

Configuration Properties 2

Bold, italicized properties are very important and most likely must be changed from their defaults during installation. **Bold properties** are likely to need modification during implementation.

Category	Property	Default	Notes
Database			
	drivers.directory	/java/drivers	determines where the build will look for your database drivers
	datasource.objb.platform	Oracle9i	used by the build to configure OJB and determine the appropriate driver and KualIDBPlatform implementations to use
	<i>datasource.username</i>		account that the application should use to access the database
	<i>datasource.password</i>		password for the application database account
	<i>oracle.datasource.url</i>	jdbc:oracle:thin:@localhost:1521:XE	JDBC URL that will be used by the application, if the <code>datasource.objb.platform</code> is set to Oracle9i
	<i>mysql.datasource.url</i>	jdbc:mysql://localhost:3306/\${datasource.username}	JDBC URL that will be used by the application, if the <code>datasource.objb.platform</code> is set to MySQL
	datasource.connection.timeout	14400	used in datasource configuration to force transactions to timeout after they've run too long
	rice.workflow.datasource.driver.class	<i>derived (see notes)</i>	this will default to the value that corresponds to the <code>datasource.objb.platform</code> specified when KFS is built
	rice.workflow.datasource.url	<i>derived (see notes)</i>	jndi url for the workflow database when <code>rice.use.standalone.workflow</code> is true. this will default to the value that corresponds to the <code>datasource.objb.platform</code> specified when KFS is built
	rice.workflow.datasource.username	\${datasource.username}	account for the workflow database when <code>rice.use.standalone.workflow</code> is true
	rice.workflow.datasource.password	\${datasource.password}	password for the workflow database account when <code>rice.use.standalone.workflow</code> is true
	rice.workflow.datasource.pool.size	\${datasource.pool.size}	# of connections to the workflow database when <code>rice.use.standalone.workflow</code> is true
	rice.workflow.datasource.connection.timeout	\${datasource.connection.timeout}	timeout in seconds for the workflow database connections when <code>rice.use.standalone.workflow</code> is true

	rice.workflow.datasource.validating.query	\${datasource.validating.query}	query that should be issued to verify the workflow database connections when <code>rice.use.s tandalone.workflow</code> is true
Application Server			
	appserver.home	/java/servers/apache-tomcat-5.5.16	determines where the build will look for your application server
	appserver.lib.dir	\${appserver.home}/common/lib	
	appserver.classes.dir	\${appserver.home}/common/classes	
	appserver.deploy.dir	\${appserver.home}/webapps	
	appserver.config.dir	\${appserver.home}/conf	
	appserver.localhost.dir	\${appserver.config.dir}/Catalina/localhost	
	appserver.work.dir	\${appserver.home}/work/Catalina/localhost	
	appserver.url	http://localhost:8080/	tells the build what the base URL is for the application server in which the application will be deployed
Directories			
	external.config.directory	/opt	the base directory for external configuration files and work directories
	security.directory	\${external.config.directory}/sa_forms/java/\${build.environment}/\${ant.project.name}	the directory where secure application property files are located (application server user must have read access)
	settings.directory	\${external.config.directory}/j2ee/\${build.environment}/\${ant.project.name}	the directory that contains the non-secure property files that are nonetheless external to the deployed application (application server user must have read access)
	logs.directory	\${external.config.directory}/logs/\${build.environment}/\${ant.project.name}	the directory for the log files output by the application (application server user must have write access)
	external.work.directory	\${external.config.directory}/work/\${build.environment}/\${ant.project.name}	the base working directory that is external to the application (used for storage of attachments, batch input files, and generated reports - application server user must have read/write access to all subdirectories)
	attachments.directory	\${external.work.directory}/attachments	the base working directory for storage of note-associated attachments that are uploaded by users
	rice.workflow.attachment.directory	\${attachments.directory}	the directory where attachments uploaded through the workflow ui are stored
	staging.directory	\${external.work.directory}/staging	the base working directory for storage of batch input files

