

Gap Analysis - Maintenance Document

Analysis JIRA:

KULRICE-8794 - Data cannot be retrieved due to an unexpected error

- KFS Customizations
- Permission Checks
- Involved Classes



Many of the items raised on the Inquiry analysis will apply to maintenance documents. They are both defined by similar data structures in the data dictionary and, ultimately, rendered by rowDisplay.tag.

Analysis notes: <https://docs.google.com/a/kuali.org/spreadsheet/ccc?key=0AqaSaSLMsdRMdEJDYUNCMFpIRTRvTEpSTWhyaFJEc2c#gid=2>

ID	KNS Feature	Implemented in KRAD	Notes	User Guide	KNS Sample	KRAD Sample
M1	<p>Ability to copy another business object by clicking a link from the lookup. Copies all the attributes except for the primary key, ver number, and object ID fields by default.</p> <p>Different implementation will be needed, since object ID and version number are no longer required.</p> <p>(Why can't we do this from a link on the inquiry page? - Would be a new feature request.)</p>			N/A	Campus	Context








M2	Upon copy, blanks out any fields determined to be "restricted" based on input from the document authorizer.	 KULRICE-10317	Restricted is defined as hidden, read-only or masked. The functionality seems confusing. When a field is set as hidden, the entire row is hidden so that the user can't set the field for the copy. This is also the case for ready-only. With read-only, the copy field is also read-only. Upon submitting the value of the read-only field will always be blank. Is this as designed?	N/A	KFS Invoice Recurrence	Travel Account
M3	Ability to delete a business object. (Though usually no link is displayed to perform this operation.)	 KULRICE-10318	Link shows up but goes to INITIATED maintenance document instead of deleting	N/A		Travel Company
M4	Ability to edit an existing business object.			N/A	Country	Travel Account
M5	Ability to pass in properties on the HTTP request to pre-populate fields on the document. If overrideKeys=field1,field2 not specified, then only parameters with the names of the primary key fields will be used.		Unable to find a KNS example.			Travel Company New - defaulted values
M6	Calls into the Maintainable to populate any blank fields which have defaults after a New or Copy document is created.	 KULRICE-10362		N/A	Country	Travel Account
M7	Automatically displays the left "Old" side when editing or copying a business object.	 KULRICE-10344		N/A	Parameter Edit	Travel Company Edit

M8	In any document where the Old/New sections are both displayed, highlight (yellow asterisk in the KNS) fields which have been changed from their "old" values.			N/A	Parameter Edit	Travel Company Edit
M9	Ability to block the creation of new objects of a given type at the data dictionary level. (allowsNewOrCopy)			N/A	KFS Asset	Labs Travel Company
M10	Knowledge of the fields and collections present in sections (tabs) so that error messages will be displayed under the correct heading.			MEH1	Country	Travel Account
M11	Auto-opening of tabs which have errors when redisplaying the document after a form submit which performs validation.			N/A	Country	Travel Account
M12		 KULRICE-10332		N/A		



Security mechanism which required the registration of all properties displayed on the form. The standard form control tags pushed their field names into a session scope variable. Upon form re-posting, only attributes registered in this way would be posted back to the business object. This was done due to the way that Struts (and now Spring) auto-bind values based on dot-notation. Without this, it is possible to create custom form widgets in the browser and change internal properties of business objects which should have been non-editable by the end user.




(A call like: `$(kfunc:registerEditableProperty(KualiForm, field.propertyName))` is used to perform this function from `rowDisplay.tag`)

	<p>The KualimaintenanceForm also registers a number of properties which are not editable, but which will be present on the form. These are used to prevent the security system from blowing the page for these standard fields. That map is also used to prevent the values on the form from being updated in the objects upon posting of the form.</p>					
M13	<p>Rendering of the "*" = required field" heading when the document was in an editable format. (Probably not needed due to dynamic feedback in KRAD.)</p>	<p> KULRICE-10345 UXI Prioritized</p>		N/A	Country	Travel Account
M14	<p>Display of maintenance document sections in collapsible tabs.</p>	<p></p>		N/A	Country	Travel Account
M15	<p>Ability to expand/collapse tabs one at a time or all at once.</p>	<p> KULRICE-10348</p>		N/A	Country	Travel Account
M16	<p>Memory of the state of the tabs on a given document across page refreshes.</p>	<p> KULRICE-10349</p>		N/A	Country	Travel Account
M17	<p>Ability to specify the default open/closed state of the document-specific tabs.</p>	<p></p>	<p>KNS: defaultOpen="true" KRAD: disclosure.defaultOpen="true"</p>	MS2	KC COI Disclosure Event Type (see Project Details)	Travel Account (see Notes)
M18	<p>Ability to include additional non-standard sections via a DD-specified file. The file would be included via the <jsp:include> tag.</p>	<p></p>	<p>Although we cannot include JSP files directly in KRAD, we can split views across multiple files.</p>	MOF1		

