

## Tech Note 1003

# Accessing InBatch Recipes In System Platform Using Recipe Automation Server (Object Hosted on Batch Server Node)

---

All Tech Notes, Tech Alerts and KBCD documents and software are provided "as is" without warranty of any kind. See the [Terms of Use](#) for more information.

Topic#: 002834

Created: January 2014

## Introduction

This *Tech Note* demonstrates the syntax used for the Recipe Object creation/release in a .NET Framework environment.

In this scenario, a Platform is deployed to an InBatch Server Node and the object containing script for accessing InBatch Recipes is hosted to this platform. The other functionality on reading/writing to the InBatch Recipe Database is documented in InBatch COM Guide.

## Application Versions

- InBatch 9.x & 10.x
- System Platform 2012 R2 (for prior versions, you have to manually build the object)

## Prerequisites

A platform must be deployed to the InBatch Server Node.

## Procedure

1. From the Archestra IDE menu, select Galaxy\Import\Script Function Library and import the file RecipeEdit.exe located at **C:\Program files(x86)\Wonderware\inBatch\Bin** on InBatch Server Node.
2. [Download and unzip the InBatchRecipeDBAccess.zip](#) file.
3. Import the \$InBatchRecipeDBAccess template aaPkg to the IDE.
4. Create an instance of this object and host it to the Platform\AppEngine\Area that is running on InBatch Server Node.
5. Deploy the new object instance and set the trigger attribute (**me.Trigger**) in ObjectViewer for this object.

See the messages in the Archestra Logger for successful connectivity, query of recipes and release of the recipeDB object.

The following graphics show the Template Object Editor, with the text in the script body (Figure 1 below), and the messages in

The screenshot displays the Archestra IDE interface for editing a script named "QueryRecipes". The main configuration area is divided into several sections:

- Aliases:** A section for defining aliases, currently empty.
- Declarations:** A section for defining variables and their types. The declarations shown are:
 

```

DIM oRecipeDB AS RecipeEditLib.IRecipe;
DIM aRecipes AS System.Array;
DIM iConnected AS Integer;
DIM i AS Integer;
            
```
- Scripts:** A section for configuring the script's execution. The "Execution type" is set to "Execute". The "Expression" is "me.Trigger". The "Trigger type" is "OnTrue". The "Trigger period" is "00:00:00.0000000". The "Deadband" is "0.0". The "Historize script state" checkbox is unchecked. The "Runs asynchronously" checkbox is checked and highlighted with a red box. The "Timeout limit" is set to "60000" ms and is also highlighted with a red box. The "Report alarm on execution error" checkbox is unchecked. The "Priority" field is empty.
- Code Editor:** The script's code is visible at the bottom of the window. The code includes:
 

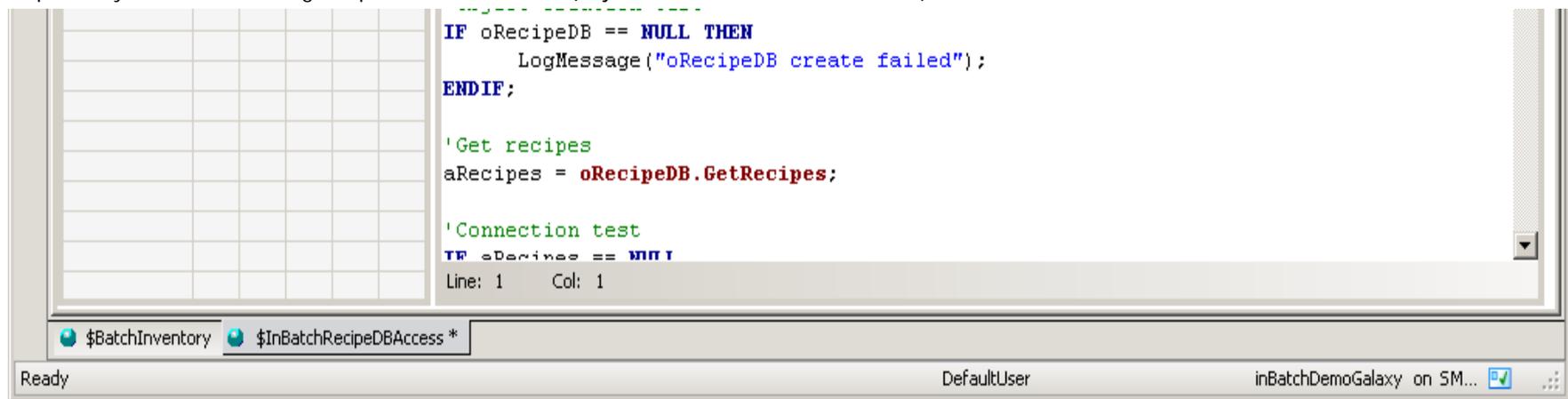
```

'object globals
me.Trigger = false;
iConnected = 0;
i = 1;

-----
'Recipe Automation Server - localhost
-----
LogMessage ("***Recipe Automation Server*** [localhost]");

'Creating and allocating memory to RecipeDB Object
oRecipeDB = NEW RecipeEditLib.wvRecipe();

'Object creation test.
            
```



