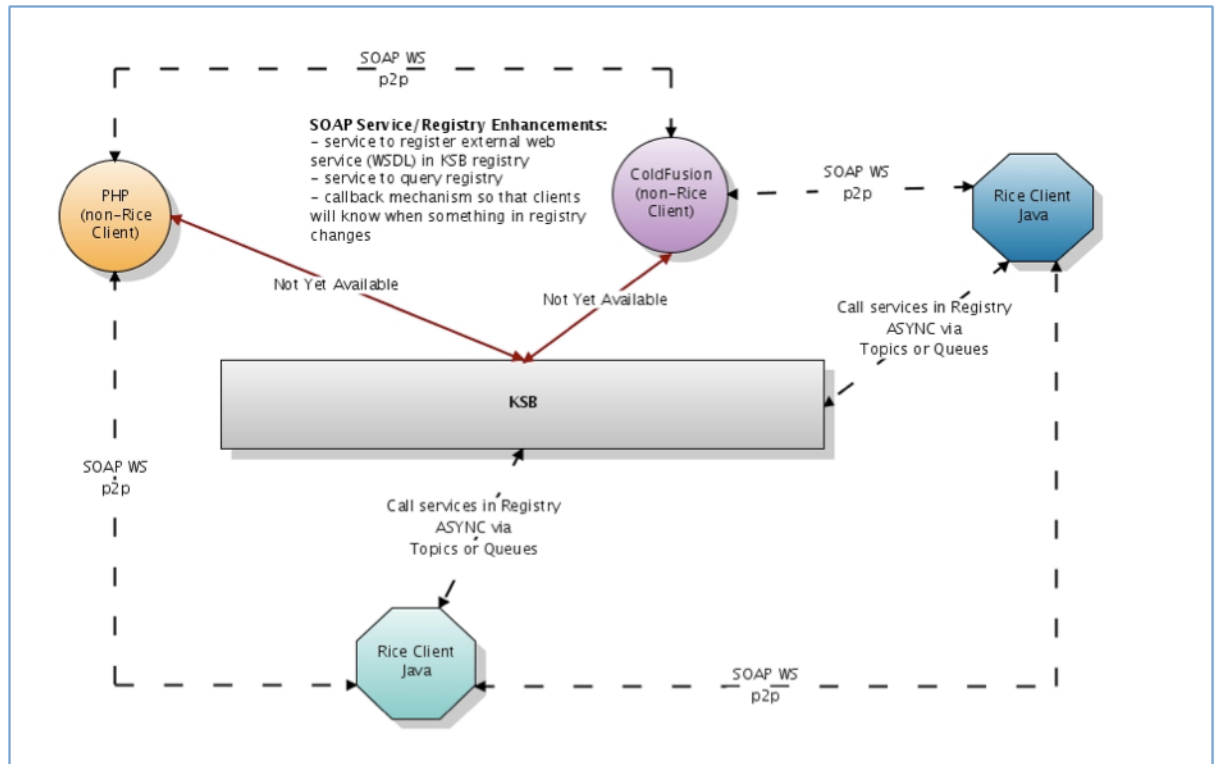
- **Topic Style Messaging** - Supports executing Java services using messaging topics. When a message is sent to a topic, all services that are listening for messages on the topic receive the message.
- **Quality of Service** - Determines how queues and topics handle messages that have problems. Time to live is supported, giving the message a configured amount of time to be handled successfully before exception handling is invoked for that message type. Messages can be given a number of retry attempts before exception handling is invoked. The delay separating each call increases. Exception handlers can be registered with each queue and topic for custom behavior when messages fail and Quality of Service limits have been reached.
- **Discovery** - Services are automatically discovered along the bus by service name. End-point URLs are not needed to connect to services.
- **Reliability** - Should problems arise, messages sent to services via queues or synchronous calls automatically fail-over to any other services bound to the same name on the bus. Services that are not available are removed from the bus until they come back online, at which time they will be rediscovered for messaging.
- **Persisted Callback** - Callback objects can be sent with any message. This object will be called each time the message is received with the response of the service (think topic as opposed to queue). In this way, we can deploy services for messaging that actually return values.
- **Primitive Business Activity Monitoring** - If turned on, each call to every service, including the parameters pass into that service, is recorded. This feature can be turned on and off at runtime.
- **Spring-Based Integration** - KSB is designed with Spring-based integration in mind. A typical scenario is making an existing Spring-based POJO available for remote asynchronous calls.
- **Programmatic Based Integration** - KSB can be configured programmatically if Spring configuration is not desired. Services can also be added and removed from the bus programmatically at runtime.

Bean-Based Services

Typically, KSB programming is centered on exposing Spring-configured beans to other calling code using a number of different protocols. Using this paradigm the client developer and the organization can rapidly build and consume services, often a daunting challenge using other buses.

Overview of Supported Service Protocols

Figure 9.2. Supported Service Protocols



This drawing is conceptual and not representative of true deployment architecture. Essentially, the KSB is a registry with service-calling behavior on the client end (Java client). All policies and behaviors (async as opposed to sync) are coordinated on the client. The client offers some very attractive messaging features:

- **Synchronization** of message sending with currently running transaction (meaning all messages sent during a transaction are ONLY sent if the transaction is successfully committed)
- **Failover** - If a call to a service comes back with a 404 (or various other network-related errors), it will try to call other services of the same name on the bus. This is for both sync and async calls.
- **Load balancing** - Clients will round-robin call services of the same name on the bus. Proxy instances, however, are bound to single machines if you want to keep a line of communication open to a single machine for long periods of time.
- **Topics and Queues**
- **Persistent messages** - When using message persistence a message cannot be lost. It will be persisted until it is sent.
- **Message Driven Service Execution** - Bind standard JavaBean services to messaging queues for message driven beans.

Message Queue

Use the Message Queue section to administer the KNS message queuing system. You can find it on the Administration menu.

It has three main sections: Current Node Info, the message filter and fetch section, and the Documents currently in route queue section.

Figure 9.3. Message Queue: Documents Currently In Route

The screenshot shows the KSB Message Queue interface. At the top, there is a 'workflow' logo and a 'Refresh Page' link. Below this is the 'Current Node Info' section, which displays the following details: IP Address: 65.60.44.250, Service Namespace: RICE, message.persistance: true, message.delivery: async, and message.off: (empty). Below the info is a filter section with input fields for Message ID, Service Name, Service Namespace, IP Number, Queue Status (a dropdown menu), App Specific Value 1, and App Specific Value 2. A 'Filter' button is located below these fields. Below the filter section is a '50 Execute Message Fetcher' button. The main section is titled 'Documents currently in route queue: 4' and shows '4 items retrieved, displaying all items.' Below this is a table with the following data:

| Message Queue Id | Service Name | Message Entity | IP Number | Queue Status | Queue Priority | Queue Date | Expiration Date | Retry Count | App Specific Value 1 | App Specific Value 2 | Actions |
|------------------|--|----------------|--------------|--------------|----------------|---------------------|---------------------|-------------|----------------------|----------------------|-------------------|
| 2340 | {TRAVEL}BlanketApproveProcessorService | TRAVEL | 129.79.44.31 | EXCEPTION | 4 | 04:43 PM 08/11/2008 | 04:43 PM 08/11/2008 | 1 | 2120 | | View Edit ReQueue |
| 63 | {TRAVEL}DocumentRoutingService | TRAVEL | 129.79.44.31 | EXCEPTION | 5 | 01:21 PM 12/22/2008 | 01:21 PM 12/22/2008 | 1 | 2694 | | View Edit ReQueue |
| 62 | {TRAVEL}DocumentRoutingService | TRAVEL | 129.79.44.31 | EXCEPTION | 5 | 01:09 PM 12/22/2008 | 01:09 PM 12/22/2008 | 1 | 2693 | | View Edit ReQueue |
| 61 | {TRAVEL}DocumentRoutingService | TRAVEL | 129.79.44.31 | EXCEPTION | 5 | 01:04 PM 12/22/2008 | 01:04 PM 12/22/2008 | 1 | 2692 | | View Edit ReQueue |

Below the table, it says '4 items retrieved, displaying all items.' At the bottom of the screenshot, there is a copyright notice: 'Copyright 2005-2007 The Kual Foundation. All rights reserved. Portions of Kual Rice are copyrighted by other parties as described in the Acknowledgments screen.'

Current Node Info

- **IP Address:** This value equals the IP address of the machine: Rice
- **message.persistance:** If true, then messages will be persisted to the datastore. Otherwise, they will only be stored in memory. If message persistence is not turned on and the server is shutdown while there are still messages in queue, those messages will be lost. For a production environment, it is recommended that message persistence be set to true.
- **message.delivery:** Can be set to either "synchronous" or "async". If this is set to synchronous, then messages that are sent in an asynchronous fashion using the KSB application interface (API) will be sent synchronously. This is useful in certain development and unit testing scenarios. For a production environment, it is recommended that message delivery be set to async.
- **message.off:** If set to "true" then asynchronous messages will not be sent. In the case that message persistence is turned on, they will be persisted in the message store and can even be picked up later using the Message Fetcher. However, if message persistence is turned off, these messages will be lost. This can be useful in certain debugging or testing scenarios.

Message Filter and Fetch

The message filter and fetch section of the Message Queue screen lets you search for, filter, and/or isolate messages in the Documents in route queue. To use the Message Filter section, enter your criteria and click the Filter button:

Figure 9.4. Message Filter Screen

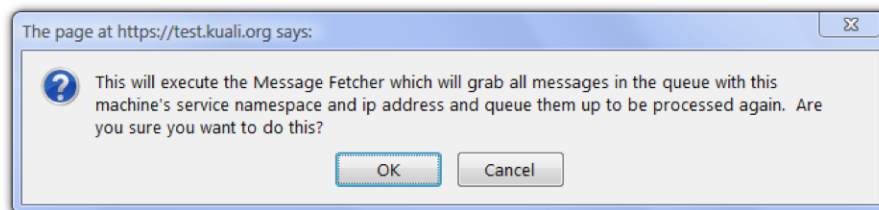
| | |
|-----------------------|---------------------------------------|
| Message ID: | <input type="text"/> |
| Service Name: | <input type="text"/> |
| Service Namespace: | <input type="text"/> |
| IP Number: | <input type="text"/> |
| Queue Status: | <input type="text" value="▼"/> |
| App Specific Value 1: | <input type="text"/> |
| App Specific Value 2: | <input type="text"/> |
| | <input type="button" value="Filter"/> |

Table 9.1. Message Filter Screen: Attributes

| Field | Description |
|----------------------|--|
| Message ID | A unique 5-digit message queue identification number |
| Service Name | The name of the service |
| Application ID | The service container's identifier |
| IP Number | The message initiator's IP address |
| Queue Status | You can sort documents by the queue status. The queue status may be: <ul style="list-style-type: none"> • QUEUED: The message is waiting for a worker thread to pick it up • ROUTING: A worker is currently working on the message. • EXCEPTION: There is a problem with the message and the route manager will ignore it. EXCEPTION status is typically set manually by the administrator to suspend a route queue entry until a problem can be diagnosed. |
| App Specific Value 1 | The specific value of a document |
| App Specific Value 2 | The specific value of a document |
| Filter Button | Click to execute the search filter |

The Execute Message Fetcher button retrieves all the messages in the route queue. You can adjust the number of messages requested by entering a number in the field left of the button.

When you click the Execute Message Fetcher button, a dialog box appears, confirming that you want to execute this command:

Figure 9.5. Execute Message Filter: Confirmation Screen

KSB displays the results of a search and/or filter at the bottom of the page in the Documents currently in route queue table.

Figure 9.6. Documents In Route Queue

| Message Queue Id | Service Name | Message Entity | IP Number | Queue Status | Queue Priority | Queue Date | Expiration Date | Retry Count | App Specific Value 1 | App Specific Value 2 | Actions |
|------------------|--|----------------|--------------|--------------|----------------|------------------------|------------------------|-------------|----------------------|----------------------|---|
| 2487 | {TRAVEL}DocumentRoutingService | TRAVEL | 129.79.44.31 | EXCEPTION | 5 | 10:54 AM 09/24/2008 | 10:54 AM 09/24/2008 | 1 | 2281 | | View Edit ReQueue |
| 2340 | {TRAVEL}BlanketApproveProcessorService | TRAVEL | 129.79.44.31 | EXCEPTION | 4 | 04:43 PM 08/11/2008 | 04:43 PM 08/11/2008 | 1 | 2120 | | View Edit ReQueue |
| 63 | {TRAVEL}DocumentRoutingService | TRAVEL | 129.79.44.31 | EXCEPTION | 5 | 01:21 PM 12/22/2008 | 01:21 PM 12/22/2008 | 1 | 2694 | | View Edit ReQueue |
| 62 | {TRAVEL}DocumentRoutingService | TRAVEL | 129.79.44.31 | EXCEPTION | 5 | 01:09 PM 12/22/2008 | 01:09 PM 12/22/2008 | 1 | 2693 | | View Edit ReQueue |
| 61 | {TRAVEL}DocumentRoutingService | TRAVEL | 129.79.44.31 | EXCEPTION | 5 | 01:04 PM 12/22/2008 | 01:04 PM 12/22/2008 | 1 | 2692 | | View Edit ReQueue |

Documents Currently in Route Queue

Table 9.2. Documents Currently in Route Queue: Attributes

| | |
|----------------------|---|
| Field | |
| Message Queue ID | A unique 5-digit message queue identification number. This is the same as the Message ID in the Message Filter section. |
| Service Name | The name of the service |
| Message Entity | |
| IP Number | The message initiator's IP address |
| Queue Status | You can sort documents by the queue status. The queue status may be: <ul style="list-style-type: none"> • QUEUED: The message is waiting for a worker thread to pick it up • ROUTING: A worker is currently working on the message. • EXCEPTION: There is a problem with the message and the route manager will ignore it. EXCEPTION status is typically set manually by the administrator to suspend a route queue entry until a problem can be diagnosed. |
| Queue Priority | The priority of the entry in the queue. Entries with the smallest number are processed first. |
| Queue Date | The date on which the queue entry should be processed. If the queue checker runs and discovers entries that have a queue date that are equal to or earlier than the current time, it processes them. The approximate time at which this screenshot was taken 4:53 PM. |
| Expiration Date | |
| Retry Count | |
| App Specific Value 1 | |
| App Specific Value 2 | |
| Actions | Click a link in this field to: <ul style="list-style-type: none"> • View: View the detail message report • Edit: Edit the settings of a Message Entry • ReQueue: Enforce the routing process |

View

When you click View in the Actions menu, KSB displays information about that message. Most of the initial information is the same as that displayed in the Documents currently in route queue table. Additional information on the View screen:

- **Message**

Table 9.3. Message: Attributes

| Field | Description |
|-----------------|------------------------------------|
| Application ID: | The service container's identifier |
| Method Name: | |

- **Payload**

Table 9.4. Payload: Attributes

| Field | Description |
|-------------------------------------|---|
| Payload Class | The class of the Payload |
| Method Name | The name of the method used in this document |
| ignoreStoreAndForward | A true and false indicator that ignores the store functions and forwards the message |
| ServiceInfo.messageEntryId | A unique 4-digit message entry identification number |
| ServiceInfo.ServiceNamespace | The application |
| ServiceInfo.serverIp | The server's IP address |
| ServiceInfo.ServiceName | The name of the service |
| ServiceInfo.endpointUrl | The web address of the service |
| ServiceInfo.queue | A true and false indicator that activates the queue or topic function: <ul style="list-style-type: none"> • "True" uses the Queue method, which sends the message to one contact at a time • "False" uses the Topic method, which sends the message to all contacts at once |
| ServiceInfo.alive | A true and false indicator that shows the activity state of the document |
| ServiceInfo.priority | The priority of the entry for execution. Entries with the smallest number are processed first |
| ServiceInfo.retryAttempts | How many times KSB will try to resend the message |
| ServiceInfo.millisToLive | An expiration indicator: <ul style="list-style-type: none"> • 1 means the message never expires |
| ServiceInfo.messageExceptionHandler | This provides a reference the service can use to call back. |
| ServiceInfo.serviceclass | The name of the service class |
| ServiceInfo.busSecurity | A true and false indicator that assigns the security function |
| ServiceInfo.credentialsType | The credential type of the document |
| Arguments | The argument of this document |

