


# Using Lookup and set a DB value action

In ConfiForms we have a number of integration points where you can supply the data from your database tables or views directly to ConfiForms.

One of such integrations is using the ConfiForms Field Definition Rules action called "Lookup and set a DB value"

 The table we use for the demo is with following structure

id	name	link
1	ConfiForms add-on for Confluence	<a href="https://marketplace.atlassian.com/plugins/com.vertuna.confluence.plugins.confiform/server/overview">https://marketplace.atlassian.com/plugins/com.vertuna.confluence.plugins.confiform/server/overview</a>
3	Smart Templates & Pages nice app	<a href="https://marketplace.atlassian.com/plugins/com.vertuna.confluence.plugins.smart-templates/cloud/overview">https://marketplace.atlassian.com/plugins/com.vertuna.confluence.plugins.smart-templates/cloud/overview</a>
2	ConfiDoc add-on for Confluence	<a href="https://marketplace.atlassian.com/plugins/com.vertuna.confluence.plugins.confirest/server/overview">https://marketplace.atlassian.com/plugins/com.vertuna.confluence.plugins.confirest/server/overview</a>

Here is an example on how to use it. We will have a simple form which will lookup the plugin link from a database table based on user selection and will set it to the text field (actually a field type called "link" to make the value clickable, but the techniques are the same)

CONFI

ConfiForms Form | formName = f

CONFI

ConfiForms Registration Control

CONFI

ConfiForms Field Definition | fieldName = choice | fieldL...

CONFI

ConfiForms Field Definition | fieldName = link | fieldLab...

The field "choice" is of type "db dropdown" and link is of type "link"

The form looks like this

## Registration

Select plugin

ConfiForms add-on for Confluence ▾

Link

<https://marketplace.atlassian.com/plugins/com.vertuna.cor>

+

Save Close

We will add a ConfiForms Field Definition Rules macro to the game

## Edit 'ConfiForms Field Definition Rules' Macro

Rules for ConfiForms Field  
Definitions [Documentation](#)

Field name

choice

Name of the field you want to track  
(optional for validation rules and for rules  
which should be executed ONLY on users  
action)

Condition

Action will be executed when condition is  
met. [Same syntax expected as in filters](#)  
The scope is current entry/record (for  
dataset validation the scope is all records).  
You can reference current records fields  
via [entry.field\_name]

☒ Execute only on user action

When checked, the rule will be  
executed ONLY on user action (not

↻ Preview

Select macro

Save Cancel

## Edit 'ConfiForms Field Definition Rules' Macro

Lookup and set DB value ▾

Choose Action to perform

Values to set

link=[entry.pluginlink]

Context is the record you have LOOKED UP. As key=value pairs, separated by &. Where key is the field name in the form where this entry should be created and value is the value to be set. You can reference record owner as [owner]; record modifier, as [modifier]; any record field as [entry.FIELD\_NAME]

SQL SELECT query

select link as "pluginlink" from cd\_c

Context is CURRENT record. You can reference record owner as [owner]; record modifier, as [modifier]; any record field as [entry.FIELD\_NAME]

Preview

Select macro

Save

Cancel

In storage format it looks like this:

```
<ac:structured-macro ac:macro-id="4b605c7d-bcdb-429d-874b-6a6817e17a79" ac:name="conform-field-definition-rules" ac:schema-version="1">
  <ac:parameter ac:name="fieldName">link=[entry.pluginlink]</ac:parameter>
  <ac:parameter ac:name="values">select link as "pluginlink" from cd_demo where id = [entry.choice.id]</ac:parameter>
  <ac:parameter ac:name="action">Lookup and set DB value</ac:parameter>
  <ac:parameter ac:name="onUserActionOnly">true</ac:parameter>
  <ac:parameter ac:name="actionFieldName">choice</ac:parameter>
  <ac:parameter ac:name="values2">7422036e-9a6f-4455-92d9-7140131fb9da</ac:parameter>
</ac:structured-macro>
```

So what we do here...

1. we set the rule to be triggered only when the value in field "choice" changes.
2. we set it to execute only on user action
3. we configure it to set the value of the result into a field called link

link=[entry.pluginlink]

4. Important note is what is this "pluginlink".. it is actually the column returned by the SQL lookup we configured (see the r)

select link as "pluginlink" from cd\_demo where id = [entry.choice.id]

5. SQL uses the value from a selected dropdown (it is a dropdown, so we need an ID. See [Accessing field values and properties](#) for more details)
6. When SQL query is executed we get the results back and use the value of a column "pluginlink" to set back to ConfiForms field link