Tech Note 905

Troubleshooting Wonderware Information Server (WIS) Part Six: ArchestrA Graphics – No Live-Data

Topic#: 002715
Created: December 2012
Updated: October 2013

Introduction

This troubleshooting guide is the sixth in a projected series.

The ArchestrA Graphics Live-data mechanism in WIS is built up on Microsoft WCF (Windows Communication Foundation) Technology. The key components in the Live-data route are LMX Publisher and WCF Service Components.

- **WCF Service Component**: Live-data Engine which sits on WIS node. It handles all the business logic, such as
  - Security verification for Writeback
  - Establish the connections for each tags to each client session
  - Push Live-data to the corresponding client session
  - Handler Mapping between WIS and IIS
- **LMX Publisher**: sets up the connection to ArchestrA Lmx and subscribes tags for bi-directional data exchange. The following diagram shows the Live-data channel.

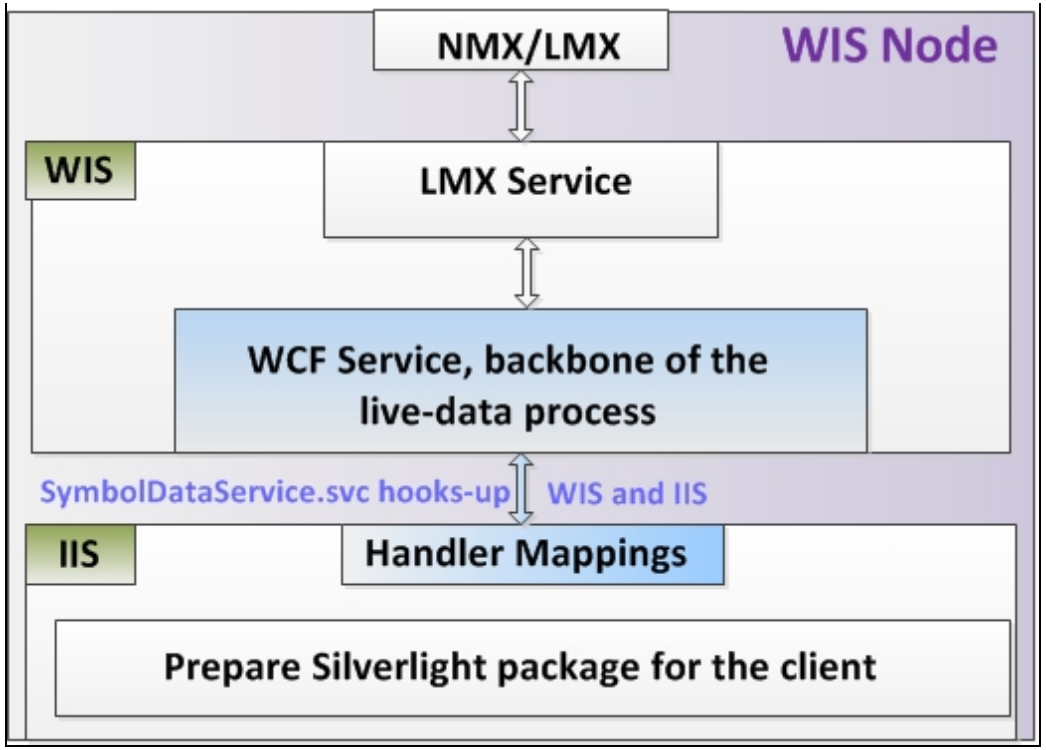

Figure 1: Live-data Channel Diagram

Note: AADataSvc is the pre-defined Virtual Directory. This is the entry point of WIS ArchestrA Graphics between IIS and IE clients. We will often use AADataSvc in this *Tech Note*.

Application Versions

- WIS 4.0 and later
- Windows 2003 Server SP2
- Windows 2008 Server or later

Case Scenarios

- **IIS Cannot Browse the Content File (.svc) Exposed by WIS WCF Service**
- **Missing .svc Path From IIS 6.0 or 7.x Handler Mappings**
- **Conflicting (Multiple) Endpoints**
- **AADataSvc Points to Wrong .NET Framework**
- **Applying HotFix Causes No Live-data Update**

- **SMC Log Flogs for Live-data**

Scenario: IIS Cannot Browse the Content File (.svc) Exposed by WIS WCF Service

Like any WCF with IIS application, WIS ArchestrA Graphics also defines the SymbolDataService.svc file which contains some WCF-specific methods that can be accessed from WIS's ASP.Net pages or their code-behind under IIS 6.0 or 7.x.

If this channel between WCF Service and IIS 6.0 or 7.x is working normally, you see the following when we send an access request from IE.

