

# Adding Workflow Building Block to Custom Planning Reports

Return to [Workflow Building Block Overview](#)

The Workflow building block offers workflow features to any arbitrary custom planning report implemented in the Jedox Report Designer.

Configuring the report to execute a workflow activity enables the end user to open it from [My Workflow Activities](#), [see full article](#). The report then automatically adapts to the workflow coordinates of the selected workflow activity. When the user changes the PoV (point of view) options, the report checks if they conform to a workflow step of a running workflow task. If so, the reports display the following workflow information:

- Current activity
- Whether the report is protected against data modification due to workflow rules
- The user groups to which the workflow activity has been assigned (in the cases where the workflow activity has been assigned to user groups other than those of the current user).
- The [Workflow Details](#) button, through which the user can inspect the details of the workflow step and proceed with the workflow.

This article describes the procedures of adding certain features to a custom planning report. We make some assumptions of your report:

- uses a range named `rngDB` (for the connection and database string)
- uses a range named `rngCube` (for the fact cube for accessing data)
- has a minimum of three combo boxes for dimensions of the cube. These three combo boxes would be used as workflow coordinates (one for each cube dimension that the workflow uses: `Version`, `Time` and `Workflowstep` dimensions).
- **Please note:** the workflow only requires these dimension *types*, but does not assume dimension names, element names, or elemental hierarchy, see [Workflow in Custom Applications](#).
- it does not use any PHP macros

If these assumptions do not apply to your report, you may need to modify these steps.

## Instructions to Add Workflow Behaviour to Custom Reports

The Workflow building block comes with the `Custom Budget.wss` report. It serves as an example of a planning report with workflow features. We also use it in the following instructions.

1.) Copy PHP macros from the `Custom Budget` report to your custom planning report

Most of the PHP macros can be copied directly from the Maco Editor, but you will want to replace the following three with those belonging to your own custom planning report:

```
[crayon-5d12f9d07b9f2336196558/]
```

Be sure to enter `updateWorkflowFromPOV();` in each macro you replace.

2.) Replace your Parameters worksheet with the one from the Custom Budget report

- Open both reports.
- In Report Designer, go to the `Parameters` worksheet in the `Custom Budget` report. Right-click the parameters tab and select `Move or copy...` from the context menu. Select `(move to end)` from the list in the popup window, tick the `Create a copy` check box, and click OK.
- Remove the `Parameter` worksheets from your custom

planning report.

- Rename `Parameters (2)` to `Parameters`.
- Open `Insert` → `Name` and fix the ranges with broken names:

Name	Refers to
rngAssignedToOthers	Parameters!\$B\$36
rngCube	Parameters!\$B\$3
rngCurrentActivityName	Parameters!\$B\$32
rngDB	Parameters!\$B\$2
rngLabelRegion	Parameters!\$B\$8
rngLabelVersion	Parameters!\$B\$9
rngLanguage	Parameters!\$B\$4
rngReadWrite	Parameters!\$B\$10
rngStepDimension	Parameters!\$B\$20
rngSubtitle	Parameters!\$B\$7
rngTime	Parameters!\$B\$14
rngTimeDimension	Parameters!\$B\$19
rngTitle	Parameters!\$B\$6
rngVersion	Parameters!\$B\$13
rngVersionDimension	Parameters!\$B\$18
rngWorkflowActivityAssignments	Parameters!\$B\$33
rngWorkflowActivityIsActive	Parameters!\$B\$35
rngWorkflowActivityIsReadwrite	Parameters!\$B\$34
rngWorkflowAssignedToMeAndReadwrite	Parameters!\$B\$4
rngWorkflowCurrentActivity	Parameter!\$B\$31
rngWorkflowDisplay	Parameters!\$B\$37
rngWorkflowReadwrite	Parameters!\$B\$4
rngWorkflowStep	Parameters!\$B\$15
rngWorkflowTaskAttributeCube	Parameters!\$B\$23
rngWorkflowTaskAttributeStepDimension	Parameters!\$B\$24
rngWorkflowTaskAttributeTime	Parameters!\$B\$28
rngWorkflowTaskAttributeTimeDimension	Parameters!\$B\$27
rngWorkflowTaskAttributeVersion	Parameters!\$B\$26
rngWorkflowTaskAttributeVersionDimension	Parameters!\$B\$28

You may wish to use the ranges `rngLanguage`, `rngTitle`, `rngSubtitle`, `rngLabelRegion`, `rngLabelVersion`, but they aren't mandatory. Use `rngReadWrite` to determine the general read-write-behaviour in your custom planning report. The formula that comes with the sample only uses the workflow related state

information `rngWorkflowActivityIsReadWrite`. You may also wish to add custom criteria to turn your report to read-only mode for other reasons.

The workflow functionality on the report understands the three workflow coordinates for version, time and workflow step, but it is not aware how these are presented to the user. If the report uses combo boxes on cube dimensions to offer the coordinates to the user, the selected elements must be translated to a form which the workflow implementation can understand. The selected values have to be placed in three named cells in the parameter sheet. When the workflow macros are executed, they expect the currently selected elements in these three named cells.