M19	Ability to configure help links in the section headers via the DD.		KNS: helpUrl propert KRAD: help bean	MS3	N/A	Travel Company
M20	Ability to hide sections conditionally via the presentation controller.		KNS: getConditionallyHiddenSectionIds of MaintenanceDocumentPresentationControllerBase KRAD: canViewGroup of MaintenanceViewPresentationControllerBase	MDPC1	KFS Asset Global	Travel Account
M21	Ability to make entire sections conditionally read-only via the presentation controller.		KNS: getConditionallyReadOnlySectionIds of MaintenanceDocumentPresentationControllerBase KRAD: canEditGroup of MaintenanceViewPresentationControllerBase	MDPC1	KFS Vendor	Travel Account
M22	Ability to include additional JS files in the page via DD configuration.		KNS: webScriptFiles KRAD: additionalScriptFiles * make sure that script still works/makes sense	MBB4	N/A	Travel Account
M23	 documentOverview.tag/rowDisplay.tag : Detection of whether the screen is a maintenance document by checking if the Struts Form class == KualiMaintenanceForm.		<#if KualiForm.viewTypeName != 'MAINTENANCE' > (ftl) This is automatic for any view that inherits from UifMaintenanceDocument	N/A		
M24	Ability to set the title of the tab sections via the DD.		KNS: title KRAD: headerText	MS1	Country	Travel Account






M25	Ability to set an ID for sections for use by the presentation controller and document authorizer when setting sections as hidden or read only. If no ID is specified, the ID defaults to the tab's title. (This is not the HTML id used in the generated code.)			MDPC2	KFS Asset Global	Travel Account
M26	Support for custom buttons on maintenance documents which only display on the "New" side and not when in read-only mode. They call back into the "refresh" method but with a custom action in one of the special URL-encoded fields. The custom action is manually inserted into the request parameter map passed to the Maintainable's processAfterPost () method.	 KULRICE-10432 UXI Prioritized	Still need to add support. Currently the best solution would still render the button on the old side as well.	 KULRICE-10432 documentation needed		
M27	Use of the control definitions in the business object DD files to render the fields.			N/A	KFS Account Type	Travel Account





M28	<p>Recognizes properties which are sensitive when passed in for the "newWithExisting" action and assumes they are encrypted, decrypting them before passing them into the new business object. (Please don't support - we should never be passing sensitive information on URLs. If there is a need to do this, have it place the sensitive data in Session scope and have the passed value be some sort of GUID to retrieve it.) (There is also use of a special prefix in places to indicate that a value is encrypted: EncryptionService. ENCRYPTION_POST_PREFIX)</p>		<p>KULRICE-10430 (not part of the equivalency) will address the enhancement for removing this feature.</p>	N/A		
M29	<p>Support for retrieving attachments which have been linked to properties on the business object itself. (I presume they have been linked to BLOB fields.) There also seems to be support for attachment objects linked to collections within the BO. (See PersistableAttachment and PersistableAttachment List)</p>	 KULRICE-10455		MA		<p>Sample Attachment</p>





M30	<p>Altering of behavior of the main Action when the generic docHandler method is used depending on the "command" passed (displayActionListView, displayDocSearchView, displaySuperUserView, displayHelpDeskActionListView, initiate)</p>	 KULRICE-10463		N/A		
M31	<p>Special handling to support multiple value lookups and mapping of the properties from the business object being looked up into the child records actually referenced by the "global" document. This is handled via the fairly overloaded "refresh" method. MV lookups only return the OBJ_ID values, so the records need to be retrieved before the properties can be mapped.</p> <p> How will this work in KRAD if the object being looked up does not implement the "GloballyUnique" interface?</p>			N/A		
M32	<p>Support for collections within business objects, with base methods for handling the addition and deletion of lines from the collection.</p>			N/A		Screenshots