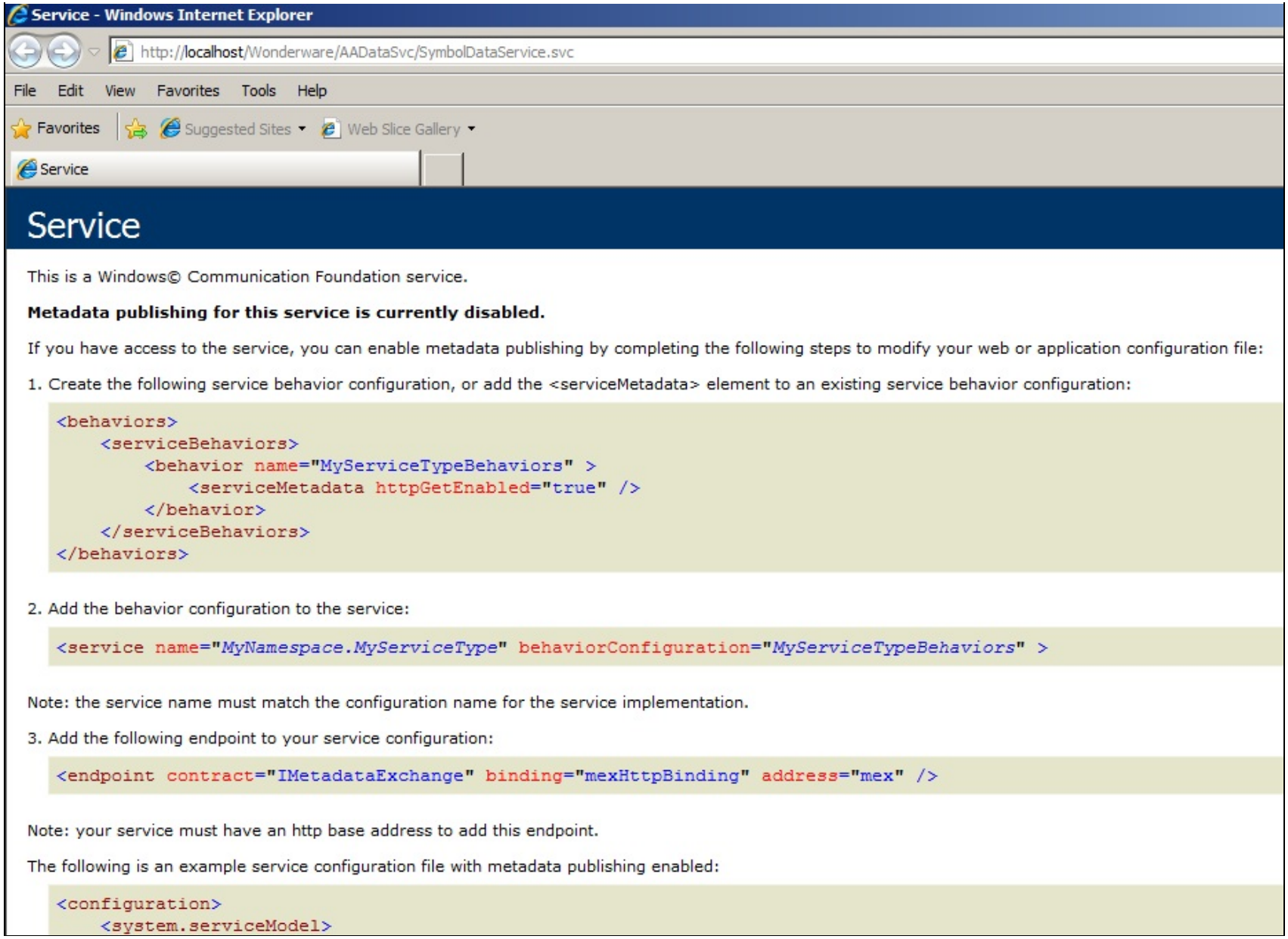

FIGURE 2: NORMAL BROWSING RESULT

The following graphics (Figures 3 and 4 below) show browsing the **SymbolDataService.svc** file in IIS 6.0 and IIS 7.x.