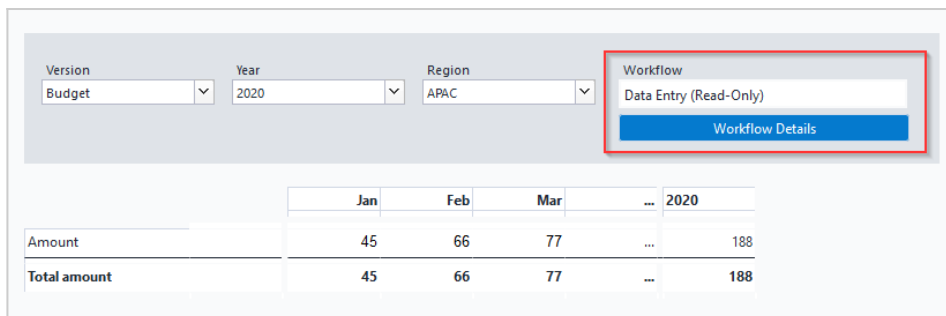
Connect your three combo boxes to their respective Cell/Range:

cmbVersion	rngVersion	Parameters!\$B\$13
cmbMonth	rngTime	Parameters!\$B\$14
cmbRegion	rngWorkflowStep	Parameters!\$B\$15

The workflow implementation will only work if the dimensions used by the report are in line with the dimensions used by the workflow step. To verify this, the names of the three dimension must be stored in another three named cells. The macros will only activate the workflow behaviour if these dimensions match the dimensions used in the definition of the workflow task. Declare the name of the three dimensions used by your report in the `rngVersionDimension`, the `rngTimeDimension`, and the `rngWorkflowStepDimension`.

### 3. Add the Indication of the Workflow

If the selection within a custom planning report matches the coordinates of a workflow task, this activates the workflow behaviour. In addition to activation, the report should also indicate information about the matching workflow step. Add the following three form elements to your report:



	Jan	Feb	Mar	...	2020
Amount	45	66	77	...	188
<b>Total amount</b>	<b>45</b>	<b>66</b>	<b>77</b>	<b>...</b>	<b>188</b>

Type	Name	Other Properties	Description	Implementation
Label	WorkflowFlow	Label Text = Text: <a href="#">Workflow</a>		
Label	WorkflowReadOnly	Label Text = Formula: <a href="#">constWorkflowStep</a> Visibility = <a href="#">constWorkflowStep</a>	Indicates an active workflow text by showing the current activity. If the report is in a read-only state due to the workflow state then "Read Only" is appended in parentheses. If the activity is not assigned to the current user, but to other user groups, these are appended in parentheses. If the workflow is not active the label is blank.	Text is taken from the named cell <a href="#">constWorkflowStep</a> on the <a href="#">Worksheet</a> . This cell is protected by a cell formula using other named cells and their macros.
Button	WorkflowDetails	Button Label: <a href="#">Workflow Details</a> Visibility = <a href="#">constWorkflowStep</a>	Shows a button to inspect details of the workflow step and to proceed with the workflow step. If the workflow is not active this button is not shown.	Opens the report <a href="#">Reports/Detail/Workflow Details Frameset</a> in a pop-up dialog.

To enable the workflow details report add a cell formula to a hidden cell (e.g. cell [A8](#)) on your report with a hyperlink to the [Workflow Detail](#) report as follows:

```
=HYPERLINK("../Reports/Detail/Workflow Detail Frameset#_new,top=50,left=50,width=690,height=650","Workflow Details","Workflow Details"&"#simple",constWorkflowTask,"@varWorkflowTask",constWorkflowStep,"@varWorkflowStep",0,"@superuser"
```

Mode" )

Modify the PHP macro to match your cell reference.

[crayon-5d12f9d07b9fd311576019/]

#### 4. Protect your Input Fields if Required by the Workflow

Add a new conditional formatting rule as follows:

Select rule type: Use a formula to determine which cell format

Format values where this formula is true: `=rngReadWrite`

- Fill → Control Background = `#F5F5F8` (or any color to show input fields in your applications)
- Protection → Locked = False

Applies to input fields (use a cell expression to cover all input fields).

They can include one or multiple comma separated ranges with multiple columns and rows e.g. `$B$4, $B$6:$D$6, $B8:$D14`.

## Opening a Planning Report for a Specific Workflow Step

Users can open the report for a specific workflow step, automatically changing the form elements (e.g. combo boxes) to the workflow step coordinates. This is implemented in teh `My Workflow Activities`

report. It is done with a HYPERLINK function transferring additional values to the named constants `constWorkflowTask` and `constWorkflowStep` while opening the report.

After that the report would:

1. Execute the `updatePOVFromWorkflow()` function in the `_load()` event. This will consume the two named constants above and (after some validation has been passed) set the named ranges `rngVersion`, `rngTime` and `rngWorkflowStep`.
2. The named ranges `rngWorkflowCurrentActivity`, `rngCurrentActivityName`, `rngWorkflowActivityAssignments`, `rngWorkflowActivityIsReadwrite`, `rngWorkflowActivityIsActive`, `rngAssignedToOthers` and `rngWorkflowDisplay` will be updated.
3. The workflow indication and protection of input cells, which refer to the above named ranges, will be updated in turn.

Please see the `updateWorkflowFromPOV()` function with its comments in the macros for more details.

---