- **Edit**

When you click **Edit** in the **Actions** menu, KSB displays the editable fields for that message. Fields on the Edit screen:

Table 9.5. Edit Screen: Attributes

| Field | Description |
|----------------|--|
| Queue Priority | Change the queue priority by entering a positive number. A smaller number has higher priority for execution. |
| Queue Status | Change the status to Queued, Routing, or Exception. |
| Retry Count | Change the number of times KSB will retry. |
| IP Number | Change the initiator's IP address. |
| Service Name | Change the name of the service. |
| Message Entity | Change the message entity. |

| Field | Description |
|----------------------|--|
| Method Name | Change the method. |
| App Specific Value 1 | Change the information for the specific value 1. |
| App Specific Value 2 | Change the information for the specific value 2. |

Functional links on the Edit page:

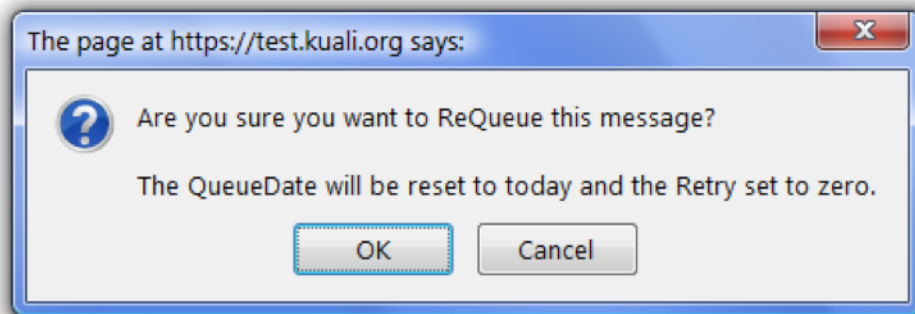
Table 9.6. Edit Screen: Links

| Field | Description |
|---------------------------|--|
| Save Changes | Save the information you just changed. |
| Save Changes and Resubmit | Save the information you changed and resubmit the message. |
| Save and Forward | Save the message and send it to the next contact. |
| Delete | Delete the message. |
| Reset | Reload the previous settings. This undoes the changes that you made on this screen, as long as you haven't yet saved them. |
| Clear Message | Clear all information fields on this page. |

- **ReQueue**

When you click ReQueue in the Actions menu, KSB displays this pop-up message:

Figure 9.7. Requeue Documents: Confirmation Screen



Thread Pool

Thread pool is a feature that improves overall system performance by creating a pool of threads which can be independently used by the system to execute multiple tasks at the same time. A task can execute immediately if there is a thread in the pool that is available. If no thread is available, the task waits for a thread to become available from the pool before executing.

The **Thread Pool** screen is accessed from the **Administration** menu. It tells you the current state of the Thread Pool and allows you to change four parameters for the Thread Pool. The core pool size, the maximum pool size, the RouteQueue.TimeIncrement and the RouteQueue.maxRetryAttempts.

Figure 9.8. Thread Pool Administration Page

Core Pool Size:

Maximum Pool Size:

Pool Size: 5

Active Count: 0

Largest Pool Size: 5

Keep Alive Time: 60000

Task Count: 112

Completed Task Count: 112

RouteQueue.TimeIncrement:

RouteQueue.maxRetryAttempts:

Execute Across All Servers with Service Namespace RICE

Table 9.7. Thread Pool: Attributes

| Field | Description |
|---|--|
| Core Pool Size | A positive number equal to the core number of threads in the pool |
| Maximum Pool Size | A positive number equal to the maximum number of threads in the pool; when the Core Pool Size is larger than the Maximum Pool Size, Maximum Pool Size automatically sets the pool size equal to the Core Pool Size |
| Pool Size | The current number of threads in the pool |
| Active Count | The approximate number of threads that are actively executing tasks |
| Largest Pool Size | Maximum number of threads allowed in the Thread Pool |
| Keep Alive Time | The amount of time which threads in excess of the core pool size may remain idle before being terminated; measured in milliseconds; for example, 60,000 milliseconds = 60 seconds |
| Task Count | Number of tasks that have been scheduled for execution |
| Completed Task Count | Number of tasks that have completed execution |
| Execute Across All Servers with Application ID RICE | When you click this checkbox, then click the Update button, the update is applied across all servers. |
| Update button | Click the Update button to execute the changes you entered in the editable fields above. |

Service Registry

The Service Registry lists published and temporary services that are available for the local machine. You cannot configure the service registry here; this is only information about the registry.

Display this page by clicking the **Service Registry** link on the Rice **Administration** page.

At the top of the page, the **Current Node Info** table shows the settings and configuration of the local machine:

The returned table of services is divided into three sections:

1. **Published Services:** Services in use by the local machine
2. **Published Temp Services:** Temporary services that are the result of Object Remoting. For more information about Object Remoting, please refer to the Object Remoting section of the KSB portion of the Technical Reference Guide.
3. **All Registry Services**

This screen print shows the top of a Service Registry page, with the **Current Node Info** table and the beginning of the **Published Services** table, as well as the refresh link and button:

Figure 9.9. Service Registry

The screenshot shows a web interface for the Service Registry. At the top left, there is a 'Current Node Info' box with the following details: IP Address: 65.60.44.250, Service Namespace: RICE, and dev.mode: false. Below this is a 'Refresh Service Registry' button. The main part of the page is a table titled 'Published Services' with the following columns: Service Name, Endpoint URL, Service Namespace, IP Number, and Alive. The table lists various services such as ImmediateEmailService, OSCacheNotificationService, and (KIM)activePrincipalRoleTypeService, all with an IP Number of 65.60.44.250 and an Alive status of true.

| Service Name | Endpoint URL | Service Namespace | IP Number | Alive |
|--|--|-------------------|--------------|-------|
| ImmediateEmailService | http://65.60.44.250:8080/kr-dev/remoting/ImmediateEmailService | RICE | 65.60.44.250 | true |
| ImmediateEmailService-forwardHandler | http://65.60.44.250:8080/kr-dev/remoting/ImmediateEmailService-forwardHandler | RICE | 65.60.44.250 | true |
| OSCacheNotificationService | http://65.60.44.250:8080/kr-dev/remoting/OSCacheNotificationService | RICE | 65.60.44.250 | true |
| OSCacheNotificationService-forwardHandler | http://65.60.44.250:8080/kr-dev/remoting/OSCacheNotificationService-forwardHandler | RICE | 65.60.44.250 | true |
| RuleCacheProcessorService | http://65.60.44.250:8080/kr-dev/remoting/RuleCacheProcessorService | RICE | 65.60.44.250 | true |
| RuleCacheProcessorService-forwardHandler | http://65.60.44.250:8080/kr-dev/remoting/RuleCacheProcessorService-forwardHandler | RICE | 65.60.44.250 | true |
| WorkgroupMembershipChangeService | http://65.60.44.250:8080/kr-dev/remoting/WorkgroupMembershipChangeService | RICE | 65.60.44.250 | true |
| WorkgroupMembershipChangeService-forwardHandler | http://65.60.44.250:8080/kr-dev/remoting/WorkgroupMembershipChangeService-forwardHandler | RICE | 65.60.44.250 | true |
| {KIM}activePrincipalRoleTypeService | http://65.60.44.250:8080/kr-dev/remoting/activePrincipalRoleTypeService | RICE | 65.60.44.250 | true |
| {KIM}activePrincipalRoleTypeService-forwardHandler | http://65.60.44.250:8080/kr-dev/remoting/activePrincipalRoleTypeService-forwardHandler | RICE | 65.60.44.250 | true |
| {KIM}documentEditorRoleTypeService | http://65.60.44.250:8080/kr-dev/remoting/documentEditorRoleTypeService | RICE | 65.60.44.250 | true |
| {KIM}documentEditorRoleTypeService-forwardHandler | http://65.60.44.250:8080/kr-dev/remoting/documentEditorRoleTypeService-forwardHandler | RICE | 65.60.44.250 | true |
| {KIM}documentOpenerRoleTypeService | http://65.60.44.250:8080/kr-dev/remoting/documentOpenerRoleTypeService | RICE | 65.60.44.250 | true |

To update the list of published services, use either the **Refresh Page** link in the header at the top of the page or the "Refresh Service Registry" button.

This screen print shows the point on a Service Registry page where KSB displays a notation that there are no published temporary services and the beginning of the **All Registry Services** table:

Figure 9.10. Service Reigstry Results

Published Services:

| Service Name | Endpoint URL |
|---|--|
| {TRAVEL}sampleAppPeopleFlowTypeService | http://dev1.rice.kuali.org/remoting/samp |
| {http://rice.kuali.org/core/v2_0}componentService | http://dev1.rice.kuali.org/remoting/soap |
| {http://rice.kuali.org/core/v2_0}coreServiceCacheAdminService | http://dev1.rice.kuali.org/remoting/soap |
| {http://rice.kuali.org/core/v2_0}namespaceService | http://dev1.rice.kuali.org/remoting/soap |
| {http://rice.kuali.org/core/v2_0}parameterRepositoryService | http://dev1.rice.kuali.org/remoting/soap |
| {http://rice.kuali.org/core/v2_0}styleRepositoryService | http://dev1.rice.kuali.org/remoting/soap |
| {http://rice.kuali.org/ken/v2_0}sendNotificationService | http://dev1.rice.kuali.org/remoting/soap |
| {http://rice.kuali.org/kew/v2_0}actionInvocationQueue | http://dev1.rice.kuali.org/remoting/soap |
| {http://rice.kuali.org/kew/v2_0}actionListCustomizationHandlerService | http://dev1.rice.kuali.org/remoting/soap |
| {http://rice.kuali.org/kew/v2_0}actionListService | http://dev1.rice.kuali.org/remoting/soap |
| {http://rice.kuali.org/kew/v2_0}actionRequestDerivedRoleTypeService | http://dev1.rice.kuali.org/remoting/soap |
| {http://rice.kuali.org/kew/v2_0}adHocReviewPermissionTypeService | http://dev1.rice.kuali.org/remoting/soap |
| {http://rice.kuali.org/kew/v2_0}documentAttributeIndexingQueue | http://dev1.rice.kuali.org/remoting/soap |
| {http://rice.kuali.org/kew/v2_0}documentOrchestrationQueue | http://dev1.rice.kuali.org/remoting/soap |

Please note, you may have permissions that allow you to click on a row's Endpoint URL to view the WSDL fiels associated with the given service. In Internet Explorer or Firefox, this WSDL will be displayed normally in a separate window. In Google Chrome or Safari, however, you will need to click the link then right click to view the frame source to see the WSDL due to current restricts in Chrome and Safari.

Quartz

The Kuali Service Bus (KSB) uses Quartz to schedule delayed tasks, including retry attempts for messages that cannot be sent the first time. By default, KSB uses an embedded quartz scheduler that can be configured by passing parameters starting with "ksb.org.quartz." into the Rice configuration.

You can inject a custom quartz scheduler if the application is already running one. See the Technical Reference Guide for KSB, Configuring Quartz for KSB for more information.

Quartz is also known as the **Exception Routing Queue**.

Figure 9.11. Exception Routing Queue

The screenshot shows a web interface titled 'workflow' with a 'Refresh Page' link. Below the header, it states '3 items retrieved, displaying all items.' A table with 6 columns is displayed: Job Name, Job Group, Description, Time to execute, FullName, and Actions. The table contains three rows of job data. Below the table, it again states '3 items retrieved, displaying all items.' At the bottom, there is a copyright notice: 'Copyright 2005-2007 The Kualii Foundation. All rights reserved. Portions of Kualii Rice are copyrighted by other parties as described in the Acknowledgments screen.'

| Job Name | Job Group | Description | Time to execute | FullName | Actions |
|----------------------------|--------------|---|------------------------------|---|----------------------|
| MessageProcessingJobDetail | KCB-Delivery | Job that handles asynchronous delivery and dismissal of MessageDeliveries | Wed May 27 08:15:40 CDT 2009 | KCB-Delivery.MessageProcessingJobDetail | Put in message queue |
| Daily Email | Email Batch | | Thu May 28 01:00:00 CDT 2009 | Email Batch.Daily Email | Put in message queue |
| Weekly Email | Email Batch | | Mon Jun 01 02:00:00 CDT 2009 | Email Batch.Weekly Email | Put in message queue |

When you click the Quartz link on the Kualii Rice Portal Administration page, KSB displays the screen shown above. The contents of the table can be sorted in ascending or descending order by clicking on a column title. This technique works for all columns except Actions. The table contains this information on each job that is scheduled:

Table 9.8. Exception Routing Queue: Attributes

| Field | Description |
|-----------------|---|
| Job Name | Unique name for the job |
| Job Group | Classification of the job |
| Description | Text description of what this job does |
| Time to execute | The scheduled date and time for the job to occur |
| FullName | A more descriptive Job Name |
| Actions | Put in message queue effectively is a button that takes that message out of quartz and sends it back into the KSB to be retried without waiting until the scheduled time. |

Security Management

For client applications to consume secured services hosted from a standalone Rice server, the implementer must generate a keystore in KSB. KSB security relies on the creation of a keystore using the JVM keytool.

Figure 9.12. Create Keystore

Copyright 2005-2007 The Kuali Foundation. All rights reserved.
Portions of Kuali Rice are copyrighted by other parties as described in the [Acknowledgments](#) screen.

To create a new Client Keystore file, complete all three fields and click the create button that is just below the fields:

Figure 9.13. Create Keystore: File Section

Create new Client Keystore file:

| | |
|---|--------------------------|
| Desired Alias (must be unique from list below): | <input type="text"/> |
| Password (will be used for keys as well as keystore): | <input type="password"/> |
| Re-Enter Password: | <input type="password"/> |
| <input type="button" value="create"/> <input type="button" value="clear fields"/> | |

The **Desired Alias** (name for the new keystore you are creating) must be unique among your keystores. KSB automatically displays a list of existing Keystore entries for your reference below the *Create new Client Keystore file* table. The data in this list can be sorted in ascending or descending order by clicking the column heading for any column except *Actions*.

Figure 9.14. Create Keystore: Existing Keystore Section

Existing Keystore Entries:
One item retrieved.

| Alias | Create Date | Type | Actions |
|-------|---------------------|-------------|---------|
| rice | 01:59 PM 10/10/2007 | Private Key | |

One item retrieved.

Table 9.9. Existing Keystore Entries: Attributes

| Field | Description |
|-------------|--|
| Alias | Keystore name |
| Create Date | Date and time the keystore was created |
| Type | The type of keystore |
| Actions | |

Chapter 10. KEW Administration Guide

Kuali Enterprise Workflow: Overview

An extension to the KEW Users Guide, this guide is aimed at providing guidance on the more powerful, administrative tools provided.

Tools covered in this guide:

- Document Operation

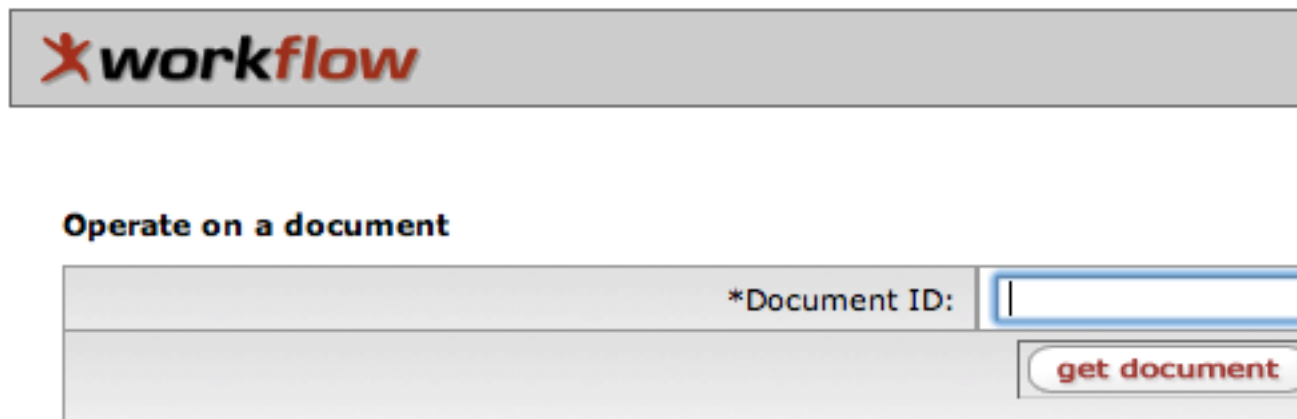
Your institution and Kuali Rice based application may use some or all of these KEW features.

