

[Tech Note 1006](#)

Managing Symbol Script Timeouts When Using WW MES API Pop-up Dialogs

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#: 002837

Created: January 2014

Introduction

Wonderware MES provides many ways to customize script or code in order to meet variable production requirements. One solution is to take advantage of Wonderware Application Server to implement custom code in script, and show a user interface within an InTouch window.

There are many concerns when utilizing ArchestraA or Symbol script for MES customization. This *Tech Note* addresses how the timeout setting affects the script behavior, and includes relevant recommendations to avoid such effect.

Application Versions

- Wonderware Application Server 3.x or higher
- Wonderware InTouch 10.x or higher
- Wonderware MES 3.5 or 4.x

Symbol Script Timeout Setting

To avoid infinite loops in a Symbol script, set a timeout limit in which FOR loops must complete execution.

If a script loop does not complete execution within the timeout limit, WindowViewer automatically terminates the loop and writes a message to the Logger. You can check or change timeout setting via **WindowMaker > Special > Configure > WindowViewer > Managed Application** tab (Figure 1 below).

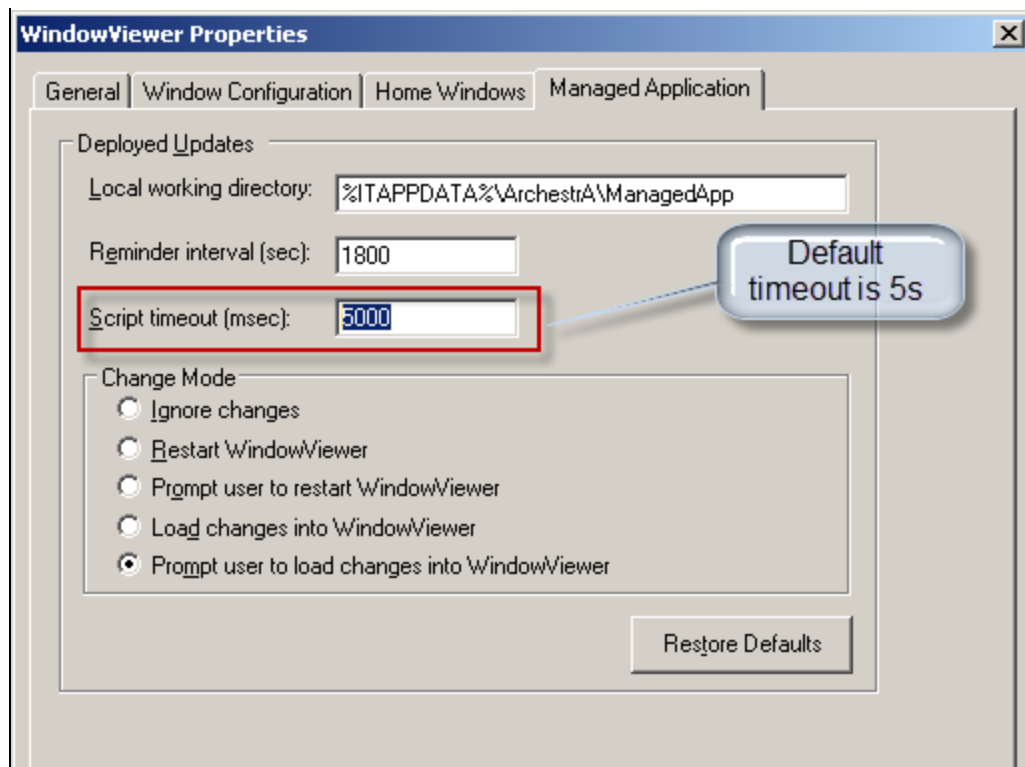


FIGURE 1: SYMBOL SCRIPT TIMEOUT SETTING IN INTOUCH

By design, the timeout setting only affects the scripts including loop logic. In most cases, there's no concern as the script is supposed to be executed successfully in few seconds. However, when the script is also including any kind of user interaction, it is most likely the script will timeout, resulting in the script halting at the next Loop logic code. It will execute other code but still generate a timeout error message.

Note: This *Tech Note* includes detailed Timeout setting information for Archestra Graphics. Detailed explanation of FOR loops will be added to Archestra Graphics User's Guide in a future release.

Sample MES Custom Code in Symbol Script

Wonderware MES provides flexible ways to customize production requirement via Stateful and Stateless APIs. Some of the popup methods for the UI are predefined for user interaction. The following list includes but is not limited to these methods:

- PopupEntitySelection()
- PopupItemReasonSelection()
- PopupUtilReasonSelection()
- PopupOperationSelection()
- PopUpAddConsDlg()

- And Others

If the custom script includes at least one of above APIs with additional loop logic inside the same script, it is likely that some of the transactions can fail due to timing out.