M33	<p>Special handling of the Struts populate command on the form to strip off the new/oldMaintainable prefix and push the resulting properties directly into the business objects. (Since the base function would to have been document.newMaintainableObject.boProperty.) This population is deferred to the maintainable's populateBusinessObject() method rather than using the Struts functions.</p> <p>Now, the existing code applies the values to the "oldMaintainable Object" as well. This probably should not have been happening, as the old object should never be changed by UI code.</p>			N/A		
M34	<p>All incoming properties are handled as String objects. Data conversion errors are handled by storing the unconverted values in a "cachedValues" Map. This map is then used when re-displaying the page so that the data the user entered is not lost even though it may not be able to be stored in the data type on the business object. (E.g., letters in a KualDecimal property.</p>	 KULRICE-10454 UXI Prioritized	Needs cleanup/improvements.	N/A		

M35	For lookups and inquiry buttons/links: Automatic passing of keys from related fields when the field is the last field in a multi-field foreign key. (E.g., retrieval and inclusion of the chartOfAccounts Code when performing a lookup on the accountNumber field.)		Lookups and inquiries are automatically hooked up if a relation ship exist (via metadata service)	N/A	KNS Chart Lookup	Travel SubAccount
M36	Automatic setting of the primary key fields on the BO to read-only when the maintenance document is in edit mode (FieldUtils).	 KULRICE-10452 UXI Prioritized		N/A		
M37	Special case of above: Automatic setting of the principal name property to read only when the related principal ID field is part of the primary key.	 KULRICE-10452 UXI Prioritized		N/A		
M38	Automatic handling of field conversion values for building the quickfinders and direct inquiry buttons on attributes. Adds the necessary "document.newMaintainableObject ." prefix.			N/A		Screenshots
M39	In maintenance document authorization checks, automatic stripping of the document/maintainable prefixes on property names to allow the permission definitions to only use the business object property name.		Invalid — prefix stripping does not happen during permission checks for things like Unmask Field (KNS and KRAD).	N/A		



M40	Automatic recognition of masked fields as sensitive. If so, it marks them as needing encryption when sent to the client and sets up their masked value for display.			N/A		Screenshots
M41	Support for specification of additional "onblur" functionality in the DD. (Assumes that any called functions exist in the standard libraries or are included in the additional script files also in the DD for the maintenance document.		KNS: webUILeaveFieldFunction KRAD: onBlurScript	MJS		Screenshots
M42	Ability to define a help URL on a per-field basis. (This should probably be retired in favor of KRAD popup help.)		KNS: System parameter ENABLE_FIELD_LEVEL_HELP_I ND enables a generic help on all fields. KRAD: Uif-Help (help property) allows custom help on each individual field.	MHF		Screenshots
M43	Ability to indicate that any property which is the code on a relationship for a BO class which implements QualiCode should display itself as a combination of Code + Name. (Used when document/field is in read-only mode.)		Enabled in View via translateCodesOnReadOnlyDisplay. Can also use readOnlyDisplaySuffixPropertyName to manually specify this on DataFields.	MADV		Screenshots
M44	Automatically clears out secure fields upon a BO copy operation.	 KULRICE-10317	Field is masked in copy, verified that result in database is null. DB Result Screenshot	N/A	KFS Main Menu ACH Bank Lookup as user ahrens (click Search and notice routing number)	Maintenance Sample - Clear secure fields on BO Copy Screenshot

M45	If a collection class under the main object implements the Inactivatable interface, then display, as part of the tab containing the collection) a show/hide inactive button.	 KULRICE-10641 UXI Prioritized		N/A		
M46	Skipped generating a required field error if the field was also marked as unconditionally read only. The assumption is that the field would be filled in by server-side code in these cases and not to inform the user since they could not do anything about it anyway.			N/A		
M47	Stores contents of all maintenance documents in a single table. (KRNS_MAINT_DOC_T) It is stored as an XStream-serialized representation of the document's maintainables. There are a number of custom tags used to wrap the business objects. See MaintenanceDocumentBase.populateMaintainablesFromXmlDocumentContents()			N/A		
M48	Ability for an application to use a custom base document class for maintenance documents.		Custom base documents (i.e. FinancialSystemMaintenanceDocument) that overrides the KNS MaintenanceDocumentBase class must be manually converted to a KRAD base class	MBC		