Document Operation

The Document Operation screen allows for low-level modifications to document data. It's available from the Administrator channel in the portal.

In certain scenarios or failure cases it may be necessary to make modifications to the document so that the state of the document in the KEW system is consistent with that of the integrating application. It may also be necessary to make modifications to the XML content of a document if, for example, there is a bug in the application integrating with KEW which results in incomplete or insufficient XML content to allow for proper routing.

Figure 10.1. Initial Screen



The screenshot shows the initial screen for document operation. At the top, there is a grey header bar with the 'workflow' logo in red and black. Below the header, the section title 'Operate on a document' is displayed in bold black text. The main content area is a light grey form. On the right side of the form, there is a label '*Document ID:' followed by an empty text input field. Below the input field, there is a button with the text 'get document' in red.

The initial screen prompts for the ID of a document to load. The administrator is then presented with a view of the document and can perform various operations on it. The screen is divided into various sections, including:

- Document Actions - Additional functions for reassigning and reprocessing document
- Document - simple data associated with the document
- Action Requests - the Action Requests associated with the document, includes requests for action which have already been satisfied

- Actions Taken - The actions that have been taken against this document (i.e. Approved by user X)
- Action Items - Items related to this document that are in users' Action Lists
- Route Node Instances - The node instances that form the document's instantiated route path
- Branch States - The branches on this document and the state of those branches
- Annotation - An annotation that will show up on the Route Log when the operation is performed.

Each of the pieces of data within the aforementioned sections has a set of radio buttons at the top that indicates Update, Delete, or No Operation. No Operation is the default. If it is desired to change or delete one of these pieces of data then the appropriate button should be selected. This is to guard against unintended or accidental changes to the document.

Each of the different sections is described in detail below.

Document Actions

Figure 10.2. Document Actions

| | | Docum |
|--------------------------------|---------------------------------------|---------------------------------------|
| Queue Document | <input type="button" value="submit"/> | |
| Index Searchable Attributes | <input type="button" value="submit"/> | |
| Queue Document Refresh | <input type="button" value="submit"/> | |
| Queue Document Blanket Approve | <input type="button" value="submit"/> | User: <input type="text"/> |
| Queue Document Move | <input type="button" value="submit"/> | Action Taken Id: <input type="text"/> |
| | | Node Names: <input type="text"/> |
| Queue Action Invocation | <input type="button" value="submit"/> | User: <input type="text"/> |
| | | Action Item Id: <input type="text"/> |
| | | Action Code: <input type="text"/> |

Several functional buttons are added under the Document Action section:

- Queue Document - Requeuing and reprocessing the document by the engine
- Index Searchable Attributes - Update searchable data of the document
- Queue Document Requeuer - Refresh document and regenerate request of current node
- Queue Document Blanket Approve - Move blanket approve document forward; User, Action Taken Id, and Node Names are required

- User - Enter initiator's network Id
- Action Taken Id - Enter an entry's action taken Id
- Node Names - Enter a comma separated list of node names
- Queue Document Move - Move document forward or backward; User, Action Taken Id, and Node Names are required
 - User - Enter initiator's network Id
 - Action Taken Id - Enter an entry's action taken Id
 - Node Names - Enter a comma separated list of node names
- Queue Action Invocation - Reassign action request based on initiator, entry ID, and action code; User, Action Item Id, and Action Code are required
 - User - Enter initiator's network Id
 - Action Item Id - Enter an entry's action item Id
 - Action Code - A, F, K, or C; A for Approve, F for FYI, K for acknowledge, and C for Complete

Document





Figure 10.3. Document

| | |
|---------------------------------|---|
| Do | |
| Document ID: 3063 | <input type="radio"/> Update <input checked="" type="radio"/> No Operation |
| * Document Version: | <input type="text" value="1"/> |
| * Initiator ID: | <input type="text" value="admin"/> |
| * Initial Route Node Instances: | <input type="text" value="3044"/> |
| * Route Status: | <input type="text" value="ENROUTE"/> |
| * Route Level: | <input type="text" value="2"/> |
| * Create Date: | <input type="text" value="02:44 PM 01/10/2012"/> |
| * Doc Status Modification Date: | <input type="text" value="02:46 PM 01/10/2012"/> |
| Approved Date: | <input type="text"/> |
| Finalized Date: | <input type="text"/> |
| Route Status Modification Date: | <input type="text"/> |
| Route Level Modification Date: | <input type="text"/> |
| Doc Type ID: | <input type="text" value="3057"/> |
| Doc Title: | <input type="text"/> |
| Application Doc ID: | <input type="text"/> |
| Doc Content: | <pre> <documentContent> <applicationContent> <data edlName="VacancyNotice"> <version current="false" date="Tue Jan <field name="oaa"> </pre> |

- Document Version - A legacy field indicating whether the document was upgraded from version 2.0 to 2.1
- Initiator ID - The workflow id of the initiator
- Initial Route Node Instances - The ID of the initial route node instance on the document
- Route Status - The current status of the document

- Route Level - A legacy field providing a numerical representation of where the document is in the route path
- Create Date - The initial date the document was created, doesn't not reflect wether the doucment was routed, saved, etc.
- Doc Status Modification Date - The date at which the document's status was last modified
- Approved Date - The date at which the document's state transitioned to APPROVED
- Finalized Date - The date at which the document's state transitioned to FINAL
- Route Status Modification Date - Legacy value, similar to Doc Status Modification Date
- Route Level Modification Date - Legacy value, no longer used
- Doc Type ID - The ID of the DocumentType definition for this document
- Doc Title - The title of the document
- Application Doc ID - A special id that can be set by client applications to associate the document to an ID in their system
- Override Indicator - Legacy value, no longer use
- Lock Code - Legacy value, no longer used
- Doc Content - The XML Content of the document







Action

| Action Request ID: 2369 | | <input type="radio"/> Update | <input type="radio"/> Delete | <input checked="" type="radio"/> No Operati |
|--------------------------------|---|------------------------------|------------------------------|---|
| * Document Version: | <input type="text" value="1"/> | | | |
| * Document ID: | <input type="text" value="3063"/> | | | |
| * Route Node Instance ID: | <input type="text" value="3045"/> | | | |
| * Action Requested: | <input type="text" value="APPROVE"/> | | | |
| * Create Date: | <input type="text" value="01/10/2012"/>  | | | |
| * Status: | <input type="text" value="DONE"/> | | | |
| * Priority: | <input type="text" value="1"/> | | | |
| * Route Level: | <input type="text" value="1"/> | | | |
| * Responsibility ID: | <input type="text" value="2145"/> | | | |
| Responsibility Description: | <input type="text" value="School/RC of BI-BUS for"/> | | | |
| Action Request Parent ID: | <input type="text"/> | | | |
| Recipient Type: | <input type="text" value="group"/> | | | |
| Person ID: | <input type="text"/>  | | | |
| Workgroup ID: | <input type="text" value="10008"/>  | | | |
| Role Name: | <input type="text"/>  | | | |
| Qualified Role Name: | <input type="text"/> | | | |
| Qualified Role Label: | <input type="text"/> | | | |
| Action Taken ID: | <input type="text" value="2337"/> | | | |
| Force Action: | <input type="text" value="No"/> | | | |
| Approve Policy: | <input type="text" value="F"/> | | | |
| Delegation Type: | <input type="text"/> | | | |
| Current Indicator: | <input type="text" value="Yes"/> | | | |
| Annotation: | <input type="text"/> | | | |
| Action Request ID: 2370 | | <input type="radio"/> Update | <input type="radio"/> Delete | <input checked="" type="radio"/> No Operati |
| * Document Version: | <input type="text" value="1"/> | | | |
| * Document ID: | <input type="text" value="3063"/> | | | |
| * Route Node Instance ID: | <input type="text" value="3046"/> | | | |
| * Action Requested: | <input type="text" value="APPROVE"/> | | | |

- Document Version - A legacy field indicating whether the request was upgraded from version 2.0 to 2.1
- Document ID - The ID of the associated document
- Route Node Instance ID - The ID of the node instance that this request is attached to
- Action Request - The type of action that is requested
- Create Date - The date the request was created
- Status - The current status of the request
- Priority - The activation priority of the request
- Route Level - A legacy field providing a numerical representation of where in the route path the request was generated
- Responsibility ID - The id of the responsibility associated with this request (relates to Rules and/or Route Modules)
- Responsibility Description - A description of the responsibility of this request
- Action Request Parent ID - ID of the parent action request if there is one
- Recipient Type - The type of recipient for this request (user, workgroup, or role)
- Person ID - If the recipient type is "user", the workflow id of the user recipient
- Workgroup ID - If the recipient type is "workgroup", the workgroup id of the workgroup recipient
- Role Name - If the recipient type is "role", the name of the role
- Qualified Role Name - If the recipient type is "role", the value of the qualified role
- Qualified Role Label - If the recipient type is "role", the label for the qualified role
- Action Taken ID - If this request has been satisfied, the id of the ActionTaken that satisfied the request
- Ignore Previous - The ignore previous indicator of the request
- Approve Policy - The approve policy of the request (only used by role requests)
- Delegation Type - If the request is a delegation, the type of delegation (primary or secondary)
- Current Indicator - Indicates if the request is "Current" or not
- Annotation - The value of the annotation on the request

Actions Taken

Figure 10.5. Actions Taken

| | | Action |
|------------------------------|---|--------|
| Action Taken ID: 2336 | <input type="radio"/> Update <input type="radio"/> Delete <input checked="" type="radio"/> No Operation | |
| * Document ID: | <input type="text" value="3063"/> | |
| * Document Version: | <input type="text" value="1"/> | |
| * Action Taken: | COMPLETED | |
| * Action Date: | <input type="text" value="01/10/2012"/>  | |
| * Action Taken Person ID: | <input type="text" value="admin"/>  | |
| Delegator Person ID: | <input type="text"/>  | |
| Delegator Workgroup ID: | <input type="text"/>  | |
| Current Indicator: | <input type="text" value="Yes"/> | |
| Annotation: | <input type="text"/> | |
| Action Taken ID: 2337 | <input type="radio"/> Update <input type="radio"/> Delete <input checked="" type="radio"/> No Operation | |
| * Document ID: | <input type="text" value="3063"/> | |
| * Document Version: | <input type="text" value="1"/> | |
| * Action Taken: | APPROVED | |
| * Action Date: | <input type="text" value="01/10/2012"/>  | |
| * Action Taken Person ID: | <input type="text" value="supervisor"/>  | |

- Document ID - The ID of the associated document
- Document Version - A legacy field indicating whether the Action Taken was upgraded from version 2.0 to 2.1
- Action Taken - the type of the action that was taken
- Action Date - the date at which the action was taken
- Action Taken Person ID - the workflow id of the user or delegate who took action
- Delegator Person ID - if this action was performed by a delegate, the workflow id of the person whose authority was delegated

- Delegator Workgroup ID - if this action was performed by a delegate, the workflow id of the Workgroup whose authority was delegated
- Current Indicator - Indicates if the Action Taken is "Current" or not, non-current actions have been revoked by an action such as ReturnToPreviousNode
- Annotation - The value of the annotation on the Action Taken

Action Items

Figure 10.6. Action Items

| | | Action |
|------------------------------|--|---|
| Action Item ID: 10227 | | <input type="radio"/> Update <input type="radio"/> Delete |
| * Document ID: | | <input type="text" value="3063"/> |
| * Doc Type Name: | | <input type="text" value="VacancyNotice"/> |
| * Doc Type Label: | | <input type="text" value="Vacancy Notice"/> |
| * Doc Handler URL: | | <input type="text" value="http://dev1.rice.kuali.or"/> |
| * Date Assigned: | | <input type="text" value="01/10/2012"/> |
| * Action Request ID: | | <input type="text" value="2370"/> |
| * Action Requested: | | <input type="button" value="APPROVE"/> |
| * Responsibility ID: | | <input type="text" value="2191"/> |
| * Person ID: | | <input type="text" value="director"/> |
| Workgroup ID: | | <input type="text" value="10005"/> |
| Role Name: | | <input type="text"/> |
| Delegator Person ID: | | <input type="text"/> |
| Delegator Workgroup ID: | | <input type="text"/> |
| Document Title: | | <input type="text"/> |

- Document ID - The ID of the associated document
- Doc Type Name - The name of the DocumentType for this item
- Doc Type Label - The label of the document type

- Doc Handler URL - The URL used to access the doc handler for this item
- Date Assigned - The creation date of the item
- Action Request ID - The id of the Action Request from which this item is derived
- Action Requested - The type of action requested by this item
- Responsibility ID - The responsibility id of the associated request
- Person ID - The workflow id of the person responsible for the item
- Workgroup ID - The workgroup id of the workgroup responsible for the item
- Role Name - If the item was derived from a role request, the name of the role
- Delegator Person ID - If the item was delegated, the workflow id of the delegating party
- Delegator Workgroup ID - If the item was delegated, the workgroup id of the delegating party
- Document Title - The title of the document

It is important to note that the ActionItem is a de-normalized representation of an Action Request on the document that is used to render the Action List in an efficient matter. Therefore, it contains some copies of data from both the document and the request itself.

Route Node Instances

Figure 10.7. Route Node Instances

| | | Route No |
|-------------------------------------|----------------------------------|---|
| Route Node Instance ID: 3045 | | <input type="radio"/> Update <input type="radio"/> Delete |
| Instance Name: | School | |
| Active Indicator: | <input type="text" value="No"/> | |
| Complete Indicator: | <input type="text" value="Yes"/> | |
| Initial Indicator: | <input type="text" value="No"/> | |
| Previous Route Node Instances: | 3044 | |
| Next Route Node Instances: | 3046 | |
| Route Node States: | None | |
| Route Node Instance ID: 3044 | | <input type="radio"/> Update <input type="radio"/> Delete |
| Instance Name: | Initiated | |
| Active Indicator: | <input type="text" value="No"/> | |

- Instance Name - The name of the node
- Active Indicator - indicates if the node is active
- Complete Indicator - indicates if the node's processing has completed
- Initial Indicator - indicates if the node has been processed by the engine yet
- Previous Route Node Instances - A comma-separated display of the IDs of the previous Route Node Instances of the node
- Next Route Node Instances - A comma-separated display of the IDs of the next Route Node Instances of the node
- Route Node States - A representation of the state attached to the node

The Route Node Instances are modeled as a Directed Acyclic Graph starting at the node instance pointed to by the Initial Route Node Instances field in the Document section. Therefore, if you delete a route node instance, it will follow all links through its set of Next Route Node Instances and delete those as well.