The sample MES custom code in a Symbol Script to verify this behavior.

In this sample code, it is simple to allow user to select the entity and show all entity attributes in the list box:

- Call **FactMES.Utility.PopupEntitySelection()** to display the popup UI for entity selection.
- Call **Entity.GetEntAttributes()** to return all entity attributes in a dataset.
- Loop through the dataset and call **ListBox.AddItem()** to display all attributes.

Sample Code

The screenshot shows the 'Edit Animations' window for 'English (United States)'. The 'Action Scripts' tab is active, showing a script for 'Button1'. The script is as follows:

```

' Get the client session
result = aaFactMES.aaClientSession.GetInstance();
clientSession = result.Value;
if ( clientSession <> null ) then
    If (clientSession.curUser <> null and clientSession.curUser.curEnt <> null) Then
        ' There is an user and entity logged in this session
        LogMessage("Begin Entity Selection");
        ' Get selected entity via popup UI
        result = aaFactMES.aaUtility.PopupEntitySelection(aaFactMES.aaUtility.aaEntityCapabilities.CanRun, entityId, entityName);
        value = result.Value;
        If (value == True) Then
            LogMessage("After Entity Selection");
            entity = clientSession.curUser.curEnt;
            ' Get entity attributes based on entity id
            result1 = entity.GetEntAttributes(entityId, null);
            valAtt = result1.Value;
            ListBox1.Clear();
            LogMessage("Begin Add Item");

```

```
LogMessage( "begin add item" );
For Each dt in valAtt.Tables
  LogMessage("Loop Tables");
  For Each dr in dt.Rows
    LogMessage("Loop Rows");
    ' Add attribute name and value to the list box
    ListBox1.AddItem(dr(5) + " : " + dr(3));
  Next;
Next;
LogMessage("After Add Item");
ErrorMsg = "Entity Selected: " + entityName;
Else
  ErrorMsg = "User Cancelled the Selection";
EndIf;
Else
  ' There is no current user or entity logged into session
  ErrorMsg = "There seems to be no user or entity logged in this session";
EndIf;
endif;
Line: 1 Col: 1
```

