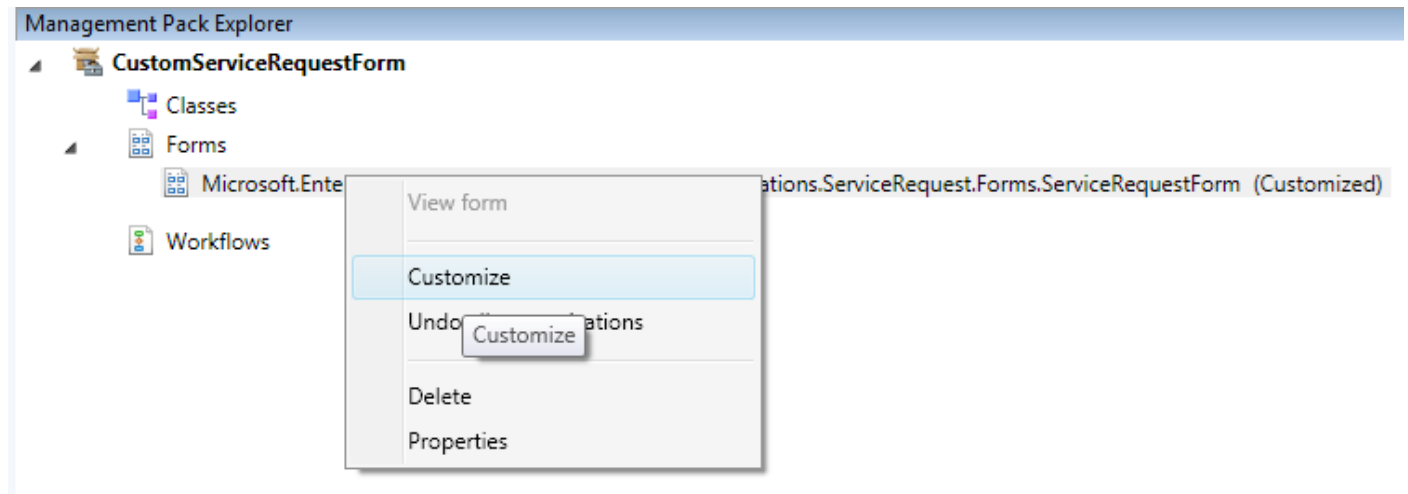


How to add Time Tracker to an existing form customisation Management Pack

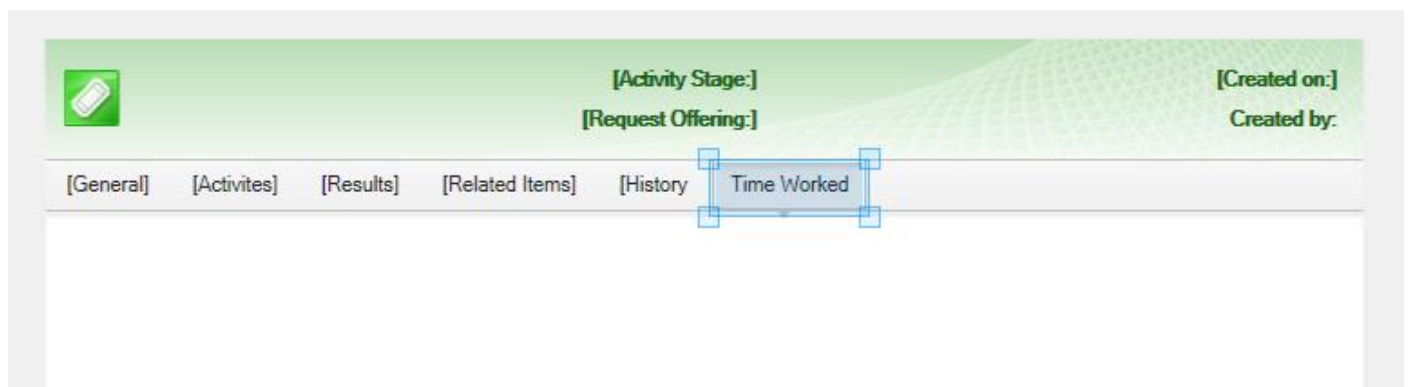
You will need:

1. Your existing unsealed form customisation Management Pack
2. The SNK file used to seal this Management Pack
3. The SCSM 2012 Authoring Tool

Open your unsealed form customisation Management Pack in the Authoring Tool and under “Forms” right-click your custom form and select “Customise”.

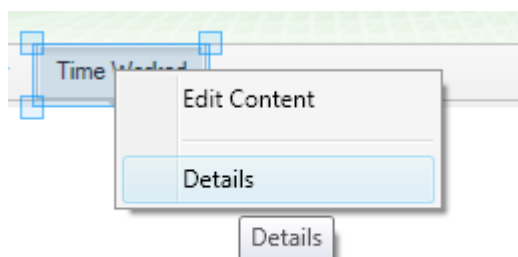


Drag and drop a new Tab Item onto your form after the “History” tab item:



Note that it is **not** possible to move this new tab so that it appears before the “History” tab.