Branch States

Figure 10.8. Branch States

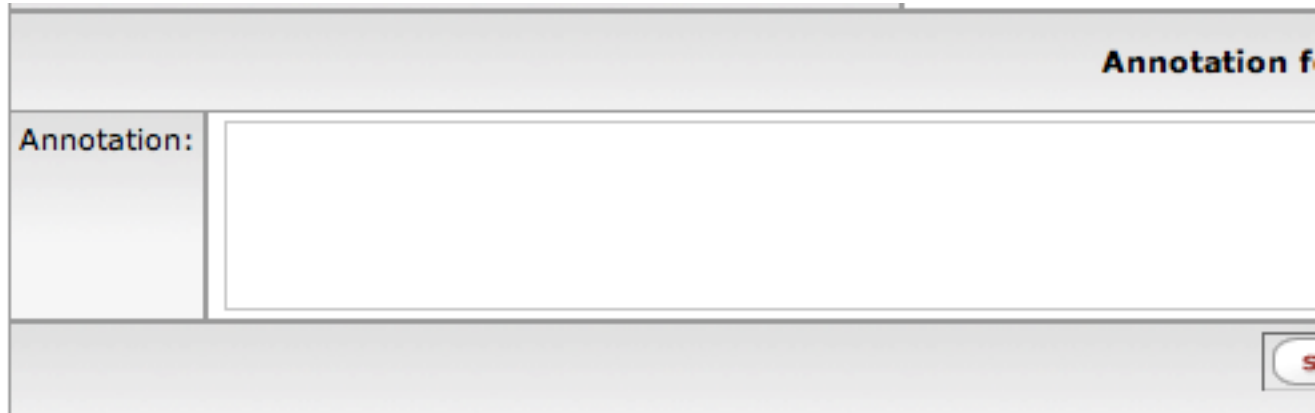
| | | |
|------------------------|--------------------------------------|---|
| | | Branch |
| Branch ID: 3043 | | <input type="radio"/> Update <input checked="" type="radio"/> No Open |
| Branch Name: | <input type="text" value="PRIMARY"/> | |
| Branch States: | None | |

- Branch Name - The name of the branch
- Branch State ID - The ID of that piece of branch state
- Branch State Key - The key of the branch state
- Branch State Value - The value of the branch state

All documents are required to have at least one branch that is named PRIMARY. Therefore, it is advisable to not rename the PRIMARY branch.

Annotation

Figure 10.9. Annotation

The image shows a screenshot of a software interface. At the top right, the text "Annotation f" is visible. Below this, there is a form with a label "Annotation:" on the left and a large, empty text input field on the right. At the bottom right of the form, there is a button with a red "S" icon, representing a "Save" function.

Here you can enter an annotation explaining the changes being made. This will be logged on the Route Log so that it can be preserved as part of the audit trail for the document.

Once the changes have been made on the document operation screen, hit the Save button and the changes will be executed on the server. Remember that in order for a change to take place the appropriate radio button must be selected on the data that requires modification.

Practical Uses of Document Operation

- Requeuing a document that was stuck
- Moving a document to FINAL
- Rolling a document back to a previous point in the route path

Examples of each are outlined below

Requeuing a document that was stuck

Moving a document to FINAL

Rolling a document back to a previous point in the route path

Chapter 11. Location

Location Overview

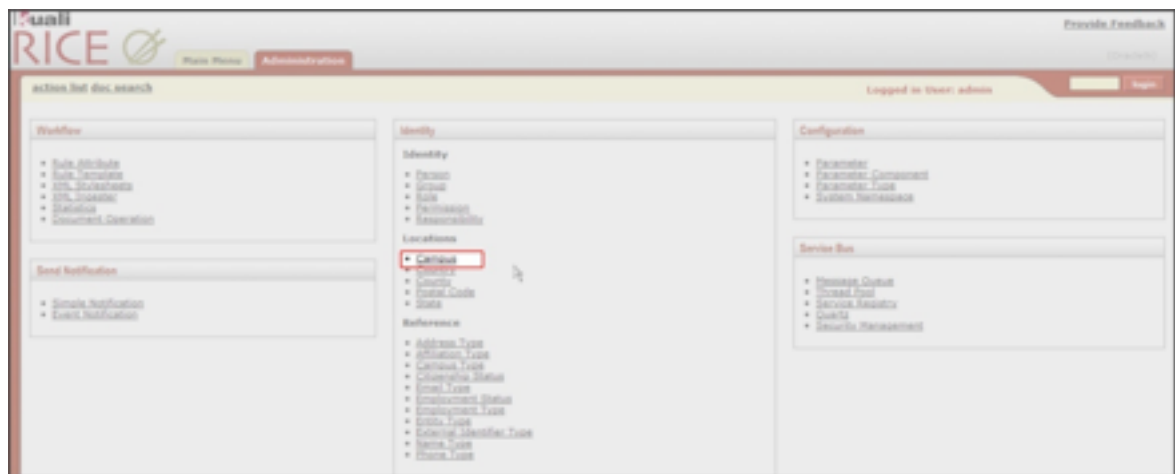
The Location module contains common "location" based options that can be used with Rice and other client applications. This currently includes screens and services for Campus, Country, State, County, and Postal Code.

Campus

Campus Lookup

The Campus Lookup screen provides detailed information about the campuses defined in your system.

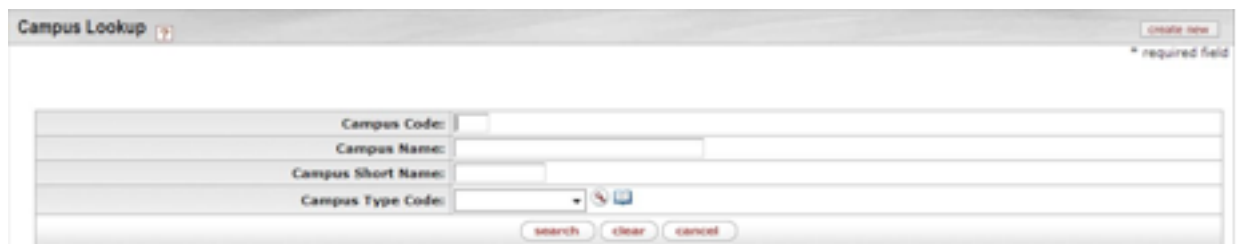
Figure 11.1. Identity Channel: Campus Link



To get to the Campus Lookup function, click **Campus** on the Rice **Administration Menu**.

This takes you to the **Campus Lookup** screen where you have the option to create a new Campus by clicking the **create new** button in the upper right corner under the Login box or to perform a search of campuses already defined in Rice. If you choose to create a new Campus, KIM displays the **CampusMaintenanceDocument**, which is discussed in the **Campus Maintenance** section below.

Figure 11.2. Campus Lookup



Another way to get to the **Campus Lookup** screen is to click the Field Lookup button next to the **Campus Code** field on any screen where it appears. If you do this, you see the Campus Lookup screen without the **create new** button.

Enter information in one or more of the fields on the Campus Lookup screen and then click the **search** button to see a list of Campuses that fit your search.

Note

Leave all of the search fields blank to get a list of all Campuses.

If you did NOT begin your Campus Lookup from the **Administration** menu, your list of search results looks like this:

Figure 11.3. Campus Lookup: Results Example

| <u>Return Value</u> | <u>Campus Code</u> | <u>Campus Name</u> | <u>Campus Short Name</u> | <u>Campus Type Code</u> |
|------------------------------|--------------------|--------------------|--------------------------|-------------------------|
| return value | BL | BLOOMINGTON | BLOOMINGTON | B |
| return value | BX | BLGTH OFF CAMPUS | BLGTH OFF CA | E |
| return value | CO | COLUMBUS | COLUMBUS | F |
| return value | EA | EAST-RICHMOND | EA-RICHMOND | B |
| return value | FW | FORT WAYNE | FORT WAYNE | B |
| return value | IN | INDIANAPOLIS | INDIANAPOLIS | B |

If you did begin your search from the **Administration** menu, you see **edit** and **copy** as links for each Campus instead of **return value**. These options are explained in the **Campus Maintenance** section below.

Click one of the underlined column titles on the search results list to sort the list by that column. Click it again to sort the list in the opposite order.

You can also export the search results list in CSV, spreadsheet, or XML format. Click the appropriate link at the bottom of the search results list to do this.

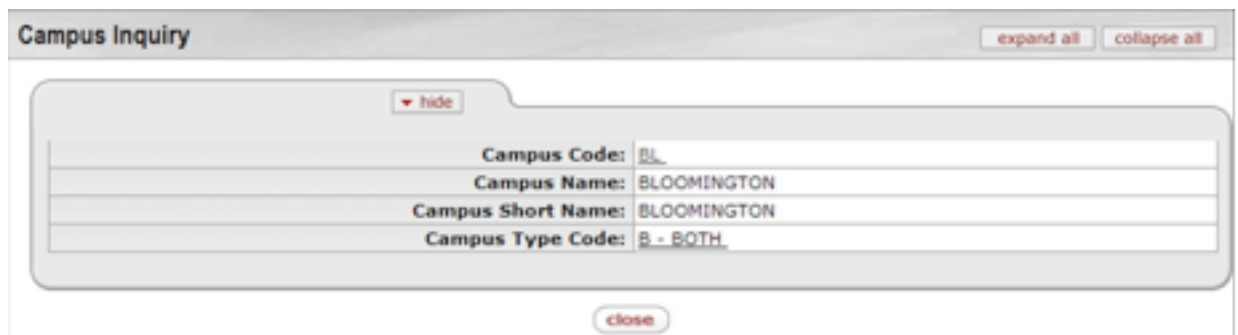
If you did NOT begin your Campus Lookup from the **Administration** menu, the **Return Value** column has a **return value** link for each campus. Click this return value link for the Campus Code for which you need information and Rice goes back to the screen you were on before and automatically enters the information about that campus on the screen for you.

You can find information about the fields on this screen under Edit Campus Tab in the Campus Maintenance section of this User Guide.

Campus Inquiry

Display this screen by clicking **Campus Code** in the search results list after you do a **Campus Lookup**:

Figure 11.4. Campus Inquiry



You can find information about the fields on this screen in the **Edit Campus Tab** section under **Campus Maintenance**, below.

Campus Maintenance

Use a **Campus** document for each of the different fiscal and physical operating entities of an institution. A campus may be identified as a fiscal entity, a physical entity, or both.

To create a new Campus

Click **Campus** on the Rice **Administration** menu for KIM to display the **Campus Lookup** screen, then click the **create new** button. KIM displays the **CampusMaintenanceDocument**.

Figure 11.5. Campue Maintenance Document

The screenshot shows a web-based form for creating a new campus. At the top right, there is a header with 'Doc. No: 2743', 'Status: INITIATED', 'Initiator: admin', and 'Created: 02:14 PM 07/01/2009'. Below this is a 'Document Overview' section with a 'Description' field and an 'Explanations' field. The main section is 'Edit Campus', which contains four required fields: 'Campus Code', 'Campus Name', 'Campus Short Name', and 'Campus Type Code'. At the bottom of the form, there are three sections: 'Notes and Attachments (0)', 'Add New Receipts', and 'Route Log', each with a 'show' button. The form concludes with a row of buttons: 'submit', 'save', 'Marked approve', 'close', and 'cancel'.

The fields on this document are described in the **Document Layout** section below. Enter information in all required fields and other fields appropriate to this campus, then click the **save** button to create your new Campus.

To edit Campus information

If you want to edit information about an existing Campus:

1. Begin at the Administration menu and do a Campus Lookup to find the campus you need to edit.
2. On the list of search results, click edit for that Campus.
3. KIM displays the CampusMaintenanceDocument for that campus:

Figure 11.6. Campus Maintenance Document: Expanded

The screenshot shows the 'CampusMaintenanceDocument' interface. At the top right, there are fields for 'Doc Nbr: 2994', 'Status: INITIATED', 'Initiator: admin', and 'Created: 04:45 PM 07/15/2009'. Below this is a 'Document Overview' section with fields for '* Description:', 'Org. Dec. #:', and 'Explanation:'. The 'Edit Campus' section is expanded, showing a comparison between 'Old' and 'New' data. The 'Old' data includes 'Campus Code: CO', 'Campus Name: COLUMBUS', 'Campus Short Name: COLUMBUS', and 'Campus Type Code: F - FISCAL'. The 'New' data includes 'Campus Code: CO', '* Campus Name: COLUMBUS', '* Campus Short Name: COLUMBUS', and '* Campus Type Code: F - FISCAL'. Below these are sections for 'Notes and Attachments (0)', 'Ad Hoc Recipients', and 'Route Log'. At the bottom, there are buttons for 'submit', 'save', 'Marked approve', 'close', and 'cancel'.

4. Change the information in the fields that you need to edit, then click the save button at the bottom of the document.

Document Layout

The Campus Maintenance Document has five sections:

- Document Overview
- Edit Campus
- Notes and Attachments
- Ad Hoc Recipients
- Route Log

You can find documentation on all of the tabs in this document except **Edit Campus** in the Common Features and Functions section of the **User Guide**.

Edit Campus Tab

Figure 11.7. Campus Maintenance Document: Edit Campus Tab

The screenshot shows the 'Edit Campus' tab. It features a comparison table between 'Old' and 'New' data. The 'Old' data includes 'Campus Code: CO', 'Campus Name: COLUMBUS', 'Campus Short Name: COLUMBUS', and 'Campus Type Code: F - FISCAL'. The 'New' data includes 'Campus Code: CO', '* Campus Name: COLUMBUS', '* Campus Short Name: COLUMBUS', and '* Campus Type Code: F - FISCAL'. The asterisks indicate required fields.

In edit mode, the **Edit Campus Tab** presents a display-only set of fields on the left and editable fields on the right in which you may enter changes. When this tab is displayed in the **Create New** process, only the editable fields are displayed.

Table 11.1. Campus Maintenance Document: Edit Campus Attributes

| Field Name | Description |
|-------------------|--|
| Campus Code | The unique identifying code assigned to a campus. |
| Campus Name | Required. The familiar name for a specific university campus. |
| Campus Short Name | Required. An abbreviated name for a specific campus; used in reports in which space is limited. |
| Campus Type Code | Required. Indicates the type of campus. Valid values are: <ul style="list-style-type: none"> • B - Both • F - Fiscal • P - Physical |

Campus Type Lookup

The **Campus Type** Lookup function displays detailed information about specific **Campus Type Codes**.

To display this screen:

1. Go to the Campus Lookup screen.
2. Select a valid Campus Type Code from the dropdown list.
3. Click the Field Lookup button next to the Campus Type Code field.

Figure 11.8. Campus Type Lookup

There are two ways to search for information using this screen:

- Enter information in one or more fields on the **Campus Type Lookup** screen and then click the **Search** button. This displays a list of campus types that match the information you entered, similar to the list in the screen print just below.
- Leave the fields on the Campus Type Lookup screen blank and click the Search button. This displays a list of all available campus types, similar to this:

Figure 11.9. Campue Type Lookup: Results Example

| Return Value | Campus Type Code | Campus Type Name | Active Indicator |
|------------------------------|------------------|------------------|------------------|
| return value | B | BOTH | Yes |
| return value | F | FISCAL | Yes |
| return value | P | PHYSICAL | Yes |

You can export the search results list in CSV, spreadsheet, or XML format. Click the appropriate link at the bottom of the search results list to do this.

The **Return Value** column has a **return value** link for each Campus Type Code. Click this return value link for the **Campus Type Code** you need, and Quali goes back to the screen you were on before and automatically enters the information about that Campus Type Code on the screen for you.

Field information on this screen can be found in the **Edit Campus Tab** section above.

Campus Type Inquiry

Use the **Campus Type Inquiry** screen to see the Campus Type Code and Name and the Active Indicator for a particular campus.

To display this screen:

1. Go to the Campus Lookup screen.
2. Select a valid Campus Type Code from the dropdown list.
3. Click the Inquiry button next to the Campus Type Code field.