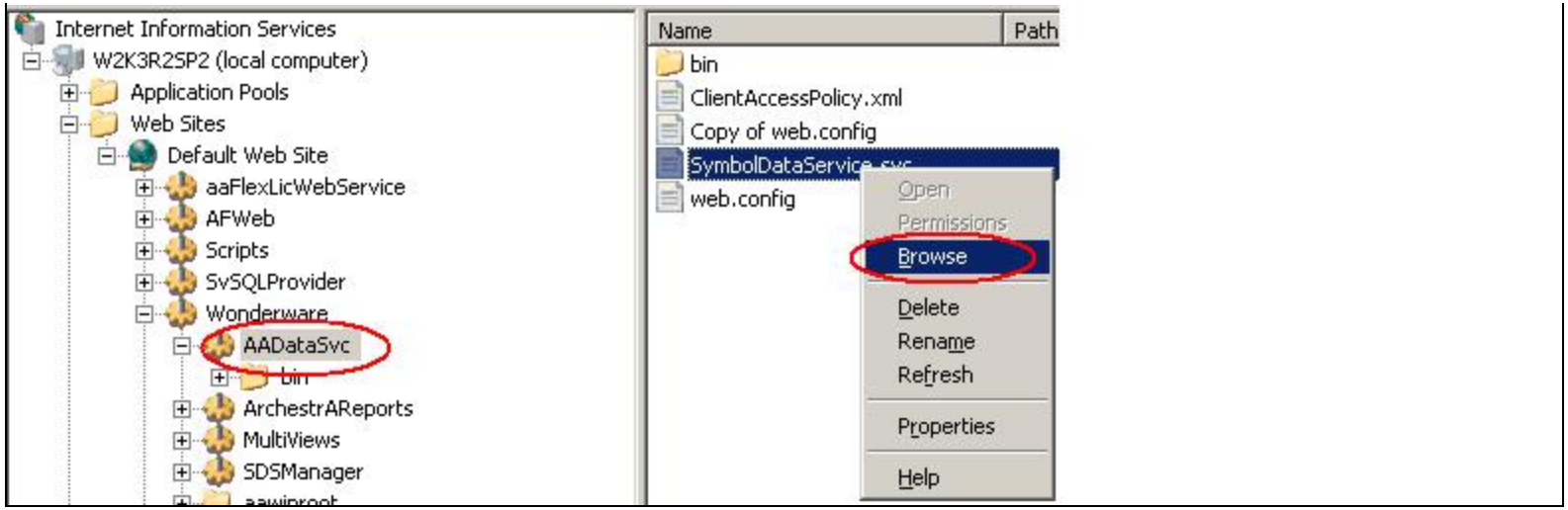

FIGURE 3: BROWSING AADATASVC FILE IN IIS 6.0

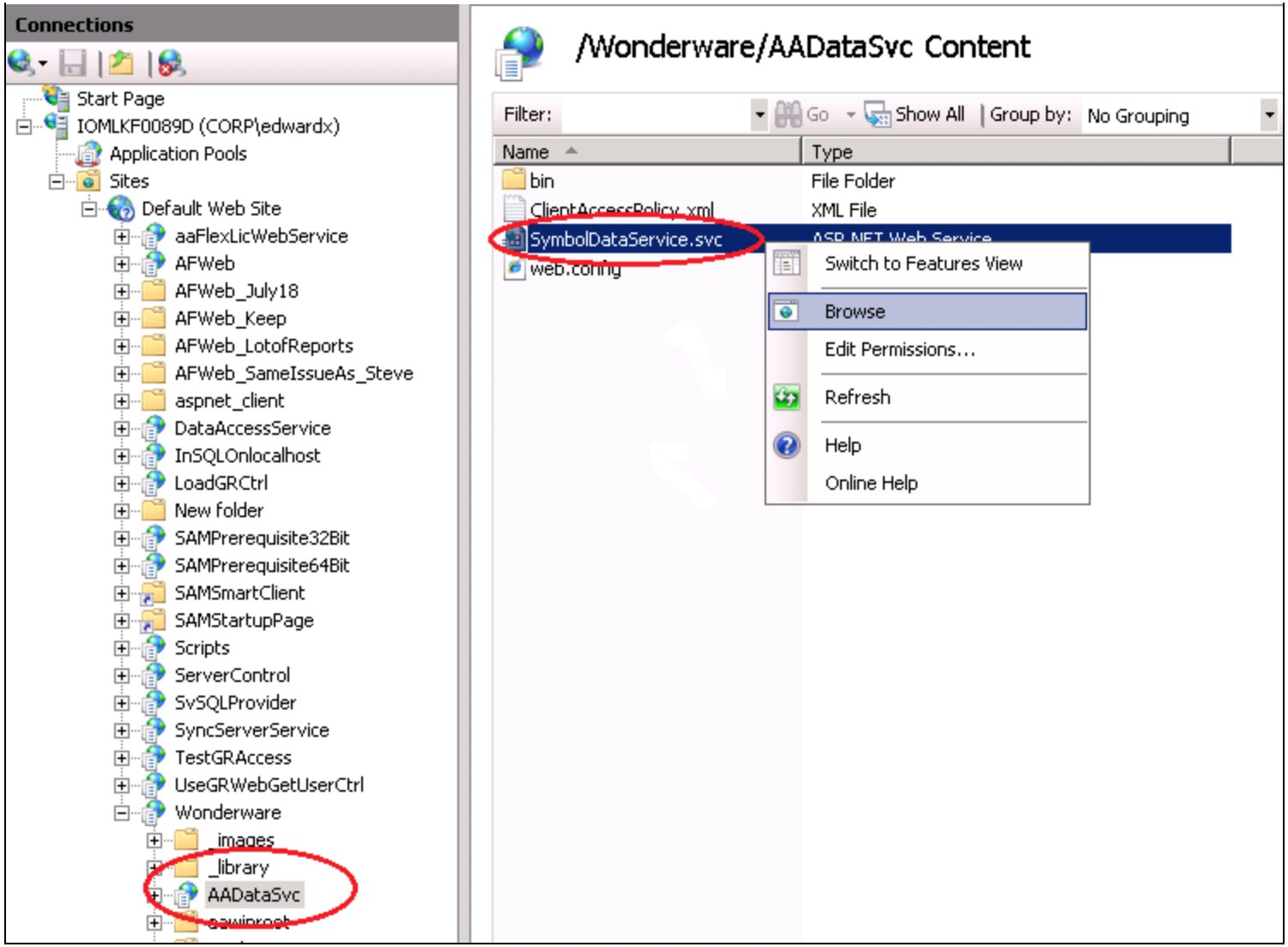

FIGURE 4: BROWSING AADATASVC IN IIS 7.X

Possible reasons for not being able to see the normal browsing result:

- For some reason, there is no authentication selected in the **AADataSvc** virtual directory. If you browse the **SymbolDataService.svc** file you see the following message:

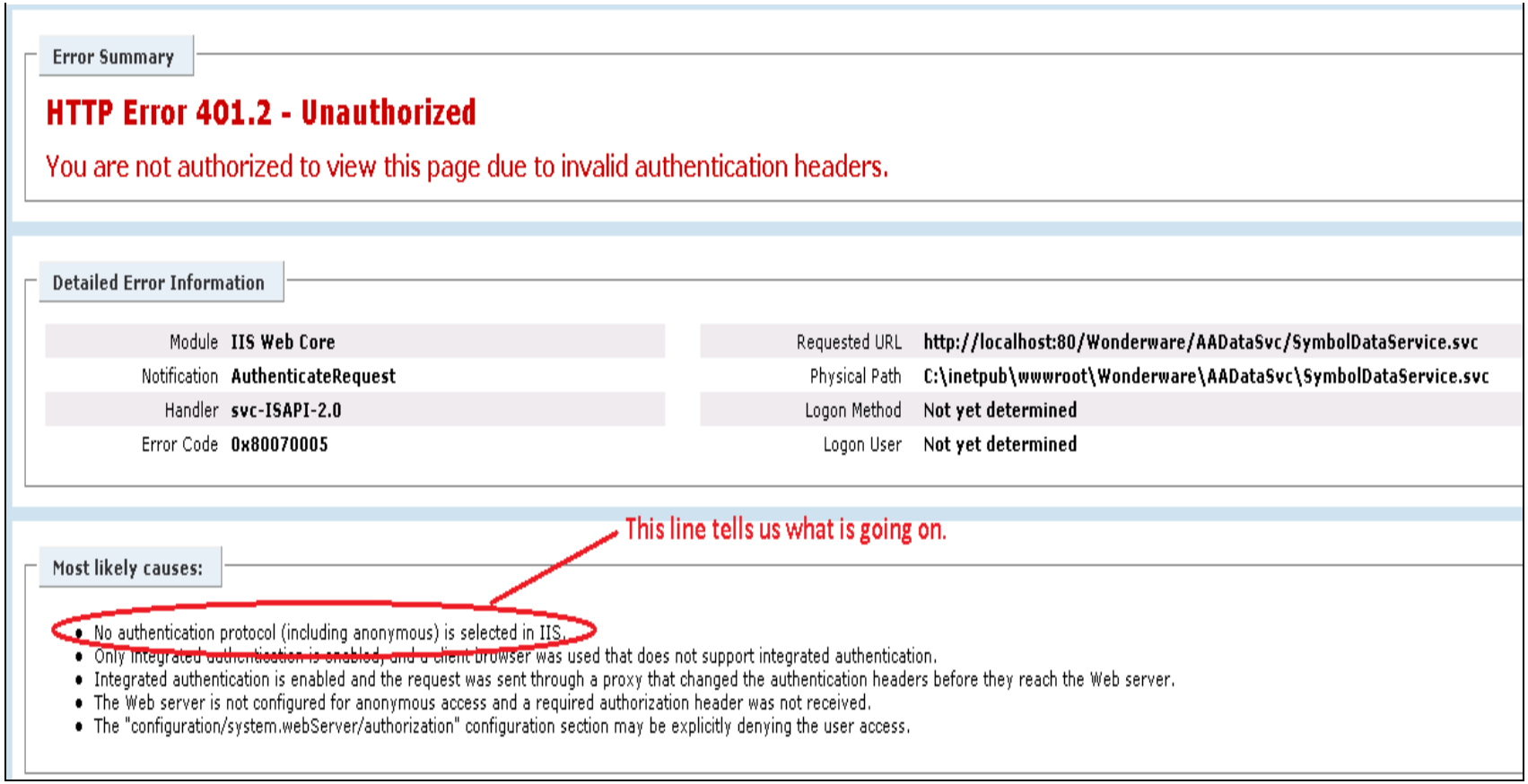

Figure 5: Error After Clicking the Browse Button

Resolution

- **IIS 6.0 and 7.x do not support Integrated Windows authentication by default.**

  See the following graphics (Figures 6 and 7 below) to check settings.

