





# Gap Analysis - Transactional Document






Analysis JIRA:










KULRICE-8795 - Data cannot be retrieved due to an unexpected error


- Involved Classes

Google Doc Gap Analysis











ID	KNS Feature	Implemented in KRAD	Notes
T1	<p>Support for a "Use Transactional Document" permission which was used to control the availability of various sections/features of the document. One of the permission details was an "editMode", which predates the KIM conversion. It was a simple string flag which was stored in the request and checked by the JSP code (and business rules) to determine what the user could do with the document at the present time.</p> <p>This feature was also used to present completely different views on a few KFS documents.</p>	<p> KULRICE-10772</p> <p>UXI Prioritized</p>	<p>Ideally, other KRAD features should be used to replace the concept of "editMode", but that may not be practical upon the initial conversion. It will need to be supported, but could be deprecated as a backwards-compatibility/conversion-support feature only.</p> <p>In KNS JSP files, these were typically checked as below:</p> <pre>&lt;c:if test="{!empty Kualiform.editingMode[amendmentEntry]}"&gt;</pre>
T2	<p>Ability to "copy" a transactional document.</p> <p>"copy" was a new action which could be taken on transactional documents. It checked the allowed actions (as given by the presentation controller (which checked the document class for the "Copyable" interface AND checked the DD for the allowsCopy attribute), and filtered by the document authorizer. (which checked the "Copy Document" permission)</p>	<p> Permission check is missing in the transactional document controller!</p>	<p>The Copyable interface is probably unnecessary. It likely pre-dates the attribute on the DD. The interface has the toCopy method. But, there is likely no reason that we could not remove that interface and add toCopy to the base Document interface and then control this from the DD. (The method is there already.)</p> <p>The main reason is that there is always code needed to copy a document. But, if we can provide a generic behavior of toCopy, that should not be a problem. (BTW - the implementation assumes that you have already loaded the document you want to copy and then want to make <b>that instance</b> into the copy. (Which means fully detaching from JPA and then changing the PK fields throughout.)</p>
T3	<p>Semi-support for non-session documents.</p> <p>org.kuali.rice.krad.document.TransactionalDocumentBase contains an isSessionDocument() method. This should be removed, as all documents are session documents these days.</p>	<p></p>	
T4	<p>Mappings of Struts Form and Action classes needed to be done via the struts-config.xml files.</p>	<p></p>	<p>Not needed in KRAD since View model is completely different and the use of Spring MVC replaces this configuration model.</p>

T5	Display of custom header fields at the top (see <code>page.tag</code> ) if added into the Form data.		<p>KFS uses this to display additional document status information as well as document linking information. (for correction documents)</p> <p><b>QUESTION (JR):</b> KRAD provides support for defining your header with whatever form fields you would like through the view configuration. KNS automatically creates the header based on the contents of <code>KualiForm.docInfo</code> (done in <code>page.tag</code>). Does this need to be included in KRAD even though it can be done through configuration of the form?</p>
T6	Display of custom header navigation tabs provided by the Form.		<p>I assume this would be replaced with standard KRAD View setup, as the use of <code>page.tag</code> will be removed.</p> <p><b>QUESTION (JR):</b> KRAD provides support for defining your header navigation with whatever form fields you would like through the view configuration. KNS automatically creates the header navigation based on the contents of <code>KualiForm.headerNavigationTabs</code> (done in <code>page.tag</code>). Does this need to be included in KRAD even though it can be done through configuration of the form?</p>
T7	Automatic display of the document title (as stored in the label of the KEW document type)	 KULRICE-10775 UXI Prioritized	
T8	Automatic display of the "template" document number. This is populated in the baseline code when a document is copied from another.	 KULRICE-10775 UXI Prioritized	
T9	Like the maintenance documents, supports an instance of <code>PromptBeforeValidation</code> and <code>DerivedValuesSetter</code> in the DD.		
T10	Like maintenance documents, can provide a list of custom JS script files in the DD.		
T11	In the DD, ability to provide a list of additional header navigation links which will be rendered automatically. See the file reference below for the attributes available on this class.		<p>Not used by KFS at all. Probably there for KC screens. I'm assuming this will be replaced by KRAD view definitions.</p> <p>However, we should make sure that it is possible to turn on/off tabs dynamically as part of the View lifecycle, since that is the reason that links like this would be built into the DD and not part of the JPS layer.</p>