	Maintainables					
M49	Ability to specify the Maintainable class for a given document in the DD.		No changes (still done via maintainableClasses property)	MBB2		
M50	Upon populating the business object (called from the action), manually performed the resolution of principal name to principal ID for all registered Person relationships.	 KULRICE-10457				
M51	Responsible for creating the UI components from the DD information and business object data.			N/A		
M52	<p>Provided customization hooks for documents for the following operations:</p> <ul style="list-style-type: none"> • Creating New BO • Editing Existing BO • Copying Existing BO • Any form POST operation • Before adding a collection line • After adding a collection line 		Custom logic needs to move to new methods.	MCH		







M53	Ability to override any default values.		Maintainable.setGenerateDefaultValues has been replaced by ViewHelperServiceImpl.applyDefaultValues which is called during the performFinalize phase before the view is displayed.	MODV		
M54	Maintained special containers for the "add" lines for collections so that the data models did not have to have "fake/non-persistent" add line attributes.		UifFormBase.newCollectionLine stores data for the add line. A custom location can be specified via addLinePropertyName and addLineBindingInfo.	MCC		
M55	Handler for and recognition of references (related objects) which could have been affected by a user's returning from a lookup with a new value.		Custom logic needs to move to new methods.	MRR		
M56	Performed for force uppercase function for the document.		Same. Can also use @ForceUppercase annotation.	DDAD5		
M57	Hook to prepare the business object as part of the document setup.		Convert the custom Maintainable.prepareBusinessObject code to use Maintainable.prepareExternalBusinessObject, though other options should be considered.	MPBO		
M58	Generation of the strings which would be stored to the KRNS_MAINT_LOCK_T table to prevent the editing of the same record on multiple documents.			N/A		






M59	Hook to allow access to and manipulation of the KNS Section objects to make tweaks. The KFS OrgReviewRole used this function to, per data available only after the user entered it on the document, to hide or mark fields as read only.		Convert the custom readOnly/hidden logic to use Uif's readOnly/hidden properties with expression.	MSC		
M60	Ability to specify (via method override) that the related document was editing an "Externalizable Business Object" - this essentially would shortcut/skip some of the behaviors which only worked when the object was a normal PersistableBusinessObject. (Only real purpose seems to be to avoid throwing errors if the object could not be loaded from the database.)		Ideally - this should not be needed after the data module conversion, as the data providers would take care of this.  Verify this.	MOF2		
M61	Hook to prepare the "old" business object after loading from the database. For non-database objects, this would be used to populate the object from whatever source was needed.		Convert the custom Maintainable.prepareBusinessObject code to use Maintainable.prepareExternalBusinessObject for BusinessObjects or Maintainable.retrieveObjectForEditOrCopy for non BusinessObjects	MPEBO		
	"Global" Maintenance Documents					

M62	Supports a mass change form of maintenance document termed a "Global". These extend the base maintainable framework by using a business object which implements a special marker interface. This object is really a document-based class complete with document number and containing one or more lists of GlobalBusinessObjectDetail objects. This and the special maintainable (KualiGlobalMaintainableImpl) result in the special behavior.	 KULRICE-10667	(Should probably choose some other term...like MassChange or BulkUpdate)	MBU		
M63	Global documents provide a hook (though it must be implemented from scratch by any implementors) to add locks to the maintenance lock table for each object being edited.	 KULRICE-10667	<p>This is a point for improvement. We should have a default implementation which, since it should know what the detail object is, to call into the maintainable for that object and generate the individual locks.</p> <p>This would however require a new API method to generate a single lock for a given business object. Can be culled from org.kuali.rice.krad.maintenance.MaintainableImpl.generateMaintenanceLocks()</p>	MBU		

M64	Calls into the GlobalBusinessObject (the document) to build the list of maintained business objects to save or delete.	 KULRICE-10667	<p>We should have a more intelligent base implementation. At the moment, everything needs to be done custom each time. It seems that it would be easy to add a method to retrieve all the contained GlobalBusinessObjectDetail objects and then ask each of those for their business object to save.</p> <p>Given that there are some hard-requirements for the implementation of the GlobalBusinessObject, we should provide a useful base class so the simple functions can be handled via configuration rather than code.</p>	MBU		
	<p>NOTE: On the maintenance document collection used for the detail objects, you must specify the global detail class as the business object and the BO class being modified as the source class. Since any instance of the global detail object needs to have that class information, this is redundant and could be removed.</p>					
	Business Rules					