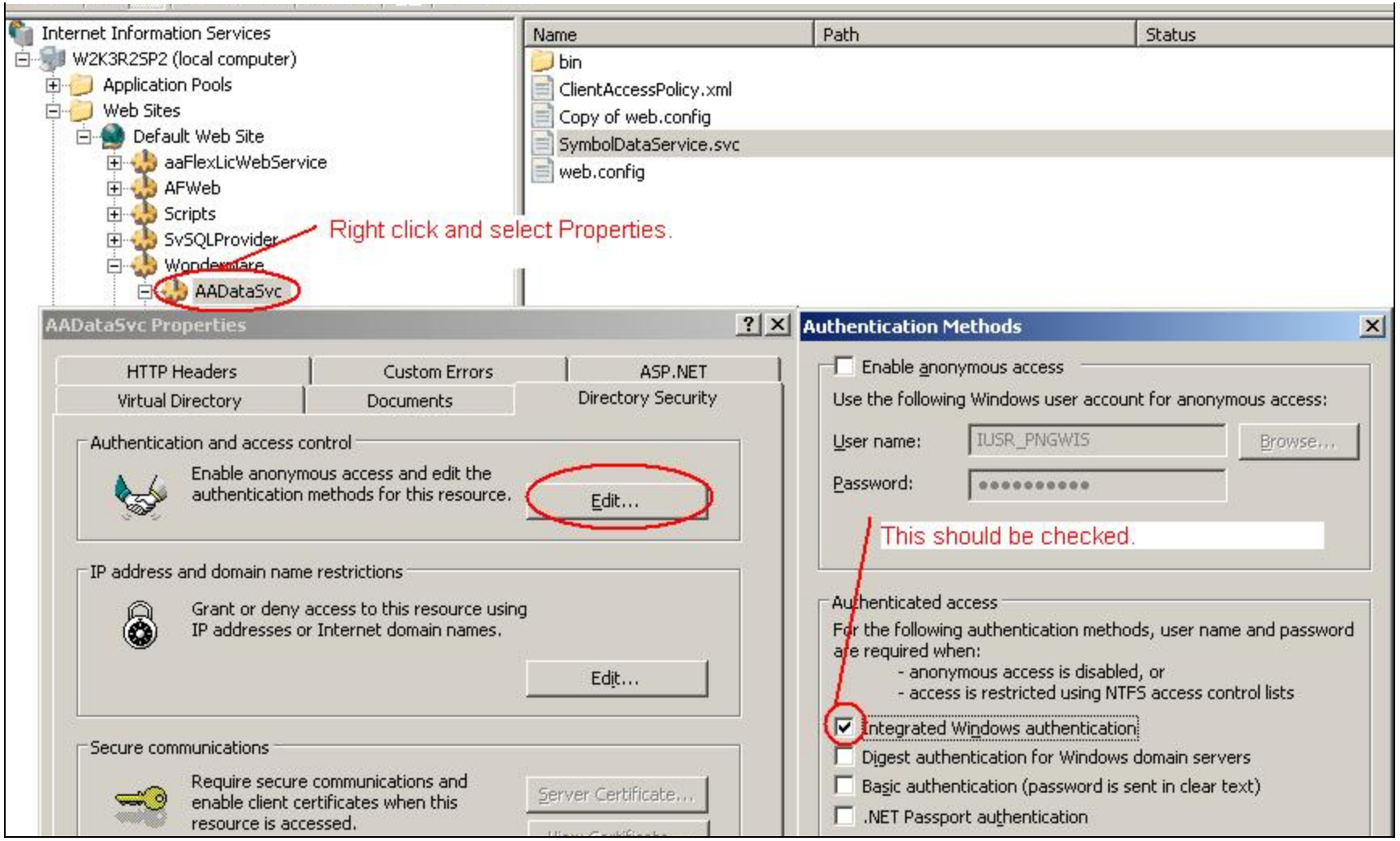

FIGURE 6: INTEGRATED WINDOWS AUTHENTICATION SETTING IN IIS 6.0

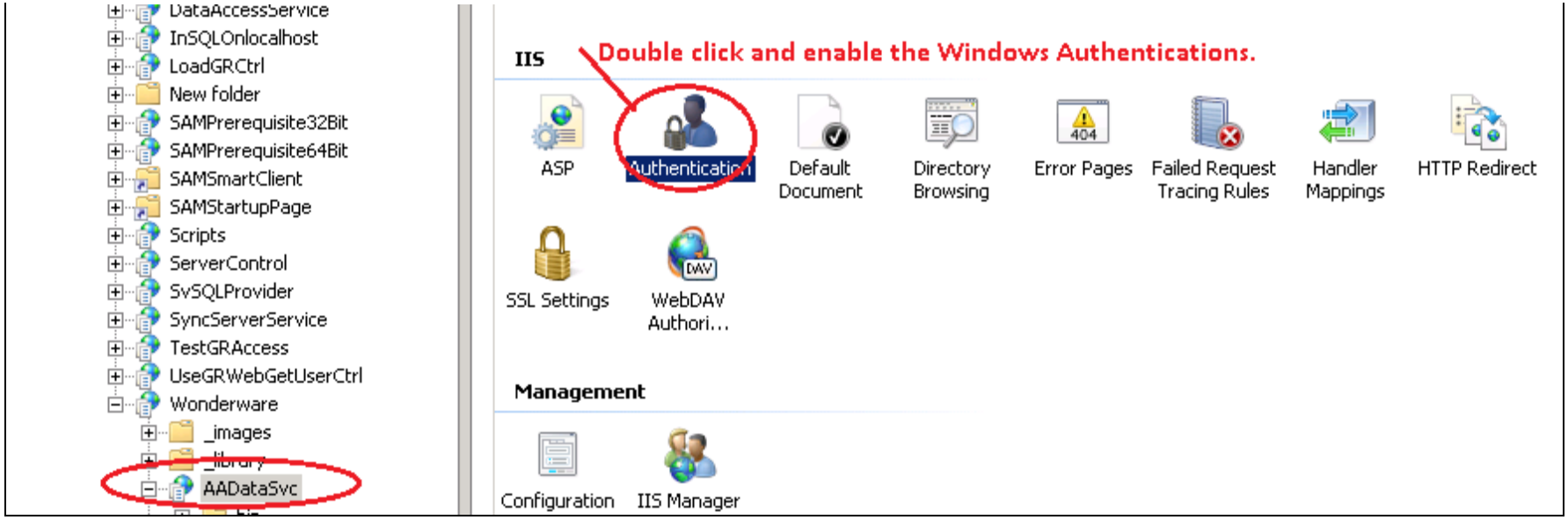

Figure 7: Integrated Windows Authentication Setting in IIS 7.x

- In the IIS configuration file, the windowsAuthentication element does not contain the NTLM value.

For IIS 6.0

1. Go to **[C]:\WINDOWS\system32\inetsrv**. Make copy of **MetaBase.xml**.
2. Open MetaBase.xml and find out the value, Location ="/LM/W3SVC/1/ROOT"
3. Add the following line: **NTAuthenticationProviders="Negotiate,NTLM"**

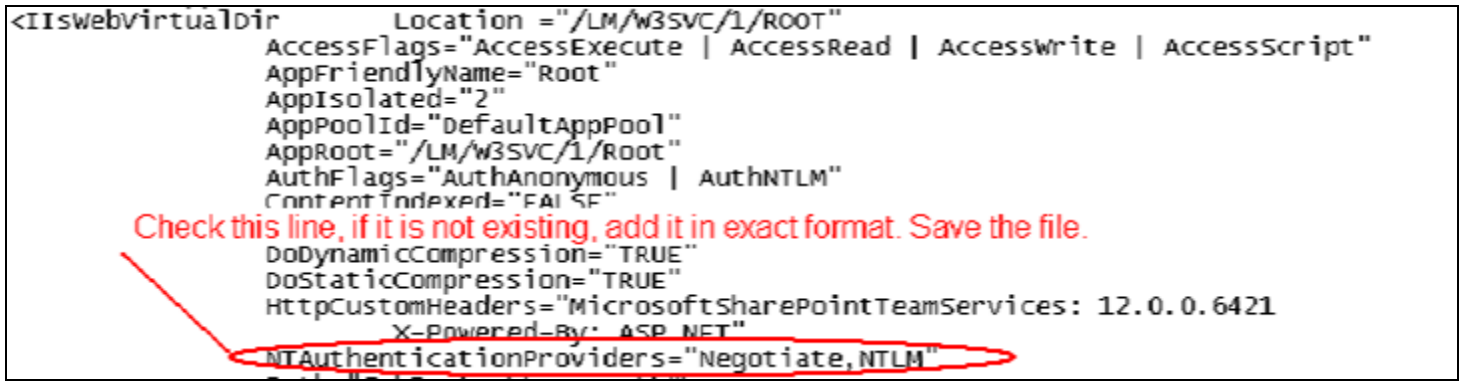

Figure 8: Add NTAuthenticationProviders Line

4. Save the changes.
5. Run cmd > **IISReset**.

For IIS 7.x

1. Go to [C]:\Windows\System32\inetsrv\config\. Make copy of **applicationHost.config**.
2. Open **applicationHost.config** and find the section shown below. Make sure the values in red circle exist.

```
</system.webServer>
<location path="" overrideMode="Allow">
    <system.webServer>
        <security>
            <authentication>
            <windowsAuthentication enabled="false">
                <providers>
                    <add value="Negotiate" />
                    <add value="NTLM" />
                </providers>
            </windowsAuthentication>

            <anonymousAuthentication enabled="true" userName="IUSR" />
            <basicAuthentication />

            <clientCertificateMappingAuthentication />

            <digestAuthentication />
            </authentication>
```

Figure 9: applicationHost.config File Section

3. You don't need to run IISReset. The change will take effect after the file is saved.

Scenario: Missing .svc path from IIS 6.0 or 7.x Handler Mappings

Before we can browse the SymbolDataService.svc, the .svc needs to be defined in Handler Mappings which map to the resources on the Web server and generate responses for requests.