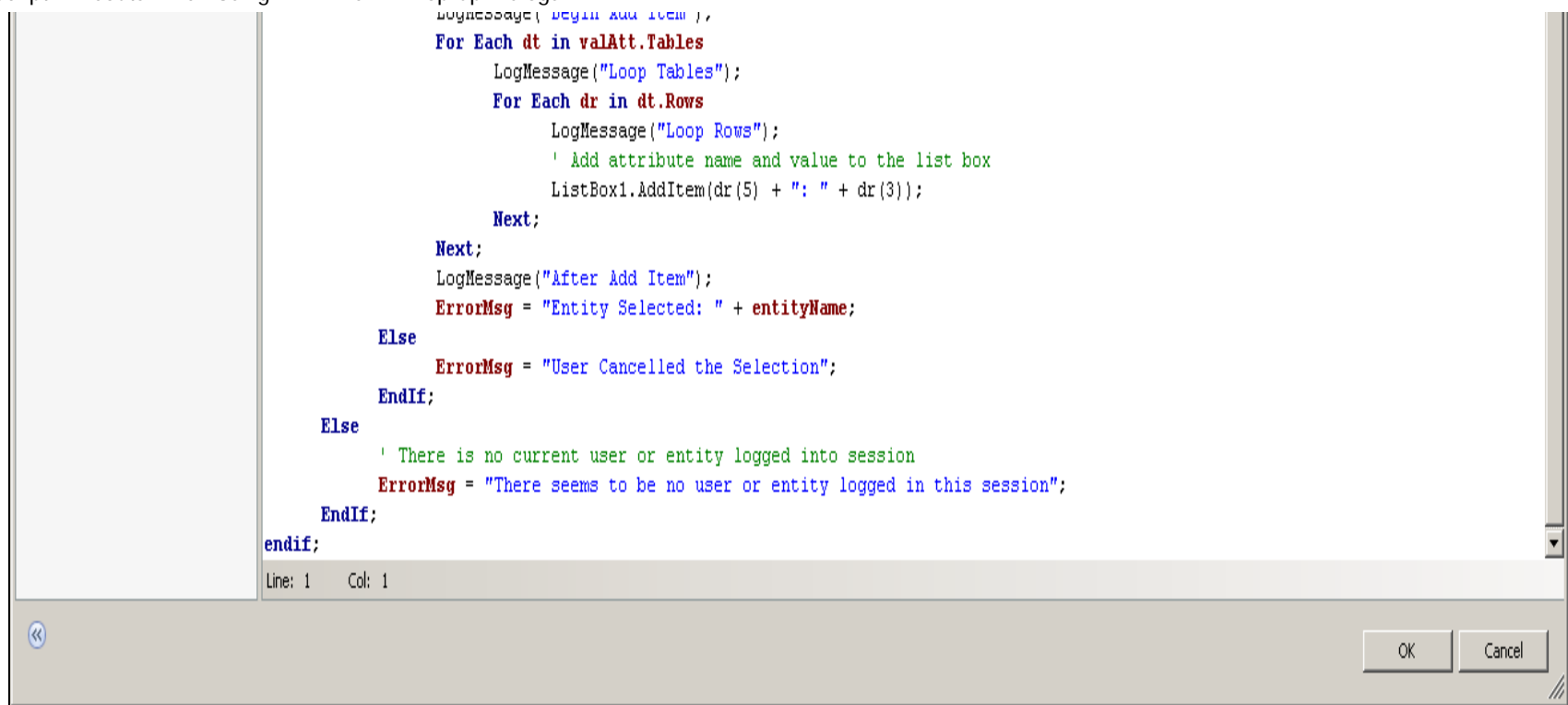


FIGURE 2: SAMPLE MES CUSTOM CODE ALL-IN-ONE

When executing the whole script in WindowViewer within default 5 seconds, the complete list of entity attributes should be displayed successfully as below:

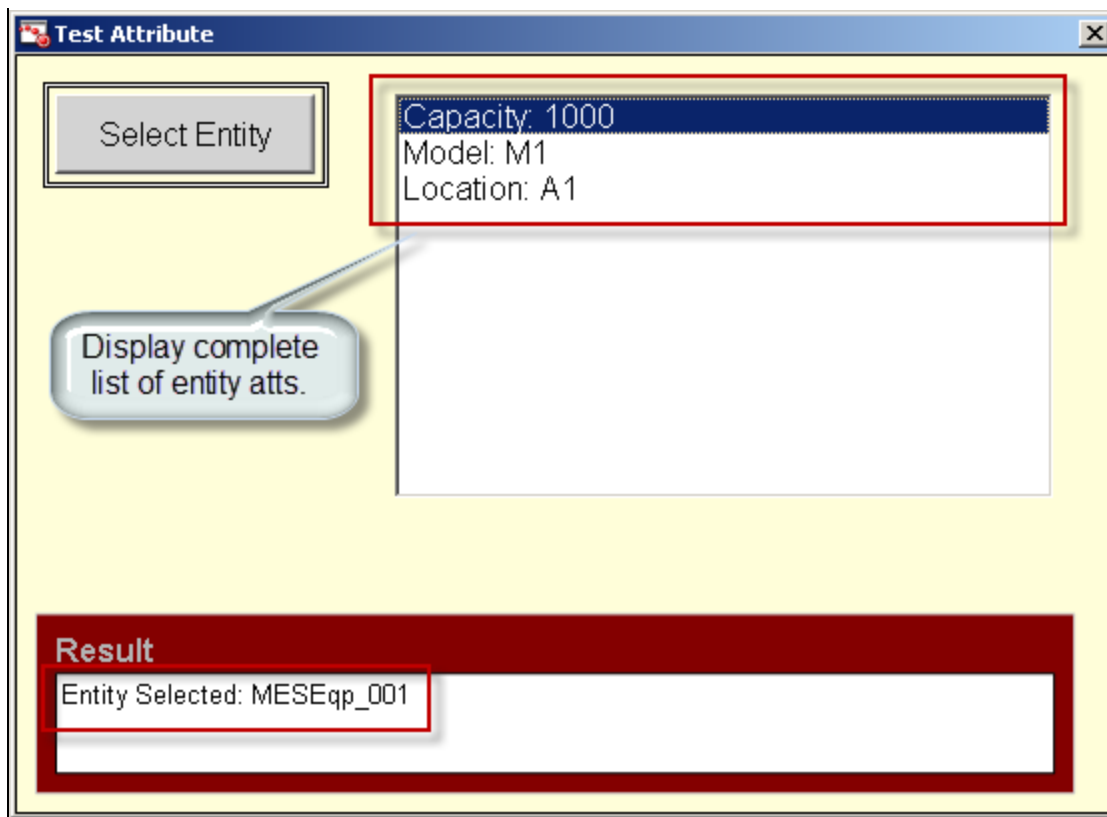


FIGURE 3: UI FOR NORMAL OPERATION

From the SMC log, we can observe all correct log lines without any error.

