CM System 7.2.0
Release Notes Guide

Copyright and Licensing Statement

Licenses and Source Code

Rhythmyx uses Mozilla's JavaScript C API. See ***http://www.mozilla.org/source.html*** for the source code. In addition, see the Mozilla Public License (***http://www.mozilla.org/source.html***).

Netscape Public License Apache Software License IBM Public License
Lesser GNU Public License

Other Copyrights

The Rhythmyx installation application was developed using InstallShield, which is a licensed and copyrighted by InstallShield Software Corporation.

The Sprinta JDBC driver is licensed and copyrighted by I-NET Software Corporation.

The Sentry Spellingchecker Engine Software Development Kit is licensed and copyrighted by Wintertree Software.

The Java™ 2 Runtime Environment is licensed and copyrighted by Sun Microsystems, Inc.
The Oracle JDBC driver is licensed and copyrighted by Oracle Corporation.

Percussion Software
600 Unicorn Park Drive
Woburn, MA 01801 U.S.A.
781.438.9900
Internet E-Mail: technical_support@percussion.com
Website: http://www.percussion.com

Contents

CM System 7.2.0 Release Notes Guide

This Guide provides implementation details for new features and bugs addressed within CM System 7.2.0. This guide should be read as a supplement to the CM System 7.2.0 Release Notes.

This guide assumes certain basic CM System knowledge provided in Percussion’s Developer Training course and can be reviewed by reading the following documentation on http://help.percussion.com/cmsystem/

- Rhythmyx Concepts Guide
- Rhythmyx Implementation Guide
- Rhythmyx Technical Reference Manual

1. Taxonomy

This feature allows developers to create, edit and manage Taxonomies and associate them to Content Type fields in the Workbench. End users can use these to tag content when creating new items in the Content Explorer. New JEXL functions allow developers to access taxonomy and taxon properties in the templates.

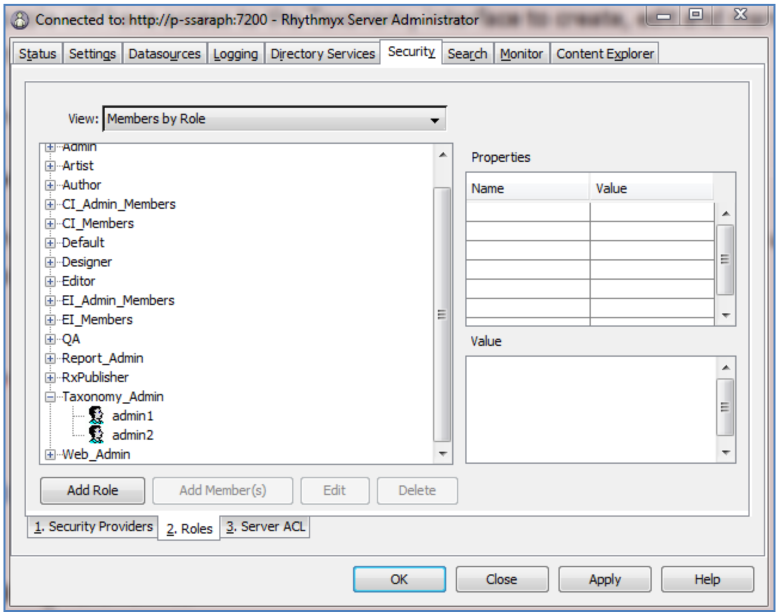
1.1 Configuring Access to Taxonomy Interface

A new Taxonomy_Admin role has been added to allow access to the taxonomy interface. Members of this role will have full access to the Taxonomy Interface to create, edit and manage taxonomies. All other users will have read only access to the taxonomy interface unless specific role based permissions are set at the taxon level by a member of the Taxonomy_Admin role.
To add users to the Taxonomy_Admin role:

1 Log in the Server Administrator

2 In the Security Tab, Select the Roles tab on the bottom.

3 Select the Taxonomy_Admin role and click on the Add Members button to add users access to the Taxonomy_Admin role.



Connected to: http://p-ssaraph:7200 - Rhythmyx Server Administrator

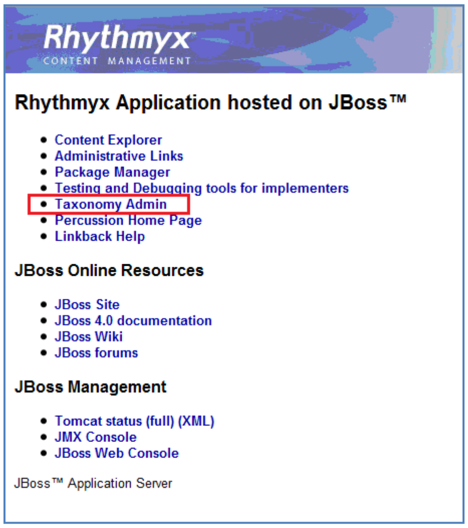
View: Members by Role

Figure 1 – Taxonomy_Admin role in Server Administrator

1.2 Taxonomy Interface in Browser

The browser based taxonomy interface allows Taxonomy_Admin members to create, edit and manage taxonomies using four pages – Index page, Create new Taxonomy page, Manage Taxonomy page and Taxon Editor page.

On the main JBoss home page, a "Taxonomy Admin" link to the taxonomy interface has been added under Rhythmyx Application hosted on JBoss.



Rhythmyx Application hosted on JBoss™

- Content Explorer
- Administrative Links
- Package Manager
- Testing and Debugging tools for implementers
- Taxonomy Admin
- Percussion Home Page
- Linkback Help

JBoss Online Resources

- JBoss Site
- JBoss 4.0 documentation
- JBoss Wiki
- JBoss forums

JBoss Management

- Tomcat status (full) (XML)
- JMX Console
- JBoss Web Console

JBoss™ Application Server

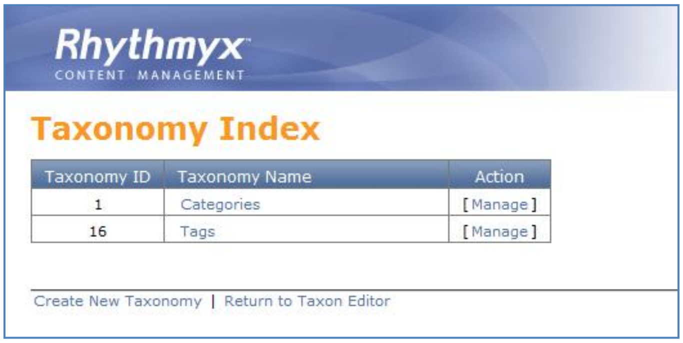
Figure 2 – Taxonomy Link on JBoss Page under Rhythmx Application hosted on JBoss

The Taxonomy Interface can be accessed directly at the following URL:

http://[CM_Server]:[Port]/Rhythmyx/taxonomy/taxonomy.html

A login will determine user access to the Taxonomy interface based on permissions set in the Server Administrator and in the taxon manager. The Taxonomy Interface opens to the home page, Taxonomy Index.

1.3 Taxonomy Index Page



Taxonomy Index

Taxonomy ID	Taxonomy Name	Action
1	Categories	[Manage]
16	Tags	[Manage]

Figure 3 – Taxonomy Index Page

The Taxonomy Index page is the taxonomy home page. It consists of a table with a list of existing taxonomies and links to the other taxonomy pages:

- **Taxon Editor** – The taxonomy names link to their respective Taxon Editor pages. The “Return to Taxon Editor” link at the bottom also links to the Taxon Editor page.

- **Manage Taxonomy** – The ‘manage’ links in the table link to the Manage Taxonomy pages for the respective Taxonomies.

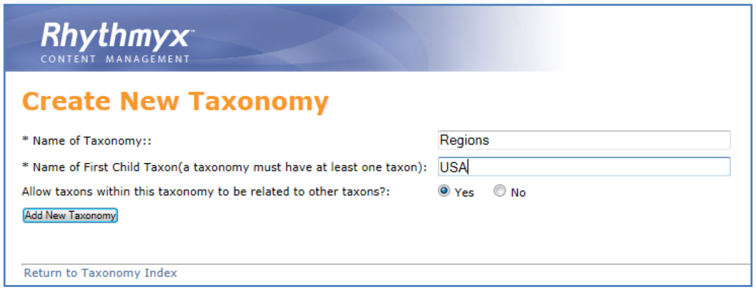
- **Create New Taxonomy** – The ‘Create New Taxonomy’ link is located at the bottom of the Index Page.

Note: For non Taxonomy_Admin roles, the ‘Manage’ links will be grayed out and the Create New Taxonomy Option will not be available.

1.4 Create New Taxonomy Page

The ‘Create New Taxonomy’ the link on the lower left corner of the Taxonomy Index page in Figure-3 links to the Create New Taxonomy page.

This page allows you to create a new taxonomy and configure some basic properties.



Create New Taxonomy

* Name of Taxonomy:: Regions

* Name of First Child Taxon(a taxonomy must have at least one taxon): USA

Allow taxons within this taxonomy to be related to other taxons?: ● Yes ○ No

Figure 4 – Create New Taxonomy

In Figure 4, you see three fields required to create a new Taxonomy:

1 Name of taxonomy – This is what your taxonomy will be called

Note: This name cannot be modified once the taxonomy is created.

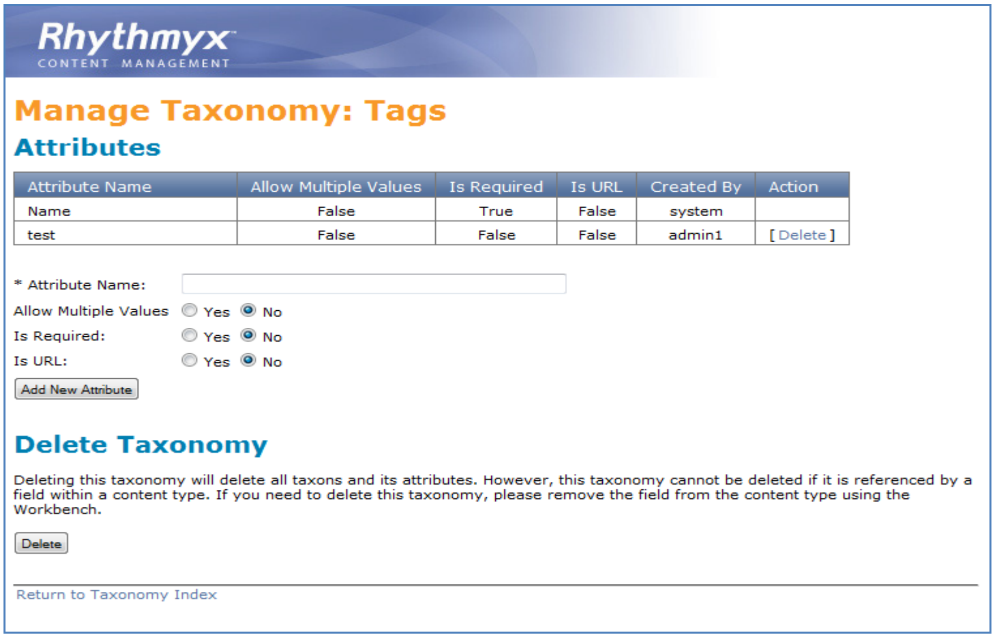
2 Name of First Child Taxon – This will be the root node in your taxonomy.

