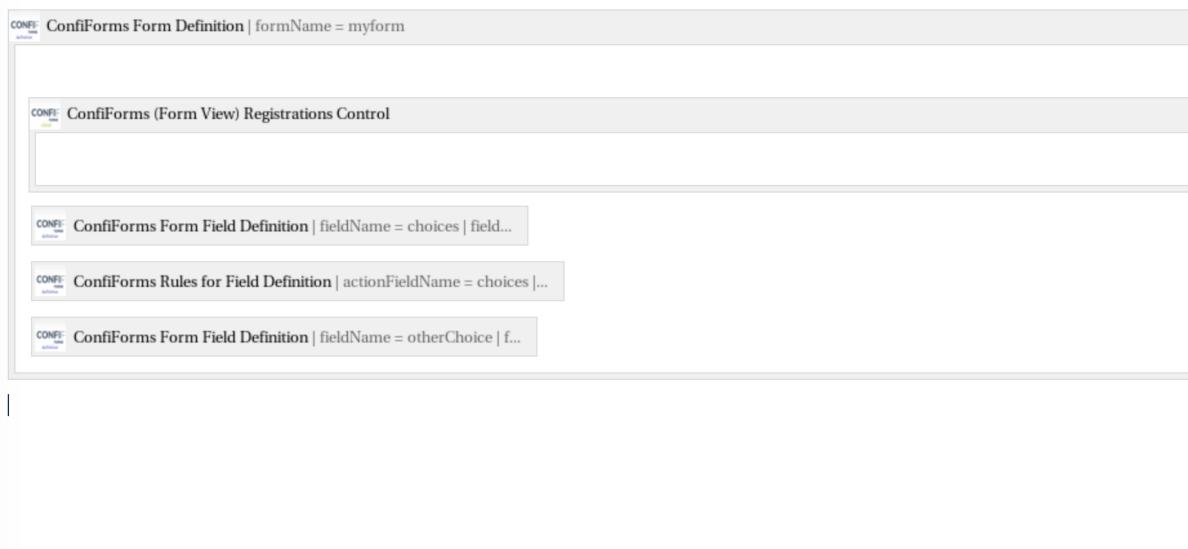


User macro to show how to make a ConfiForms dropdown field with possibility to choose other option

This is a quick demo on how you can reuse your ConfiForms configurations by creating user macros

Consider the following form, where the user can choose an option and if an "other" option is selected then the text field is shown to capture the value for the "other"



What we can do is to take out the 3 macros that define the fields and their behaviour and create a user macro that can be reused whenever needed

We take out the storage format for these 3 macro and create a "cf-choices" macro like this

Edit User Macro

Macro Browser Information

Macro Name* Macro names are converted to lower case

Visibility Visible to all users in the Macro Browser Visible only to system administrators in the Macro Browser

Macro Title*

Description

Categories Administration Communication Confluence content Development External content

Icon URL Absolute URL (e.g. <http://mysite.com/myimage.png>), or relative to the Confluence base URL (e.g. [/images/logo/confluence_48_white.png](#))

Documentation URL

Definition of User Macro

Macro Body No macro body
Processing Escaped
 Unrendered
You should use this option for bodies that are processed within the template before being output. Ensure that HTML is ultimately output by the template.
 Rendered
The body will be rendered so most HTML entered will be passed to the template unmodified but Confluence specific mark up such as macro definitions will be rendered.

Template*

```
<p>
<ac:structured-macro ac:macro-id="574fb50b-f05d-4fd6-bbaf-941028cdd456" ac:name="confiform-field-definition" ac:schema-version="1">
<ac:parameter ac:name="fieldName">choices</ac:parameter>
<ac:parameter ac:name="fieldLabel">Choices</ac:parameter>
<ac:parameter ac:name="values">false[1=one|2=two|3=three|4=other]</ac:parameter>
<ac:parameter ac:name="type">select</ac:parameter>
</ac:structured-macro>
</p>
<p>
<ac:structured-macro ac:macro-id="b355cde9-2e2c-4ffe-8c16-9d7a14c05c0e" ac:name="confiform-field-definition-rules" ac:schema-version="1">
<ac:parameter ac:name="condition">!choices:4</ac:parameter>
<ac:parameter ac:name="fieldName">otherChoice</ac:parameter>
<ac:parameter ac:name="action">Hide field</ac:parameter>
<ac:parameter ac:name="actionFieldName">choices</ac:parameter>
<ac:parameter ac:name="withReverseRule">true</ac:parameter>
</ac:structured-macro>
</p>
<p>
```

```
<ac:structured-macro ac:macro-id="574fb50b-f05d-4fd6-bbaf-941028cdd456" ac:name="confiform-field-
definition" ac:schema-version="1">
<ac:parameter ac:name="fieldName">choices</ac:parameter>
<ac:parameter ac:name="fieldLabel">Choices</ac:parameter>
<ac:parameter ac:name="values">false[1=one|2=two|3=three|4=other]</ac:parameter>
<ac:parameter ac:name="type">select</ac:parameter>
</ac:structured-macro>
</p>
<p>
<ac:structured-macro ac:macro-id="b355cde9-2e2c-4ffe-8c16-9d7a14c05c0e" ac:name="confiform-field-
definition-rules" ac:schema-version="1">
<ac:parameter ac:name="condition">!choices:4</ac:parameter>
<ac:parameter ac:name="fieldName">otherChoice</ac:parameter>
<ac:parameter ac:name="action">Hide field</ac:parameter>
<ac:parameter ac:name="actionFieldName">choices</ac:parameter>
<ac:parameter ac:name="withReverseRule">true</ac:parameter>
</ac:structured-macro>
</p>
<p>
```

So, in the end the macro configuration for the user looks like this



See the online demo below