In this case, browsing to the **SymbolDataService.svc** file generates the following error (Figure 10 below):

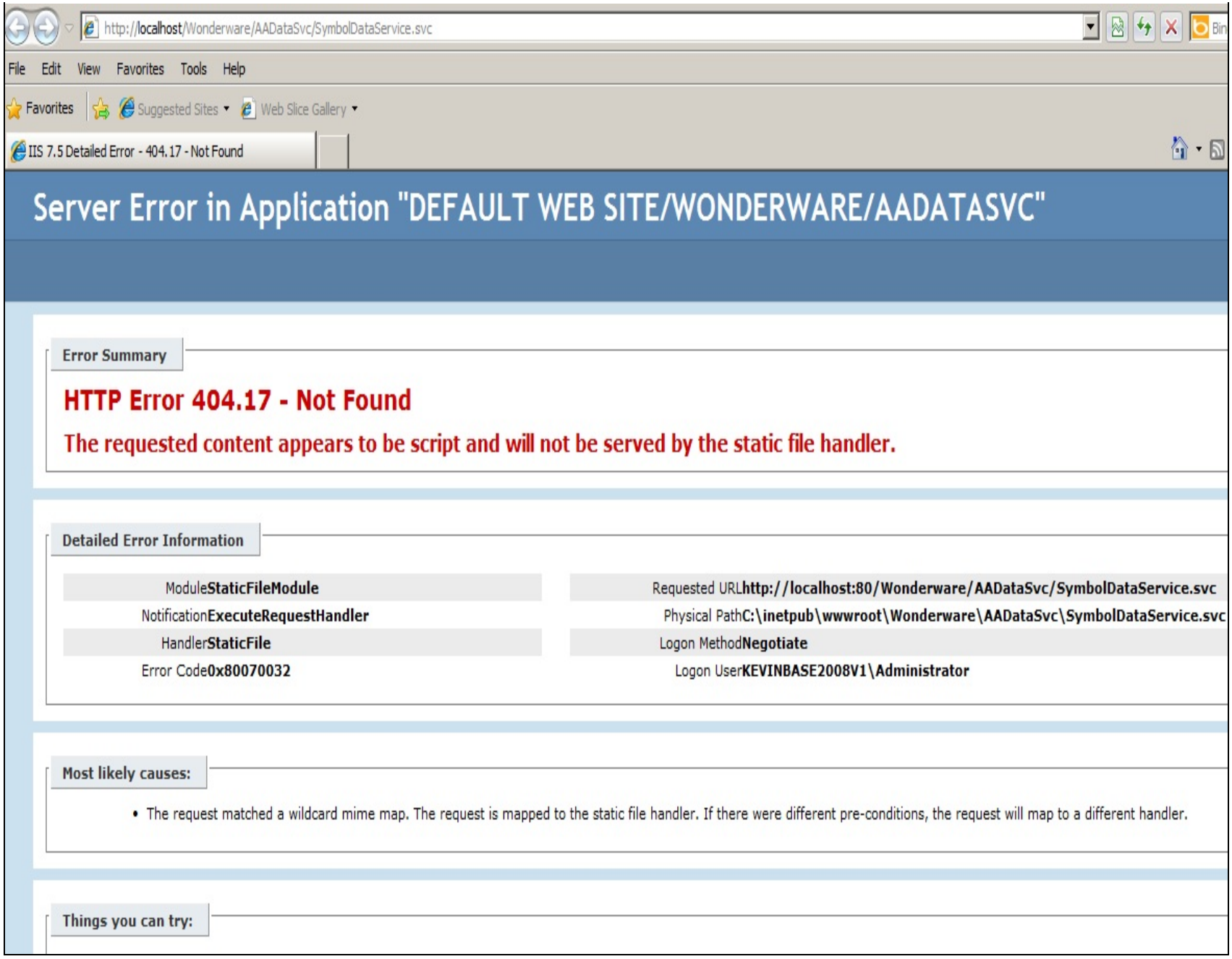

Figure 10: Error After Clicking the Browse Button

Resolution

Task: Check the Application Pool that AADataSvc belongs to.

For IIS7

This step is only necessary when you have IIS 7.x. IIS 6.0 only supports 32-bit Windows.