3 Allow taxons within this taxonomy to be related to other taxons – If set to “Yes,” this feature allows developers to relate a taxon to other taxons within that taxonomy. These relationships can be leveraged in templates by using JEXL functions.

Note: This property cannot be changed once the taxonomy is created. ‘Add New Taxonomy will save the new taxonomy and redirect to the corresponding Manage Taxonomy Page.

1.5 Manage Taxonomy Page

The manage links in the ‘Action’ column on the Taxonomy Index page link to the Manage Taxonomy pages of the corresponding taxonomies in the same row.
The Manage Taxonomy page allows you to add, configure and delete taxon attributes and the delete the selected taxonomy.



Manage Taxonomy: Tags

Attributes

Attribute Name	Allow Multiple Values	Is Required	Is URL	Created By	Action
Name	False	True	False	system	
test	False	False	False	admin1	[Delete]

* Attribute Name:
Allow Multiple Values ○ Yes ● No
Is Required: ○ Yes ● No
Is URL: ○ Yes ● No

Delete Taxonomy

Deleting this taxonomy will delete all taxons and its attributes. However, this taxonomy cannot be deleted if it is referenced by a field within a content type. If you need to delete this taxonomy, please remove the field from the content type using the Workbench.

Figure 5 – Manage Taxonomy Page

Creating New Attributes:
To create a new attribute fill out the 4 fields on the Manage Taxonomy Page (Figure-5):

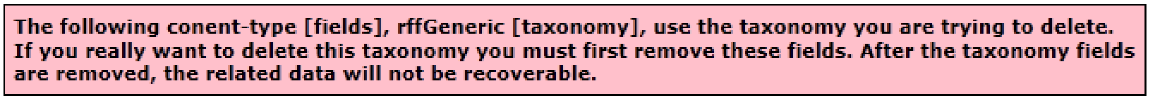
1. Attribute Name – This will be the name of your attribute.
2. Allow multiple values – If ‘Yes’ is selected, users will be able to add multiple values for that attribute.
3. Is required – If ‘Yes’ is selected, the taxons in the selected taxonomy will not save unless this attribute has been populated.
4. Is URL – If ‘Yes’ is selected, the value will be validated for valid URL strings.

Deleting Attributes:
To delete attributes, click the ‘Delete’ link for the corresponding attribute name in the Attributes table on the Manage Taxonomy Page (Figure-5).
Upon deletion, the attribute will be removed from all taxons in the taxonomy and all populated values, if any, will be lost. This action cannot be undone.

Deleting Taxonomy:

To delete a taxonomy, click the 'Delete' button on the Manage Taxonomy Page of the desired taxonomy (Figure-5).

If this taxonomy is being referenced by local or shared fields on any content types, the taxonomy cannot be deleted until this association is removed. Attempting to remove a taxonomy that is being referenced on a content type will result in the error message in Figure-6. (See Associating Taxonomies to Content Type fields in the Workbench for details).

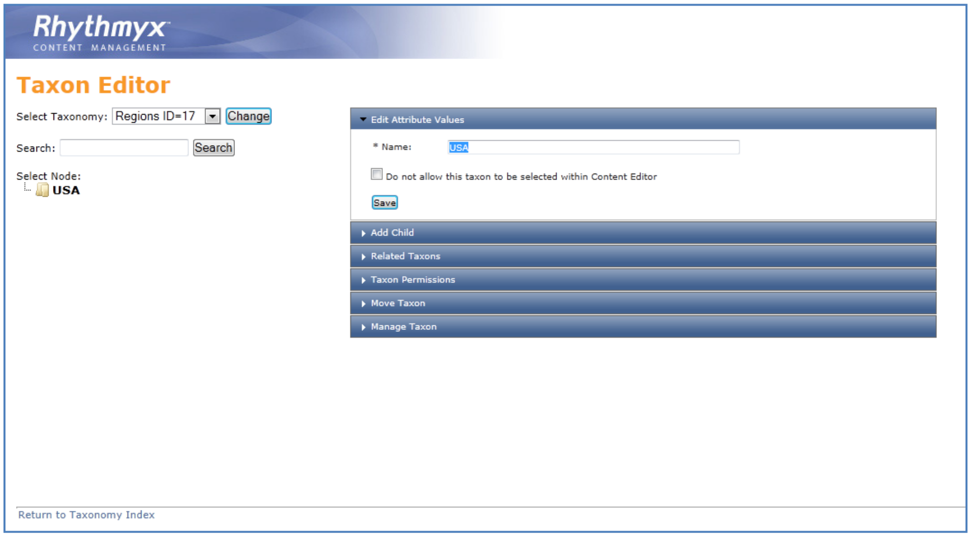


The following conent-type [fields], rffGeneric [taxonomy], use the taxonomy you are trying to delete. If you really want to delete this taxonomy you must first remove these fields. After the taxonomy fields are removed, the related data will not be recoverable.

Figure 6 - Deleting Used Taxonomy Error

1.6 Taxon Editor Page

The Taxonomy names in the Taxonomy Index page link to their respective Taxon Editor pages. The Taxon Editor page allows you to select a taxonomy and browse through the taxonomy hierarchy. Selecting a taxon within the hierarchy opens an accordion menu which allows users to add new taxon children, edit their attribute values, set taxon permissions, move taxons within the hierarchy, relate taxons to other taxons within the taxonomy and archive or delete taxons.



Select Taxonomy: Regions ID=17

* Name: USA

☐ Do not allow this taxon to be selected within Content Editor

Figure 7 – Taxon Editor

- **Edit Attribute Values:** Edit Attribute Values Tab, allows users to add and edit the selected taxon's attribute values. Validation on these values will occur based on the properties selected for that attribute in the Manage Taxonomy page.

- **Add Child:** Add Child tab, is used to create new taxons within the taxonomy. Add child will create a new taxon under the selected taxon. In addition to the taxon attributes for this new child taxon, there are two additional checkboxes:
  - Move node to top/root level – This will make the new taxon a top level taxon at level 1 of the hierarchy.
  - Do not allow this taxon to be selected within Content Editor – This will mark the taxon as a "category" taxon that cannot be selected when tagging items in the content explorer.

- **Taxon Permissions:** Taxon Permissions tab is used to set role based permissions on taxons so that members of the selected roles are able to log into the Taxon Editor and have full read, write permissions to the selected taxon.

Note: Users given write access to taxons can also modify taxon permissions. However, the Taxonomy_Admin role will always have read write permissions that cannot be removed at the taxon level.

The final checkbox, 'Apply permissions to child taxons as well,' will cascade these permissions down to all child taxons without overriding their existing permissions.

- **Move Taxon:** Move Taxon tab is used to change the selected taxon's position in the Taxonomy hierarchy. Move taxon to top/root level will make the new taxon a top level taxon at level 1 of the hierarchy. Browse to search for a new parent taxon allows you to browse through the taxonomy hierarchy and pick a taxon under which the selected taxon will be placed.

- **Manage Taxon:** Manage Taxon tab is used to archive or delete taxons. Archive Taxon – Archiving a taxon moves it to a state where it is still saved in the CMS but can no longer be used to tag content or pull information on templates. Archived taxons can be un-archived.
- **Delete Taxon** – Deleting a taxon permanently removes the taxon and any tags and relationships to it. This action cannot be undone.

1.7 Associating Taxonomies to Content Type fields in the Workbench

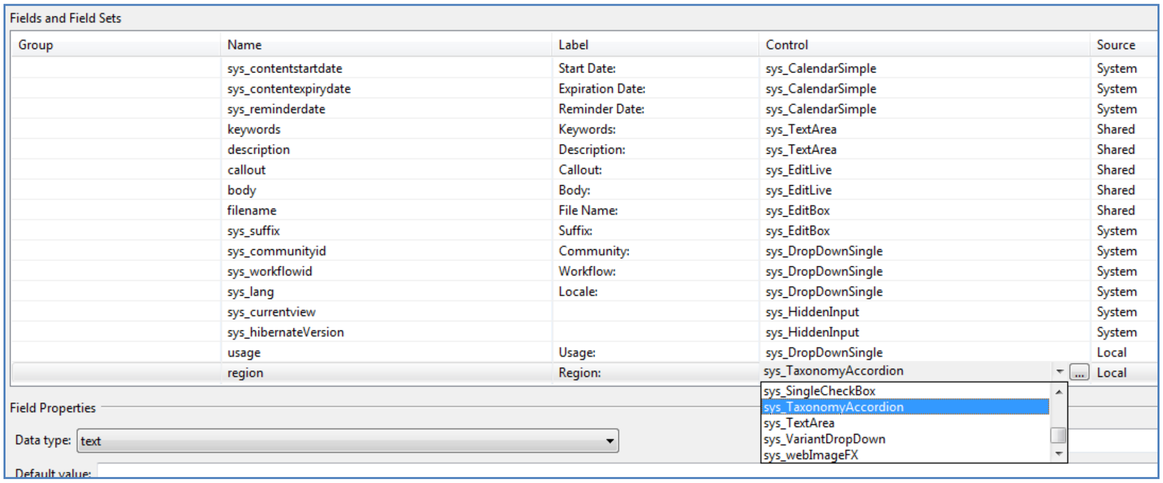
Adding a taxonomy accordion field on a content type allows content items of that type to be tagged by the taxonomy vocabulary for the specified taxonomy.

To associate a Taxonomy to a field in the Workbench:

1. Login to the Workbench, and open the Content Design Tab
2. Open the Content Type or Shared Field file where the field is to be added.
3. Create a new field and select the sys_TaxonomyAccordian for control as seen in Figure 8. Like other fields, this can be set to be a required field.

Note: Like other fields, this may be set to be a required field.

4. Select the ellipsis in the control to edit the control properties. In the Name column, double click to access the dropdown menu and select "taxonomy_name." In the Value column, enter the taxonomy name for the Taxonomy to be associated.



Fields and Field Sets

Group	Name	Label	Control	Source
	sys_contentstartdate	Start Date:	sys_CalendarSimple	System
	sys_contentexpirydate	Expiration Date:	sys_CalendarSimple	System
	sys_reminderdate	Reminder Date:	sys_CalendarSimple	System
	keywords	Keywords:	sys_TextArea	Shared
	description	Description:	sys_TextArea	Shared
	callout	Callout:	sys_EditLive	Shared
	body	Body:	sys_EditLive	Shared
	filename	File Name:	sys_EditBox	Shared
	sys_suffix	Suffix:	sys_EditBox	System
	sys_communityid	Community:	sys_DropDownSingle	System
	sys_workflowid	Workflow:	sys_DropDownSingle	System
	sys_lang	Locale:	sys_DropDownSingle	System
	sys_currentview		sys_HiddenInput	System
	sys_hibernateVersion		sys_HiddenInput	System
	usage	Usage:	sys_DropDownSingle	Local
	region	Region:	sys_TaxonomyAccordion	Local

Field Properties

Data type: text

Default value:

Figure 8 – Taxon Editor

Note: Once a taxonomy has been associated to a field, the taxonomy cannot be deleted unless this association is removed or the field is deleted.

1.8 Tagging Content in the Content Explorer

Once a taxonomy field has been added to a content type, items of that type can be tagged using the associated taxonomy.

To tag content items in the Content Explorer:

1. Login to the Content Explorer, and right click on a folder to create a new item of any type that has a Taxonomy control field.
2. The Taxonomy filed will be displayed as an accordion menu with two tabs – Quick Tag and Browsing (See Figure-9).
   a. Quick Tag enables searching for taxons in the taxonomy using type ahead.
   b. Browsing tab, allows users to hierarchically expand taxonomy.