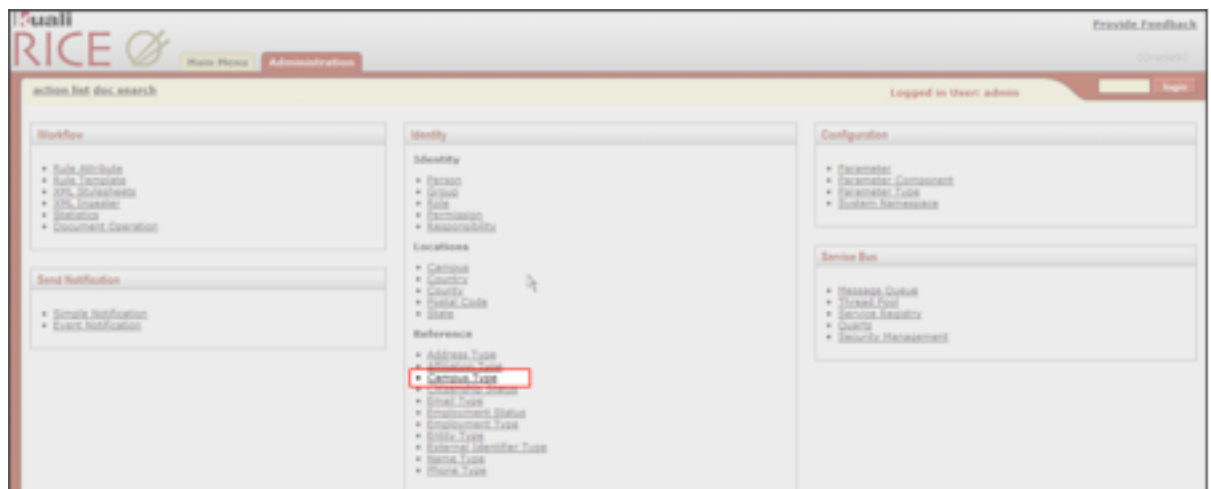
Figure 11.10. Campus Type Inquiry

| | |
|-------------------|------|
| Campus Type Code: | B |
| Campus Type Name: | BOTH |
| Active Indicator: | Yes |

Field information on this screen can be found in the **Edit Campus Tab** section above.

Campus Type Maintenance

Figure 11.11. Identity Channel: Campus Type Link



To create or perform maintenance on a Campus Type, click **Campus Type** on the **Administration Menu**.

Figure 11.12. Campus Type Lookup

Campus Type Lookup * required field

Campus Type Code:

Campus Type Name:

Active Indicator: Yes No Both

From the **Campus Type Lookup** screen, you may either create a new **Campus Type** or search for an existing one.

Note

If you don't have a **create new** button in the top right corner of the **Campus Type Lookup** screen, you did not open the screen from the **Administration** menu.

When you search for and find an existing Campus Type (see instructions above), you have the option to edit or copy it. After you click the **create new**, **edit**, or **copy** button, KIM displays the **Campus Type Maintenance** document. This document lets you add and update Campus Types.

Figure 11.13. Campus Maintenance Document

CampusMaintenanceDocument Doc. No: 1743 Status: INITIATED
Initiator: admin Created: 02:14 PM 01/07/2009

Document Overview * required field

Document Overview

* Description:

Grs. Desc. #:

Explanation:

Edit Campus * tabs

New

* Campus Code:

* Campus Name:

* Campus Short Name:

* Campus Type Code:

Notes and Attachments (0)

Ad Hoc Receipts

Route Log

Document Layout

The **CampusTypeMaintenance** document contains four standard tabs and the **Edit Campus Type** tab. The standard tabs are documented in the Common Features and Functions section of this User Guide. The Edit Campus Type tab is described below.

Figure 11.14. Campus Maintenance Document: Overview Tab

Edit Campus Type Tab

The Edit Campus Type tab displays the fields that are unique to each Campus Type.

Figure 11.15. Campus Maintenance Document: Edit Campus Type Tab

Table 11.2. Campus Maintenance Document: Edit Campus Type Attributes

| Field | Description |
|------------------|---|
| Campus Type Code | Required. The code used by Rice to identify this type of Campus. Valid values are: <ul style="list-style-type: none"> • B - Both • F - Fiscal • P - Physical |
| Campus Type Name | Required. The descriptive name for this Campus Type |
| Active Indicator | Required. Indicates whether the Campus Code is active or inactive. Remove the checkmark to deactivate a Campus Type. Removing the checkmark will also remove this Campus Type from the dropdown list on Campus documents. |

Postal Code

Postal Code Lookup

The Postal Code Lookup screen provides detailed information about the postal codes (zip codes) defined in KIM.

You access the **Postal Code Lookup** screen by clicking **Postal Code** in the **Identity** section of the **Administration** menu.

Figure 11.16. Identity Channel: Postal Code Link

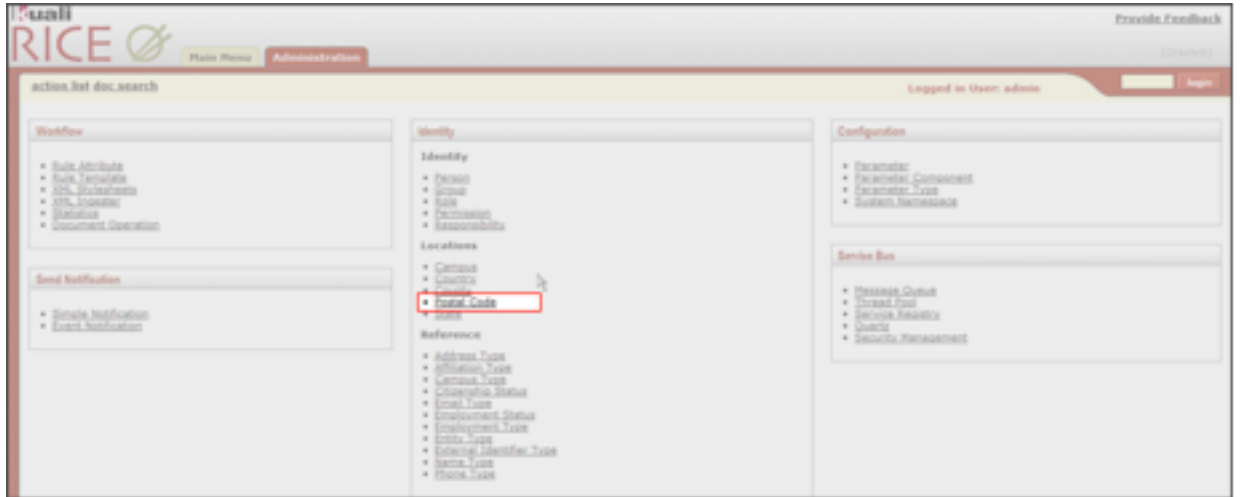


Figure 11.17. Postal Code Lookup

The screenshot shows the 'Postal Code Lookup' form. It contains several input fields: 'Country Code', 'Postal Code', 'State', 'City Name', and another 'Country Code'. There is also an 'Active Indicator' field with radio buttons for 'Yes', 'No', and 'Both'. A 'search' button is located at the bottom right of the form.

A search produces a list similar to the example below.

Figure 11.18. Postal Code Lookup: Results Example

| Actions | Country Code | Postal Code | State | City Name | Active Indicator |
|-----------|--------------|-------------|-------|-----------|------------------|
| edit copy | US | 45243 | OH | MADEIRA | Yes |

Note

You may also enter a city and state to search for the associated zip code(s).

Descriptions of the fields in this list can be found in the Postal Code Maintenance section below.

Clicking **edit** takes you to the **Postal Code Maintenance** screen.

Clicking **copy** also takes you to the **Postal Code Maintenance** screen and populates some of the fields in the **Edit Postal Codes** tab with information from the item you copied.

Clicking a **Country Code** from the list takes you to the **Country Code Inquiry** screen.

Clicking a **Postal Code** from the list takes you to the **Postal Code Inquiry** screen.

Clicking a **State Code** from the list takes you to the **State Code Inquiry** screen.

Postal Code Inquiry

Figure 11.19. Postal Code Inquiry

Information about these fields can be found in the Postal Code Maintenance section below.

Postal Code Maintenance

The **Postal Code Maintenance Document** defines the postal (zip) code by city and state. When you click the Postal Code option on the Administration screen, KIM displays the **Postal Code Lookup** screen. At this point you can perform a lookup and select a postal code to edit or click the **create new** button, KIM displays the **PostalCodeMaintenanceDocument** screen.

Figure 11.20. Postal Code Manintenance Document

To create a new Postal Code, enter information in the required fields in the **Document Overview** and **Edit Postal Codes** tabs, then click the **submit** button at the bottom of the screen.

Document Layout

All of the tabs on this screen are standard, except the **Edit Postal Codes** tab. You can find information on standard tabs in the Common Features and Functions section of this **User Guide**.

Edit Postal Codes tab

In edit mode, the **Edit Postal Codes** tab presents a display-only set of fields on the left and editable fields on the right, in which you may enter changes. The tab looks like this:

Figure 11.21. Postal Code Maintenance Document: Edit Postal Codes Tab

In create mode, the **Edit Postal Codes** tab presents only a set of fields in which you may enter information.

Figure 11.22. Postal Code Maintenance Document: Create Postal Codes

Table 11.3. Postal Code Maintenance Document: Create Postal Codes Attributes

| Field | Description |
|------------------|---|
| Country Code | The country code for the country associated with the zip (postal) code |
| Postal Code | A Postal Service zip code |
| State | Required. The state associated with the postal (zip) code |
| County Code | The KIM identifying code for the county associated with the postal (zip) code |
| City Name | Required. The name of the city associated with the postal (zip) code |
| Active Indicator | Indicates whether the postal code is active or inactive. Remove the checkmark to deactivate this postal code. |

County

County Lookup

The **County Lookup** screen provides information on counties defined in KIM.

Access the **County Code Lookup** screen by clicking **County** in the **Identity** section of the **Administration Menu**.

Figure 11.23. Identity Channel: County Lin

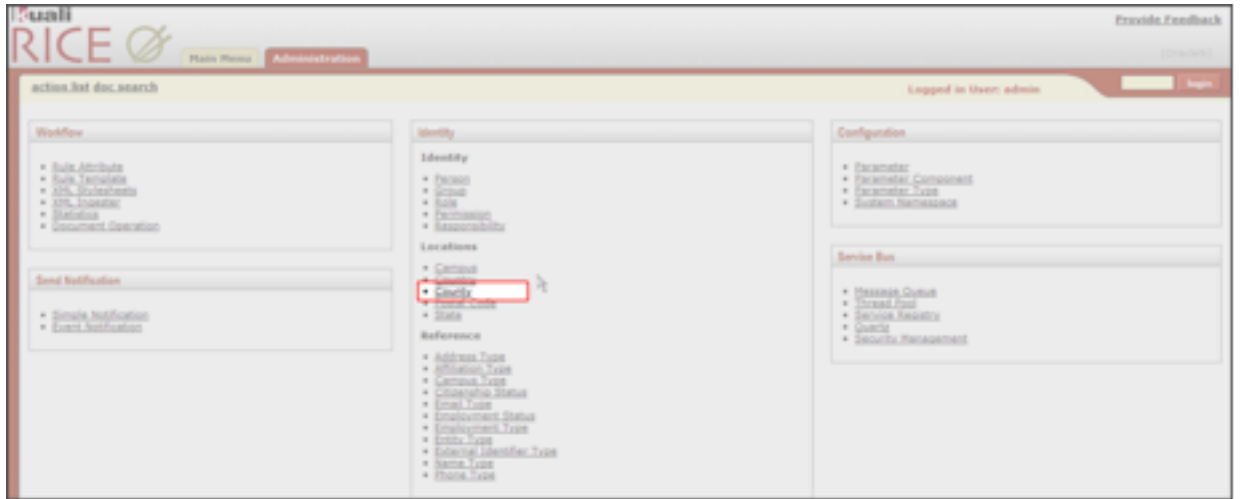
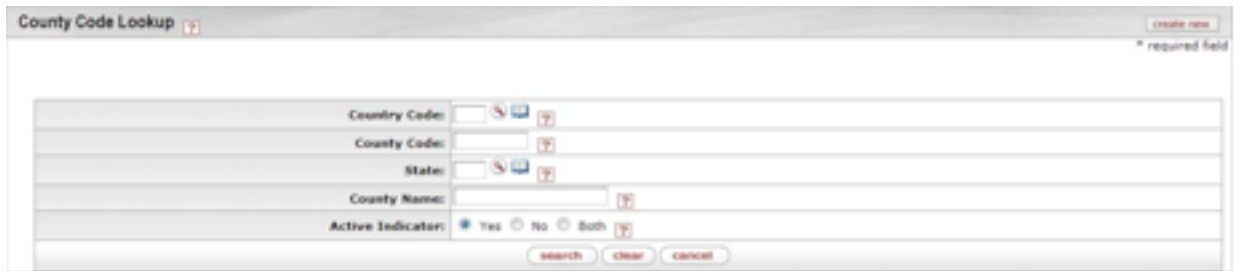


Figure 11.24. County Code Lookup



Click the **create new** button to go to the **County Maintenance** function, where you can create a new County in Rice.

Conducting a search from this screen returns a list similar to this:

Figure 11.25. County Code Lookup: Results Example

| Action | Country Code | County Code | State | County Name | Active Indicator |
|-----------|--------------|-------------|-------|-------------|------------------|
| edit copy | 100 | 101 | GA | county 101a | yes |

Clicking the **edit** or **copy** link in the Action column allows you to edit the county information you selected or to create a new county in Rice, beginning with a copy of the information from this county. This link takes you to the **County Maintenance Document**.

Clicking a **Country Code** takes you to the **Country Inquiry** function for that code.

Clicking a **County Code** takes you to the **County Inquiry** function for that code.

Clicking a **State** code takes you to the **State Inquiry** function for that code.

County Inquiry

This screen provides a snapshot of the important information about a **County**:

Figure 11.26. County Inquiry

The screenshot shows a web form titled "County Inquiry". It contains several input fields: "Country Code" (with a dropdown arrow), "County Code" (with a dropdown arrow), "State" (with a dropdown arrow), "County Name" (with a dropdown arrow), and "Active Indicator" (with a dropdown arrow). The "Active Indicator" field is currently set to "Yes". There is a "Check" button at the bottom of the form. In the top right corner, there are "expand all" and "collapse all" buttons.

You can find information about the fields on this screen in the **County Maintenance** section below.

County Maintenance

Use the **County Maintenance** document to assign specific identifying codes to county names. When you click the County option on the Administration screen, KIM displays the **County Lookup** screen. After you select to edit a county or click the **create new** button, KIM displays the **CountyMaintenanceDocument** screen:

Figure 11.27. County Maintenance Document

The screenshot shows the "CountyMaintenanceDocument" screen. At the top right, there is a header with "Doc. No.: 2710", "Status: INITIATED", "Initiator: admin", and "Created: 02-04 PM 01/05/20". Below this is a "Document Overview" tab with fields for "DESCRIPTION" and "Org. Div. #". To the right of these fields is an "Explanation" field. Below the "Document Overview" is an "Edit Counties" tab with a "New" section containing fields for "Country Code", "County Code", "State", "County Name", and "Active Indicator". At the bottom of the screen, there are three tabs: "Notes and Attachments (0)", "Ad Hoc Receipts", and "Route Log".

To create a new County, enter valid information in the required fields in the **Document Overview** and **Edit Counties** tabs, then click the **Submit** button at the bottom of the screen.

Document Layout

All of the tabs on this screen are standard, except the **Edit Counties** tab. Information on standard tabs can be found in the Common Features and Functions section of this User Guide.

Edit Counties Tab

In edit mode, the **Edit Counties** tab presents a display-only set of fields on the left and editable fields on the right, in which you may enter changes. The tab looks like this:

Figure 11.28. County Maintenance Document: Edit Counties Tab

In create mode, the **Edit Counties** tab presents only a set of fields in which you may enter information.

Figure 11.29. County Maintenance Document: Create County

Table 11.4. County Maintenance Document: Create County Attributes

| Field | Description |
|------------------|---|
| Country Code | Required. The country code for the country in which this county is located. |
| County Code | Required. A unique identifying code assigned to this county. |
| State | Required. The state abbreviation assigned by the U.S. Postal Service to the state in which this county is located. |
| County Name | Required. The actual name of this county. |
| Active Indicator | Optional. Indicates whether the county code is <i>active</i> or <i>inactive</i> . Remove the check mark to deactivate this county code. |

State

State Code Lookup

The State Code Lookup function provides a convenient way to find key information pertaining to states that are defined in Rice.

You can display the **State Code Lookup** screen by clicking **State** on the Rice **Administration** menu.