M65	Provides a base method to allow customization of the setup of convenience objects (and anything else which will be used by the various rule methods.) The base implementation just extracts them from the old and new maintainable objects		Custom logic needs to move to new methods.	MRC		
M66	The base rule class provides explicit hooks for adding functionality to the default validation operations shared by all maintenance documents. (The processCustomXxxx methods) The base has hooks for the following operations: <ul style="list-style-type: none"> • save • route • approve • add collection line 	 KULRICE-10655	Custom logic needs to move to new methods.	 documentation needed		
M67	Performs a set of global (base) validations for the document as a whole. These are always run and usually prevent the operation on the document if they fail. (doc description, filling out the PK)			N/A		
M68	The base classes provide special support for linking errors to the correct form field path so that the error messages appear in the correct location.			N/A		

M69	Upon document routing, performs a permission check via the document authorizer to ensure that the initiator can edit that particular business object.			N/A		
M70	If the object is being inactivated, then checks for InactivationBlocking configuration referencing this class. If a blocking record is found, it stops the document submission.		This function does not block the deletion of business objects and probably should. (It was created before the delete function was incorporated.) KULRICE-10656 created to address this.	N/A		
	DD Properties					
M71	Ability to control whether an BO can be deleted via maintenance document at the DD level. (I.e., disallow it regardless of permissions.)			N/A	KC Sponsor	Travel Account Delete (attempt)
M72	Allow for PK values to be preserved when copying a business object.			N/A	KFS OrgReviewRole Copy	
M73	Ability to control whether new BOs of a given type can be created via documents. (if not, you can only edit existing records.)		JK confirmed that this feature is a duplicate to M9	N/A	duplicate to M9	duplicate to M9
M74	Automatic detection, upon document submit, if there is already an existing document ENROUTE for the same business object.			N/A	KFS Asset Global (locked by documentNumber)	Screenshots

M75	Ability to define a set of key fields used for detecting the above locking. (E.g., when the object has a user-visible unique key which is separate from the underlying primary key.)		When not specified, the locking keys default to blank, but they probably should default to the primary key. KULRICE-10662 was created to address this.	N/A	KFS Asset Global (locked by documentNumber)	Travel Mileage Rate (locked by mileageRateCd and mileageRateName)
M76	Ability to set the maintenance document base class. (Defaults to MaintenanceDocumentBase - but KFS does override this.)			N/A		
M77	Ability to pre-process the HTTP request and manipulate the Struts form object directly rather than relying on the business object population. (org.kuali.rice.kns.web.derivedvaluesetter.DerivedValuesSetter)		KNS: DerivedValuesSetter.setDerivedValues KRAD: p:refreshWhenChangedPropertyNames="field1, field2"	MDD1		
M78	Ability to specify a class which can ask questions prior to routing the document. (org.kuali.rice.kns.rule.PromptBeforeValidation)		KNS: promptBeforeValidationClass in xml KRAD: Use KRAD's dialog framework	MDD2	KFS, Main Menu, Vendor, search for Vendor of type Purchase Order. Edit link for Basic Vendor Estate/Trust > B2B Contract Vendor. Change the Parent Indicator from No to Yes Screenshot	Lab Screenshot
M79	Ability to set a field as required (MaintainableFieldDefinition) - this was used to render the asterisk on the UI and perform DD validation when the document was submitted.			N/A		

M80	Ability to set a field as "lookup-only", meaning that it could only be populated by use of a lookup - never entered directly. This was generally done when special processing was needed after entering a value. (E.g., to load other fields.)	✓	KNS: LookupReadOnly ControlDefinition KRAD: InputField with p:widgetInputOnly="true" and quickfinder defined	MDD3	KFS: OrgReview Lookup (Document Type field)	Edit TravelExpensitem (Travel Company Name)
M81	Ability to specify a class (ValueFinder) which will obtain the default value for a field.	✓		MDD4		
M82	Ability to set a field as "unconditionally read only" so that not even permissions can make it editable. This also causes the UI components to simply not render the <input> tag for that field.	✓	KNS: p:unconditionally ReadOnly="true" KRAD: p:readOnly="true"	MDD5	KFS PurchaseOrderContractLanguage Edit (see Contract Language Create Date)	new TravelAccount (Date Created is uneditable. Set to today by default)
M83	Ability to note that a field (within a collection) should be read only after its been added. (Used to lock down collection PK fields after the user enters them.	✓	KNS: p:readOnlyAfterAdd="true" KRAD: p:readOnly="@(! #isAddLine)"	MDD5	KFS AccountGlobal Maintenance (accountNumber field after adding an account to List of Accounts)	TravelAccount Edit (Sub Account fields readOnly after added to collection)
M84	Ability to specify an alternate attribute to use for display purposes when the document is in read-only mode.	✓	KNS: p:alternateDisplayAttributeName= "MyCustomDisplay" KRAD: p:readOnlyDisplayReplacement= "MyCustomDisplay"	MDD6		This is covered in the KRAD Sample App --> Library --> Fields --> Data Field --> Replace Property with Field It can be tested by verifying that the Data Field 1 is 'My Book Title' and not 1001.