1. Right-click **AADataSvc** and click **Manage Application / Advanced Settings**.

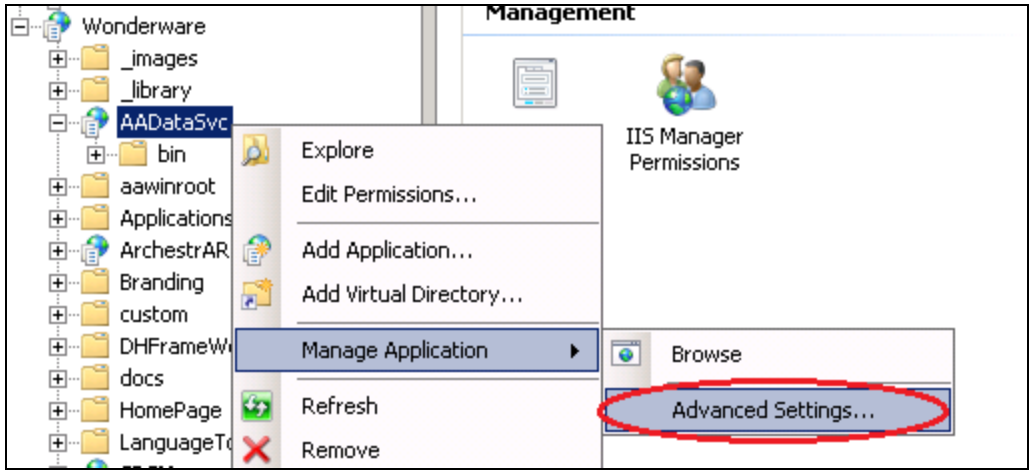

Figure 11: Advanced Settings for Application

2. Check to see that **SVAppPool** is visible for the Application Pool (Figure 11 below).

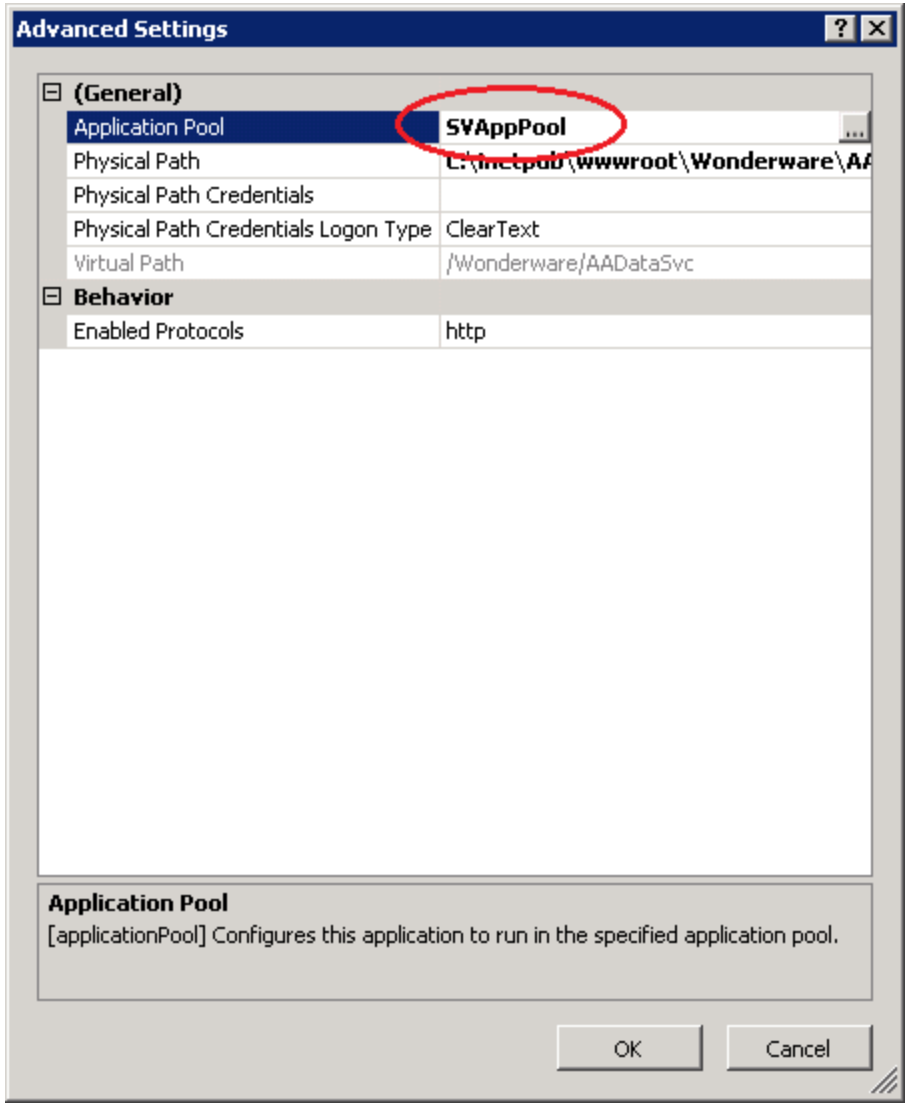

FIGURE 12: AADATASVC BELONGS TO SVAPPPOOL

In the **Connections** panel, click Application Pools, then right-click **SVAppPool** and click **Advanced Settings**.