Figure 11.30. Identity Channel: State Link

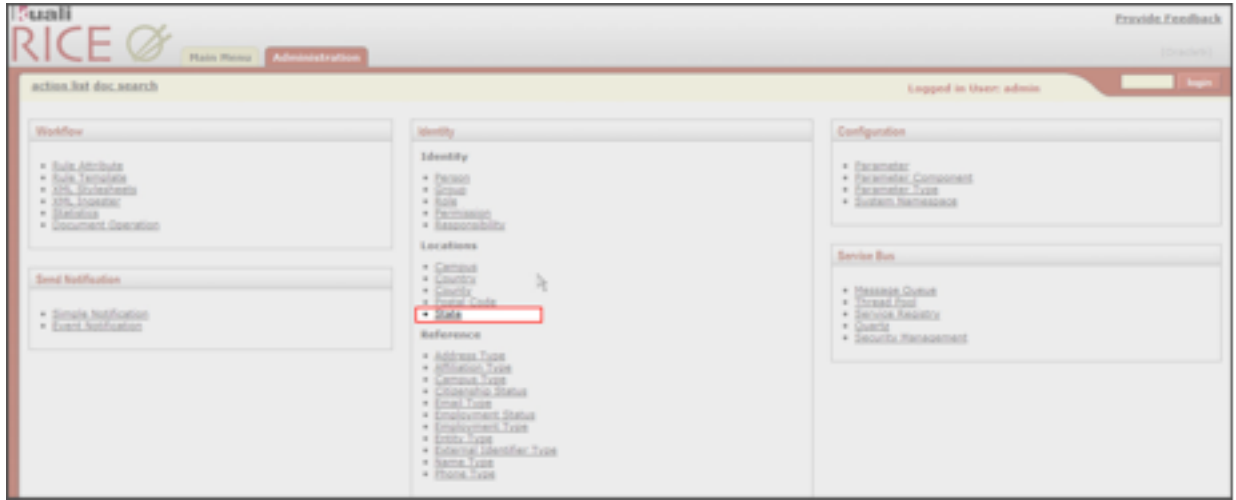


Figure 11.31. State Lookup



Enter information in one or more of the fields on the **State Code Lookup** screen and then click the **search** button to see a list of States that fit your criteria.

Note

Leave all of the search fields blank to display a list of all States.

Figure 11.32. State Lookup: Results Example

| Actions | Country Code | State Abbreviation | State Name | Active Indicator |
|---|--------------|--------------------|--|------------------|
| edit copy | <u>US</u> | -- | OUT OF COUNTRY | Yes |
| edit copy | <u>US</u> | AA | ARMED FORCES AMERICAS (EXCEPT CANADA) | Yes |
| edit copy | <u>US</u> | AE | ARMED FORCES EURO/CANADA/AFRICA/MSD EAST | Yes |
| edit copy | <u>US</u> | AK | ALASKA | Yes |
| edit copy | <u>US</u> | AL | ALABAMA | Yes |
| edit copy | <u>US</u> | AP | ARMED FORCES PACIFIC | Yes |
| edit copy | <u>US</u> | AR | ARKANSAS | Yes |

Click **edit** or **copy** in the **Actions** column to go to the **StateMaintenanceDocument** screen, where you can edit the existing **State** or use a copy of it as a base for establishing a new **State** in KIM.

Click an underlined **Country Code** to go to the **Country Inquiry** screen for that **Country Code**.

You can also export the search results list in CSV, spreadsheet, or XML format. Click the appropriate link at the bottom of the search results list to do this.

State Inquiry

The **State Inquiry** screen provides a snapshot of the most important information defined in Rice for a **State**.

Figure 11.33. State Inquiry

State Maintenance

The State document defines the U.S. Postal Service codes used to identify states. When you choose the State option on the Administration menu, KIM displays the **State Code Lookup** screen. After you click **edit** for a State or click the **create new** button, KIM displays the **StateMaintenanceDocument**.

Figure 11.34. State Maintenance Document

Document Layout

This screen contains five tabs:

- Document Overview
- Edit States
- Notes and Attachments
- Ad Hoc Recipients
- Route Log

The **Edit States** tab is unique to this screen and is described below. The other four tabs are described in the Common Features and Functions section of this User Guide.

Edit States Tab

The Edit States tab can be displayed with two different sets of fields. Which you see depends on whether you are creating a new State or editing an existing State.

Figure 11.35. State Maintenance Document: Add State

The version of the tab above is displayed when you are creating a new State.

Figure 11.36. State Maintenance Document: Edit States Tab

Pictured above in **edit** mode, the **Edit States** tab presents a display-only set of fields on the left and editable fields on the right, in which you may enter changes.

Table 11.5. States Maintenance Document: Edit States Attributes

| Field | Description |
|--------------------|---|
| Country Code | A unique identifying code assigned to a country Required for a new State Display-only when editing a State |
| State Abbreviation | The state abbreviation assigned by the U.S. Postal Service Required for a new State Display-only when editing a State |
| State Name | The full name of the state Required for a new State When editing, this field can be modified but not deleted. |
| Active Indicator | Optional. Indicates whether the State code is active or inactive. Remove the checkmark to deactivate the State code. |

Country

Country Lookup

This screen provides information about **Country Codes** and is accessed directly off the Administration menu.

Figure 11.37. Country Lookup

Performing a search on a **Country Code** produces results similar to this:

Figure 11.38. Country Lookup: Results Example

| Actions | Country Code | Country Code | State | Country Name | Active Indicator |
|---|--------------------|--------------------|-------|---------------|------------------|
| edit copy | US | US | US | United States | Yes |

Selecting **edit** or **copy** in the **Actions** column takes you to the **Country Maintenance Document** screen, where you can edit the existing **Country Code** or use a copy of it as a base for establishing a new **Country Code**.

Selecting an underlined **Country Code** takes you to the **Country Inquiry** screen for that **Country Code**.

You can also export the search results list in CSV, spreadsheet, or XML format. Click the appropriate link at the bottom of the search results list to do this.

Country Inquiry

The **Country Inquiry** screen displays details about a Country in Rice. You get to this screen by clicking a **Country Code** in the list of search results after you do a **Country Lookup**.

Figure 11.39. Country Inquiry

The screenshot shows a 'Country Inquiry' window with the following details:

- Country Code: US
- Country Name: UNITED STATES
- Country Restricted Indicator: No
- Active Indicator: Yes

Buttons for 'hide', 'expand all', 'collapse all', and 'close' are visible.

Country Maintenance

The Country document is used to assign specific identifying codes to country names.

When you choose the Country option on the Administration menu, KIM displays the **Country Lookup** screen. After you select a country or click the **create new** button, KIM presents the **CountryMaintenanceDocument** screen.

Figure 11.40. Country Maintenance Document

The screenshot shows the 'CountryMaintenanceDocument' screen with the following details:

- Doc. No.: 2740
- Status: INITIATED
- Initiator: admin
- Created: 01:57 PM 03/01/2009

The main form area includes sections for:

- Document Overview:** Description, Dist. Doc. #, and Explanations.
- Edit Country:** Fields for Country Code, Country Name, Country Restricted Indicator, and Active Indicator.
- Notes and Attachments (0)**
- Ad Hoc Recipients**
- Route Log**

Buttons for 'submit', 'save', 'cancel approval', 'close', and 'cancel' are at the bottom.

Document Layout

All of the tabs in the **Country Maintenance Document** are standard with the exception of the **Edit Country** tab. Documentation on standard tabs can be found in the Common Features and Functions section of the User Guide.

Edit Country tab

Figure 11.41. Country Maintenance Document: Edit Country Tab

The screenshot shows the 'Edit Country' tab interface. It features a 'New' button and four input fields: 'Country Code', 'Country Name', 'Country Restricted Indicator', and 'Active Indicator'. Each field has a red question mark icon to its right. The 'Country Restricted Indicator' and 'Active Indicator' fields include checkboxes.

Table 11.6. Country Maintenance Document: Edit Country Attributes

| Field | Description |
|------------------------------|--|
| Country Code | A unique identifying code assigned to a country. |
| Country Name | Required. The actual name of a specific country. |
| Country Restricted Indicator | Required. Check the box to indicate if the country is restricted from use in procurement. Clear the check box if it is not restricted. |
| Active Indicator | Indicates whether this country code is active or inactive in Rice. Remove the checkmark to deactivate this country code. |

Glossary

A

| | |
|--------------------------|---|
| Action List | A list of the user's notification and workflow items. Also called the user's Notification List. Clicking an item in the Action List displays details about that notification, if the item is a notification, or displays that document, if it is a workflow item. The user will usually load the document from their Action List in order to take the requested action against it, such as approving or acknowledging the document. |
| Action List Type | This tells you if the Action List item is a notification or a more specific workflow request item. When the Action List item is a notification, the Action List Type is "Notification." |
| Action Request | A request to a user or Workgroup to take action on a document. It designates the type of action that is requested, which includes: <ul style="list-style-type: none">• Approve: requests an approve or disapprove action.• Complete: requests a completion of the contents of a document. This action request is displayed in the Action List after the user saves an incomplete document.• Acknowledge: requests an acknowledgment by the user that the document has been opened - the doc will not leave the Action List until acknowledgment has occurred; however, the document routing will not be held up and the document will be permitted to transaction into the processed state if necessary.• FYI: a notification to the user regarding the document. Documents requesting FYI can be cleared directly from the Action List. Even if a document has FYI requests remaining, it will still be permitted to transition into the FINAL state. |
| Action Request Hierarchy | Action requests are hierarchical in nature and can have one parent and multiple children. |
| Action Requested | The action one needs to take on a document; also the type of action that is requested by an Action Request. Actions that may be requested of a user are: <ul style="list-style-type: none">• Acknowledge: requests that the users states he or she has reviewed the document.• Approve: requests that the user either Approve or Disapprove a document.• Complete: requests the user to enter additional information in a document so that the content of the document is complete.• FYI: intended to simply makes a user aware of the document. |
| Action Taken | An action taken on a document by a Reviewer in response to an Action Request. The Action Taken may be: <ul style="list-style-type: none">• Acknowledged: Reviewer has viewed and acknowledged document.• Approved: Reviewer has approved the action requested on document. |

- Blanket Approved: Reviewer has requested a blanket approval up to a specified point in the route path on the document.
- Canceled: Reviewer has canceled the document. The document will not be routed to any more reviewers.
- Cleared FYI: Reviewer has viewed the document and cleared all of his or her pending FYI(s) on this document.
- Completed: Reviewer has completed and supplied all data requested on document.
- Created Document: User has created a document
- Disapproved: Reviewer has disapproved the document. The document will not be routed to any subsequent reviewers for approval. Acknowledge Requests are sent to previous approvers to inform them of the disapproval.
- Logged Document: Reviewer has added a message to the Route Log of the document.
- Moved Document: Reviewer has moved the document either backward or forward in its routing path.
- Returned to Previous Node: Reviewer has returned the document to a previous routing node. When a Reviewer does this, all the actions taken between the current node and the return node are removed and all the pending requests on the document are deactivated.
- Routed Document: Reviewer has submitted the document to the workflow engine for routing.
- Saved: Reviewer has saved the document for later completion and routing.
- Superuser Approved Document: [Superuser](#) has approved the entire document, any remaining routing is cancelled.
- Superuser Approved Node: Superuser has approved the document through all nodes up to (but not including) a specific node. When the document gets to that node, the normal Action Requests will be created.
- Superuser Approved Request: Superuser has approved a single pending Approve or Complete Action Request. The document then goes to the next routing node.
- Superuser Cancelled: Superuser has canceled the document. A Superuser can cancel a document without a pending Action Request to him/her on the document.
- Superuser Disapproved: Superuser has disapproved the document. A Superuser can disapprove a document without a pending Action Request to him/her on the document.

| | |
|---------------------|--|
| | <ul style="list-style-type: none">• Superuser Returned to Previous Node: Superuser has returned the document to a previous routing node. A Superuser can do this without a pending Action Request to him/her on the document. |
| Activated | The state of an action request when it has been sent to a user's Action List. |
| Activation | The process by which requests appear in a user's Action List |
| Activation Type | Defines how a route node handles activation of Action Requests. There are two standard activation types: <ul style="list-style-type: none">• Sequential: Action Requests are activated one at a time based on routing priority. The next Action Request isn't activated until the previous request is satisfied.• Parallel: All Action Requests at the route node are activated immediately, regardless of priority |
| Active Indicator | An indicator specifying whether an object in the system is active or not. Used as an alternative to complete removal of an object. |
| Ad Hoc Routing | A type of routing used to route a document to users or groups that are not in the Routing path for that Document Type. When the Ad Hoc Routing is complete, the routing returns to its normal path. |
| Annotation | Optional comments added by a Reviewer when taking action. Intended to explain or clarify the action taken or to advise subsequent Reviewers. |
| Approve | A type of workflow action button. Signifies that the document represents a valid business transaction in accordance with institutional needs and policies in the user's judgment. A single document may require approval from several users, at multiple route levels, before it moves to final status. |
| Approver | The user who approves the document. As a document moves through Workflow, it moves one route level at a time. An Approver operates at a particular route level of the document. |
| Attachment | The pathname of a related file to attach to a Note. Use the "Browse..." button to open the file dialog, select the file and automatically fill in the pathname. |
| Attribute Type | Used to strongly type or categorize the values that can be stored for the various attributes in the system (e.g., the value of the arbitrary key/value pairs that can be defined and associated with a given parent object in the system). |
| Authentication | The act of logging into the system. The Out of the box (OOTB) authentication implementation in Rice does not require a password as it is intended for testing purposes only. This is something that must be enabled as part of an implementation. Various authentication solutions exist, such as CAS or Shibboleth, that an implementer may want to use depending on their needs. |
| Authorization | Authorization is the permissions that an authenticated user has for performing actions in the system. |
| Author Universal ID | A free-form text field for the full name of the Author of the Note, expressed as "Lastname, Firstname Initial" |

B

| | |
|---------------------------|--|
| Base Rule Attribute | <p>The standard fields that are defined and collected for every Routing Rule. These include:</p> <ul style="list-style-type: none"> • Active: A true/false flag to indicate if the Routing Rule is active. If false, then the rule will not be evaluated during routing. • Document Type: The Document Type to which the Routing Rule applies. • From Date: The inclusive start date from which the Routing Rule will be considered for a match. • Force Action: a true/false flag to indicate if the review should be forced to take action again for the requests generated by this rule, even if they had taken action on the document previously. • Name: the name of the rule, this serves as a unique identifier for the rule. If one is not specified when the rule is created, then it will be generated. • Rule Template: The Rule Template used to create the Routing Rule. • To Date: The inclusive end date to which the Routing Rule will be considered for a match. |
| Blanket Approval | <p>Authority that is given to designated Reviewers who can approve a document to a chosen route point. A Blanket Approval bypasses approvals that would otherwise be required in the Routing. For an authorized Reviewer, the Doc Handler typically displays the Blanket Approval button along with the other options. When a Blanket Approval is used, the Reviewers who are skipped are sent Acknowledge requests to notify them that they were bypassed.</p> |
| Blanket Approve Workgroup | <p>A workgroup that has the authority to Blanket Approve a document.</p> |
| Branch | <p>A path containing one or more Route Nodes that a document traverses during routing. When a document enters a Split Node multiple branches can be created. A Join Node joins multiple branches together.</p> |
| Business Rule | <ol style="list-style-type: none"> 1. Describes the operations, definitions and constraints that apply to an organization in achieving its goals. 2. A restriction to a function for a business reason (such as making a specific object code unavailable for a particular type of disbursement). Customizable business rules are controlled by Parameters. |

C

| | |
|-------------|---|
| Campus | <p>Identifies the different fiscal and physical operating entities of an institution.</p> |
| Campus Type | <p>Designates a campus as physical only, fiscal only or both.</p> |
| Cancel | <p>A workflow action available to document initiators on documents that have not yet been routed for approval. Denotes that the document is void and should be disregarded. Canceled documents cannot be modified in any way and do not route for approval.</p> |

| | |
|--------------------------------------|---|
| Canceled | A routing status. The document is denoted as void and should be disregarded. |
| CAS - Central Authentication Service | http://www.jasig.org/cas - An open source authentication framework. Kuali Rice provides support for integrating with CAS as an authentication provider (among other authentication solutions) and also provides an implementation of a CAS server that integrates with Kuali Identity Management. |
| Client | A Java Application Program Interface (API) for interfacing with the Kuali Enterprise Workflow Engine. |
| Client/Server | The use of one computer to request the services of another computer over a network. The workstation in an organization will be used to initiate a business transaction (e.g., a budget transfer). This workstation needs to gather information from a remote database to process the transaction, and will eventually be used to post new or changed information back onto that remote database. The workstation is thus a Client and the remote computer that houses the database is the Server. |
| Close | A workflow action available on documents in most statuses. Signifies that the user wishes to exit the document. No changes to Action Requests, Route Logs or document status occur as a result of a Close action. If you initiate a document and close it without saving, it is the same as canceling that document. |
| Comma-separated value | A file format using commas as delimiters utilized in import and export functionality. |
| Complete | A pending action request to a user to submit a saved document. |
| Completed | The action taken by a user or group in response to a request in order to finish populating a document with information, as evidenced in the Document Route Log. |
| Country Restricted Indicator | Field used to indicate if a country is restricted from use in procurement. If there is no value then there is no restriction. |
| Creation Date | The date on which a document is created. |
| CSV | See comma-separated value |
| D | |
| Date Approved | The date on which a document was most recently approved. |
| Date Finalized | The date on which a document enters the FINAL state. At this point, all approvals and acknowledgments are complete for the document. |
| Deactivation | The process by which requests are removed from a user's Action List |
| Delegate | A user who has been registered to act on behalf of another user. The Delegate acts with the full authority of the Delegator. Delegation may be either Primary Delegation or Secondary Delegation . |
| Delegate Action List | A separate Action List for Delegate actions. When a Delegate selects a Delegator for whom to act, an Action List of all documents sent to the Delegator is displayed. |

For both [Primary](#) and [Secondary Delegation](#) the Delegate may act on any of the entries with the full authority of the Delegator.