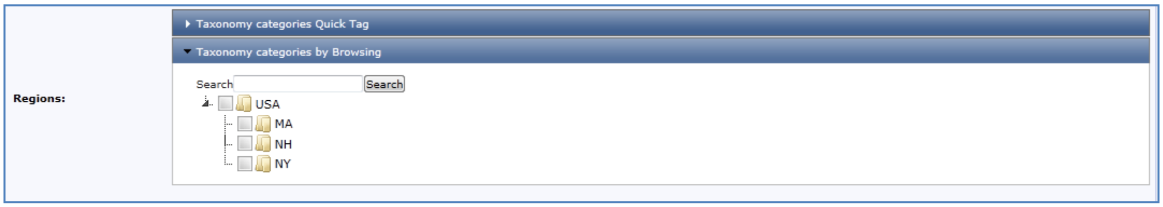

Regions:
Taxonomy categories Quick Tag
Taxonomy categories by Browsing
Search
Search
USA
MA
NH
NY

Figure 9 – Taxonomy field with Browsing tab selected in Content Explorer

3. Once a node is selected, the content item is tagged with that taxon (See Fig 10). It can be removed by clicking the "x" in the Quick Tag tab or by unselecting the node in the Browsing tab.

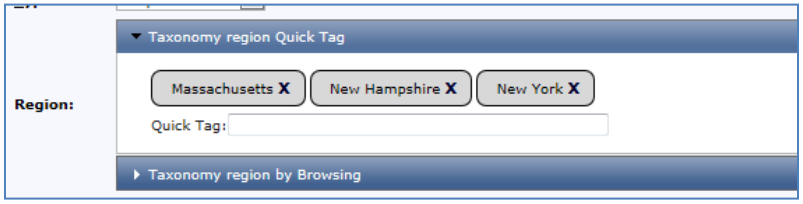


Figure 10 – Content item tagged with OOB Region taxonomy

1.9 Taxonomy JEXL in Templates

Taxonomy JEXLs can be used to retrieve information about taxonomies, taxons, attributes and tagged content.

The 4 taxonomy JEXLs are:

```
$rx.perc.taxonomy.getNodesFromIds(node_ids) - Returns a list of taxonomy nodes
(taxons) from a string of node IDs

$rx.perc.taxonomy.getRelatedNodes(node_id,relationship_type_id) - Returns a list of
related nodes (taxons) for a given node_id and relationship_type_name
$rx.perc.taxonomy.getTaxonomyAtrributes(taxonomy_name) - Returns a list of taxon
attributes types for a given taxonomy_name

$rx.perc.taxonomy.getTaxonomyNodes(taxonomy_name) - Returns a list of taxonomy nodes
(taxons) for a given taxonomy_name
```

To see additional methods available for Taxonomy, refer to the Javadocs. These can be accessed through the help menu in the workbench.:

Help>HelpContents>Rhtyhmyx Workbench Help> Rhtyhmyx API JavaDoc

Under All Classes, there are two taxonomy related docs:

- com.percussion.taxonomy
- com.percussion.taxonomy.jexl

Examples:
Below are example macros built using the taxonomy JEXL and methods available in the Javadoc.
To test these macros, the following need to be configured in the Workbench and Content Explorer:

1. Create a new content type (Taxonomy-test) for testing and add 2 fields ‘region’ and ‘tags’ with the Taxonomy_Accordion control.

2. Associate the out of box ‘Region’ and ‘Tags’ taxonomies respectively to the above fields.

   a. If these taxonomies do not exist in the testing environment, associate any two taxonomies. The taxonomy associated with the ‘tags’ field should have the relate feature on.

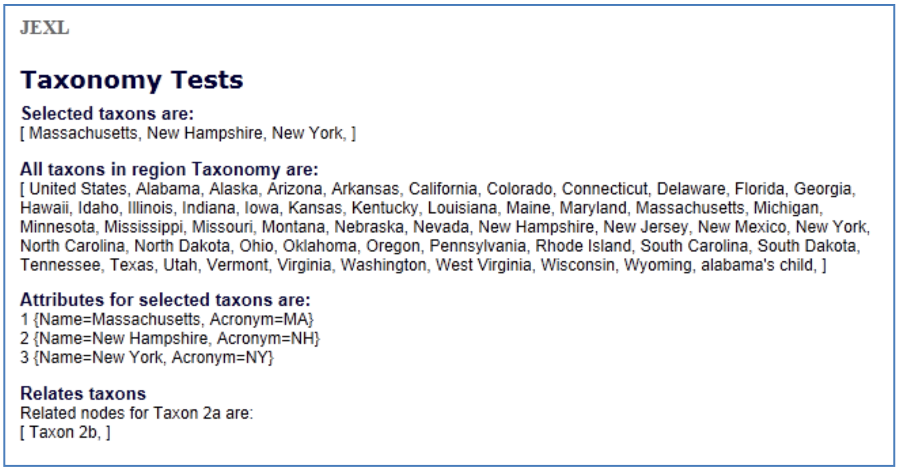
3. Edit a template (Taxonomy-test-template) associated to the Taxonomy-test content type and add the following velocity macros:

```
## This macro outputs all the taxons selected in the content item
#macro (get_selected_taxon_names)
    #set($selected =
$rx.percTaxonomyJexl.getNodesFromIds($sys.item.getProperty("region").Stri
ng))
       <h3>Selected taxons are:</h3> <p> [
       #foreach($item in $selected)
              $item.Name,
       #end
       ]</p>
#end


## This macro outputs a list of all taxons for a given taxonomy
#macro(get_all_taxons_in_taxonomy $taxonomyName)
       #set($nodes =
$rx.percTaxonomyJexl.getTaxonomyNodes($taxonomyName))
       <h3> All taxons in $taxonomyName Taxonomy are: </h3> <p>[
       #foreach($node in $nodes)
              $node.Name,
       #end
       ] </p>
#end

## This macro outputs a list of the attribute values for all the taxons
selected for the ‘region’ field in the content item
#macro(get_attributes_for_selected_taxons)
       <h3>Attributes for selected taxons are: </h3> <p>
       #set($selected =
$rx.percTaxonomyJexl.getNodesFromIds($sys.item.getProperty("region").Stri
ng))
              #foreach($item in $selected )
                     $velocityCount $item.getAttributes() <br>
              #end
       </p>
#end


## This macro outputs a list of related taxons for all selected taxons in
the content item
#macro(get_related_taxons)
       <h3>Relates taxons</h3>
       #set($selected =
$rx.percTaxonomyJexl.getNodesFromIds($sys.item.getProperty("tags").String
))
       #foreach($item in $selected)
              Related nodes for $item.Name are: <p>[
           #set($related=$rx.percTaxonomyJexl.getRelatedNodes($item.Id,
"Related To"))
              #foreach($item in $related)
                     $item.Name,
              #end
              ]</p>
       #end

#end

## These are the calls to the above defined macros
  <h1>Taxonomy Tests </h1>
  #get_selected_taxon_names()
  #get_attributes_for_selected_taxons()
  #get_attributes_for_selected_taxons()
  #get_related_taxons()
```

1. In the Taxon Editor in the Content Explorer, edit the Tags taxonomy to add additional 2nd level taxons. Relate the new taxons to the first one.

2. In the Content Explorer, create a new content item of the Taxonomy-Test content type and tag the item using both taxonomy fields. For the ‘tags’ field, make sure to tag using the taxon that had other taxons related to it.

3. Preview the content item using the taxonomy-test-template. You should see a list of selected taxons, all taxons in selected taxonomy, attributes for selected taxons and related nodes for selected taxons. See example in Figure-11. In previewed example, the content has been tagged using MA, NH and NY for the ‘region’ field and Taxon 2a for the ‘tags’ field.



JEXL

Taxonomy Tests

Selected taxons are:
[Massachusetts, New Hampshire, New York,]

All taxons in region Taxonomy are:
[United States, Alabama, Alaska, Arizona, Arkansas, California, Colorado, Connecticut, Delaware, Florida, Georgia, Hawaii, Idaho, Illinois, Indiana, Iowa, Kansas, Kentucky, Louisiana, Maine, Maryland, Massachusetts, Michigan, Minnesota, Mississippi, Missouri, Montana, Nebraska, Nevada, New Hampshire, New Jersey, New Mexico, New York, North Carolina, North Dakota, Ohio, Oklahoma, Oregon, Pennsylvania, Rhode Island, South Carolina, South Dakota, Tennessee, Texas, Utah, Vermont, Virginia, Washington, West Virginia, Wisconsin, Wyoming, alabama's child,]

Attributes for selected taxons are:
1 {Name=Massachusetts, Acronym=MA}
2 {Name=New Hampshire, Acronym=NH}
3 {Name=New York, Acronym=NY}

Relates taxons
Related nodes for Taxon 2a are:
[Taxon 2b,]

Figure 11–Taxonomy template preview of test macros.

1.10 Additional Information

MSM cannot be used to move taxonomies across servers. The recommended best practice is to establish the same taxonomy structure on the target server before using MSM to move any content types that reference taxonomies. Note that the taxonomies should be referenced by taxonomy_name (not taxonomy_id) on the content type fields for them to correctly associate with the expected taxonomies.

If Taxonomy is already installed prior to upgrading to CM System 7.2.0 please refer to *Upgrading Taxonomy* for more information.

Note: this feature is automatically enabled in new installations.

2. Link Automation

This feature allows content contributors to use a new menu under Tools in Ephox Edit-live fields to automatically convert non-managed, hyperlinks in Ephox fields to inline templates to managed items in the CMS. For URLs with the same domain/ alias of managed sites, the URL structure is used to either identify and link to items if they already exist in the expected folder path in the CMS, or create new items of type Managed Auto Link in Folders in Content Explorer if they don't. If a Managed Auto Link item is created in Folders, once its corresponding item is created in the expected folder location under a managed site and transitioned to a workflow state, a workflow action associated to a transition identifies the original Managed Autolink item, deletes it, and replaces the inline template link to point to the new item instead. For hyperlinks in Ephox linking to external links, the Generate Managed Links creates inline templates either linking to existing External Link items in Folders in the Content Explorer or it creates new items in the folder path determined by the URL structure.

2.1 Enabling and Configuring Link Automation

To enable Link Automation, the parameters in the autoLinkGeneration.properties file need to be filled out:

({root} /rxconfig/Server/autoLinkGeneration.properties)

Each property has a detailed description of the parameter's function.

A few of these properties require additional set up in the Workbench:

1 **managedAutoLink.ContentTypeName** – The name of an 'external link' content type to be used when creating links to pages that fall under any of the sites listed under managedSites parameter and are not currently in the CMS system (they are expected to be imported at a later time.) If empty or missing, or the content type doesn't exist, these types of links will not be transformed.

2 **externalLink.ContentTypeName -** The name of an 'external link' content type to be used when creating links to pages that do not fall under any of the sites listed under managedSites parameter. (They are not under management and are never expected to be.) If empty or missing, or the content type doesn't exist, these types of links will not be transformed.

The above two parameters require content types that have three parameters, sys_title, displaytitle and url, and no other required fields. The same external link content type (eg: the out of box rffExternalLink) may be used for both parameters. However, it is recommended that two different content types be used so that it is easier to distinguish between managedAutoLink items which temporarily substitute items that will be migrated into the CMS at a later time and ExternalLink items which store URLs to sites never to be managed in the CMS.

templateName - The name of the template to use when creating inline templates Required if externalLink.ContentTypeName or managedAutoLink.ContentTypeName is specified.