M85	Ability to specify an additional attribute for use during read-only display.		KNS: p:additionalDisplayAttributeName="myReadOnlySuffix" KRAD: p:readOnlyDisplaySuffix="myReadOnlySuffix"	MDD7		This is covered in the KRAD Sample App --> Library --> Fields --> Data Field --> Append Property with Field It can be tested by verifying that the Data Field 1 is '1001 - My Book Title' and not 1001.
M86	On maintainable collections, to support multiple value lookup, you can specify what the source class is that you are looking up. (i.e., on the account global document, the detail object is AccountGlobalDetail - part of the document - but the source object is Account - the business object which gets referenced by that document class)			MMVL	KFS AccountGlobal Maintenance (List of Accounts section)	This is covered in the KRAD Sample App --> Library --> Collection Features --> Multi Value Lookup You should test the lookup/add multiple lines link.
M87	Ability to control the existence of the "add" line via the collection definition. (i.e. make sure that the add line doesn't show on the submission confirmation screen)			MDD8		This is covered in the KRAD Sample App --> Library --> Containers --> Collection Group --> Table Layout Readonly (dropdown) It can be tested by verifying that the add line is not shown. You can find a sample addline the the KRAD Sample App --> Containers --> Collection Features --> Addline example.
M88	Ability to control whether to have a multiple value lookup.			MMVL		

M89	Ability to specify whether items can be deleted from the collection. (Default is to only allow deletion of newly added lines.)	 KULRICE-10684	In KRAD defaults to delete any line but only newly added lines can be specified.	MDD9	KC: Organization Maintenance allow all deletion: (see Org Type) only add lines: see(Audit collection)	all (default): TermSpecification (see Context collection) only add lines: Travel Account (see sub account)
M90	Ability to specify which field (singular) should be highlighted if there is a key violation between collection records.	 KULRICE-10684 UXI Prioritized		 documentation needed		
M91	Ability to set a title for the summary line that appears at the top of each collection "row" (Applies when using the KNS-style old/new vertical alignment for collection rows.)	 KULRICE-10685 UXI Prioritized		 documentation needed		
M92	Ability to set fields which will be included after the summary title. (Applies when using the KNS-style old/new vertical alignment for collection rows.)	 KULRICE-10685 UXI Prioritized				
M93	Ability to specify the list of fields which form the unique key within a collection.	 KULRICE-10710		MC4	KFS Account Global	Travel Account
M94	List data object attributes which should be reference checked for existence. These will be checked that they exist in the database and are active. You can (must?) also indicate which field you want the framework to highlight if the validation fails.	 Bugs - KULRICE-12849 KULRICE-12850		N/A	KNS Postal Code Maintenance Country, State, and County Screenshot	Existence Checks Demo Page



M95	Ability to configure workflow attribute information for the document, both for searching and for workflow node processing. The DataDictionaryQualifierResolver handles the workflow attributes.			N/A		
-----	---	---	--	-----	--	--





KFS Customizations

These items are out of scope for Rice, but information for KFS customizations.