Disapprove A workflow action that allows a user to indicate that a document does not represent a valid business transaction in that user's judgment. The initiator and previous approvers will receive Acknowledgment requests indicating the document was disapproved.

Disapproved A status that indicates the document has been disapproved by an approver as a valid transaction and it will not generate the originally intended transaction.

Doc Handler The Doc Handler is a web interface that a Client uses for the appropriate display of a document. When a user opens a document from the Action List or Document Search, the Doc Handler manages access permissions, content format, and user options according to the requirements of the Client.

Doc Handler URL The URL for the [Doc Handler](#).

Doc Nbr See [Document Number](#).

Document Also see [E-Doc](#).

An electronic document containing information for a business transaction that is routed for Actions in KEW. It includes information such as Document ID, Type, Title, Route Status, Initiator, Date Created, etc. In KEW, a document typically has XML content attached to it that is used to make routing decisions.

Document Id See [Document Number](#).

Document Number A unique, sequential, system-assigned number for a document

Document Operation A workflow screen that provides an interface for authorized users to manipulate the XML and other data that defines a document in workflow. It allows you to access and open a document by Document ID for the purpose of performing operations on the document.

Document Search A web interface in which users can search for documents. Users may search by a combination of document properties such as Document Type or Document ID, or by more specialized properties using the Detailed Search. Search results are displayed in a list similar to an Action List.

Document Status See also [Route Status](#).

Document Title The title given to the document when it was created. Depending on the Document Type, this title may have been assigned by the Initiator or built automatically based on the contents of the document. The Document Title is displayed in both the Action List and Document Search.

Document Type The Document Type defines the routing definition and other properties for a set of documents. Each document is an instance of a Document Type and conducts the same type of business transaction as other instances of that Document Type.

Document Types have the following characteristics:

- They are specifications for a document that can be created in KEW

- They contain identifying information as well as policies and other attributes
- They defines the Route Path executed for a document of that type (Process Definition)
- They are hierarchical in nature may be part of a hierarchy of Document Types, each of which inherits certain properties of its [Parent Document Type](#).
- They are typically defined in XML, but certain properties can be maintained from a graphical interface

| | |
|-------------------------|---|
| Document Type Hierarchy | A hierarchy of Document Type definitions. Document Types inherit certain attributes from their parent Document Types. This hierarchy is also leveraged by various pieces of the system, including the Rules engine when evaluating rule sets and KIM when evaluating certain Document Type-based permissions. |
| Document Type Label | The human-readable label assigned to a Document Type. |
| Document Type Name | The assigned name of the document type. It must be unique. |
| Document Type Policy | These advise various checks and authorizations for instances of a Document Type during the routing process. |
| Drilldown | A link that allows a user to access more detailed information about the current data. These links typically take the user through a series of inquiries on different business objects. |
| Dynamic Node | An advanced type of Route Node that can be used to generate complex routing paths on the fly. Typically used whenever the route path of a document cannot be statically defined and must be completely derived from document data. |

E

| | |
|-----------------|--|
| ECL | <ol style="list-style-type: none">1. An acronym for Educational Community License.2. All Quali software and material is available under the Educational Community License and may be adopted by colleges and universities without licensing fees. The open licensing approach also provides opportunities for support and implementation assistance from commercial affiliates. |
| E-Doc | An abbreviation for electronic documents, also a shorthand reference to documents created with eDocLite. |
| eDocLite | A framework for quickly building workflow-enabled documents. Allows you to define document screens in XML and render them using XSL style sheets. |
| Embedded Client | A type of client that runs an embedded workflow engine. |
| Employee Status | Found on the Person Document; defines the employee's current employment classification (for example, "A" for Active). |
| Employee Type | Found on the Person Document; defines the employee's position classification (for example, "P" for Professional). |

| | |
|--------------------------|---|
| Entity | An Entity record houses identity information for a given Person, Process, System, etc. Each Entity is categorized by its association with an Entity Type. |
| Entity Attribute | Entities have directory-like information called Entity Attributes that are associated with them Entity Attributes make up the identity information for an Entity record. |
| Entity Type | Provides categorization to Entities. For example, a “System” could be considered an Entity Type because something like a batch process may need to interface with the application. |
| Exception | A workflow routing status indicating that the document routed to an exception queue because workflow has encountered a system error when trying to process the document. |
| Exception Messaging | The set of services and configuration options that are responsible for handling messages when they cannot be successfully delivered. Exception Messaging is set up when you configure KSB using the properties outlined in KSB Module Configuration. |
| Exception Routing | A type of routing used to handle error conditions that occur during the routing of a document. A document goes into Exception Routing when the workflow engine encounters an error or a situation where it cannot proceed, such as a violation of a Document Type Policy or an error contacting external services. When this occurs, the document is routed to the parties responsible for handling these exception cases. This can be a group configured on the document or a responsibility configured in KIM. Once one of these responsible parties has reviewed the situation and approved the document, it will be resubmitted to the workflow engine to attempt the processing again. |
| Extended Attributes | Custom, table-driven business object attributes that can be established by implementing institutions. |
| Extension Rule Attribute | One of the rule attributes added in the definition of a rule template that extends beyond the base rule attributes to differentiate the routing rule. A Required Extension Attribute has its "Required" field set to True in the rule template. Otherwise, it is an Optional Extension Attribute. Extension attributes are typically used to add additional fields that can be collected on a rule. They also define the logic for how those fields will be processed during rule evaluation. |

F

| | |
|--------------------------------|---|
| Field Lookup | The round magnifying glass icon found next to fields throughout the GUI that allow the user to look up reference table information and display (and select from) a list of valid values for that field. |
| Final | A workflow routing status indicating that the document has been routed and has no pending approval or acknowledgement requests. |
| Flexible Route Management | A standard KEW routing scheme based on rules rather than dedicated table-based routing. |
| FlexRM (Flexible Route Module) | The Route Module that performs the Routing for any Routing Rule is defined through FlexRM. FlexRM generates Action Requests when a Rule matches the |

data value contained in a document. An abbreviation of "Flexible Route Module."
A standard KEW routing scheme that is based on rules rather than dedicated table-based routing.

Force Action A true/false flag that indicates if previous Routing for approval will be ignored when an [Action Request](#) is generated. The flag is used in multiple contexts where requests are generated (e.g., rules, ad hoc routing). If Force Action is False, then prior Actions taken by a user can satisfy newly generated requests. If it is True, then the user needs to take another Action to satisfy the request.

FYI A workflow action request that can be cleared from a user's Action List with or without opening and viewing the document. A document with no pending approval requests but with pending Acknowledge requests is in Processed status. A document with no pending approval requests but with pending FYI requests is in Final status. See also [Ad Hoc Routing](#) and [Action Request](#).

G

Group A Group has members that can be either [Principals](#) or other Groups (nested). Groups essentially become a way to organize Entities (via Principal relationships) and other Groups within logical categories.

Groups can be given authorization to perform actions within applications by assigning them as members of [Roles](#).

Groups can also have arbitrary identity information (i.e., [Group Attributes](#) hanging from them. Group Attributes might be values for "Office Address," "Group Leader," etc.

Groups can be maintained at runtime through a user interface that is capable of workflow.

Group Attribute Groups have directory-like information called Group Attributes hanging from them. "Group Phone Number" and "Team Leader" are examples of Group Attributes.

Group Attributes make up the identity information for a Group record.

Group Attributes can be maintained at runtime through a user interface that is capable of workflow.

H

Hierarchical Tree Structure A hierarchical representation of data in a graphical form.

I

Initialized The state of an Action Request when it is first created but has not yet been Activated (sent to a user's Action List).

Initiated A workflow routing status indicating a document has been created but has not yet been saved or routed. A Document Number is automatically assigned by the system.

Initiator A user role for a person who creates (initiates or authors) a new document for routing. Depending on the permissions associated with the Document Type, only certain users may be able to initiate documents of that type.

Inquiry A screen that allows a user to view information about a business object.

J

Join Node The point in the routing path where multiple branches are joined together. A Join Node typically has a corresponding [Split Node](#) for which it joins the branches.

K

KC - Kualii Coeus TODO

KCA - Kualii Commercial Affiliates A designation provided to commercial affiliates who become part of the Kualii Partners Program to provide for-fee guidance, support, implementation, and integration services related to the Kualii software. Affiliates hold no ownership of Kualii intellectual property, but are full KPP participants. Affiliates may provide packaged versions of Kualii that provide value for installation or integration beyond the basic Kualii software. Affiliates may also offer other types of training, documentation, or hosting services.

KCB – Kualii Communications Broker KCB is logically related to KEN. It handles dispatching messages based on user preferences (email, SMS, etc.).

KEN - Kualii Enterprise Notification A key component of the Enterprise Integration layer of the architecture framework. Its features include:

- Automatic Message Generation and Logging
- Message integrity and delivery standards
- Delivery of notifications to a user’s Action List

KEW – Kualii Enterprise Workflow Kualii Enterprise Workflow is a general-purpose electronic routing infrastructure, or workflow engine. It manages the creation, routing, and processing of electronic documents (eDocs) necessary to complete a transaction. Other applications can also use Kualii Enterprise Workflow to automate and regulate the approval process for the transactions or documents they create.

KFS – Kualii Financial System Delivers a comprehensive suite of functionality to serve the financial system needs of all Carnegie-Class institutions. An enhancement of the proven functionality of Indiana University's Financial Information System (FIS), KFS meets GASB and FASB standards while providing a strong control environment to keep pace with advances in both technology and business. Modules include financial transactions, general ledger, chart of accounts, contracts and grants, purchasing/accounts payable, labor distribution, budget, accounts receivable and capital assets.

KIM – Kualii Identity Management A Kualii Rice module, Kualii Identity Management provides a standard API for persons, groups, roles and permissions that can be implemented by an institution. It also provides an out of the box reference implementation that allows for a university to use Kualii as their Identity Management solution.

| | |
|--|--|
| KNS – Kuali Nervous System | A core technical module composed of reusable code components that provide the common, underlying infrastructure code and functionality that any module may employ to perform its functions (for example, creating custom attributes, attaching electronic images, uploading data from desktop applications, lookup/search routines, and database interaction). |
| KPP - Kuali Partners Program | The Kuali Partners Program (KPP) is the means for organizations to get involved in the Kuali software community and influence its future through voting rights to determine software development priorities. Membership dues pay staff to perform Quality Assurance (QA) work, release engineering, packaging, documentation, and other work to coordinate the timely enhancement and release of quality software and other services valuable to the members. Partners are also encouraged to tender functional, technical, support or administrative staff members to the Kuali Foundation for specific periods of time. |
| KRAD - Kuali Rapid Application Development | TODO |
| KRMS - Kuali Rules Management System | TODO |
| KS - Kuali Student | Delivers a means to support students and other users with a student-centric system that provides real-time, cost-effective, scalable support to help them identify and achieve their goals while simplifying or eliminating administrative tasks. The high-level entities of person (evolving roles-student, instructor, etc.), time (nested units of time - semesters, terms, classes), learning unit (assigned to any learning activity), learning result (grades, assessments, evaluations), learning plan (intentions, activities, major, degree), and learning resources (instructors, classrooms, equipment). The concierge function is a self-service information sharing system that aligns information with needs and tasks to accomplish goals. The support for integration of locally-developed processes provides flexibility for any institution's needs. |
| KSB – Kuali Service Bus | Provides an out-of-the-box service architecture and runtime environment for Kuali Applications. It is the cornerstone of the Service Oriented Architecture layer of the architectural reference framework. The Kuali Service Bus consists of: <ul style="list-style-type: none"> • A services registry and repository for identifying and instantiating services • Run time monitoring of messages • Support for synchronous and asynchronous service and message paradigms |
| Kuali | <ol style="list-style-type: none"> 1. Pronounced "ku-wah-lee". A partnership organization that produces a suite of community-source, modular administrative software for Carnegie-class higher education institutions. See also Kuali Foundation 2. (n.) A humble kitchen wok that plays an important role in a successful kitchen. |
| Kuali Foundation | Employs staff to coordinate partner efforts and to manage and protect the Foundation's intellectual property. The Kuali Foundation manages a growing portfolio of enterprise software applications for colleges and universities. A lightweight Foundation staff coordinates the activities of Foundation members for critical software development and coordination activities such as source code control, release engineering, packaging, documentation, project management, |

software testing and quality assurance, conference planning, and educating and assisting members of the Kualo Partners program.

Kualo Rice

Provides an enterprise-class middleware suite of integrated products that allow both Kualo and non-Kualo applications to be built in an agile fashion, such that developers are able to react to end-user business requirements in an efficient manner to produce high-quality business applications. Built with Service Oriented Architecture (SOA) concepts in mind, KR enables developers to build robust systems with common enterprise workflow functionality, customizable and configurable user interfaces with a clean and universal look and feel, and general notification features to allow for a consolidated list of work "action items." All of this adds up to providing a re-usable development framework that encourages a simplified approach to developing true business functionality as modular applications.

L

Last Modified Date

The date on which the document was last modified (e.g., the date of the last action taken, the last action request generated, the last status changed, etc.).

M

Maintenance Document

An e-doc used to establish and maintain a table record.

Message

The full description of a [notification message](#). This is a specific field that can be filled out as part of the Simple Message or Event Message form. This can also be set by the programmatic interfaces when sending notifications from a client system.

Message Queue

Allows administrators to monitor messages that are flowing through the Service Bus. Messages can be edited, deleted or forwarded to other machines for processing from this screen.

N

Namespace

A Namespace is a way to scope both [Permissions](#) and [Entity Attributes](#). Each Namespace instance is one level of scoping and is one record in the system. For example, "KRA" or "KC" or "KFS" could be a Namespace. Or you could further break those up into finer-grained Namespaces such that they would roughly correlate to functional modules within each application. Examples could be "KRA Rolodex", "KC Grants", "KFS Chart of Accounts".

Out of the box, the system is bootstrapped with numerous Rice namespaces which correspond to the different modules. There is also a default namespace of "KUALO".

Namespaces can be maintained at runtime through a maintenance document.

Note Text

A free-form text field for the text of a Note

Notification Content

This section of a [notification message](#) which displays the actual full message for the notification along with any other content-type-specific fields.

Notification Message The overall Notification item or Notification Message that a user sees when she views the details of a notification in her Action List. A Notification Message contains not only common elements such as Sender, Channel, and Title, but also content-type-specific fields.