284971	2013/12/27	上午 10:40:34	5820	5744	Info	view	TestAtt1.Button1: Begin Entity Selection
284972	2013/12/27	上午 10:40:37	5820	5744	Info	view	TestAtt1.Button1: After Entity Selection
284973	2013/12/27	上午 10:40:37	5820	5744	Info	view	TestAtt1.Button1: Begin Add Item
284974	2013/12/27	上午 10:40:37	5820	5744	Info	view	TestAtt1.Button1: Loop Tables
284975	2013/12/27	上午 10:40:37	5820	5744	Info	view	TestAtt1.Button1: Loop Rows
284976	2013/12/27	上午 10:40:37	5820	5744	Info	view	TestAtt1.Button1: Loop Rows
284977	2013/12/27	上午 10:40:37	5820	5744	Info	view	TestAtt1.Button1: Loop Rows
284978	2013/12/27	上午 10:40:37	5820	5744	Info	view	TestAtt1.Button1: After Add Item

FIGURE 4: SMC LOG FOR NORMAL OPERATION

When Timeout Triggered

When the script does not execute within the (default) 5 seconds, the user cannot complete the action in Entity Selection popup UI in time. Nothing is displayed and no error message is displayed in the UI.

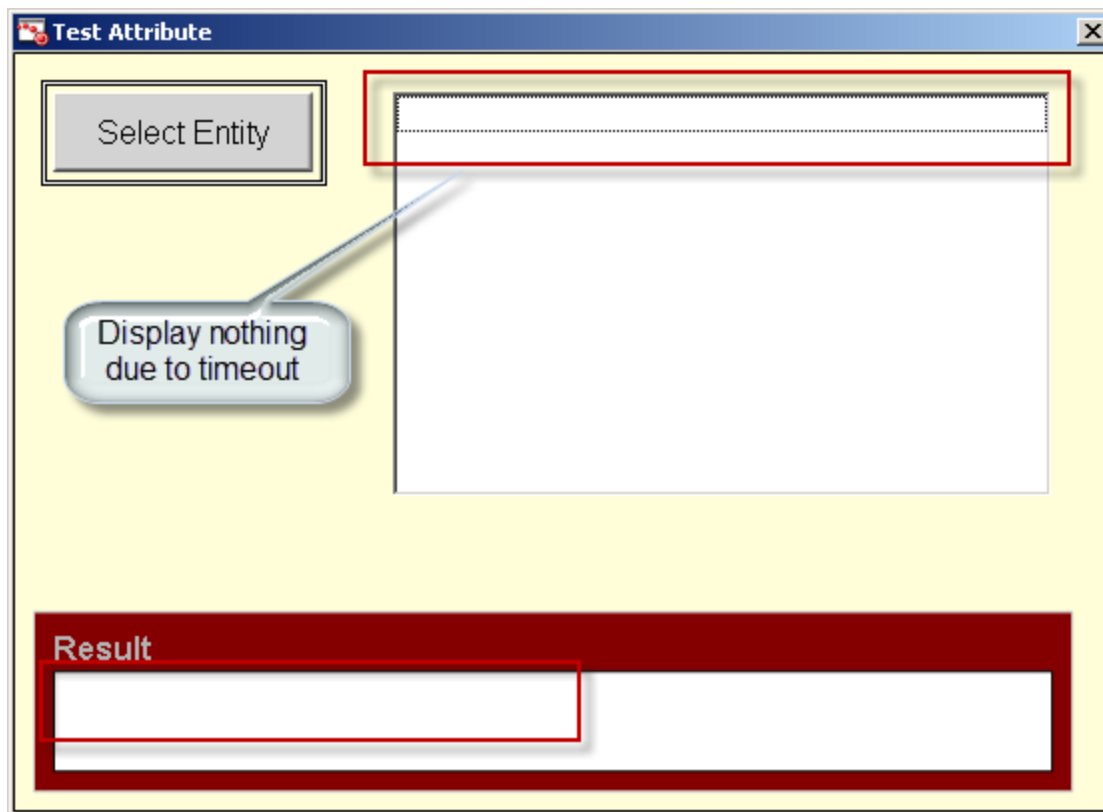


FIGURE 5: UI FOR TIMEOUT OPERATION

This is because the timeout limit is triggered. From the SMC we can observe uncompleted log lines with additional timeout warnings.