KFS/Rice Override Function	Types of Changes
org.kuali.rice.kns.maintenance.KualiMaintainableImpl.getCoreSections()	Hacking of the field conversions for a specific field. Postal code in the case I'm looking at. It seems to be an attempt to make it populate the city and state name upon return from the lookup.
org.kuali.rice.kns.maintenance.KualiMaintainableImpl.refresh()	Performing additional field derivations after the superclass actions.
org.kuali.rice.kns.maintenance.KualiMaintainableImpl.processAfterNew()	Building additional data structures (business objects) for use by the document.
org.kuali.rice.kns.maintenance.KualiMaintainableImpl.processAfterCopy()	Altering/blanking of additional key fields on the new (copied) object.
org.kuali.rice.kns.maintenance.KualiMaintainableImpl.getSections()	Programatic removal or manipulation of fields defined in the maintenance document XML.
org.kuali.rice.kns.maintenance.KualiMaintainableImpl.saveBusinessObject()	Additional persistence actions - usually follow-up to the actual persistence.
org.kuali.rice.kns.maintenance.KualiMaintainableImpl.generateMaintenanceLocks()	Customization of lock text / automatic registration of other affected business objects
org.kuali.rice.kns.maintenance.KualiMaintainableImpl.getDocumentTitle()	Allow customization of how the document title appears in document search.

Permission Checks

ID	KIM Permission	Check Description	Location	Supported in KRAD	User Guide
M96	KR-NS / Full Unmask Field	Checks if a security-masked field can be displayed in the clear	BusinessObjectAuthorizationServiceImpl.canFullyUnmaskField		MPC1
M97	KR-NS / Partial Unmask Field	Checks if a security-masked field can be partially displayed	BusinessObjectAuthorizationServiceImpl.canPartiallyUnmaskField		MPC2

M98	KR-NS / Create / Maintain Record(s)	Allows the control of the editing or creation of individual records via permissions. It's sometimes used to control access to specific records based on role qualifiers, though that is not directly supported. (subclasses add the PK values of the object being edited)	canCreate canMaintain canCreateOrMaintain		N/A
M99	KR-NS / Modify Maintenance Document Section	Uses the DD-provided section ID to check this permission. This permission is keyed by the Business Object and section ID being edited. This must be the "master" business object being edited by the document. (Not a child object also being edited by the document.)	BusinessObjectAuthorizationServiceImpl. considerMaintenanceDocumentAuthorizer		MPC3
M100	KR-NS / View Inquiry or Maintenance Document Section	Uses the DD-provided section ID to check this permission. This permission is keyed by the Business Object and section ID being edited. This must be the "master" business object being edited by the document. (Not a child object also being edited by the document.)	BusinessObjectAuthorizationServiceImpl. considerInquiryOrMaintenanceDocumentAuthorizer		MPC4
M101	KR-NS / Modify Maintenance Document Field	This permission is keyed by the Business Object and property name being edited. This must be the "master" business object being edited by the document. (Not a child object also being edited by the document.)	BusinessObjectAuthorizationServiceImpl. considerBusinessObjectFieldModifyAuthorization		MPC5

M102	KR-NS / View Inquiry or Maintenance Document Field	This permission is keyed by the Business Object and property name being edited. This must be the "master" business object being edited by the document. (Not a child object also being edited by the document.)	BusinessObjectAuthorizationServiceImpl.considerBusinessObjectFieldViewAuthorization	✓	MPC6
M103	KR-NS / Edit Document	A (usually) automatically determined permission to ensure that the proper users can edit the documents at the right times. When this is false, the document should generally be completely non-editable. Not Maint Doc specific: Only here to ensure that the interaction between Edit Document and the Modify Field permissions is preserved.		✓	N/A
M104	KR-NS / Open Document		DocumentAuthorizerBase.canOpen	✓	N/A
M105	KR-NS / Perform Custom Maintenance Document Function	When a custom button is supposed to be present on a maintenance document, this permission is checked for the id of the button. (property name) If the user does not have the permission, the button is not shown.	BusinessObjectAuthorizationServiceImpl.considerCustomButtonFieldAuthorization	✓	MPC7



Delete permission check is insufficient. Just because a user can maintain an object, doesn't mean they should be able to delete it. A new permission template should be introduced for business object deletion.




Additionally, any record deletion should be within a try/catch which handles any FK violations and reports back by disapproving the document rather than throwing to exception.