All inline templates created by the Link Automation process are generated using this template. Therefore, this template should be allowed for the content types specified in externalLink.ContentTypeName and managedAutoLink.ContentTypeName properties as well as all other content types that may be referenced by inline templates in Ephox fields.

To allow this template for the above content types:

1 In Workbench, in the Content Design tab, expand each content type node

2 Expand the ‘Allowed Templates and XSL Variants’ and drag the template specified in the templateName property and drop it in.

3 Repeat this process for all content types that may be referenced by inline templates

Note: This template is now being used for regular content types as well as external link content types with url fields. This template needs to be modified to handle either case – generate $pagelinks to items within the CMS as well as $pagelinks to the values in the URL fields of the ExternalLink and managedAutoLink content types.

Additionally the binding needs to account for anchors in the url. Below is an example where the out of box rffSnLink template has been modified to work with both, regular as well as ExternalLink content types:

In the Workbench, open the template specified in the templateName property (rffSnLink)
In the Bindings tab, modify the $pagelink binding:

```
$if ($sys.item.hasProperty('rx:url').String)  {
$sys.item.getProperty('rx:url').String)+$rx.asmhelper.getAnchor($sys.item
,true).;  } else {
$rx.location.generate($sys.assemblyItem);  }
```

This checks to see if there is a URL field on the content type and sets the $pagelink binding to its value if it exists. If not, it sets the $pagelink binding’s value to the item location in the CMS.

1 In User velocity macro file, create a modified fieldLink Macro that replaces filed value with selected text if applicable:

```
#macro(select_fieldLink $fieldname $pagelink)##
#startAAField($fieldname $pagelink)##
#if ($sys.part.render)##
$tools.esc.xml($rx.asmhelper.getSelectedText(params,$sys.item.getProperty
($fieldname).String))##
#end##
#endAAField($fieldname)##
#end
```

2. In the template specified in templateName, the link can be inserted using the following macro call:

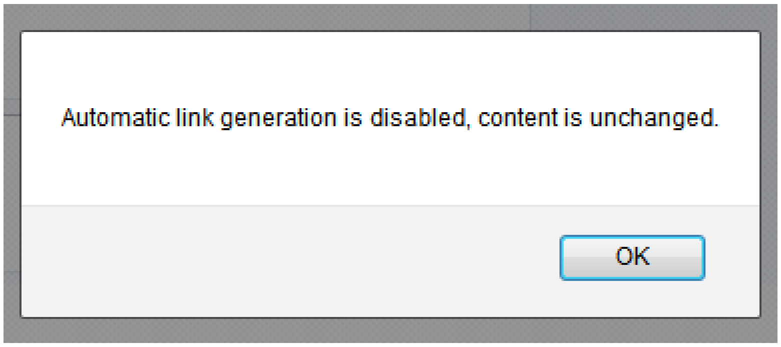
```
#select_fieldLink("displaytitle" $pagelink)
```

Once this template has been updated, and allowed on all content types, the out of box inline template slot needs to be updated to allow this template:

1. In Workbench, in the Assembly Design tab, open the sys_inline_variant slot

2. Ensure that all content types that can be used in inline templates (including the content types specified in properties managedAutoLink.ContentTypeName and externalLink.ContentTypeName) are allowed with the new, modified template specified in property templateName.

If the properties in the autoLinkGeneration.properties are not correctly or completely filled in, the error message in Figure-12 will be displayed in the Content Explorer when Link Automation is run in Ephox.



Automatic link generation is disabled, content is unchanged.

Figure 12 – Error message when properties in autoLinkGeneration.properties are not set

2.2 Updating the Ephox Configuration File

This step is for upgrades only. Out of box installations can skip to the next step.

On upgrades, by default the Link Automation Menu is not available in Ephox. To activate this feature, the elj_config.xml file needs to be updated:
/rx_resources/ephox/elj_config.xml

Uncomment the following line:

```
<customMenuItem action="raiseEvent"
imageURL="../rx_resources/ephox/images/managedLinks.gif"
name="GenerateManagedLinks" text="Generate Managed Links"
value="RxEphoxGenerateManagedLinks"/>
```

Update the imgURL value to:

```
="../sys_resources/ephox/images/managedLinks.gif"
```

2.3 Enabling True Inline Templates

True inline templates need to be enabled so that there are no breaks in the text when Link Automation inserts Inline Templates. To enable true inline templates:

1. Ensure that rxEditLiveFormEncodeDecode.disabled is renamed to rxEditliveFormEncodeDecode.xml under {installation root}\rx_resources\ephox\plugins

2. Open all editlive configuration files that are in use and add the attribute inlineSpanPriority="true" to the htmlFilter element. (This will force span tags to have priority order within an all inline set of tags).

3. Open the server.properties file located in {installation root}\rxconfig\Server and set the property allowTrueInlineTemplates=true

4. Re-start server

5. Clear browser cache and java plugin cache

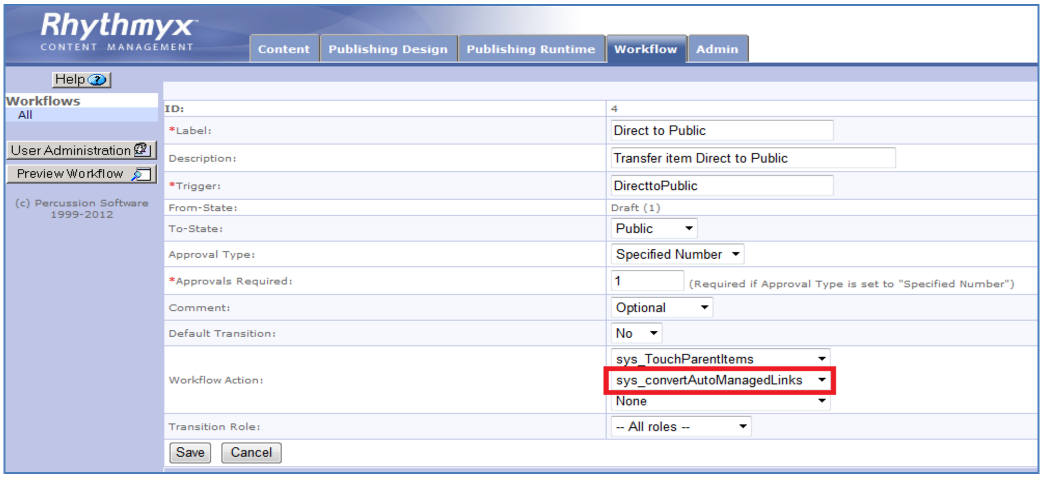
Templates that should be inline, should not contain block elements and must have a <span> tag as the root element.

True "inline" inline templates will be indicated by two arrow images (> inline stuff <) and can be removed by double clicking to invoke the context menu with a delete option.

2.4 Adding the Workflow Action for Auto Link Generation

A new workflow action, sys_convertAutoManagedLinks, examines all the inline templates of an item, and for each one that links to a managedAutoLink item, if the dependent item is now under management in the expected folder location, the original reference to the managedAutoLink will be replaced with a reference directly to the new item. Once this change has successfully been made, the managedAutoLink item under folders is deleted from the system.

It is recommended that you add this workflow action to all transitions going to the public state. A modification has been made to allow three workflow actions per transition in the workflow. When adding the sys_ convertAutoManagedLinks workflow action to workflow transitions, make sure not to replace other workflow actions associated to that transition. See Figure-13 for reference.



ID:	4
*Label:	Direct to Public
Description:	Transfer item Direct to Public
*Trigger:	DirecttoPublic
From-State:	Draft (1)
To-State:	Public
Approval Type:	Specified Number
*Approvals Required:	1 (Required if Approval Type is set to "Specified Number")
Comment:	Optional
Default Transition:	No
Workflow Action:	sys_TouchParentItems sys_convertAutoManagedLinks None
Transition Role:	-- All roles --

Figure 13 – Example of sys_onvertAutoManagedLinks workflow action added to the Direct to Public transition without replacing the sys_TouchParentItems workflow action.

2.5 Updating Location Schemes and sys_title to Supported Format

Link Automation uses the url structure to determine folder path and file name when searching for and creating items. The sys_title field, required for all content types is used by Link Automation as the filename at the end of the url.

New character restrictions must be enforced for Folder Name and $sys_title field to support Link Automation:

- No spaces or special characters are allowed.
- $sys.title fields must all have the file extensions appended at the end (eg .html, .gov, .edu)
- Similarly, the location scheme must conform to these specifications to support Link Automation. No regular expressions can be used to insert or replace any characters or spaces and must use folder path and sys_title only to determine the url structure:

```
$sys.pub_path + $sys.item.getProperty('rx:sys_title’).String
```

2.6 Using Link Automation in the Content Editor

Once Link Automation is enabled and configured, this feature is accessed in the Ephox EditLive editor by clicking Tools>Generate Managed Links. This menu runs Link Automation which parses the content for non-managed links which are processed and convert to a managed links.

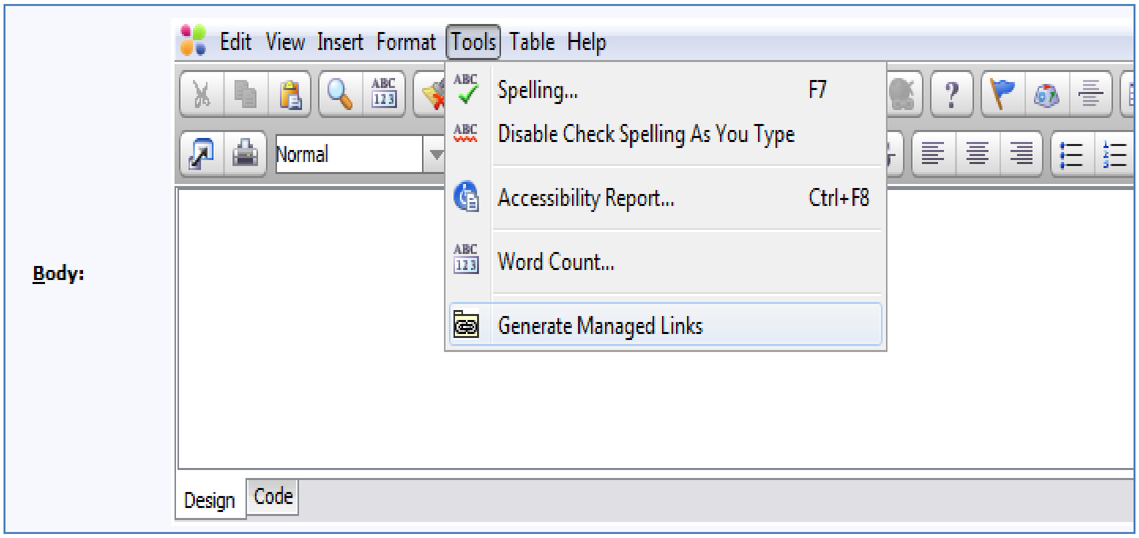

Edit View Insert Format Tools Table Help
Spelling... F7
Disable Check Spelling As You Type
Accessibility Report... Ctrl+F8
Word Count...
Generate Managed Links
Normal
Body:
Design
Code

Figure 14 – Link Automation menu in Ephox

All <a> tags are examined and handled except ones with image tags and already managed content which are skipped. The remaining <a> tags are examined and processed in the following manner:

- If the href is page or site relative, or the href's domain matches that of a managed site or its alias, it determines if an item exists in the expected folder path with the expected $sys_title and links to it. If not, it searches for a managedAutolink item with the expected $sys_title in //Folders/{Rx virtual root}/{full path of item}/{item name} and links to it. If no items exists in either of the above locations, it creates a new managedAutoLink item and links to that instead.
- If the href is an external url , it check under //Folders external/{domain}/{full path of item}/{item name} to see if an ExternalLink content item exists in the expected folder path with the expected $sys_title and links to it. If not, it creates a new ExternalLink item and links to it.

