

Workflow Document Type 2

To define a workflow document type, we create an XML document and then ingest that document into the Workflow Ingestor. [Excellent documentation of the precise details of the XML document exists](#), but we can still look at highlights from a specific KEW document type. Because it is so basic, let's take a look at the Internal Billing KEW document type.

```
<?xml version="1.0" encoding="UTF-8" ?>
<data xmlns="ns:workflow" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="ns:workflow resource:WorkflowData">
  <documentTypes xmlns="ns:workflow/DocumentType"
xsi:schemaLocation="ns:workflow/DocumentType resource:DocumentType">
    <documentType>
      <name>InternalBillingDocument</name>
      <parent>KualiFinancialDocument</parent>
      <label>Internal Billing</label>

<postProcessorName>org.kuali.workflow.postprocessor.KualiPostProcessor</postProcessorName>

  <superUserWorkgroupName>KUALI_ROLE_SUPERVISOR</superUserWorkgroupName>

<blanketApproveWorkgroupName>KUALI_ROLE_ADMINISTRATOR</blanketApproveWorkgroupName>

<defaultExceptionWorkgroupName>KUALI_ROLE_SUPERVISOR</defaultExceptionWorkgroupName>

<docHandler>${application.url}/financialInternalBilling.do?methodToCall=docHandler</docHandler>
  <active>true</active>
  <routingVersion>1</routingVersion>
  <routePaths>
    <routePath>
      <start name="Adhoc Routing" nextNode="Account Review" />
      <requests name="Account Review" nextNode="Org Review" />
      <requests name="Org Review" nextNode="Subfund" />
      <requests name="Subfund" />
    </routePath>
  </routePaths>
  <routeNodes>
    <start name="Adhoc Routing">
      <activationType>S</activationType>
      <mandatoryRoute>false</mandatoryRoute>
      <finalApproval>false</finalApproval>
    </start>
    <requests name="Account Review">
      <activationType>S</activationType>
      <ruleTemplate>KualiAccountTemplate</ruleTemplate>
      <mandatoryRoute>false</mandatoryRoute>
      <finalApproval>false</finalApproval>
    </requests>
    <requests name="Org Review">
      <activationType>S</activationType>
      <ruleTemplate>KualiOrgReviewTemplate</ruleTemplate>
      <mandatoryRoute>false</mandatoryRoute>
      <finalApproval>false</finalApproval>
    </requests>
    <requests name="Subfund">
      <activationType>S</activationType>
      <ruleTemplate>KualiSubFundTargetTemplate</ruleTemplate>
    </requests>
  </routeNodes>
</docHandler>
</documentTypes>
</data>
```

```
    <mandatoryRoute>false</mandatoryRoute>
    <finalApproval>false</finalApproval>
  </requests>
</routeNodes>
</documentType>
```

```
</documentTypes>
</data>
```

Here we can see three major divisions: the basic information for the document type, the route path, and the route nodes. Two more advanced sections are not used by the Internal Billing document, policies and search attributes; we'll cover those separately.

KEW Document Type Basics

The basic section contains the name of the document type. This name must be a unique identifier among all workflow document types. Ingesting two document types with the same document type name means that KEW will make the second loaded definition the canonical document type definition for both documents, which could cause some confusion. We'll cover naming conventions below.

Document types, of course, can inherit certain attributes from other document types, and therefore, many document types declare a parent document type that they inherit from, in this example from `KualiFinancialDocument`, which acts as the parent document type of practically all the KFS financial processing documents. The basic section also includes the label that KEW should show for the document type to users. Then the document type then defines the post processor class used, the super user, blanket approve, and default exception handling workgroups. Those values tend to be pretty standard but, of course, can be customized.

Then we get the url for the "docHandler" - that is, what page should be loaded to display the document at hand. For most of the KFS transactional docs, determining that value is pretty easy, as most transactional documents have Struts pages defined for each. For maintenance documents, the docHandler tag should always be:

```
<docHandler>${application.url}/kr/maintenance.do?methodToCall=docHandler</docHandler>
```

Finally, we have tags to set whether the document type is actually active in KEW currently and the version of the routing path.

Naming conventions for document types

To avoid uniqueness issues with document type names and for some consistency throughout the system, there are a couple guidelines to follow:

1. Transactional document type names should be in the form of `{document name}Document`. Maintenance document type names should be in the form of `{business object to maintain}MaintenanceDocument`. Global maintenance document type names should be in the form of `{business object to maintain}GlobalMaintenanceDocument`.
2. "Kuali" should not precede any document type name. Therefore, the name "InternalBillingDocument" follows the convention, but the name "KualiInternalBillingDocument" does not.
3. Document type names should be in camel case and should not have any spaces within them.

These conventions assure well-defined document type names throughout the system.

Routing & Searching On Document Types



For details on creating the rule templates and search attributes that are referenced in the XML below, see the [rule and search attributes documentation](#).