Involved Classes

Class	Notes	Finished Analysis
Struts Classes		

org.kuali.rice.kns.web.struts.action.KualiMaintenanceDocumentAction		✓
org.kuali.rice.kns.web.struts.action.KualiDocumentActionBase		✓
org.kuali.rice.kns.web.struts.form.KualiMaintenanceForm		✓
org.kuali.rice.kns.web.struts.form.KualiDocumentFormBase		✓
org.kuali.rice.kns.web.struts.form.pojo.PojoFormBase		✓
General Classes		
org.kuali.rice.kns.document.MaintenanceDocumentBase	Mostly support for the PersistableAttachment framework. Left in the KNS because its objects use PersistableBusinessObject. Needs to be converted to KRAD/JPA.	✓
org.kuali.rice.krad.maintenance.MaintenanceDocumentBase	This is most of the KNS function and is working as-is.	✓
org.kuali.rice.kns.rules.MaintenanceDocumentRule		✓
org.kuali.rice.kns.maintenance.rules.MaintenanceDocumentRuleBase		✓
org.kuali.rice.kns.rule.PromptBeforeValidation		✓
org.kuali.rice.kns.web.derivedvaluesetter.DerivedValuesSetter		✓
org.kuali.rice.kns.util.KNSGlobalVariables	Support for a MessageList component which could add messages to the top of the KNS screens after form submission. (E.g., Document is successfully routed)	✓
org.kuali.rice.kns.util.FieldUtils		✓
org.kuali.rice.kns.web.ui.FieldBridge	Converts the field metadata into UI objects used by rowDisplay.tag	✓
org.kuali.rice.kns.web.ui.SectionBridge	Converts the section metadata into UI objects used by rowDisplay.tag	✓
org.kuali.rice.kns.util.MaintenanceUtils		✓
JSP/Tag Files		
/rice-web/src/main/webapp/kr/WEB-INF/jsp/KualiMaintenanceDocument.jsp		✓
/rice-web/src/main/webapp/WEB-INF/tags/kr/page.tag		✓
/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		✓
/rice-web/src/main/webapp/WEB-INF/tags/kr/rowDisplay.tag	Handles the actual display of the data cells in the document.	✓
Maintainable		

org.kuali.rice.kns.maintenance.KualiGlobalMaintainableImpl		
org.kuali.rice.krad.bo.GlobalBusinessObject		
org.kuali.rice.krad.bo.GlobalBusinessObjectDetail		
org.kuali.rice.krad.bo.GlobalBusinessObjectDetailBase		
org.kuali.rice.kns.maintenance.KualiMaintainableImpl		
org.kuali.rice.krad.maintenance.MaintainableImpl		
Data Dictionary Classes		
org.kuali.rice.kns.service.impl.MaintenanceDocumentDictionaryServiceImpl	<p>Support service for accessing the DD. Mainly proxied all calls, handling nulls where needed. Also handled calling the ValueFinder instances for obtaining default values for fields.</p> <ul style="list-style-type: none"> Performed the required field validation for maint docs and the existence check for person fields. (This latter seems to assume that the Person object had already been populated by upstream code. Checks for duplicates in collections. 	
org.kuali.rice.kns.datadictionary.MaintainableSectionDefinition		
org.kuali.rice.kns.datadictionary.MaintainableSubSectionHeaderDefinition		
org.kuali.rice.kns.datadictionary.MaintainableItemDefinition	Dummy superclass which just provided a name.	
org.kuali.rice.kns.datadictionary.MaintainableFieldDefinition		
org.kuali.rice.kns.datadictionary.MaintainableCollectionDefinition		
org.kuali.rice.kns.datadictionary.validation.MaintenanceDocumentAttributeValueReader	Legacy support class.	
org.kuali.rice.kns.datadictionary.MaintenanceDocumentEntry		
org.kuali.rice.kns.datadictionary.KNSDocumentEntry		
org.kuali.rice.krad.datadictionary.MaintenanceDocumentEntry		
Presentation Controller and Authorization		
org.kuali.rice.krad.document.DocumentPresentationControllerBase		
org.kuali.rice.kns.document.authorization.MaintenanceDocumentPresentationControllerBase		
org.kuali.rice.kns.document.authorization.MaintenanceDocumentAuthorizerBase		

org.kuali.rice.kns.document.authorization.MaintenanceDocumentRestrictionsBase	DTO class which contains the results of all presentation controller and authorizer checks on document sections and fields. Used by the KNS UI classes to alter the contents of the document.	
org.kuali.rice.kns.document.authorization.InquiryOrMaintenanceDocumentRestrictionsBase		
org.kuali.rice.kns.bo.authorization.BusinessObjectAuthorizerBase		
org.kuali.rice.kns.service.impl.BusinessObjectAuthorizationServiceImpl	Handles the building of the "restrictions" objects by calling the methods on the document authorizer.	