Note that all managedAutoLink items are created under //Folders/{Rx virtual root}/ and all ExternalLink items are created under //Folders external/. See Figure - 15

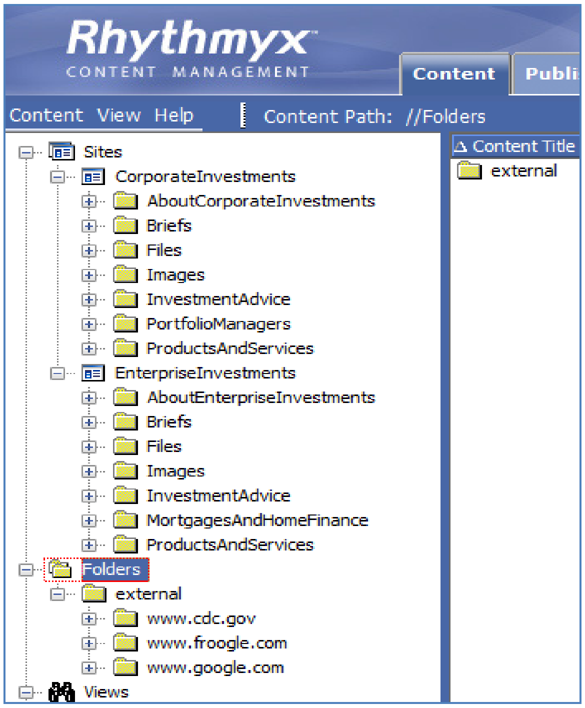
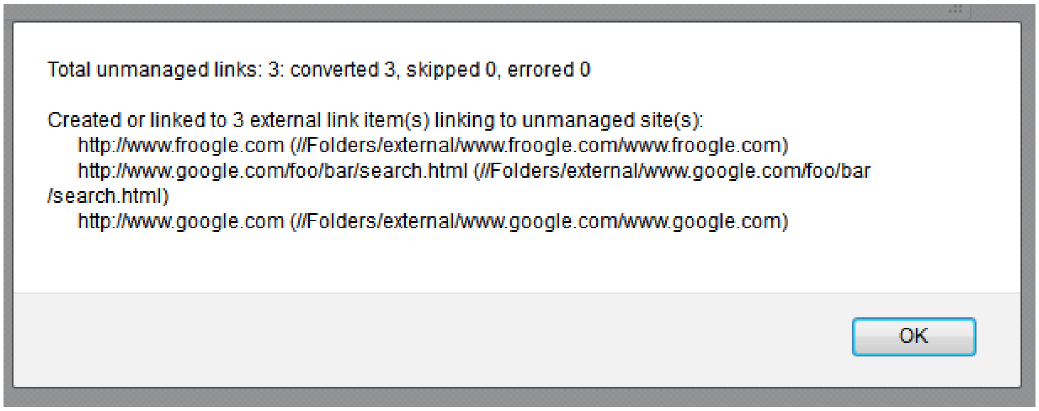


Figure 15 – ExternalLink items created under //Folder/external

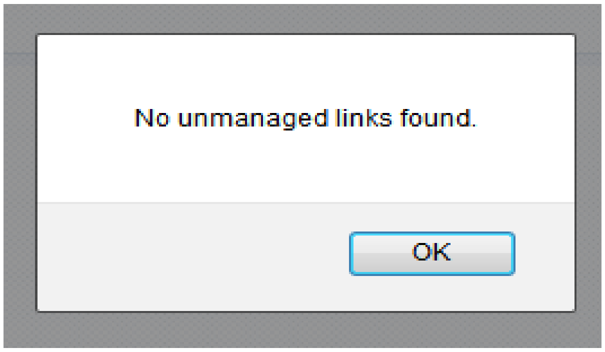
When 'Generate Managed Links' is run, it present a dialog with a list of links that were processed. See Figure-16,17 :



```
Total unmanaged links: 3: converted 3, skipped 0, errored 0

Created or linked to 3 external link item(s) linking to unmanaged site(s):
    http://www.froogle.com (//Folders/external/www.froogle.com/www.froogle.com)
    http://www.google.com/foo/bar/search.html (//Folders/external/www.google.com/foo/bar
/search.html)
    http://www.google.com (//Folders/external/www.google.com/www.google.com)
```

Figure 16 – Link Automation dialog

Or



No unmanaged links found.

Figure 17 – Link Automation dialog when no links found

3. Revision Removal

This functionality enables users in the Admin role to delete specified revisions of content items. This is accomplished by creating a scheduled task using the sys_PurgeRevisions extension. This feature allows purging of older revisions for content items of all content types and currently does not support the ability to choose a specific content type. This new extension, sys_PurgeRevisions, can be configured using the following 4 parameters:

- *alwaysKeepMinNumberOfRevs:* The minimum number of revisions to keep in the CMS.
- *alwaysKeepRevsYoungerThanDays:* Always keep revisions in the CMS younger than these many days.
- *deleteRevsAboveCount:* Delete revisions in the CMS above this number of revisions. The Scheduled Task will delete (*deleteRevsAboveCount* + 1) to the oldest revision that exists for that content item.
- *deleteRevsOlderThanDays:* Delete revisions in the CMS older than these many days.

The above 4 parameters are required and the scheduled task will not execute without them. The scheduled task will only execute if the following conditions are met by these 4 parameters:

- *alwaysKeepMinNumberOfRevs* < *deleteRevsAboveCount*
- *alwaysKeepRevsYoungerThanDays* < *deleteRevsOlderThanDays*.

Revisions that fall within parameters *alwaysKeepMinNumberOfRevs* and *alwaysKeepRevsYoungerThanDays* will always be kept in the CMS. Zero is a valid value for all the above parameters and ensures that the current latest revision is kept.

The Revision Removal process has a dedicated log, *revisionPurge.log.* This log will have a detailed description of all the errors encountered. On success, information about the number of revisions removed in total and the number of revisions removed for each content type.

3.1 Keep Outranks Delete

The new extension will iterate through all content items, and mark revisions either for Delete or Keep. Revisions marked for Delete will be deleted from within the CMS. If an item gets marked for Delete AND Keep, it will be kept. If a revision is not marked for either Delete or Keep, it will be kept in the CMS.

For example, consider a content item with multiple revisions:

- Revision1 – 8/1/2011
- Revision2 – 8/12/2011
- Revision3 – 9/21/2011
- Revision4 – 3/12/2012
- Revision5 – 6/1/2012
- Revision6 – 6/12/2012
- Revision7 – 6/20/2012 (current date) – Last

Assuming that the revision removal task is scheduled and has the following parameters defined:

- *alwaysKeepMinNumberOfRevs:* 2
- *alwaysKeepRevsYoungerThanDays:* 30
- *deleteRevsAboveCount:* 4
- *deleteRevsOlderThanDays:* 45

Revisions are marked as Keep/Delete in the following manner and Final Outcome:

Revision	**Date**	**alwaysKeepMinNumberOfRevs** **(2)**	**alwaysKeepRevsYoungerThanDays** **(30)**	**deleteRevsAboveCount** **(4)**	**deleteRevsOlderThanDays** **(45)**	**Final Outcome**
1	8/1/2011	Delete	Delete	Delete	Delete	Delete
2	8/12/2011	Delete	Delete	Delete	Delete	Delete
3	9/21/2011	Delete	Delete	Delete	Delete	Delete
4	3/12/2012	Delete	Delete	**Keep**	Delete	Delete
5	6/1/2012	**Keep**	**Keep**	**Keep**	**Keep**	**Keep**
6	6/12/2012	**Keep**	**Keep**	**Keep**	**Keep**	**Keep**
7	6/20/2012	**Keep**	**Keep**	**Keep**	**Keep**	**Keep**

Delete Revisions if the following logic is met:

Not(alwaysKeepMinNumberOfRevs OR alwaysKeepRrevsYoungerThanDays) AND (deleteRevsAboveCount OR deleteREvsOlderThanDays)

Final Outcome: Revisions 4 through 7 are kept in the CMS and revisions 1 through 3 are deleted.

3.2 Revision Removal Process

1 Install the Revision Removal Patch

Shut down the server. Install the Revision Removal patch (CMS-XXX).

2 Set the parameter in server.properties

Create a new parameter ***allowPurgeRevisionsScheduledTask*** in /rxconfig/Server/server.properties and set its value to ***true***. This parameter indicates whether or not the Revision Removal process is enabled or disabled. The scheduled task will require this property to be enabled and will not run if this property is not set to 'true'.

3 Start the server and Schedule a task

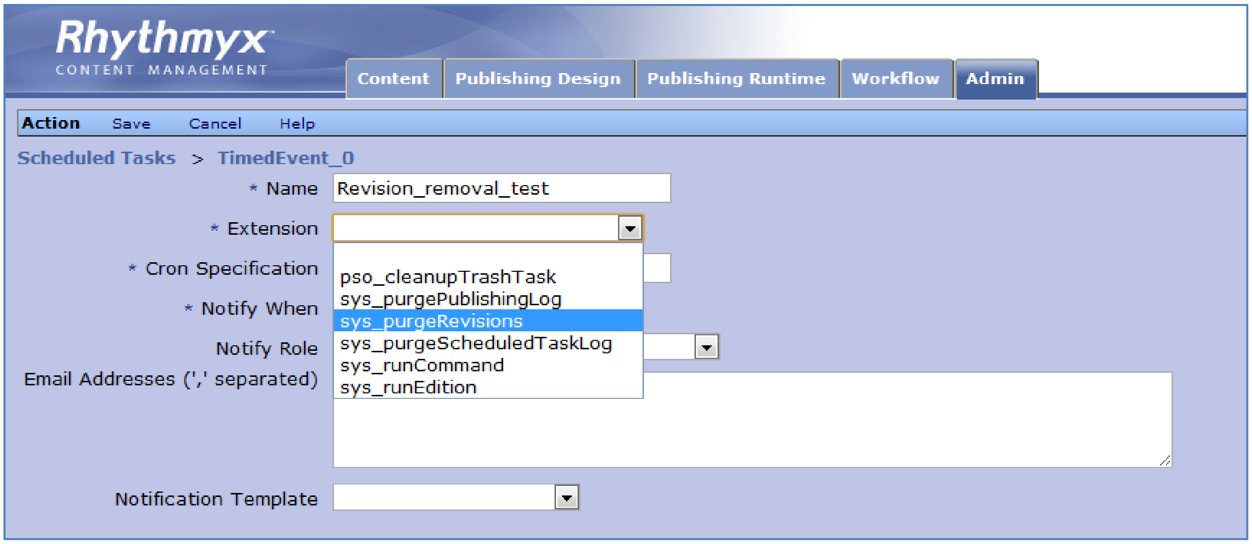
Once the server is up, log into the Content Explorer as an Admin role.

