

Upgrading Rhythmyx 6.0/6.1 to Rhythmyx 6.5.2

System Requirements

Rhythmyx Version 6.5.2 has the same system requirements as Rhythmyx Version 6.0/6.1.

Upgrade to sys_EditLive Control

In Rhythmyx Version 6.5.2, the Ephox EditLive rich text editor is upgraded to EditLive Version 6.3. In Content Editors, users will notice that fields using the sys_EditLive control are not available until they click in the field.

The sys_EditLiveDynamic control is deprecated in this release. Any field using the sys_EditLiveDynamic control should be modified to use the sys_EditLive control.

The sys_EditLiveDynamic control should not be used in new Content Editor fields (although it is still listed as an available control for fields in the Workbench).

New Active Assembly Interface

Rhythmyx Version 6.5.2.2 introduces a new Active Assembly interface with enhanced usability. This functionality is automatically available for assembly using Velocity Templates. (XSL Variants must be converted to Velocity Templates before the new Active Assembly interface will be available.)

The new Active Assembly interface requires modifications to Velocity macros. Standard Velocity macros shipped with Rhythmyx will be converted automatically during the upgrade. Any custom macros you have implemented must be manually modified to enable the new Active Assembly interface functionality in those Templates. You must include the #startAA<object> and #endAA<object> macros appropriate to that type of object:

- Page
#startAAPage
#endAAPage
- Field
#startAAFfield
#endAAFfield
- Slot
#startAASlot
#endAASlot
- Snippet
#startAASnippet
#endAASnippet

For example, the #field macro released with Rhythmyx Version 6.0 used the following code:

```
#macro(field $fieldname)#__handlefieldaa($fieldname)
$sys.item.getProperty($fieldname).String#end
```

In Rhythmyx Version 6.5.2, the #field macro uses this code:

```
#macro(field $fieldname)#startAAfield($fieldname "")#if
($sys.part.render)$sys.item.getProperty($fieldname).String#end
#endAAfield($fieldname)#end
```

Updating Content Editor Extensions

Note: The following information is only applicable to systems that upgraded to Rhythmyx Version 6.0/6.1 before upgrading to Version 6.5.2. This information is not relevant to systems originally implemented on Rhythmyx Version 6.0/6.1.

In Rhythmyx Version 6.0, additional Java interfaces were added for extensions specific to Content Editors (e.g., Field Validations, Visibility Rules, Input and Output Transformers). For example, in Rhythmyx Version 5.x, a Field Validation was a UDF. In Rhythmyx Version 6.0 and later, a Field Validation is treated specifically as a Field Validation extension.

In Rhythmyx Version 6.0 and 6.1, on a Content Type editor dialog, both the Content Type extensions and UDFs were listed. In other word, the Field Validation editor listed both Field Validation extensions and UDFs.

In Version 6.5.2, the Rhythmyx Workbench is more strict. Only the correct Content Type extensions are listed. For backwards compatibility, a Workbench preference, **Show legacy interfaces for Content Type extensions**, was added. The preference allows the Version 6.5.2 Rhythmyx Workbench to display both Content Type extensions and UDFs as in Rhythmyx Version 6.0/6.1.

When upgrading from Rhythmyx Version 6.0/6.1, the preference is not checked. Thus, if you have a system that was upgraded from Rhythmyx Version 5.7 or earlier to Rhythmyx Version 6.0/6.1, then upgraded to Rhythmyx Version 6.5.2, the preference will not be checked. Some extensions that were previously available may seem to disappear or may appear to be registered incorrectly. Check the **Show legacy interfaces for Content Type extensions** preference manually to resolve this problem.

Extension code should be updated to use the correct Java interfaces:

Content Type Extension	Interface
Field Input Transformer	com.percussion.extension.IPSFieldInputTransformer
Field Output Transformer	com.percussion.extension.IPSFieldOutputTransformer
Field Read Only Rule	com.percussion.extension.IPSFieldEditabilityRule
Field Validations	com.percussion.extension.IPSFieldValidator
Field Visibility Rules	com.percussion.extension.IPSFieldVisibilityRule
Item Input Transformer	com.percussion.extension.IPSItemInputTransformer
Item Ouptut Transformer	om.percussion.extension.IPSItemOutputTransformer
Item Validation	com.percussion.extension.IPSItemValidator

After updating the extension code, you should uncheck the **Show legacy interfaces for Content Type extensions** preference.