|  |
| --- |
| **Pass data between plug-ins** |
| [See Also](ms-its:C:\Users\home\Desktop\2016%20and%202015%20SDK\SDK\CrmSdk2016.chm::/html/b3f4fbd0-cede-4e68-8908-39a1f3419ca9.htm#seeAlsoNoToggle) |

The message pipeline model for Microsoft Dynamics CRM defines a parameter collection of custom data values in the execution context that is passed through the pipeline and shared among registered plug-ins, even from different 3rd party developers. This collection of data can be used by different plug-ins to communicate information between plug-ins and enable chain processing where data processed by one plug-in can be processed by the next plug-in in the sequence and so on. This feature is especially useful in pricing engine scenarios where multiple pricing plug-ins pass data between one another to calculate the total price for a sales order or invoice. Another potential use for this feature is to communicate information between a plug-in registered for a pre-event and a plug-in registered for a post-event.

The name of the parameter that is used for passing information between plug-ins is [SharedVariables](ms-its:C:\Users\home\Desktop\2016%20and%202015%20SDK\SDK\CrmSdk2016.chm::/html/P_Microsoft_Xrm_Sdk_IExecutionContext_SharedVariables.htm). This is a collection of key\value pairs. At run time, plug-ins can add, read, or modify properties in the **SharedVariables** collection. This provides a method of information communication among plug-ins.

This sample shows how to use **SharedVariables** to pass data from a pre-event registered plug-in to a post-event registered plug-in.

|  |  |
| --- | --- |
| **C#** | ms-its:C:\Users\home\Desktop\2016%20and%202015%20SDK\SDK\CrmSdk2016.chm::/local/copycode.gifCopy Code |
| using System;  // Microsoft Dynamics CRM namespace(s)  using Microsoft.Xrm.Sdk;  namespace Microsoft.Crm.Sdk.Samples  {  /// <summary>  /// A plug-in that sends data to another plug-in through the SharedVariables  /// property of IPluginExecutionContext.  /// </summary>  /// <remarks>Register the PreEventPlugin for a pre-operation stage and the  /// PostEventPlugin plug-in on a post-operation stage.  /// </remarks>  public class PreEventPlugin : IPlugin  {  public void Execute(IServiceProvider serviceProvider)  {  // Obtain the execution context from the service provider.  Microsoft.Xrm.Sdk.IPluginExecutionContext context = (Microsoft.Xrm.Sdk.IPluginExecutionContext)  serviceProvider.GetService(typeof(Microsoft.Xrm.Sdk.IPluginExecutionContext));  // Create or retrieve some data that will be needed by the post event  // plug-in. You could run a query, create an entity, or perform a calculation.  //In this sample, the data to be passed to the post plug-in is  // represented by a GUID.  Guid contact = new Guid("{74882D5C-381A-4863-A5B9-B8604615C2D0}");  // Pass the data to the post event plug-in in an execution context shared  // variable named PrimaryContact.  context.SharedVariables.Add("PrimaryContact", (Object)contact.ToString());  }  }  public class PostEventPlugin : IPlugin  {  public void Execute(IServiceProvider serviceProvider)  {  // Obtain the execution context from the service provider.  Microsoft.Xrm.Sdk.IPluginExecutionContext context = (Microsoft.Xrm.Sdk.IPluginExecutionContext)  serviceProvider.GetService(typeof(Microsoft.Xrm.Sdk.IPluginExecutionContext));  // Obtain the contact from the execution context shared variables.  if (context.SharedVariables.Contains("PrimaryContact"))  {  Guid contact =  new Guid((string)context.SharedVariables["PrimaryContact"]);  // Do something with the contact.  }  }  }  } | |

It is important that any type of data added to the shared variables collection be serializable otherwise the server will not know how to serialize the data and plug-in execution will fail.

For a plug-in registered in stage 20 or 40, to access the shared variables from a stage 10 registered plug-in that executes on create, update, delete you must access the [ParentContext](ms-its:C:\Users\home\Desktop\2016%20and%202015%20SDK\SDK\CrmSdk2016.chm::/html/P_Microsoft_Xrm_Sdk_IPluginExecutionContext_ParentContext.htm).**SharedVariables** collection. For all other cases, [IPluginExecutionContext](ms-its:C:\Users\home\Desktop\2016%20and%202015%20SDK\SDK\CrmSdk2016.chm::/html/T_Microsoft_Xrm_Sdk_IPluginExecutionContext.htm).**SharedVariables** contains the collection.