i. In the Admin tab, click *Scheduled Tasks* in the left navigation menu.
ii. *Action > Create Task*
iii. Enter a name with no spaces for the Scheduled tasks. For example, *Revision_removal_test*
iv. In the dropdown for *Extension,* select *sys_purgeRevisions.* Refer to fig. 1.0
v. Once *sys_PurgeRevisions* extension is selected, four parameters, *alwaysKeepMinNumberOfRevs, alwaysKeepRevsYoungerThanDays, deleteRevsAboveCount, and deleteRevsOlderThanDays* will appear. See fig. 1.1
vi. Enter in numerical values into these parameters and ensure that *alwaysKeepMinNumberOfRevs* < *deleteRevsAboveCount* and *alwaysKeepRevsYoungerThanDays < deleteRevsOlderThanDays.* If the above conditions are not met, the scheduled task will not execute. The server log will show that there were invalid values entered and that the scheduled task did not execute.
vii. Enter the *Cron Specification* for the task. Rhythmyx uses the Quartz Enterprise Job Scheduler (http://www.opensymphony.com/quartz/). Eg. 0 30 15 0 0 0 *. This will run the scheduled job every day at 3:30pm.
viii. Set up any notifications if required. Either *Always*, to get notified whenever the revision removal scripts run or *On Failure,* to get notified when there the revision removal process fails to execute. Select the desired Notification Template, preferably *Task_Template.*
ix. Save the newly created Schedule Task

Once the task has been scheduled wait for the task to run at the scheduled time or the Admin user can also run the task by clicking on the task name and once the editor for the task opens, *Action>Run Now.*

To see the status of *Revision_removal_test* task, click on *Task Log* for the left side menu option. The Task Log mentions the Name of the scheduled task, Start time and status of the task, whether it was executed successfully or failed.

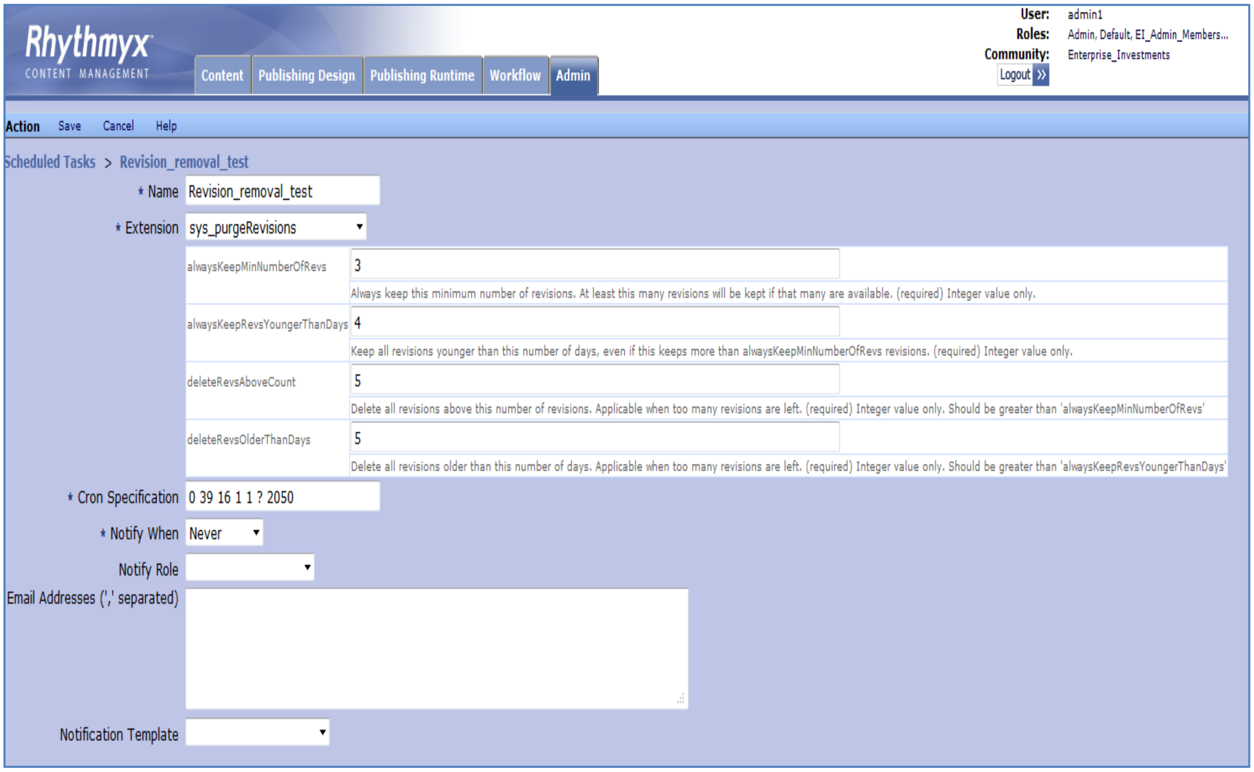
Check the *revisionPurge.log* for detailed logging of the task.



Scheduled Tasks > TimedEvent_0

* Name	Revision_removal_test
* Extension	
* Cron Specification	
* Notify When	
Notify Role	
Email Addresses (',' separated)	
Notification Template	

Figure 18



User: admin1
Roles: Admin, Default, EI_Admin_Members...
Community: Enterprise_Investments

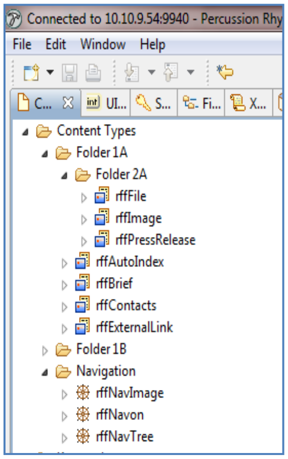
Scheduled Tasks > Revision_removal_test

* Name	Revision_removal_test	
* Extension	sys_purgeRevisions	
	alwaysKeepMinNumberOfRevs	3 Always keep this minimum number of revisions. At least this many revisions will be kept if that many are available. (required) Integer value only.
	alwaysKeepRevsYoungerThanDays	4 Keep all revisions younger than this number of days, even if this keeps more than alwaysKeepMinNumberOfRevs revisions. (required) Integer value only.
	deleteRevsAboveCount	5 Delete all revisions above this number of revisions. Applicable when too many revisions are left. (required) Integer value only. Should be greater than 'alwaysKeepMinNumberOfRevs'
	deleteRevsOlderThanDays	5 Delete all revisions older than this number of days. Applicable when too many revisions are left. (required) Integer value only. Should be greater than 'alwaysKeepRevsYoungerThanDays'
* Cron Specification	0 39 16 1 1 ? 2050	
* Notify When	Never	
Notify Role		
Email Addresses (',' separated)		
Notification Template		

Figure 19

4. Grouping of Content Types

In the Workbench, developers can organize Content Types in folders so that end users can view these folders as submenus within New Item action menu when creating content items in the Content Explorer. Only Content Types visible in the logged in community will be visible and by default, the action menus will reflect the exact folder structure created within the Workbench. However, there is an option that allows you to skip certain folders. See Implementation Document for details.



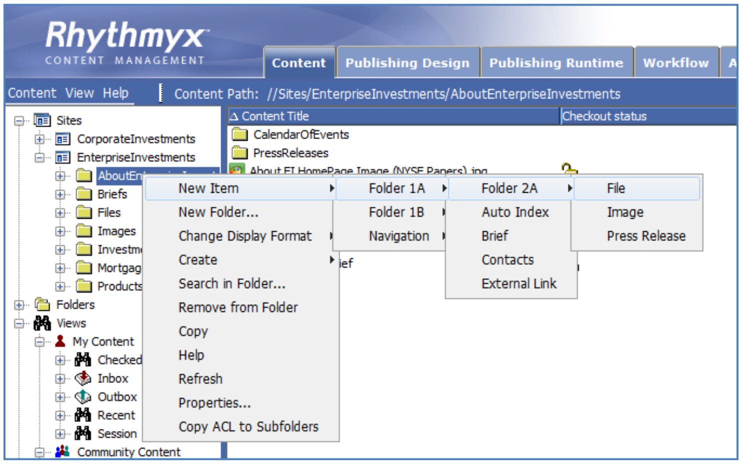


Figure 20 - Grouped Content Types in Workbench

Figure 21 - Resulting submenus in Content Explorer

4.1 Creating Content Type Groups in Workbench

To organize Content Types within folders:

1. Login to the Workbench, and open the Content Design tab.
2. Expand the Content Types node.
3. Create desired sub-folders to organize content types. These folders will determine the sub menus that end-users will see within Create > New Item action menu in the Content Explorer.

Content Types that are not placed within sub-folders will be displayed directly within the Create > New Item action menu. All Content Types and group names will be sorted alphabetically.
Note that the Navigation folder in the Content Design tab cannot be deleted or moved. As a result, for all communities where navigation content types are visible, there will be a top level navigation submenu.

4.2 Configuring the Content Type Grouping Feature

Content Type Grouping can be configured using 2 new properties within server.properties located at ({root}\rxconfig\Server\server.properties) .

1. To enable grouping, add a new property called ‘contentTypeGroupingInCX’ and set its value to true

```
        contentTypeGroupingInCX=true
```

2. To enable skipping of multi-tiered folder structure where folders have only one sub-folder, add a new property called ‘skipFoldersIfOnlySingleSubMenuInCX’ and set its value to true. This will compress a highly tiered folder-structure where if only one sub-folder exists within a folder that folder it will be skipped when displaying it as an action menu. Note that the grouping property must be enabled in order for this to work.

```
        skipFoldersIfOnlySingleSubMenuInCX =true
```

By default, if this property is not set, the entire folder structure will be shown in the Create > New Item action menu.

To demonstrate how grouping of content types behave, below is an example that explores the action menus generated in the content explorer depending on the folder structure in workbench, Content Type visibility by community, and whether the folder skipping property is enabled:

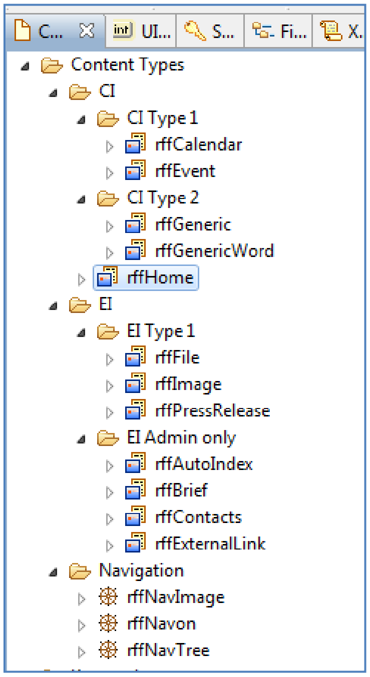
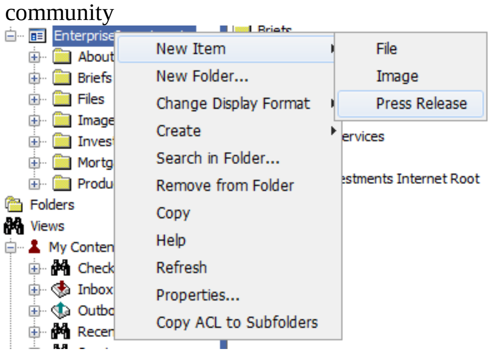
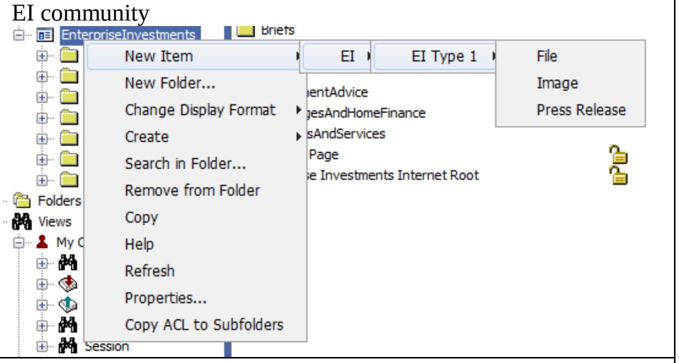
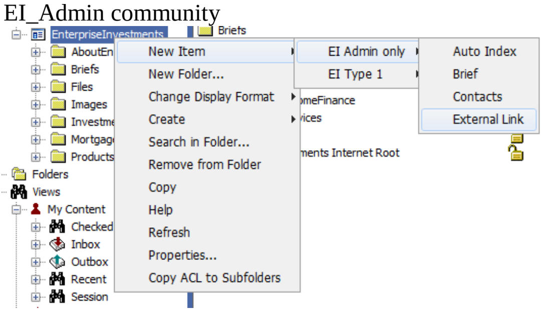
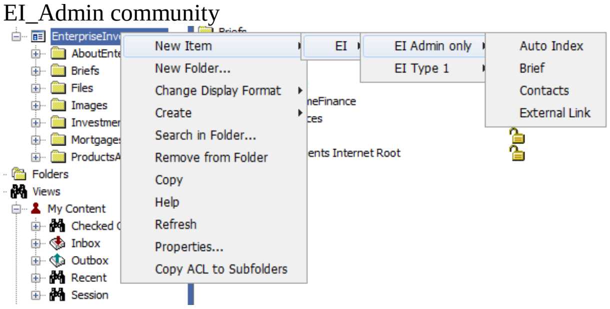
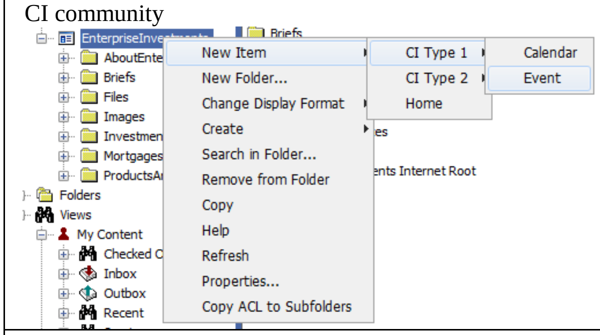
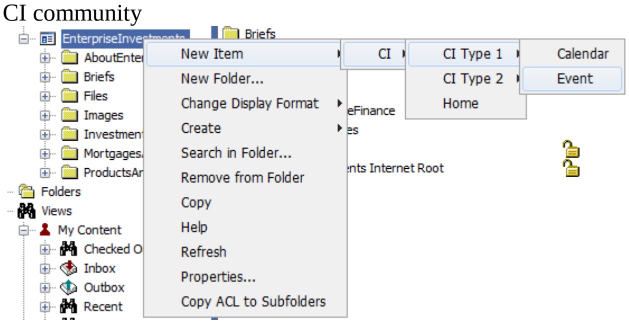
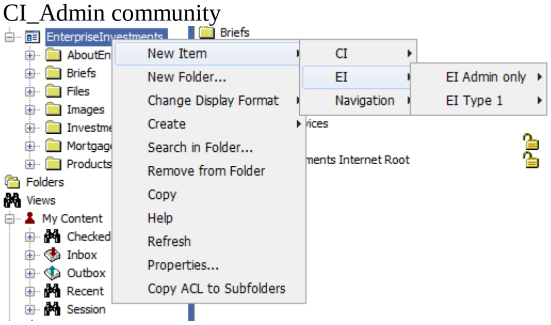
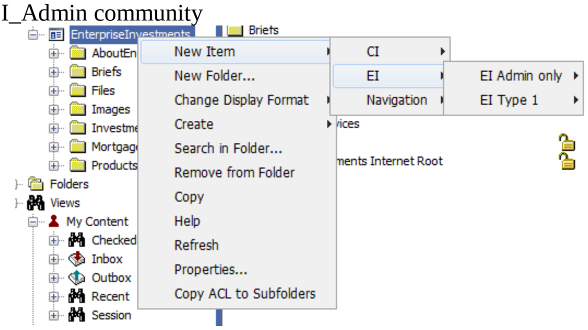


Figure 22 - Grouped Content Types in Workbench

There are 4 communities with the following visibility settings:

Community Name	**Visible Content Types**
EI	All Content Types in folder EI-Type1: `rffFile, rffImage, rffPressRelease`
EI_Admin	All Content Types in folder EI: `rffFile, rffImage, rffPressRelease, rffAutoIndex, rffBrief, rffContacts, rffExternalLink`
CI	All Content Types in folders CI-Type1 and CI-Type2: `rffCalendar, rffEvent, rffGeneric, rffGenericWord, rffHome`
CI_Admin	All Content Types in folder Content Types: `rffFile, rffImage, rffPressRelease, rffAutoIndex, rffBrief, rffContacts, rffExternalLink, rffCalendar, rffEvent, rffGeneric, rffGenericWord, rffHome`

Recorded below is the resulting submenu structure in each community

SkipFoldersIfOnlySubMenuInCX = true	SkipFoldersIfOnlySubmenuInCX = false
EI community  Enterprise Briefs About Briefs Files Image Inves Mortg Produ Folders Views My Conten Check Inbox Outbo Recen New Item New Folder... Change Display Format Create Search in Folder... Remove from Folder Copy Help Refresh Properties... Copy ACL to Subfolders File Image Press Release ervices estments Internet Root	EI community  EnterpriseInvestments Briefs Folders Views My C Session New Item New Folder... Change Display Format Create Search in Folder... Remove from Folder Copy Help Refresh Properties... Copy ACL to Subfolders EI EI Type 1 File Image Press Release entAdvice gesAndHomeFinance sAndServices Page e Investments Internet Root
EI_Admin community  EnterpriseInvestments Briefs AboutEn Briefs Files Images Investme Mortgag Products Folders Views My Content Checked Inbox Outbox Recent Session New Item New Folder... Change Display Format Create Search in Folder... Remove from Folder Copy Help Refresh Properties... Copy ACL to Subfolders EI Admin only EI Type 1 Auto Index Brief Contacts External Link omeFinance vices ments Internet Root	EI_Admin community  EnterpriseInve AboutEnt Briefs Files Images Investmer Mortgage ProductsA Folders Views My Content Checked C Inbox Outbox Recent Session New Item New Folder... Change Display Format Create Search in Folder... Remove from Folder Copy Help Refresh Properties... Copy ACL to Subfolders EI EI Admin only EI Type 1 Auto Index Brief Contacts External Link neFinance ces ents Internet Root
CI community  EnterpriseInve Briefs AboutEnte Briefs Files Images Investmen Mortgages ProductsA Folders Views My Content Checked C Inbox Outbox Recent New Item New Folder... Change Display Format Create Search in Folder... Remove from Folder Copy Help Refresh Properties... Copy ACL to Subfolders CI Type 1 CI Type 2 Home Calendar Event es ents Internet Root	CI community  EnterpriseInvestments Briefs AboutEnte Briefs Files Images Investment Mortgages ProductsAr Folders Views My Content Checked O Inbox Outbox Recent New Item New Folder... Change Display Format Create Search in Folder... Remove from Folder Copy Help Refresh Properties... Copy ACL to Subfolders CI CI Type 1 CI Type 2 Home Calendar Event eFinance es nts Internet Root
CI_Admin community  EnterpriseInvestments Briefs AboutEn Briefs Files Images Investme Mortgag Products Folders Views My Content Checked Inbox Outbox Recent Session New Item New Folder... Change Display Format Create Search in Folder... Remove from Folder Copy Help Refresh Properties... Copy ACL to Subfolders CI EI Navigation EI Admin only EI Type 1 vices ments Internet Root	CI_Admin community  EnterpriseInvestments Briefs AboutEn Briefs Files Images Investme Mortgag Products Folders Views My Content Checked Inbox Outbox Recent Session New Item New Folder... Change Display Format Create Search in Folder... Remove from Folder Copy Help Refresh Properties... Copy ACL to Subfolders CI EI Navigation EI Admin only EI Type 1 ices ments Internet Root

Note: The 'skipFoldersIfOnlySingleSubMenuInCX' property within server.properties is used for Grouping of Content Types as well as Templates. If this property is set to true, Template's Preview and Active Assembly action menus will behave similar to the New Item action menu of Content Types. Refer to Grouping Templates in Preview feature for details on the Template Grouping feature.

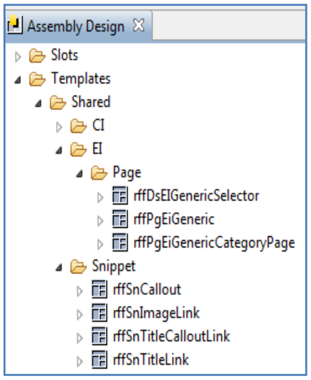
4.3 Additional Information

When using MSM, if the source server is enabled with grouping of content types feature and has content types organized within folder structures in Workbench, the recommended best-practice is to establish the same folder structure on the target server before using MSM to move any content types to the target server.

Note: this feature is automatically enabled in new installations.

5. Grouping of Templates

In the Workbench, developers can organize templates in folders so that end users can view these folders as submenus within the Active Assembly and the Preview action menu in the Content Explorer. Only templates visible in the logged in community will be displayed and by default, the submenus will reflect the exact folder structure created within the Workbench. There is an option that allows you to skip certain folders. See Implementation Document for details.



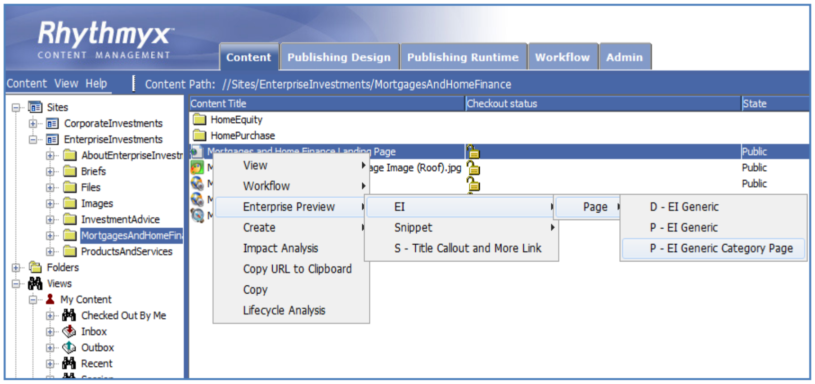


Figure 23 - Grouped Templates in Workbench

Figure 24 - Resulting submenus in Content Explorer

5.1 Creating Template Groups in Workbench

To organize templates within folders:

1. Login to the Workbench, and open the Assembly Design tab.
2. Expand the Shared node. Only folders in the Shared node (not in the Type Specific node) will be picked up for submenus in the Content Explorer.
3. Create desired sub-folders to organize templates. These folders will determine the sub menus that end-users will see within the Preview and Active Assembly action menus.

Templates that are not placed within sub-folders under the Shared node and templates in the Type Specific node will be displayed directly within the Preview and Active Assembly action menus. All templates and group names will be sorted alphabetically.