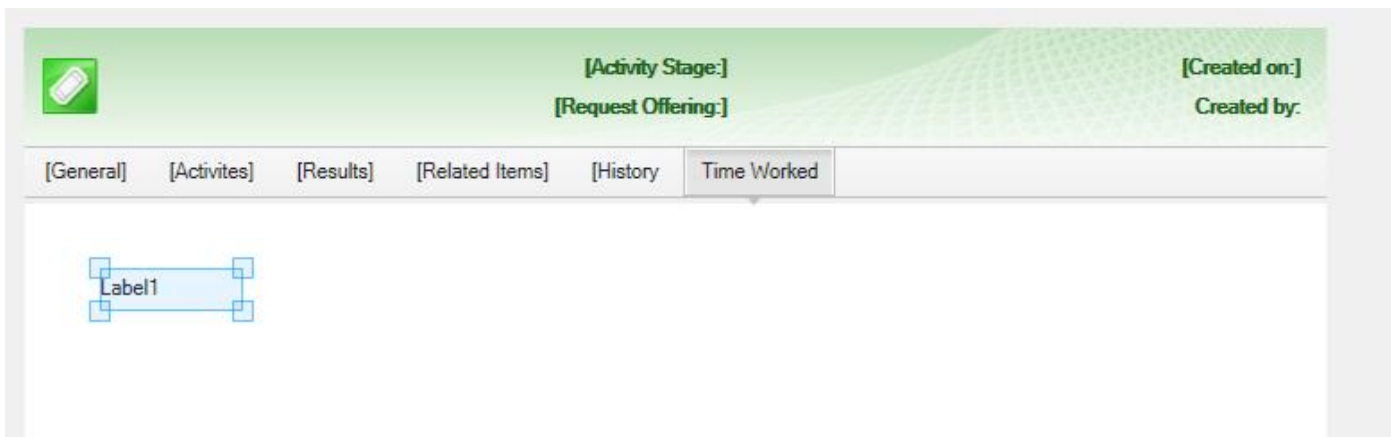
You can change the name of the tab item by right-clicking it and selecting “Details”:



Then change the “Content” property to something suitable:

Name	TabItem_1
Text	
Content	Time Worked

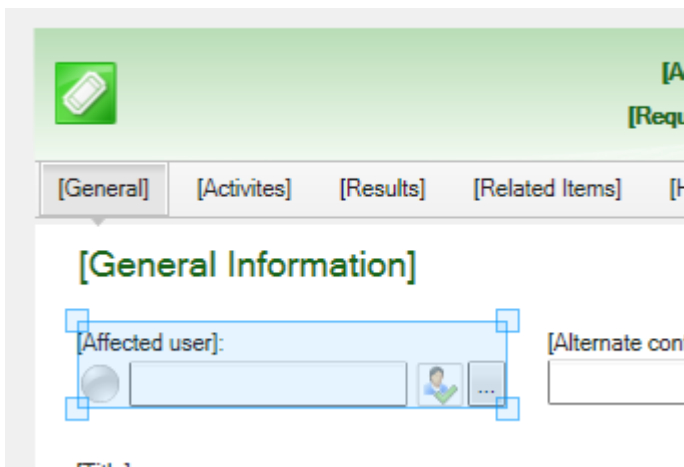
Select “View\Form Customisation Toolbox” and drag and drop a label onto your new tab:



This is only a place holder, it does not matter where you put it and it will be removed shortly.

Next, SCSM will default to the tab last edited using the Authoring Tool, so now you must edit the General tab:

Select “Affected User”:



Right-click and select Details:

Width	Auto
Margin	
Bottom	0
Left	0
Right	0
Top	1
Miscellaneous	
Flow Direction	Left to right

Change the “Top” margin to 1, then to 0 again (or whatever the original value was). This ensures that the General tab will be displayed first when you edit a Work Item within the SCSM console using this form.

Save your Management Pack and exit the Authoring Tool.

Now open your Management Pack in any XML editor such as Visual Studio and search for “</Customization>” to find the end of your custom form. Your recently made customisations will be here.

Find the “AddControl” block for the label you just added, for example:

```
<AddControl Parent="Grid_1" Assembly="PresentationFramework, Version=3.0.0.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35" Type="System.Windows.Controls.Label" Left="84.5" Top="41"
Right="563" Bottom="847" Row="0" Column="0" />
```

Note that depending on the form being customised, the values for “Parent” will be different. Change the “Assembly” value to:

```
Assembly="Cireson.TimeTracker, Version=7.5.0.0, Culture=neutral, PublicKeyToken=98ba2176e2a9efbc"
```

Change the “Type” value to:

```
Type="Cireson.TimeTracker.Controls.TimeTrackerFormControl"
```

If you made any changes to the label (such as moving it), you will need to remove them before proceeding.

Scroll to the top of your “<Customization>” section. Just above that is the “<Form ID=” line. Look at the value for the “Target”. This is the Type Projection that your custom form uses to display relationship data.

