# Microsoft Dynamics CRM SDK

## ServiceBus Custom Workflow Activity Readme

## This folder contains sample code demonstrating how to write a Microsoft Dynamics CRM 2015 Workflow which posts its context to the Windows Azure Service Bus.

## Instructions

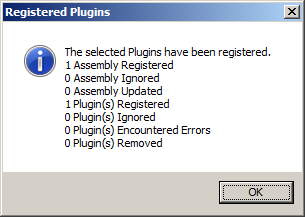
## Service Bus Custom Workflow Activity Project

## Prerequisite: Complete the REST Listener or Two-Way Listener samples as one of these will be needed to verify this custom workflow is working.

## How to Install, Build, and Run the Sample

1. In Windows Explorer, launch Microsoft Visual Studio 2012 or 2013.
2. Open the WindowsAzure solution.
3. Compile the library by pressing Ctrl+ Shift + b

## Configuring Microsoft Dynamics CRM to post to Windows Azure Service Bus on creation of an account

1. Using the PluginRegistration Tool in the Microsoft Dynamics CRM SDK, connect to the Microsoft Dynamics CRM organization which will post to the Windows Azure Service Bus endpoint.
2. Press the ‘Register’ button.
3. Select ‘Register New Plugin’
4. In the ‘Step #1: Specify the Location of the Assembly to Analyze’ field enter the location of the AzureAwareWorkflowActivity assembly built in the previous section.
5. Press the ‘Load Assembly’ button
6. Expand the ‘(Assembly) AzureAwareWorkflowActivity’ in the Step #2 section and ensure there is a Microsoft.Crm.Sdk.Samples.AzureAwareWorkflowActivity node which is checked.
7. Ensure in the ‘Step #3: Specify the Isolation Mode’ None is selected, and in ‘Step #4: Specify the Location where the Assembly should be stored’ Database is selected.
8. Press ‘Register Selected Plugins’
9. The following dialog box should appear:  
   
10. Use Internet Explorer to navigate to the Microsoft Dynamics CRM 2015 environment to have the plugin run.
11. Click the Settings area.
12. Under the ‘Process Center’ node click Processes.
13. Press the ‘New’ button.
14. For Process name enter testingAzureWorkflow
15. Ensure Entity is set to Account and Category is set to Workflow.
16. Press the ‘OK’ button which should bring up a new window editing the process.
17. Ensure ‘Start when: Record is created’ is checked
18. Click the ‘Add Step’ button.
19. Hover over the AppFabricWorkflowActivity.PostContextActivity menu item entry which should pop up a sub menu.
20. Click the guid listed in the sub menu to add it as a step. The guid in this step corresponds to the friendly name and name fields from when the assembly was registered in the Plugin Registration Tool.
21. In the ‘Type a step description here.’ field change the text to ‘onCreateOfAccount’
22. Click ‘Set Properties’
23. For the property named ‘Input id’ in the value column click the magnifying glass icon to select a configuration to use. Select the serviceendpoint corresponding to the REST listener or Two Way Listener.
24. Press the ‘Save and Close’ button to close the ‘Set Custom Step Input Properties – Webpage Dialog’
25. Press the Activate button to activate the workflow.
26. On the confirmation dialog press ‘OK’
27. Press the ‘Save and Close’ button to close the new process workflow.
28. Run the two way listener or REST listener project to listen for messages posted to azure.
29. Create a new account to cause the registered workflow to post a message to azure.
30. The details of the created account should display in the console window as shown in the next section.

**Expected Results**

You should see:  


## Troubleshooting

## Build error(s)

## Ensure the proper references are added and the project is targeting the .NET Framework 4.5.2 and not the .NET Framework 4.5.2 Client Profile.

## Run-time errors

## If when running the project it hangs while the host is opening check if Fiddler or some other proxy is preventing the sample from working.

## If everything seems to run fine but upon creating a new email nothing shows up, open Microsoft Dynamics CRM, navigate to Settings, System Jobs, and look for a job regarding the created email with ‘Waiting’ for Status Reason. Double click that job and view details to get troubleshooting information.

## If you are experiencing errors when running the samples, check the following.

## You have followed all instructions in this document correctly.

## The server, organization, and logon credentials you entered are correct.

## Your account on the CRM server has the required privileges to perform the sample’s intended operation.

## Important Notes

* The sample files should not be used in a production environment without prior testing. You should deploy an application that uses this sample code to a test environment and examine it for interaction or interference with other parts of the system.
* Before you deploy applications that use this sample code to a production environment, make sure that you consider the existing customizations that you may have implemented in Microsoft Dynamics CRM.

## Copyright

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

© 2014 Microsoft Corporation. All rights reserved.

Microsoft, Microsoft Dynamics, Active Directory, Outlook, Visual Studio, Windows, Windows Server, and Windows Vista are trademarks of the Microsoft group of companies.

All other trademarks are property of their respective owners.