

Getting started with TFS Event Workflows

How to create a custom workflow in Visual Studio 2010 and install it on the Team Foundation Server

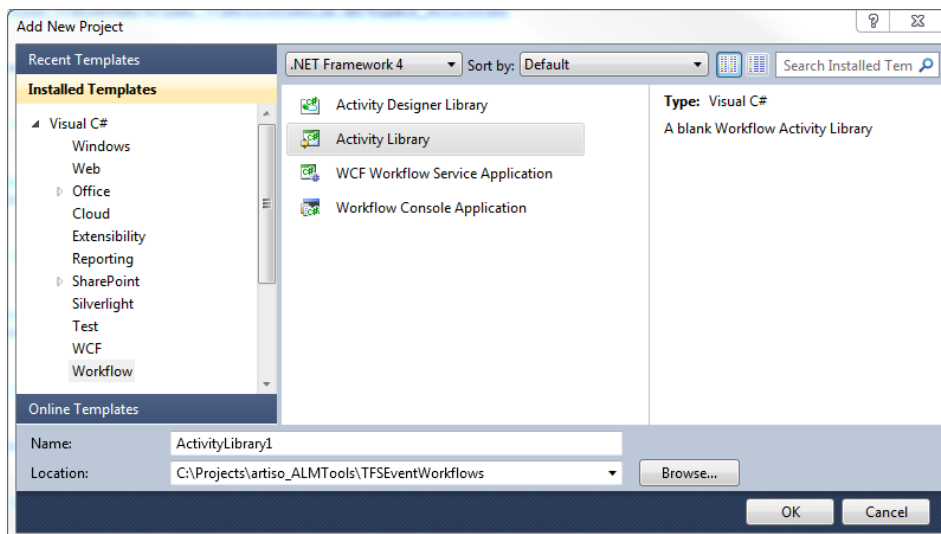
This sample shows how to execute a workflow when a work item is saved.

First we create a workflow which handles the WorkitemChangedEvent. The workflow will add a new work item of the type task to a new user story. So each time a new user story is created by a user a new task will be added as child.

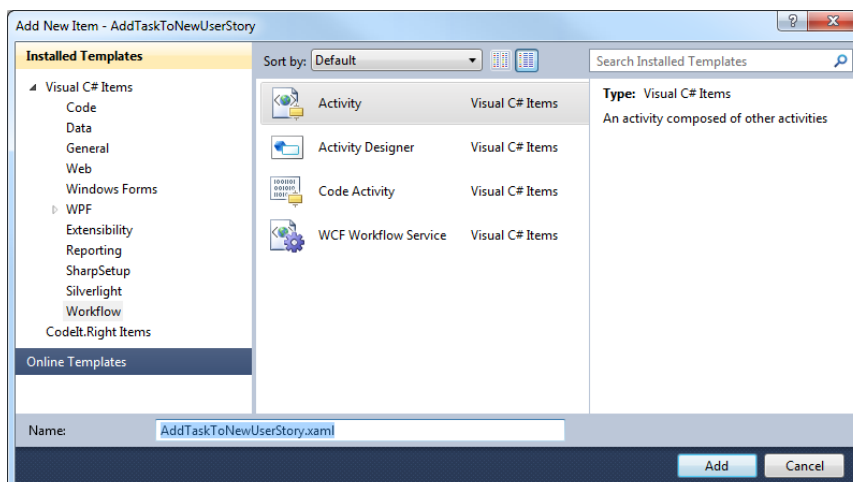
1. Creating a custom workflow with the TFSActivitiesLib

First step is to create the workflow.

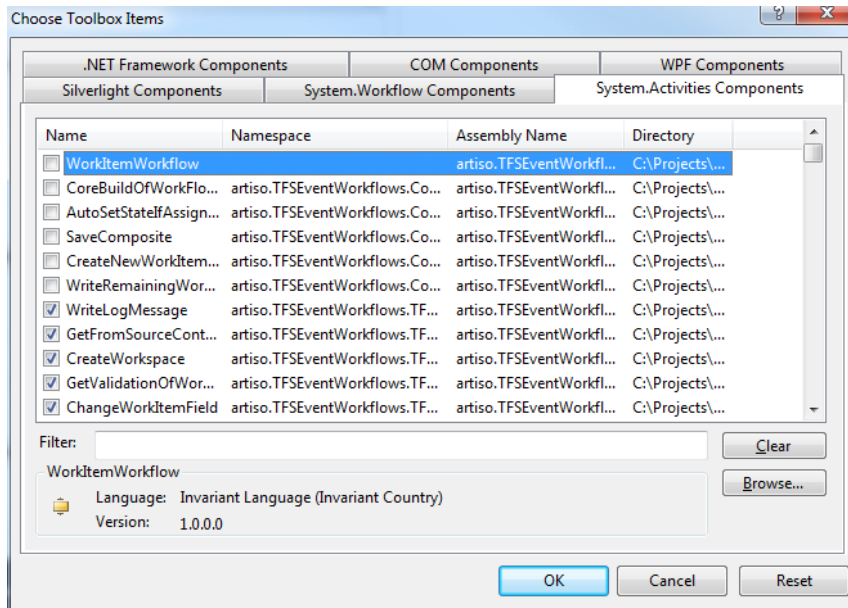
1. Create a new solution. Add a new "Activity Library" project from the "Workflow" template.