5.2 Configuring the Template Grouping Feature

Template Grouping can be configured using 2 new properties within server.properties located at ({root}\rxconfig\Server\server.properties).

1 To enable grouping, add a new property called ‘templatesGroupingInCX’ and set its value to true.

```
templatesGroupingInCX=true
```

Note: this feature is automatically enabled in new installations.

2 To enable skipping of multi-tiered folder structure where folders have only one sub-folder, add a new property called ‘skipFoldersIfOnlySingleSubMenuInCX’ and set its value to true. This will compress a highly tiered folder-structure where if only one sub-folder exists within a folder that folder it will be skipped when displaying it as an action menu. Note that the grouping property must be enabled in order for this to work.

```
skipEmptyFolderMenusInCx=true
```

By default, if this property is not set, the full folder structure will be shown in the Preview submenu.
To demonstrate how grouping of Templates behave, below is an example that explores the action menus generated in the content explorer depending on the folder structure in workbench, Template visibility by community, and whether the folder skipping property is enabled:

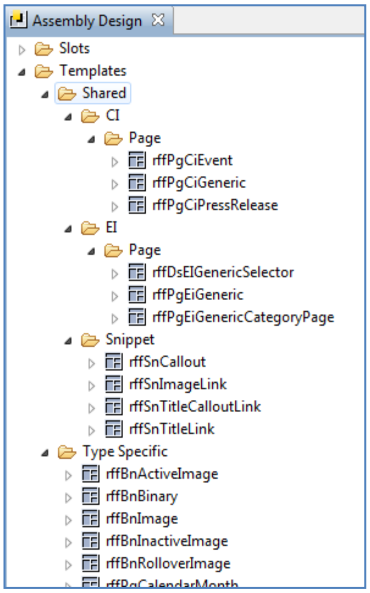


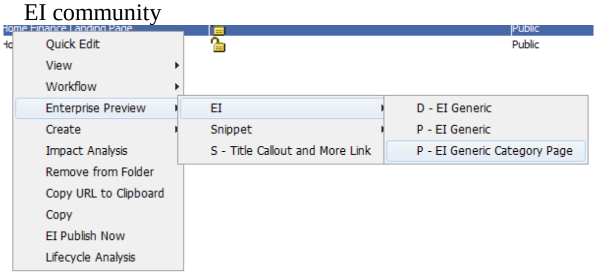
Figure 25 - Grouped Templates in Workbench

There are 3 communities with the following visibility settings:

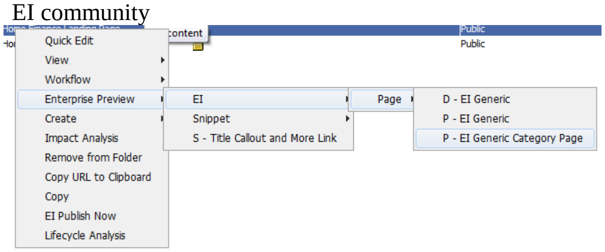
Community Name	Visible Templates
EI	All Templates in folder EI and Snippet Some templates in folder Type Specific
CI	All Templates in folders CI and Snippet Some templates in folder Type Specific
Admin	All Templates in folders Shared and Type Specific

Recorded below is the resulting submenu structure in each community

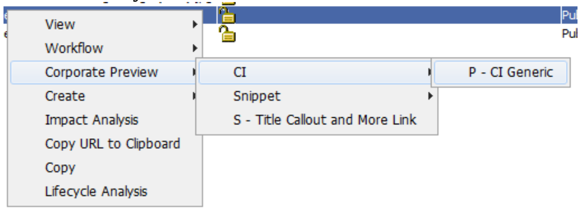
<table>
<tr><th>SkipFoldersIfOnlySubMenuInCX = true</th><th>SkipFoldersIfOnlySubmenuInCX = false</th></tr>
<tr><td>
EI community



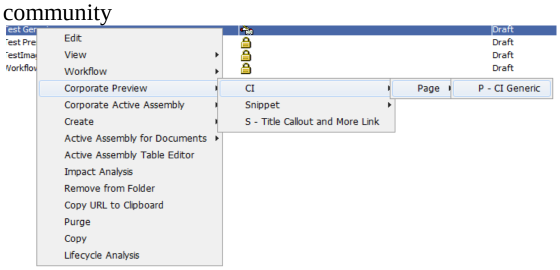
</td><td>
EI community



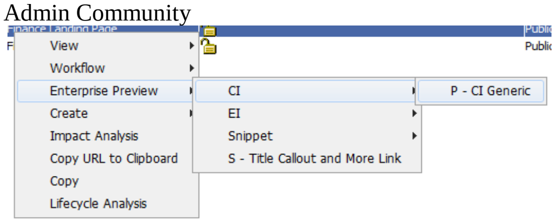
</td></tr>
<tr><td>
CI community



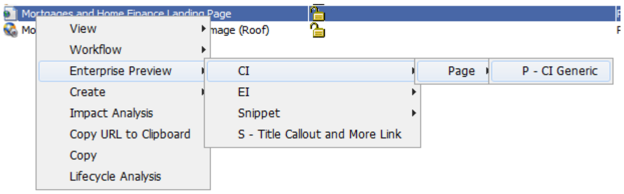
</td><td>
CI community



</td></tr>
<tr><td>
Admin Community



</td><td>
Admin community



</td></tr>
</table>

Note that the ‘skipFoldersIfOnlySingleSubMenuInCX’ property within server.properties is used for Grouping of Content Types and Templates. If this property is set to true, Content Types’ ‘New Item’ Action menu will behave similar to that of Templates. Refer to Grouping Content Types feature for details on the Content Types Grouping feature.

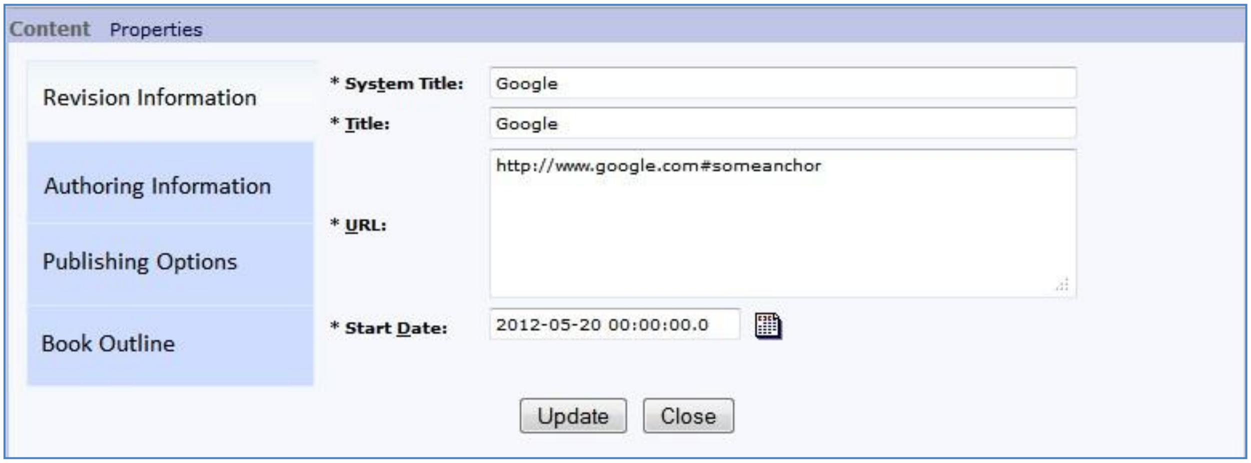
5.3 Additional Information

When using MSM, if the source server is enabled with grouping of templates feature and templates organized within folder structures in Workbench, then the recommended practice is to establish the same folder structure on the target server before using MSM to move any templates.

This feature cannot be implemented for type-specific templates because folders cannot be created within Type Specific templates in Workbench. As a result all type-specific templates available for a content item will be listed under Preview and Active Assembly without any cascaded menu or sub-menu.

6. Vertical Tab for Field Grouping

Developers can set a property that determines whether grouped fields are rendered to contributors as collapsible sections or vertical tabs when editing/creating items in the Content Explorer. Once this property is set, it will render the same for all items in the CMS in the edit, and view>content action menus.



* System Title: Google

* Title: Google

* URL: http://www.google.com#someanchor

* Start Date: 2012-05-20 00:00:00.0

6.1 Configuring field grouping property

Grouped field rendering can be configured in the server.properties file ({root}\rxconfig\Server\server.properties).

To render grouped fields as collapsible sections:

> Set the parameter `contentEditorGroupRenderingType` to `collapsible`
> `contentEditorGroupRenderingType =collapsible`

To render grouped fields as vertical tabs,

> Set the parameter `contentEditorGroupRenderingType` to `tabsLeft`
> `contentEditorGroupRenderingType =tabsLeft`

By default, if this property is not set, the grouped fields will render “collapsed.”

6.2 Additional Information

In the event of validation errors on submission, the label on every tab that has errors will be changed to indicate it contains 1 or more errors and the first tab that contains errors will be activated.
Refer to the CM System 7.1 release notes for more information on the Grouping of Fields feature.

7. Formatting in Help Text

In the Workbench, developers can add formatted text that contains various html tags, static images and static links to helptext associated to fields. The following tags are allowed within helptext:

- <h1>.. </h1>
- <h2>.. </h2>
- <h3>.. </h3>
- <p>.. </p>
- <b>.. </b>
- <i>.. </i>
- <u>.. </u>
- <a href=’’>.. </a>
- <img src=’’ />

7.1 Configuring Helptext bubble to accommodate HTML tags

The helptext feature has been extended to support some additional HTML tags including:

- <h1>.. </h1>
- <h2>.. </h2>
- <h3>.. </h3>
- <p>.. </p>
- <b>.. </b>
- <i>.. </i>
- <u>.. </u>
- <a href=’’>.. </a>
- <img src=’’ />

With the addition of image and list tags to the helptext, it is likely that they spill out of the helptext bubble. The size of the helptex bubble can be configured in the tooltip.css file ({root}\rx_resources\css\tooltip.css).

To configure the size of the helptext bubble:

1. Locate the “.tooltip”

2. Update the default width of the helptext bubble dialog to the desired width value and save

```
width: desired value here;
```

7.2 Additional Information

Refer to the CM System 7.1 Release Notes Guide for more information on the Contextual Helptext for Field feature.

8. Webdav for Content Types with hash fields

Content Types with hash fields can be imported into the CMS using webdav.

Webdav has been configured to accept Content Types with hash fields as well as out of box binary fields.

For Content Types whose binary fields have been replaced with hash fields, the Webdav configuration does not need to be changed. By default, Webdav first checks for binary {fieldname}. If it is no longer assigned to the content type, it checks instead for {fieldname}_hash. So, for example, for the Image Content Type, Webdav will first look for img1 field and if it is not found will look for img1_hash instead. If desired, the Webdav configuration can be updated to check directly for {fieldname} _hash.

When using hash fields, the size and type field mappings are no longer required in the Webdav configuration or on the content type. To be able to remove these fields, the Webdav configuration should be updated to directly specify the {fieldname}_hash. To continue to use these fields on the Content Type, they will be populated as before if the mappings exist.

For Content Types with both, binary as well as hash fields, the file will be uploaded to both. However, it is recommended that as soon as the de-duplication process has been run, the old, redundant binary fields be removed from the Content Types.