The screenshot displays a 'Message Details' dialog box and a log window. The dialog box shows a warning message: "Failed for the following reason: timed out". A callout bubble points to this message with the text "Interaction more than 5s". The log window shows a sequence of messages for "TestAtt1.Button1" including "Begin Entity Selection", "After Entity Selection", and "Begin Add Item". A callout bubble points to the log entry for "Begin Add Item" with the text "Not completed!".

No.	Date	Time	Process ID	Thread ID	Log Flag	Message
284987	2013/12/27	上午 11:01:44	5820	5744	Warning	Failed for the following reason: timed out
284984	2013/12/27	上午 11:01:36	5820	5744	Info	TestAtt1.Button1: Begin Entity Selection
284985	2013/12/27	上午 11:01:44	5820	5744	Info	TestAtt1.Button1: After Entity Selection
284986	2013/12/27	上午 11:01:44	5820	5744	Info	TestAtt1.Button1: Begin Add Item
284987	2013/12/27	上午 11:01:44	5820	5744	Warning	Failed for the following reason: timed out

FIGURE 6: SMC LOG FOR TIMEOUT OPERATION

There's a big concern that not all of the MES transactions are completed, resulting in data inconsistency in MES database. However, it does not make sense to limit user to interact the popup UI (e.g. Entity Selection in this example) within this limited interval.

Managing the Scripts

It is possible to change the timeout setting for slow running scripts with no user interaction to a maximum of 360 seconds. The recommended strategy for the pop-up dialogs from WW MES is to split them into multiple scripts. Setting a long timeout interval results in potential performance issues or deadlocks if infinite loops are encountered. In this case, you can show the popup dialog in one script and then have it start a second script that contains the loop logic.

The following procedure shows how to separate the scripts.