Routing

As we saw in the `InternalBillingDocument` document type, defining the workflow routing for a document type includes two parts: defining the path of the document from node to node and defining each of the nodes themselves.

The `InternalBillingDocument` document type above configures a route path, made out of route nodes, and then it proceeds to define each node. Many of the node definitions are standard: most every document goes through Ad Hoc routing and Organization Review. Financial documents tend to go through Account Review as well. If those were the only routes a document was going through, its routing definition would look like so:

```

<routePaths>
  <routePath>
    <start name="Adhoc Routing" nextNode="Account Review" />
    <requests name="Account Review" nextNode="Org Review" />
    <requests name="Org Review" />
  </routePath>
</routePaths>

```

Here, we see that there is one routePath defined, which starts at Adhoc routing (so defined because it uses the "start" tag instead of the "requests" tag), and then continues to Account Review and Org Review. The Org Review request has no "next" attribute, and therefore, KEW stops it there. Then the route nodes just named are defined:

```

<routeNodes>
  <start name="Adhoc Routing">
    <activationType>S</activationType>
    <mandatoryRoute>>false</mandatoryRoute>
    <finalApproval>>false</finalApproval>
  </start>
  <requests name="Account Review">
    <activationType>S</activationType>
    <ruleTemplate>KualiAccountTemplate</ruleTemplate>
    <mandatoryRoute>>false</mandatoryRoute>
    <finalApproval>>false</finalApproval>
  </requests>
  <requests name="Org Review">
    <activationType>S</activationType>
    <ruleTemplate>KualiOrgReviewTemplate</ruleTemplate>
    <mandatoryRoute>>false</mandatoryRoute>
    <finalApproval>>false</finalApproval>
  </requests>
</routeNodes>

```

Here we note that once again, the "Adhoc Routing" node is defined with a "start" tag, while all subsequent nodes are defined with "requests" tags. The name for each node is the same as in the path tags above. The activation type can either be "S" for serial, one at a time nodes, or "P" for parallel nodes. Non Adhoc Routing requires a KEW rule template to be named for the node. Again, KualiAccountTemplate for Account Review and KualiOrgReviewTemplate for Organization Review are predefined, and KFS has several other useful rule templates for routing, include templates that route the document to an award workgroup, for verification of Contracts & Grants account activity, and templates that route through chart, subfund, and sub account verification (the InternalBillingDocument document type, defined above, uses the subfund group template, for instance). All rule templates available in KFS can be seen at the Rule Template screen, on the Administration tab; likewise, implementors can create their own rule templates.

This leaves us with the mandatoryRoute and finalApproval tags. If the finalApproval tag is set to true means that no approvals can come after the document was approved at the given node. Other nodes can have FYI power over the document, but no later node can ask for approval on the document. Most documents don't need this level of control, and so most of the time, that tag is set to false. The mandatoryRoute tag requires an approval request at the given node. So, for instance, let's say that for the Internal Billing document, the account review went to user ROENFRO and the organization review also went to user ROENFRO. Because the Internal Billing document has mandatoryRoute set to false for both of these nodes, user ROENFRO only needs to approve the document once for both route nodes to be satisfied. If mandatoryRoute was true, user ROENFRO would get two approval requests, one for each node, and would be required to approve the document twice. Again, most of the time, this tag is set to false.

Now, the document type has its routing. Let's examine what other definitions we can give within the KEW document template.

Searching

Most of the rule attributes we associate with document types are searching attributes, either XML Searchable Attributes or Java Searchable Attributes.

We can add as many searchable attributes as we want to a document, simply listing each name within attributes tags, like so:

```
<attributes>
  <attribute>
    <name>KualiDocumentHeaderTotalAmountRange</name>
  </attribute>
  <attribute>
    <name>KualiPayeeIdSearchableAttribute</name>
  </attribute>
  <attribute>
    <name>KualiPayeeNameSearchableAttribute</name>
  </attribute>
  <attribute>
    <name>KualiPaymentReasonSearchableAttribute</name>
  </attribute>
  <attribute>
    <name>KualiDisbursementVoucherPdpDates</name>
  </attribute>
</attributes>
```

These are (at the time of this writing, at least), the searchable rule attributes associated with the Disbursement Voucher document. This means that a DV search will show, either in the results or the search fields or both, the total amount of the document, the id and name of the payee of the voucher, the payment reason, and the extraction, cancellation, and payment dates for the document. All of these make it much easier to find Disbursement Voucher documents within the KEW document search.

Policies

The final area that KEW document types let us define are policies. Policies give special control to certain document types over how KEW handles them. For instance, for a document, we can set a policy that says that only initiators can cancel the document:

```
<policies>
  <policy>
    <name>INITIATOR_MUST_CANCEL</name>
    <value>true</value>
  </policy>
</policies>
```

[Here's a full list of the policies that KEW supports.](#) Most documents don't go to the trouble of redefining their policies, instead using KEW default behavior. But this is a further level of customization that KEW provides for document types.