For example:

```
<Forms>
  <Form ID="CustomForm_55e37944_4d62_4c0a_b8c0_984e3c98de40" Accessibility="Public"
  Target="CustomForm_55e37944_4d62_4c0a_b8c0_984e3c98de40_TypeProjection"
```

Copy the value for “Target”.

Scroll to the top of your XML. Search for the copied value.

You should find the Type Projection definition:

```
<TypeProjections>
  <TypeProjection ID="CustomForm_55e37944_4d62_4c0a_b8c0_984e3c98de40_TypeProjection"
  <Component Path="$Context/Path[Relationship='Alias_af535d94_6959_4f7e_96b6_ccd852:
  <Component Path="$Context/Path[Relationship='Alias_af535d94_6959_4f7e_96b6_ccd852:
  <Component Path="$Context/Path[Relationship='Alias_af535d94_6959_4f7e_96b6_ccd852:
```

Scroll to end of this Type Projection and after the last “<Component>” line but before the “</TypeProjection>” line add a new component of:

```
<Component Path="$Target/Path[Relationship='WI!System.WorkItemHasBillableTime']$"
Alias="BillableTime">
  <Component
  Path="$Target/Path[Relationship='WI!System.WorkItem.BillableTimeHasWorkingUser']$"
  Alias="WorkingUser"/>
</Component>
```

Do not change the values for “Alias”.

For example:

```

<Component Path= $Context/Path[Relationship= Alias_af535d94_6959_4f7e_96b6_ccd852262ab7!System.WorkItemHasCommentLog '!
<Component Path="$Target/Path[Relationship='WI!System.WorkItemHasBillableTime']$" Alias="BillableTime">
  <Component Path="$Target/Path[Relationship='WI!System.WorkItem.BillableTimeHasWorkingUser']$" Alias="WorkingUser"/>
</Component>
</TypeProjection>
</TypeProjections>
</EntityType>

```

The Alias here of “WI” must be changed to match your Management Pack. In the “<References>” section at the start of the XML find the reference for “System.WorkItem.Library”:

```

<Reference Alias="Alias_af535d94_6959_4f7e_96b6_ccd852262ab7">
  <ID>System.WorkItem.Library</ID>
  <Version>7.5.2905.0</Version>
  <PublicKeyToken>31bf3856ad364e35</PublicKeyToken>
</Reference>

```

Copy the value for “Alias” and replace “WI” in your component with it (leave the “!” after the alias”):

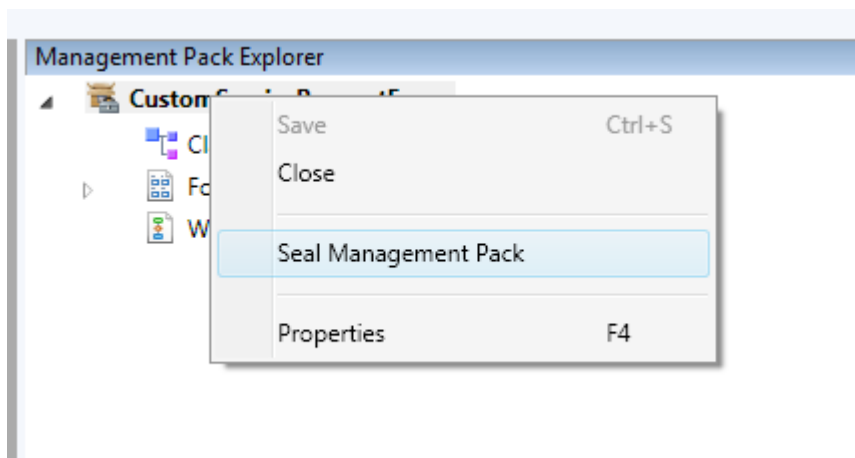
```

<Component Path="$Target/Path[Relationship='Alias_af535d94_6959_4f7e_96b6_ccd852262ab7!System
  <Component Path="$Target/Path[Relationship='Alias_af535d94_6959_4f7e_96b6_ccd852262ab7!System
</Component>
</TypeProjection>

```

Save and close your XML and open it in the Authoring Tool once more.

Seal your Management Pack using the same SNK it was originally sealed with:



Complete the "Seal Management Pack" dialogue and click "Seal":

Seal Management Pack

A key file (.snk) and a company name are required to seal a management pack. The key file, which contains a public-private key pair, is used to create a strong name signature for the management pack. Sealing a management pack allows other management packs to reference objects within the sealed management pack. Sealed versions of a management pack cannot be modified or unsealed.

Management Pack:
CustomServiceRequestForm

Output Directory:
C:\mps\TimeTracker

Key File:
C:\MPs\TimeTracker\seal.snk

Company:
Your Company Name

Copyright:

Lastly, import your new sealed Management Pack (.mp) into SCSM using the console under "Administration\Management Packs". Restart your SCSM console.

Time Tracker will now appear on your new tab for the class you have customised:

General Activities Results Related Items Service Level History Time Worked

Time Worked

Hours 0 Minutes Add Subtract Add Timer

Analyst	Last Updated	Time Worked
---------	--------------	-------------

00:20 Total Time: 0 Minutes