1. Create the necessary custom properties in ArchestraA Graphic as public variables, e.g. **EntityID** and **FillLB**.

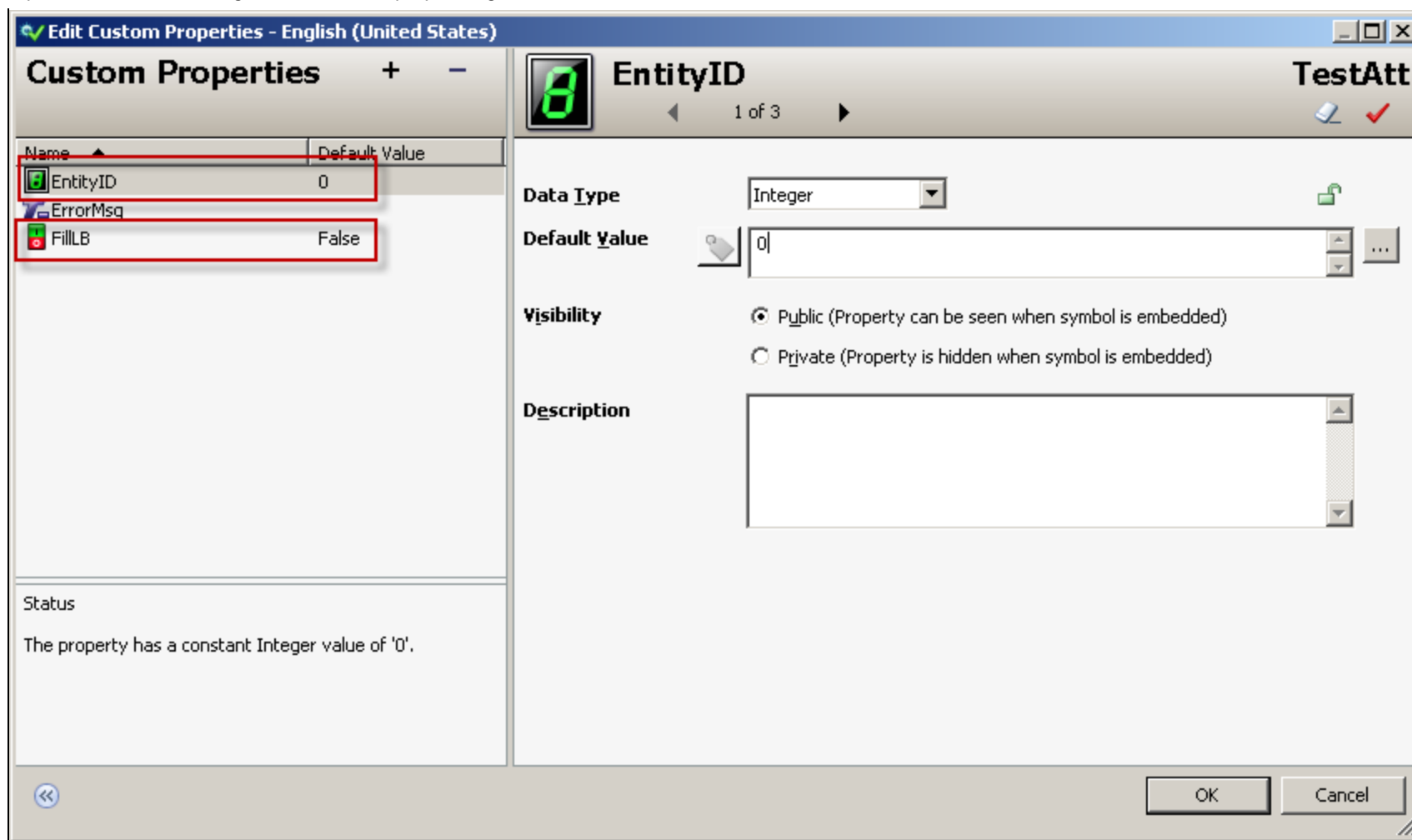


FIGURE 7: CREATE CUSTOM PROPERTIES AS PUBLIC VARIABLES

2. Include the loop logic in another named script to fill in list box, e.g. **FillListBox**.

Edit Scripts

Symbol Scripts + -

FillListBox 2 of 2

Predefined Scripts

Predefined Scripts

Named Scripts

FillListBox

Expression FillLB

Trigger OnTrue Quality changes Period: 1000 ms

Deadband 0

```

' Get the client session
result = aaFactMES.aaClientSession.GetInstance();
clientSession = result.Value;
if ( clientSession <> null ) then
    If (clientSession.curUser <> null and clientSession.curUser.curEnt <> null) Then
        ' There is an user and entity logged in this session
        entity = clientSession.curUser.curEnt;
        ' Get entity attributes based on entity id
        result1 = entity.GetEntAttributes(EntityId, null);
        valAtt = result1.Value;
        ListBox1.Clear();
        LogMessage("Begin Add Item");
        For Each dt in valAtt.Tables
            LogMessage("Loop Tables");
            For Each dr in dt.Rows
                LogMessage("Loop Rows");
                ' Add attribute name and value to the list box
                ListBox1.AddItem(dr(5) + ": " + dr(3));
            Next;
        Next;
        LogMessage("After Add Item");
    Else
        ' There is no current user or entity logged into session
        ErrorMsg = "There seems to be no user or entity logged in this session";
    EndIf;
endif;
    
```

Entity Id could be got from another script even timeout

Line: 40 Col: 1

OK Cancel

FIGURE 8: INCLUDE LOOP LOGIC IN NAMED SCRIPT

3. Delete the loop logic from original script only to include popup dialog and necessary transactions. Call the named script **FillListBox** (triggered by **FILLB**) instead.