O

OOTB Stands for "out of the box" and refers to the base deliverable of a given feature in the system.

Optimistic Locking A type of "locking" that is placed on a database row by a process to prevent other processes from updating that row before the first process is complete. A characteristic of this locking technique is that another user who wants to make modifications at the same time as another user is permitted to, but the first one who submits their changes will have them applied. Any subsequent changes will result in the user being notified of the optimistic lock and their changes will not be applied. This technique assumes that another update is unlikely.

Optional Rule Extension Attribute An Extension Attribute that is not required in a Rule Template. It may or may not be present in a [Routing Rule](#) created from the Template. It can be used as a conditional element to aid in deciding if a Rule matches. These Attributes are simply additional criteria for the Rule matching process.

Org Doc # The originating document number.

Organization Refers to a unit within the institution such as department, responsibility center, campus, etc.

Organization Code Represents a unique identifier assigned to units at many different levels within the institution (for example, department, responsibility center, and campus).

P

Parameter Component Code Code identifying the parameter Component.

Parameter Description This field houses the purpose of this parameter.

Parameter Name This will be used as the identifier for the parameter. Parameter values will be accessed using this field and the namespace as the key.

Parameter Type Code Code identifying the parameter type. Parameter Type Code is the primary key for its' table.

Parameter Value This field houses the actual value associated with the parameter.

Parent Document Type A Document Type from which another [Document Type](#) derives. The child type can inherit certain properties of the parent type, any of which it may override. A Parent Document Type may have a parent as part of a hierarchy of document types.

Parent Rule A Routing Rule in KEW from which another Routing Rule derives. The child Rule can inherit certain properties of the parent Rule, any of which it may override. A Parent Rule may have a parent as part of a hierarchy of Rules.

Permission Permissions represent fine grained actions that can be mapped to functionality within a given system. Permissions are scoped to [Namespace](#) which roughly correlate to modules or sections of functionality within a given system.

A developer would code authorization checks in their application against these permissions.

Some examples would be: "canSave", "canView", "canEdit", etc.

Permissions are aggregated by [Roles](#).

Permissions can be maintained at runtime through a user interface that is capable of workflow; however, developers still need to code authorization checks against them in their code, once they are set up in the system.

Attributes

1. Id - a system generated unique identifier that is the primary key for any Permission record in the system
2. Name - the name of the permission; also a human understandable unique identifier
3. Description - a full description of the purpose of the Permission record
4. Namespace - the reference to the associated [Namespace](#)

Relationships

1. Permission to [Role](#) - many to many; this relationship ties a Permission record to a Role that is authorized for the Permission
2. Permission to [Namespace](#) - many to one; this relationship allows for scoping of a Permission to a Namespace that contains functionality which keys its authorization checking off of said

| | |
|---------------------|---|
| Person Identifier | The username of an individual user who receives the document ad hoc for the Action Requested |
| Person Role | Creates or maintains the list used in selection of personnel when preparing the Routing Form document. |
| Pessimistic Locking | A type of lock placed on a database row by a process to prevent other processes from reading or updating that row until the first process is finished. This technique assumes that another update is likely. |
| Plugins | A plugin is a packaged set of code providing essential services that can be deployed into the Rice standalone server. Plugins usually contains only classes used in routing such as custom rules or searchable attributes, but can contain client application specific services. They are usually used only by clients being implemented by the 'Thin Client' method |
| Post Processor | A routing component that is notified by the workflow engine about various events pertaining to the routing of a specific document (e.g., node transition, status change, action taken). The implementation of a Post Processor is typically specific to a particular set of Document Types. When all required approvals are completed, the engine notifies the Post Processor accordingly. At this point, the Post Processor is responsible for completing the business transaction in the manner appropriate to its Document Type. |

| | |
|------------------------|--|
| Posted Date/Time Stamp | A free-form text field that identifies the time and date at which the Notes is posted. |
| Postal Code | Defines zip code to city and state cross-references. |
| Preferences | User options in an Action List for displaying the list of documents. Users can click the Preferences button in the top margin of the Action List to display the Action List Preferences screen. On the Preferences screen, users may change the columns displayed, the background colors by Route Status, and the number of documents displayed per page. |
| Primary Delegation | The Delegator turns over full authority to the Delegate. The Action Requests for the Delegator only appear in the Action List of the Primary Delegate. The Delegation must be registered in KEW or KIM to be in effect. |
| Principal | <p>A Principal represents an Entity that can authenticate into the system. One can roughly correlate a Principal to a login username. Entities can exist in KIM without having permissions or authorization to do anything; therefore, a Principal must exist and must be associated with an Entity in order for it to have access privileges. All authorization that is not specific to Groups is tied to a Principal.</p> <p>In other words, an Entity is for identity while a Principal is for access management.</p> <p>Also note that an Entity is allowed to have multiple Principals associated with it. The use case typically given here is that a person may apply to a school and receive one log in for the application system; however, once accepted, they may receive their official login, but use the same identity information set up for their Entity record.</p> |
| Processed | A routing status indicating that the document has no pending approval requests but still has one or more pending acknowledgement requests. |

R

| | |
|-----------------------------------|---|
| Recipient Type | The type of entity that is receiving an Action Request. Can be a user, workgroup, or role. |
| Required Rule Extension Attribute | An Extension Attribute that is required in a Rule Template. It will be present in every Routing Rule created from the Template. |
| Responsibility | See Responsible Party . |
| Responsibility Id | A unique identifier representing a particular responsibility on a rule (or from a route module). This identifier stays the same for a particular responsibility no matter how many times a rule is modified. |
| Responsible Party | The Reviewer defined on a routing rule that receives requests when the rule is successfully executed. Each routing rule has one or more responsible parties defined. |
| Reviewer | A user acting on a document in his/her Action List and who has received an Action Request for the document. |
| Rice | An abbreviation for Kualu Rice. |
| Role | Roles aggregate Permissions . When Roles are given to Entities (via their relationship with Principals) or Groups an authorization for all associated Permissions is granted. |

| | |
|-----------------|---|
| Route Header Id | Another name for the Document Id . |
| Route Log | Displays information about the routing of a document. The Route Log is usually accessed from either the Action List or a Document Search. It displays general document information about the document and a detailed list of Actions Taken and pending Action Requests for the document. The Route Log can be considered an audit trail for a document. |
| Route Module | A routing component that the engine uses to generate action requests at a particular Route Node . FlexRM (Flexible Route Module) is a general Route Module that is rule-based. Clients can define their own Route Modules that can conduct specialized Routing based on routing tables or any other desired implementation. |
| Route Node | <p>Represents a step in the routing process of a document type. Route node "instances" are created dynamically as a document goes through its routing process and can be defined to perform any function. The most common functions are to generate Action Requests or to split or join the route path.</p> <ul style="list-style-type: none">• Simple: do some arbitrary work• Requests: generate action requests using a Route Module or the Rules engine• Split: split the route path into one or more parallel branches• Join: join one or more branches back together• Sub Process: execute another route path inline• Dynamic: generate a dynamic route path |
| Route Path | The path a document follows during the routing process. Consists of a set of route nodes and branches. The route path is defined as part of the document type definition. |
| Route Status | <p>The status of a document in the course of its routing:</p> <ul style="list-style-type: none">• Approved: These documents have been approved by all required reviewers and are waiting additional postprocessing.• Cancelled: These documents have been stopped. The document's initiator can 'Cancel' it before routing begins or a reviewer of the document can cancel it after routing begins. When a document is cancelled, routing stops; it is not sent to another Action List.• Disapproved: These documents have been disapproved by at least one reviewer. Routing has stopped for these documents.• Enroute: Routing is in progress on these documents and an action request is waiting for someone to take action.• Exception: A routing exception has occurred on this document. Someone from the Exception Workgroup for this Document Type must take action on this document, and it has been sent to the Action List of this workgroup.• Final: All required approvals and all acknowledgements have been received on these documents. <u>No changes are allowed to a document that is in Final status.</u> |

- **Initiated:** A user or a process has created this document, but it has not yet been routed to anyone's Action List.
- **Processed:** These documents have been approved by all required users, and is completed on them. They may be waiting for Acknowledgements. No further action is needed on these documents.
- **Saved:** These documents have been saved for later work. An author (initiator) can save a document before routing begins or a reviewer can save a document before he or she takes action on it. When someone saves a document, the document goes on that person's Action List.

Routed By User The user who submits the document into routing. This is often the same as the Initiator. However, for some types of documents they may be different.

Routing The process of moving a document through its route path as defined in its Document Type. Routing is executed and administered by the workflow engine. This process will typically include generating Action Requests and processing actions from the users who receive those requests. In addition, the Routing process includes callbacks to the Post Processor when there are changes in document state.

Routing Priority A number that indicates the routing priority; a smaller number has a higher routing priority. Routing priority is used to determine the order that requests are activated on a route node with sequential activation type.

Routing Rule A record that contains the data for the [Rule Attributes](#) specified in a [Rule Template](#). It is an instance of a Rule Template populated to determine the appropriate Routing. The Rule includes the Base Attributes, Required Extension Attributes, Responsible Party Attributes, and any Optional Extension Attributes that are declared in the Rule Template. Rules are evaluated at certain points in the routing process and, when they fire, can generate Action Requests to the responsible parties that are defined on them.

Technical considerations for a Routing Rules are:

- Configured via a GUI (or imported from XML)
- Created against a Rule Template and a Document Type
- The Rule Template and its list of Rule Attributes define what fields will be collected in the Rule GUI
- Rules define the users, groups and/or roles who should receive action requests
- Available Action Request Types that Rules can route
 - Complete
 - Approve
 - Acknowledge
 - FYI
- During routing, Rule Evaluation Sets are "selected" at each node. Default is to select by Document Type and Rule Template defined on the Route Node

- Rules match (or ‘fire’) based on the evaluation of data on the document and data contained on the individual rule
- Examples
 - If dollar amount is greater than \$10,000 then send an Approval request to Joe.
 - If department is “HR” request an Acknowledgment from the HR.Acknowledgers workgroup.

Rule Attribute

Rule attributes are a core KEW data element contained in a document that controls its Routing. It participates in routing as part of a Rule Template and is responsible for defining custom fields that can be rendered on a routing rule. It also defines the logic for how rules that contain the attribute data are evaluated.

Technical considerations for a Rule Attribute are:

- They might be backed by a Java class to provide lookups and validations of appropriate values.
- Define how a Routing Rule evaluates document data to determine whether or not the rule data matches the document data.
- Define what data is collected on a rule.
- An attribute typically corresponds to one piece of data on a document (i.e dollar amount, department, organization, account, etc.).
- Can be written in Java or defined using XML (with matching done by XPath).
- Can have multiple GUI fields defined in a single attribute.

Rule QuickLinks

A list of document groups with their document hierarchies and actions that can be selected. For specific document types, you can create the rule delegate.

Rule Template

A Rule Template serves as a pattern or design for the routing rules. All of the Rule Attributes that include both Required and _Optional_ are contained in the Rule Template; it defines the structure of the routing rule of FlexRM. The Rule Template is also used to associate certain Route Nodes on a document type to routing rules.

Technical considerations for a Rule Templates are:

- They are a composition of Rule Attributes
- Adding a ‘Role’ attribute to a template allows for the use of the Role on any rules created against the template
- When rule attributes are used for matching on rules, each attribute is associated with the other attributes on the template using an implicit ‘and’ logic attributes
- Can be used to define various other aspects to be used by the rule creation GUI such as rule data defaults (effective dates, ignore previous, available request types, etc)

S

| | |
|---------------------------|---|
| Save | A workflow action button that allows the Initiator of a document to save their work and close the document. The document may be retrieved from the initiator's Action List for completion and routing at a later time. |
| Saved | A routing status indicating the document has been started but not yet completed or routed. The Save action allows the initiator of a document to save their work and close the document. The document may be retrieved from the initiator's action list for completion and routing at a later time. |
| Searchable Attributes | <p>Attributes that can be defined to index certain pieces of data on a document so that it can be searched from the Document Search screen.</p> <p>Technical considerations for a Searchable Attributes are:</p> <ul style="list-style-type: none">• They are responsible for extracting and indexing document data for searching• They allow for custom fields to be added to Document Search for documents of a particular type• They are configured as an attribute of a Document Type• They can be written in Java or defined in XML by using Xpath to facilitate matching |
| Secondary Delegation | <p>The Secondary Delegate acts as a temporary backup Delegator who acts with the same authority as the primary Approver/the Delegator when the Delegator is not available. Documents appear in the Action Lists of both the Delegator and the Delegate. When either acts on the document, it disappears from both Action Lists.</p> <p>Secondary Delegation is often configured for a range of dates and it must be registered in KEW or KIM to be in effect.</p> |
| Service Registry | Displays a read-only view of all of the services that are exposed on the Service Bus and includes information about them (for example, IP Address, or Endpoint URL). |
| Simple Node | A type of node that can perform any function desired by the implementer. An example implementation of a simple node is the node that generates Action Requests from route modules. |
| SOA | An acronym for Service Oriented Architecture. |
| Special Condition Routing | This is a generic term for additional route levels that might be triggered by various attributes of a transaction. They can be based on the type of document, attributes of the accounts being used, or other attributes of the transaction. They often represent special administrative approvals that may be required. |
| Split Node | A node in the routing path that can split the route path into multiple branches. |
| Spring | The Spring Framework is an open source application framework for the Java platform. |
| State | Defines U.S. Postal Service codes used to identify states. |
| Status | On an Action List; also known as Route Status. The current location of the document in its routing path. |

Submit A workflow action button used by the initiator of a document to begin workflow routing for that transaction. It moves the document (through workflow) to the next level of approval. Once a document is submitted, it remains in 'ENROUTE' status until all approvals have taken place.

Superuser A user who has been given special permission to perform Superuser Approvals and other Superuser actions on documents of a certain Document Type.

Superuser Approval Authority given Superusers to approve a document of a chosen Route Node. A Superuser Approval action bypasses approvals that would otherwise be required in the Routing. It is available in Superuser Document Search. In most cases, reviewers who are skipped are not sent Acknowledge Action Requests.

Superuser Document Search A special mode of Document Search that allows Superusers to access documents in a special Superuser mode and perform administrative functions on those documents. Access to these documents is governed by the user's membership in the Superuser Workgroup as defined on a particular Document Type.

T

Thread pool A technique that improves overall system performance by creating a pool of threads to execute multiple tasks at the same time. A task can execute immediately if a thread in the pool is available or else the task waits for a thread to become available from the pool before executing.

Title A short summary of the notification message. This field can be filled out as part of the Simple Message or Event Message form. In addition, this can be set by the programmatic interfaces when sending notifications from a client system.

This field is equivalent to the "Subject" field in an email.

U

URL An acronym for Uniform Resource Locator.

User A person who can log in and use the application. This term is synonymous with "Principal" in KIM. "Whereas Entity Id represents a unique Person, Principal Id represents a set of login information for that Person."

V

Viewer A user(s) who views a document during the routing process. This includes users who have action requests generated to them on a document.

W

Web Service Client A type of client that connects to a standalone KEW server using Web Services.

Wildcard A character that may be substituted for any of a defined subset of all possible characters.

Workflow Electronic document routing, approval and tracking. Also known as Workflow Services or Kualu Enterprise Workflow (KEW). The Kualu infrastructure service

that electronically routes an e-doc to its approvers in a prescribed sequence, according to established business rules based on the e-doc content. See also [Kuali Enterprise Workflow](#).

Workflow Engine

The component of KEW that handles initiating and executing the route path of a document.

Workflow QuickLinks

A web interface that provides quick navigation to various functions in KEW. These include:

- Quick EDoc Watch: The last five Actions taken by this user. The user can select and repeat these actions.
- Quick EDoc Search: The last five EDocs searched for by this user. The user can select one and repeat that search.
- Quick Action List: The last five document types the user took action with. The user can select one and repeat that action.

X

XML

See also [XML Ingestor](#).

1. An acronym for Extensible Markup Language.
2. Used for data import/export.

XML Ingestor

A workflow function that allows you to browse for and upload XML data.

XML RuleAttribute

Similar in functionality to a RuleAttribute but built using XML only