	rice.workflow.xml.directory	\${staging.directory}/workflow	the directory where the xml files that define your workflow data should be stored when they are uploaded
	reports.directory	\${external.work.directory}/reports	the base working directory for oltp and batch report generation
Configuration Files			
	security.property.file	\${security.directory}/\${security.file}	the single secure property file currently used by the application
	log4j.settings.file	\${settings.directory}/\${log4j.file}	the property file that contains the log4j configuration for the application
	rice.workflow.log4j.settings.path	\${log4j.settings.file}	the path to the file containing the log4j configuration for workflow
	encrypt.attributes.properties.file	fieldsToEncrypt.properties	the file that lists the tables and columns that should be processed by the PostDataLoad EncryptionStep
URLs			
	application.url	\${appserver.url}\${ant.project.name}-\${build.environment}	the base url of the application
	rice.url	\${application.url}/\${rice.application}	the base url for the rice struts module
	rice.ksb.url	\${application.url}/\${ksb.application}	the base url for the service bus struts module
	rice.workflow.url	\${application.url}/\${workflow.application}	the base url for the workflow struts module or the workflow application on the central server
	htdocs.url	\${appserver.url}\${ant.project.name}/\${build.environment}/	the base url for environment-based static content that is external to the application
	htdocs.logs.url	\${htdocs.url}logs/	the base url for the logs directory, if accessible via the web
	htdocs.staging.url	\${htdocs.url}staging/	the base urls for the staging directory, if accessible via the web
	externalizable.static.content.url	\${application.url}/static	the base url for static content like images and help that could reside within the application or be maintained by other group(s) on other server(s)
	externalizable.help.url	\${externalizable.static.content.url}/help/	the base url for help content
	externalizable.images.url	\${externalizable.static.content.url}/images/	the base url for images that can reside outside the application
	rice.externalizable.images.url	\${application.url}/\${rice.application}/static/images/	the base url for rice images that can reside outside the application
	externalizable.xml.url	\${externalizable.static.content.url}/xml/	the base url for xml (xsd, xsl, etc.) related files that can reside outside the application
Miscellaneous Configuration			

	is.local.build	true	this property determines whether the build will do a local deployment or package for a deployment on the test node
	build.environment	dev	this is the kual and workflow environment
	rice.workflow.build.environment	\${build.environment}	the workflow deployment environment that this deployment of the application should use (can only differ from build.environment if rice.use.standalone.workflow is true)
	production.environment.code	prd	determines what build.environment the KualConfigurationService considers your production environment
	tests.includes	**/*Test.java	this property determines what tests are included when the test target is executed
	memory.monitor.threshold	.85	determines when the MemoryMonitor considers the application to have exceeded reasonable memory usage (decimal represents percentage of tenured allocation)
Logging			
	use.p6spy	false	this property can be used to turn on logging of the SQL generated and executed by OJB
	appenders	INFO, LogFile, StdOut	this property determines what log statements will be printed by default and whether the logs go to the log file or standard out
	log4j.reload.minutes	5	determines how frequently the logging framework (Log4jConfigurer) will check for changes to log4j.properties
Batch			
	use.quartz.scheduling	true	determines whether quartz (SchedulerService and scheduleJob) should be used for batch job scheduling and execution - impacts whether the batch scheduling ui works
	batch.schedule.cron.expression	00 00 23 ? * *	the Quartz cron expression for the scheduleJobTrigger (in KualSpringBeansKfs.xml), i.e. when should the standard batch schedule run
Service Bus			
	rice.message.delivery	async	determines whether the workflow processes that would normally be performed asynchronously are asynchronous (async) or are synchronous instead