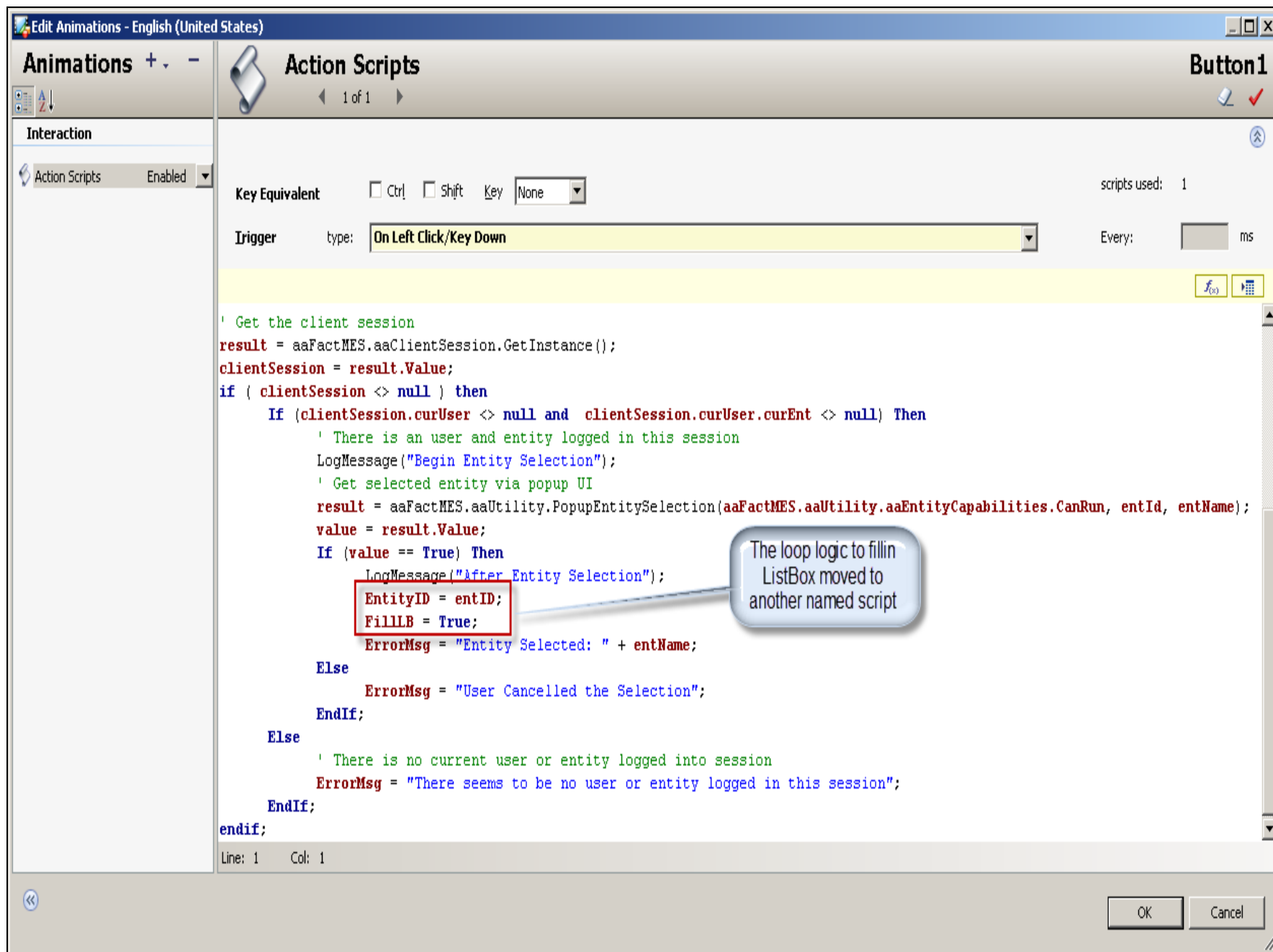


FIGURE 9: INCLUDE ONLY POPUP DIALOG IN ORIGINAL SCRIPT

4. Re-execute the graphic in WindowViewer even when exceeding the timeout interval. You would still be expected to see additional timeout warning in SMC log by design. However, the complete list of entity attributes should display successfully.