**FIGURE1: ENSURE THE SCRIPT RUNS ASYNCHRONOUSLY**

The following section contains the script (if you using a prior version of System Platform 2012 R2, copy the following text and paste it to your object, then declare variables in the declaration section as shown in Figure 1 (above).

```

'script globals and Trigger is a Boolean UDA
me.Trigger = false;
iConnected = 0;
i = 1;
-----
'Recipe Automation Server - localhost
-----
LogMessage("***Recipe Automation Server*** [localhost]");

'Creating and allocating memory to RecipeDB Object
oRecipeDB = NEW RecipeEditLib.wvRecipe();

'Object creation test
IF oRecipeDB == NULL THEN
LogMessage("oRecipeDB create failed");
ENDIF;

'Get recipes
aRecipes = oRecipeDB.GetRecipes;

'Connection test
IF aRecipes == NULL
THEN iConnected = 0;
LogMessage("oRecipeDB connection state: Not connected " + iConnected);
ELSE iConnected = 1;
LogMessage("oRecipeDB connection state: Connected " + iConnected);
ENDIF;
FOR i = aRecipes.GetLowerBound(0) to aRecipes.GetUpperBound(0)
LogMessage(" Recipe[" + i + "]: " + aRecipes.GetValue(i));
NEXT;

'note - to prevent multiple copies, shutdown recipe server
LogMessage("***Recipe Automation Server*** release");
IF oRecipeDB <> NULL
THEN System.Runtime.InteropServices.Marshal.ReleaseComObject( oRecipeDB );
oRecipeDB = NULL;
ENDIF;
-----

```

No.	Date	Time	Process ID	Thread ID	Log Flag	Component	Message
813585	12/9/2013	10:18:17 AM	9412	12032	Info	ScriptRuntime	InBatchRecipeDBAccess_002.QueryRecipes: ***Recipe Automation Server*** [localhost]
813586	12/9/2013	10:18:17 AM	5844	6576	Info	RecipeEdit	Startup - Version 10.0.000
813587	12/9/2013	10:18:17 AM	5844	13136	Info	RecipeEdit	License feature 'INBATCH_SERVER' of version 10.000 has been acquired, expiration date =
813588	12/9/2013	10:18:18 AM	9412	12032	Info	ScriptRuntime	InBatchRecipeDBAccess_002.QueryRecipes: oRecipeDB connection state: Connected 1
813589	12/9/2013	10:18:18 AM	9412	12032	Info	ScriptRuntime	InBatchRecipeDBAccess_002.QueryRecipes: Recipe[1]: CB1000 Hold
813590	12/9/2013	10:18:18 AM	9412	12032	Info	ScriptRuntime	InBatchRecipeDBAccess_002.QueryRecipes: Recipe[2]: CB1000 Mix
813591	12/9/2013	10:18:18 AM	9412	12032	Info	ScriptRuntime	InBatchRecipeDBAccess_002.QueryRecipes: ***Recipe Automation Server*** release
813592	12/9/2013	10:18:19 AM	5844	6576	Info	RecipeEdit	Shutdown

FIGURE 2: LOGGER MESSAGES

S. Mariyala

*Tech Notes* are published occasionally by Wonderware Technical Support. Publisher: Invensys Systems, Inc., 26561 Rancho Parkway South, Lake Forest, CA 92630. There is also technical information on our software products at [Wonderware Technical Support](#).

For technical support questions, send an e-mail to [wwsupport@invensys.com](mailto:wwsupport@invensys.com).

 [Back to top](#)

©2014 Invensys Systems, Inc. All rights reserved. No part of the material protected by this copyright may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying, recording, broadcasting, or by any information storage and retrieval system, without permission in writing from Invensys Systems, Inc.

[Terms of Use](#).