	rice.route.queue.max.retry.attempts	3	how many times the service bus will try to send a message when it is encountering an exception that may be due to a resource outage, for example
	rice.route.queue.time.increment	1000	the length of time the service bus will wait before retrying after each failed attempt to send a message
	rice.thread.pool.size	20	the number of threads available to process service bus messages, i.e. deal with queued workflow documents
Workflow			
	rice.use.standalone.workflow	false	are multiple applications at your institution using workflow, and do you therefore point to a central workflow deployment that they all use? note: if you change this property value to true, you should carefully review all properties with the prefix <code>rice.workflow</code> and read the detailed configuration instructions . also, properties with the prefix <code>rice.workflow.datasource</code> will only be used when this property is set to true. you may also need to change the value of <code>spring.plugin.files</code> .
	rice.workflow.xml.poll.interval.seconds	30	the frequency with which workflow checks for new files in the xml directory
	rice.workflow.xml.initial.delay.seconds	10	the length of time workflow waits to do the first check of the xml directory when the application starts up
Customization Hooks			
	spring.source.files	org/kuali/kfs/KualiSpringBeansKfs.xml, org/kuali/module/cg/KualiSpringBeansCg.xml, org/kuali/module/chart/KualiSpringBeansChart.xml, org/kuali/module/financial/KualiSpringBeansFinancial.xml, org/kuali/module/gl/KualiSpringBeansGl.xml, org/kuali/module/kra/KualiSpringBeansKra.xml, org/kuali/module/labor/KualiSpringBeansLabor.xml, org/kuali/module/pdp/KualiSpringBeansPdp.xml, org/kuali/module/purap/KualiSpringBeansPurap.xml, org/kuali/module/vendor/KualiSpringBeansVendor.xml	the files that define the spring context for the application - note that further configuration files may be specified in the module definitions
	spring.test.files	SpringBeansTest.xml,org/kuali/kfs/KualiSpringBeansKfsTest.xml	the files that are included in the application spring context for unit testing

	spring.plugin.files	SpringRicePluginBeans.xml	the files that are included in the application spring context when running inside of a workflow plugin
	property.files	classpath:ApplicationResources.properties,file: \${security.property.file}	the configuration property files that should be available in the KualConfigurationService context, along with configuration.properties
	property.test.files	classpath:test-configuration.properties	the configuration property files that should be added to the KualConfigurationService context for unit testing
	struts.config.files	/WEB-INF/struts-config.xml	the files that define the struts context for the application
	rice.struts.config.files	/kr/WEB-INF/struts-config.xml	the files that define the struts context for the rice struts module
	rice.workflow.struts.config.files	/en/WEB-INF/struts-config.xml	the files that define the struts context for the workflow struts module
	rice.ksb.struts.config.files	/ksb/WEB-INF/struts-config.xml	the files that define the struts context for the ksb struts module
	struts.message.resources	ApplicationResources	the files containing user messages that will be displayed via struts tags in jsps
	jstl.constants.classnames	org.kuali.kfs.KFSConstants,org.kuali.module.gl.GLConstants,org.kuali.module.kra.KraConstants,org.kuali.module.purap.PurapConstants	the classes that wrap java constants for jsps and tags (see ApplicationInitializeListener)
	jstl.constants.main.class	org.kuali.kfs.KFSConstants	used to determine what class from the above (jstl.constants.classnames) list is bound to the "Constants" name in JSTL
	javascript.files	kr/scripts/core.js,kr/scripts/dhtml.js, kr/scripts/my_common.js, kr/scripts/jscalendar-1.0/calendar.js, kr/scripts/jscalendar-1.0/lang/calendar-en.js, kr/scripts/jscalendar-1.0/calendar-setup.js, dwr/engine.js, dwr/util.js, dwr/interface/UserService.js, kr/scripts/objectInfo.js	the javascript files that should be available to every page in the application
	css.files	kr/css/kuali.css,kr/scripts/jscalendar-1.0/calendar-win2k-1.css	the css files that should be available to every page in the application
	portal.javascript.files	kr/scripts/my_common.js	the javascript files that should be available to the portal
	portal.css.files	css/portal.css	the css files that should be available to the portal
Service Configuration			
Authentication			

	require.https	false	determines whether cas requires that the application be secured via ssl
	validate.password	false	determines whether the password field appears on the cas login screen and is validated and controls the appearance of the password field on the user document when <code>maintain.users.locally</code> is true
User			
	maintain.users.locally	true	determines whether the institutional user data is ever editable on the user document
Encryption			
	encryption.key	71C64w6ksLU	used by the <code>DemonstrationGradeEncryptionService</code> and the <code>workflowEncryptionService</code> to encrypt / decrypt data in the database
Mail			
	mail.relay.server		determines the mail relay server used by <code>MailService</code>
	batch.mailing.list		from email address for mail sent by the batch framework (<code>JobListener</code> and other) and to email address for job failure messages

Unable to render {include} The included page could not be found.