2. Add a new workflow activity with the name AddTaskToNewUserStory.xaml to the project.



3. Add a reference to the TFSActivitiesLib (artiso.TFSEventWorkflows.TFSActivitiesLib.dll).
4. Open the AddTaskToNewUserStory.xaml editor and add the TFS Event Workflows activities to the toolbox. Right-click on the Toolbox and select "Add Tab". Type in a name for the tab like "TFSEventWorkflows". Right-click on the new tab and select "Choose Items..."



5. In the "Choose Toolbox Items" window click "Browse..." and select the artiso.TFSEventWorkflows.TFSActivitiesLib.dll.
6. The activities from the TFSActivitiesLib are in the Toolbox and can be dragged into the workflow designer.

Now we can design the workflow.

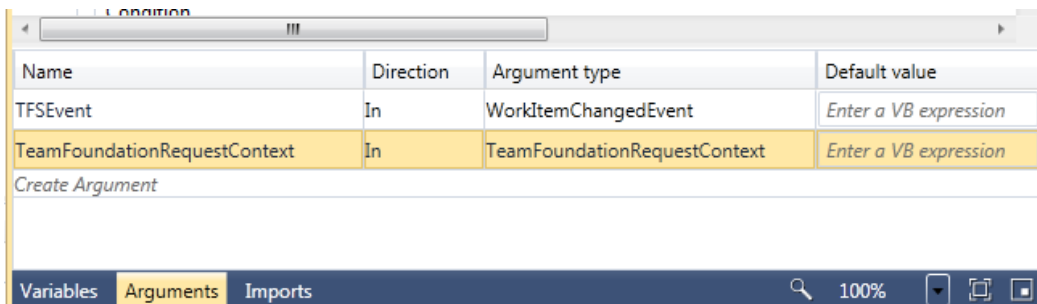
7. First we need to define the InArguments for the workflow. A WorkItemChangedEvent workflow takes two arguments, the WorkItemChangedEvent and TeamFoundationRequestContext.

Add a reference to:

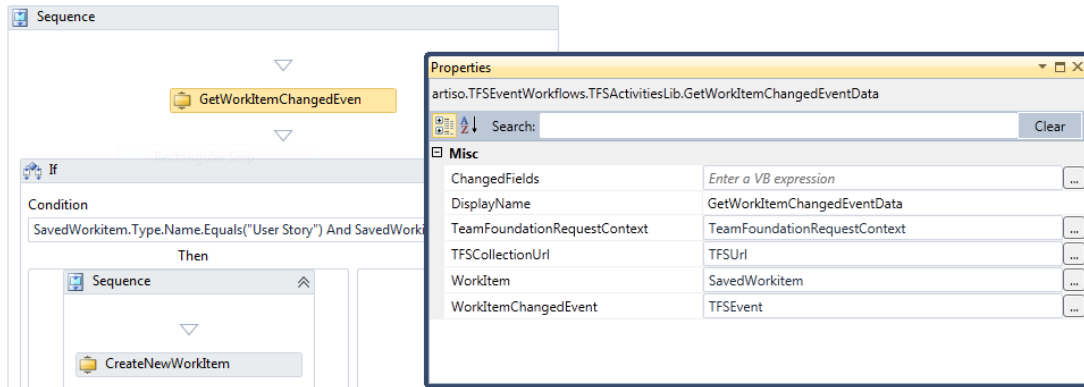
Microsoft.TeamFoundation.WorkItemTracking.Server.Dataaccesslayer.dll

Microsoft.TeamFoundation.Framework.Server.dll

Microsoft.TeamFoundation.WorkItemTracking.Client



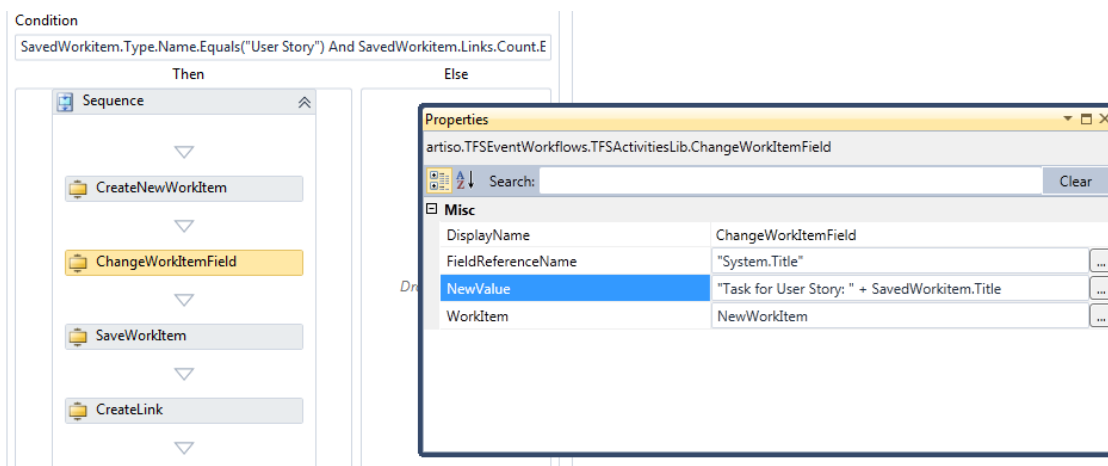
8. To process the InArguments we drag the activity GetWorkitemChangedEventData to the workflow. The activity takes the InArguments and returns the saved work item, the TFS collection URL and the changed fields of the work item.



