Customizations 2

KFS and Rice include many methods of customization. The first point of functional customization is the system parameters table, which allows control of valid values for document fields, configuration of batch processes, groups used to limit access to certain actions, and specification of relative document-level help URLs. The first points of technical customization are the configuration properties file and the web.xml file. Many configuration properties open the door to further customizations.

The single most powerful configuration property is `spring.source.files`. This hook facilitates service and other spring bean overrides. One of the most powerful override options is the module definition. Module definitions are what control the tables that the application knows about, the DWR files that can be used to expose services to javascript, the code that is responsible for authorizing access to the lookups, inquiries, and documents in that module, etc. Overriding a module definition also paves the way to overriding data dictionary entries, which control the organization and behavior of the user interface, validation of user input, authorization to initiate documents, etc. Extended attributes bring all of these technical configuration options together to allow you to actually add fields to delivered tables, without modifying delivered code.

One type of customization option that is noticeably absent in the above paragraph is workflow customization: routing and document searches. Although the data dictionary can be used to control what gets put in the XML that is given to workflow for each document, what workflow will do with that XML is determined by another type of XML configuration that is given to KEW itself and resides in the database. See the *Workflow* menu on the *Administration* menu group of the KFS portal for related searches and access to the XML Ingester user interface, which will allow you to upload these configuration files. Configuration files may also be placed in the directory corresponding to the `rice.workflow.xml.directory` configuration property, so that the workflow engine will automatically load them on startup.

> ⚠ Before you decided to use any technical customization options, you should familiarize yourself with our upgrade strategy and recommendations on customization.