T12	Retaining of tab open/closed state between requests.		
T13	Ability to render fields and labels using standard custom tags by passing in the appropriate AttributeDefinition entry. (Or Map representation thereof.)		See: <pre>&lt;kul:htmlAttributeHeader Cell /&gt; &lt;kul:htmlAttributeLabel /&gt; &lt;kul:htmlAttributeHeader Cell /&gt; &lt;kul:htmlControlAttribut e /&gt;</pre>
T14	Ability to add custom methods to the action/controller class which can be invoked via the standard "methodToCall" syntaxes.		
T15	Ability to override baseline action/controller methods to add behavior. (Overriding code is still responsible for calling the superclass.)		Consider allowing the registration of additional customization "hooks" which can be registered for baseline events. This would allow for custom methods to be called and not rely on superclass functions being called. This would allow the baseline code to always run and then call through the list of custom hooks at the appropriate time(s). (before/after hooks?)
T16	Ability to intercept the methodToCall processing and tweak before the requested method gets called. (Struts execute() method.)		
T17	Connection to PessimisticLockService to block pre-routing document actions if the check fails.		
T18	Support of the concept of "document exiting"-type actions which change behavior. Values are stored in the session which allow actions to be taken after the action is complete.		It seems to be related to clearing the Form objects from memory (the UserSession) and partly for pessimistic locking clearing.
T19	Support for prompting users for a note/workflow annotation when taking certain types of workflow actions (cancel/disapprove/etc...)	 KULRICE-10756 UXI Prioritized	
T20	Ability to intercept/customize the form field to data object bindings by overriding the populate() method on the Struts Form class.		

T21	Ability to override the base "parent" beans used in most DD definitions for the client applications to add/change defaults and add new properties and behaviors.		For examples, see KFS:  work/src/org/kuali/kfs/sys/document/datadictionary/FinancialSystemDataDictionaryBaseTypes.xml  work/src/org/kuali/kfs/sys/document/datadictionary/FinancialSystemTransactionalDocument.xml org.kuali.kfs.sys.document.datadictionary.FinancialSystemTransactionalDocumentEntry
-----	--	---	---

## Involved Classes

Class	Notes	Finished Analysis
<b>Struts Classes</b>		
org.kuali.rice.kns.web.struts.action.KualiTransactionalDocumentActionBase		
org.kuali.rice.kns.web.struts.action.KualiDocumentActionBase		
org.kuali.rice.kns.web.struts.form.KualiTransactionalDocumentFormBase	Most of the methods on this class are left-over cruft from KFS. (formatReversalDate)  forcedReadOnlyFields is only used in a single place in all of KFS   Performs special setting of empty select lists and checkboxes to make sure they get blanked or set to false even when the attribute was not set by the browser.	
org.kuali.rice.kns.web.struts.form.KualiDocumentFormBase		
org.kuali.rice.kns.web.struts.form.KualiForm		
org.kuali.rice.kns.web.struts.form.pojo.PojoFormBase		
<b>General Classes</b>		
org.kuali.rice.krad.document.TransactionalDocumentBase		
org.kuali.rice.kns.web.ui.HeaderField	UI Holder for page.tag to use to display the document header fields.	
org.kuali.rice.kns.util.ActionFormUtilMap	A class which should die! Was placed in the Form to allow for configuration of some items which were difficult in the KNS. It should no longer be necessary in KRAD.	
<b>JSP/Tag Files</b>		
(No JSP Files - The master JSP for each transactional document was custom. They would use the base custom tag files plus those added by the client applications.)		

/rice-web/src/main/webapp/WEB-INF/tags/kr/documentControls.tag	Displayed the appropriate buttons at the bottom of the document page. Also had the ability for extra buttons to be displayed before the standard buttons if they were added to some properties on the base Form class.	✓
/rice-web/src/main/webapp/WEB-INF/tags/kr/documentOverview.tag		✓
/rice-web/src/main/webapp/WEB-INF/tags/kr/documentPage.tag		✓
/rice-web/src/main/webapp/WEB-INF/tags/kr/page.tag	Displayed the document header information (doc #, initiator, etc...)	✓
/rice-web/src/main/webapp/WEB-INF/tags/kr/documentOverview.tag		✓
/rice-web/src/main/webapp/WEB-INF/tags/kr/documentControls.tag		✓
/rice-web/src/main/webapp/WEB-INF/tags/kr/tab.tag		✓
<b>Data Dictionary Classes</b>		
org.kuali.rice.kns.datadictionary.Transaction alDocumentEntry		✓
org.kuali.rice.kns.datadictionary.HeaderNavigation	<b>protected</b> String headerTabMethodToCall; <b>protected</b> String headerTabNavigateTo; <b>protected</b> String headerTabDisplayName; <b>protected</b> HelpDefinition helpDefinition; <b>protected boolean</b> disabled = <b>false</b> ;	✓
org.kuali.rice.kns.datadictionary.KNSDocumentEntry		✓
<b>Presentation Controller and Authorization</b>		
org.kuali.rice.krad.document.DocumentPresentationControllerBase		✓
org.kuali.rice.kns.document.authorization.Transaction alDocumentPresentationControllerBase	(empty placeholder)	✓
org.kuali.rice.kns.document.authorization.Transaction alDocumentAuthorizerBase		✓
org.kuali.rice.kns.document.authorization.DocumentAuthorizerBase		✓