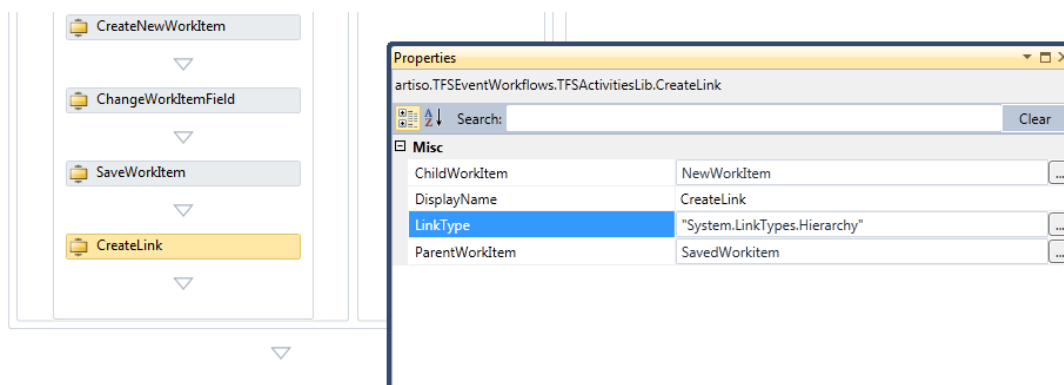
9. After we get the saved workitem from the GetWorkitemChangedEventData activity we check if the work item type is a "User Story" and if the revision of the work item is 1, which means the work item is new.

`SavedWorkitem.Type.Name.Equals("User Story") And SavedWorkitem.Revision.Equals(1)`

10. If the condition is true we create a new work item with the CreateNewWorkItem activity, change the title of the new work item with the ChangeWorkItemField activity and save the new work item.



11. After the save we create a hierarchy link between the user story (parent) and the new task (child) with the CreateLink activity.



2. How to install the TFS Event Workflows on the TFS and get the WorkItemChangedEvent workflow running

1. Open the TFS Event Workflows project, build the solution and copy the assemblies from the output to the plugins folder of the TFS.

Copy all files to both directories:

...\Microsoft Team Foundation Server 2010\Application Tier\Web Services\bin\Plugins

...\Microsoft Team Foundation Server 2010\Application Tier\TFSJobAgent\plugins

Required assemblies:

- artiso.TFSEventWorkflows.LoggingLib.dll
- artiso.TFSEventWorkflows.TFSActivitiesLib.dll
- artiso.TFSEventWorkflows.TFSEventWorkflowsServerPlugin.dll
- artiso.TFSEventWorkflows.TFSEventWorkflowsServerPlugin.dll.config
- log4net.dll
- MyWorkflowFile.xaml

TFS Event Workflows Setup:

The TFS Event Workflow can also be installed by executing the "TFSEventWorkflowsSetup.msi". Set the installation path to the plugins folder of the TFS.

(...\Microsoft Team Foundation Server 2010\Application Tier).

2. Copy the workflow file AddTaskToNewUserStory.xaml to the plugins directory.
3. To subscribe to the WorkItemChangedEvent must be configured in the configuration file (artiso.TFSEventWorkflows.TFSEventWorkflowsServerPlugin.dll.config).

In the tfsEventConfig section the following values must be set:

name:

A name for the entry. Must be unique.

fullTypeName:

The full name of the TFS event.

eventAssemblyName:

The full name of the assembly which contains the TFS event.

workflowFileName:

The path of the workflow file which will be executed with the event. If the workflow file is in the same directory as the server plugin dll the workflow file name is sufficient (e.g. AddTaskToNewUserStory.xaml).

```
<tfsEventConfig>
  <tfsEvents>

    <add name="workItemChangedEvent1"
      fullTypeName="Microsoft.TeamFoundation.WorkItemTracking.Server.WorkItemChangedEvent"
      eventAssemblyName="Microsoft.TeamFoundation.WorkItemTracking.Server.Dataaccesslayer"
      workflowFileName="AddTaskToNewUserStory.xaml" />

  </tfsEvents>
</tfsEventConfig>
```

4. Save the configuration file and create a new user story.