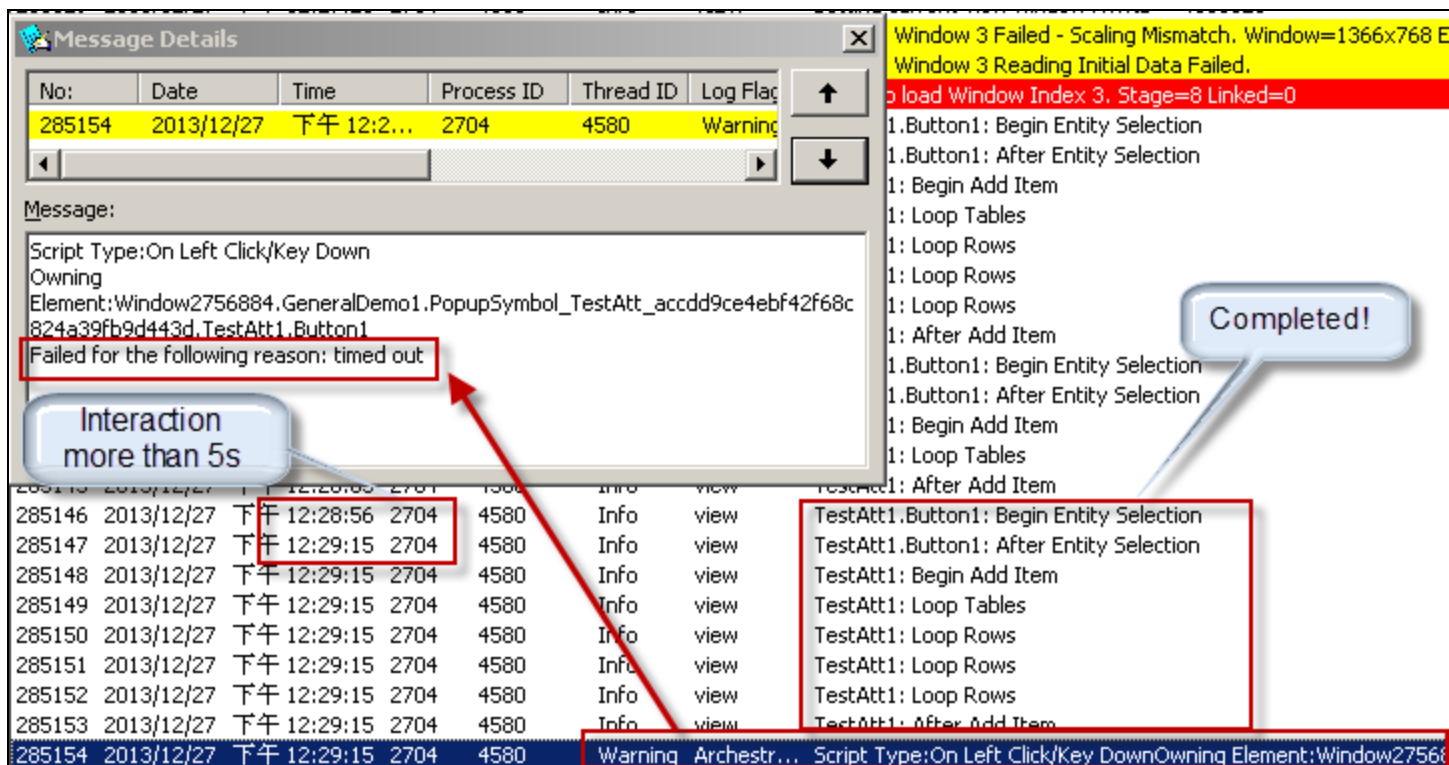


FIGURE 10: SMC LOG FOR TIMEOUT OPERATION AFTER SEPARATING ORIGINAL SCRIPT

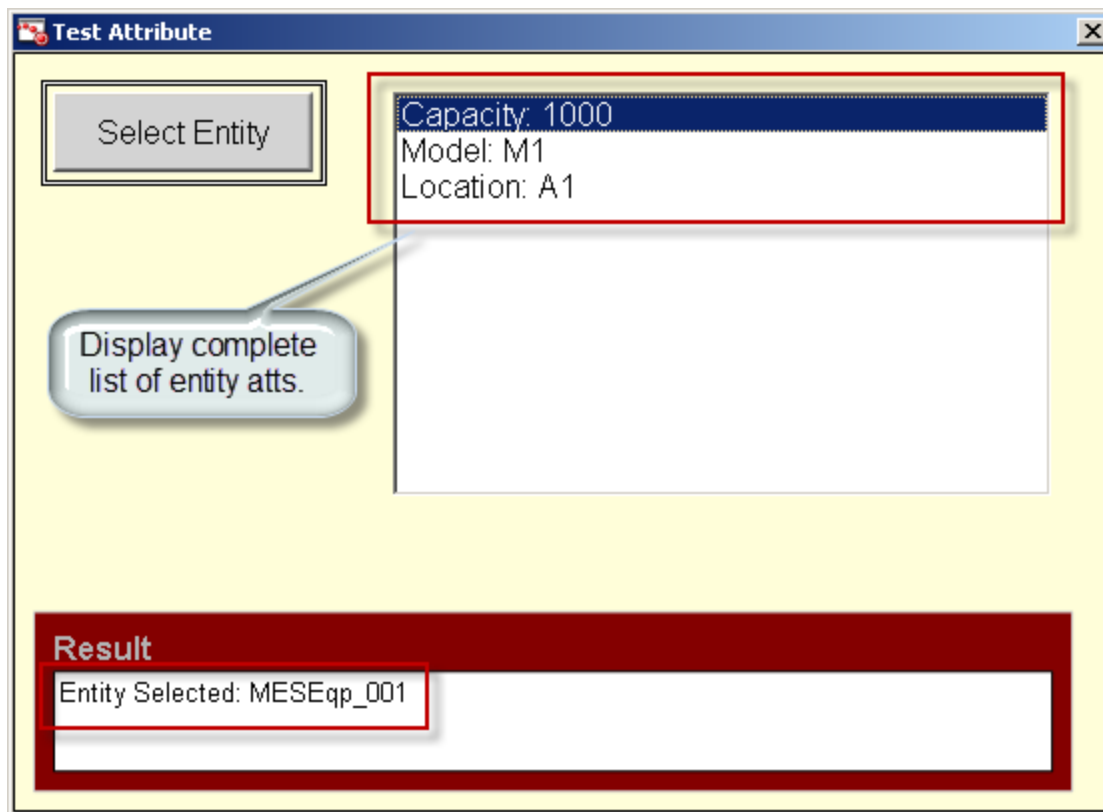


FIGURE 11: UI FOR TIMEOUT OPERATION AFTER SEPARATING ORIGINAL SCRIPT

S. Hong

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.

 [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](#).