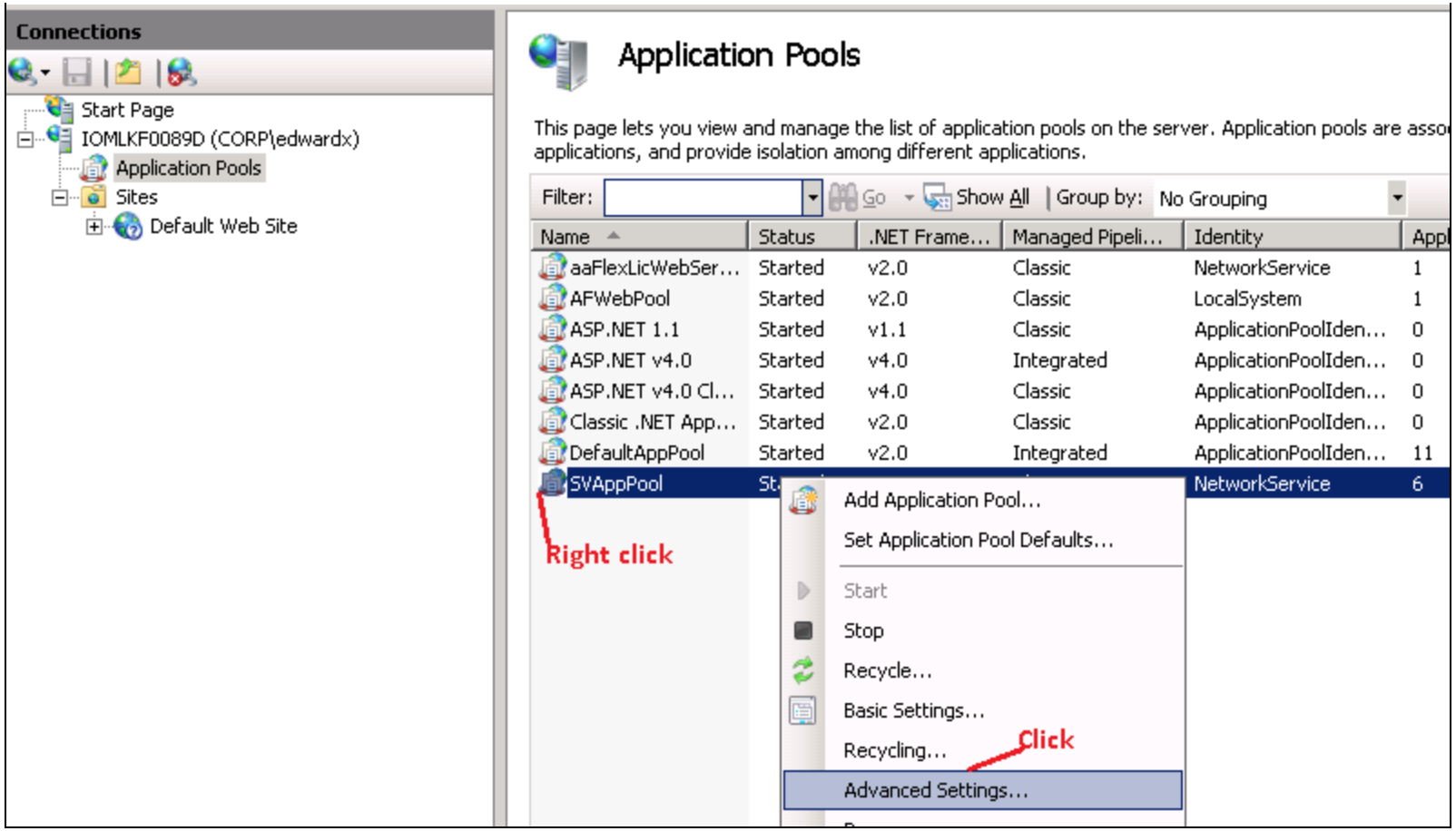

FIGURE 13: STEPS TO OPEN SVAPPPOOL'S ADVANCED SETTINGS

Note: When using Information Server Versions prior to 2012 R2 (WIS 5.0), verify that the **(General)** Advanced Settings list appears as shown in Figure 14 (below).

4. Verify that the **(General)** Advanced Settings list appears as shown in Figure 14 (below).

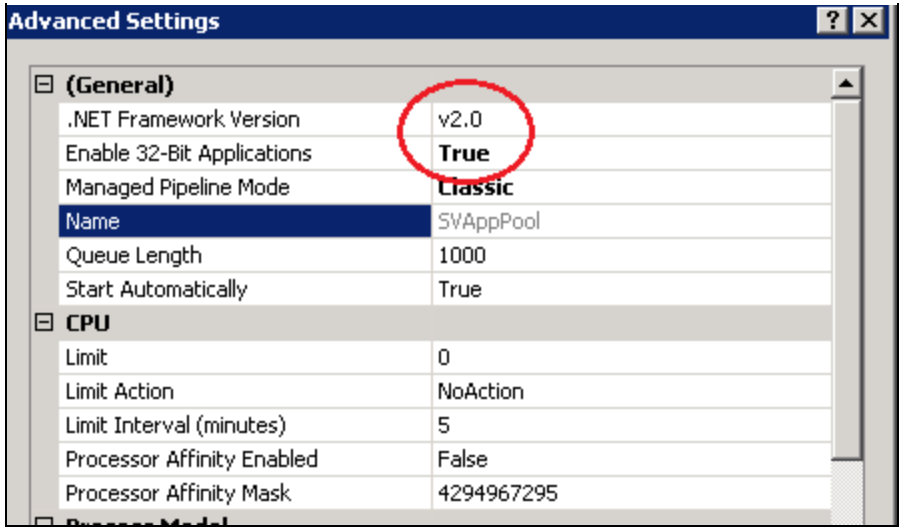

Figure 14: Make Sure That .NET 2.0 and Enable 32-Bit Application are Set

Figure 14 (above) shows that **AADataSvc** is under .NET Framework 2.0 and 32-Bit environment.

Note: When using Information Server Versions 2012 R2 and later (WIS 5.0), verify that the **(General)** Advanced Settings list appears as shown in Figure 15 (below).

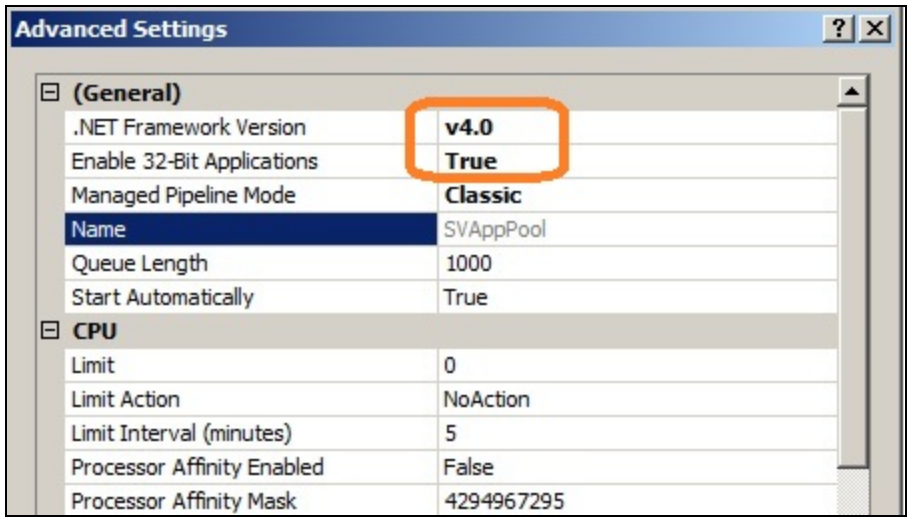

Figure 15: Ensure That .NET 4.0 and Enagle 32-bit Applications Options are Set

Task: Define a Handler in Mappings List for AADataSvc

- **Using Information server Versions prior to 2012 R2 (WIS 5.0)**

For IIS 6.0

1. Right-click AADataSvc and click **Properties**.
2. Click the **Configuration** button.
3. In the Mappings tab, check the list for the **.svc** extension.

   Figure 16 (below) shows that .svc is missing/not defined.

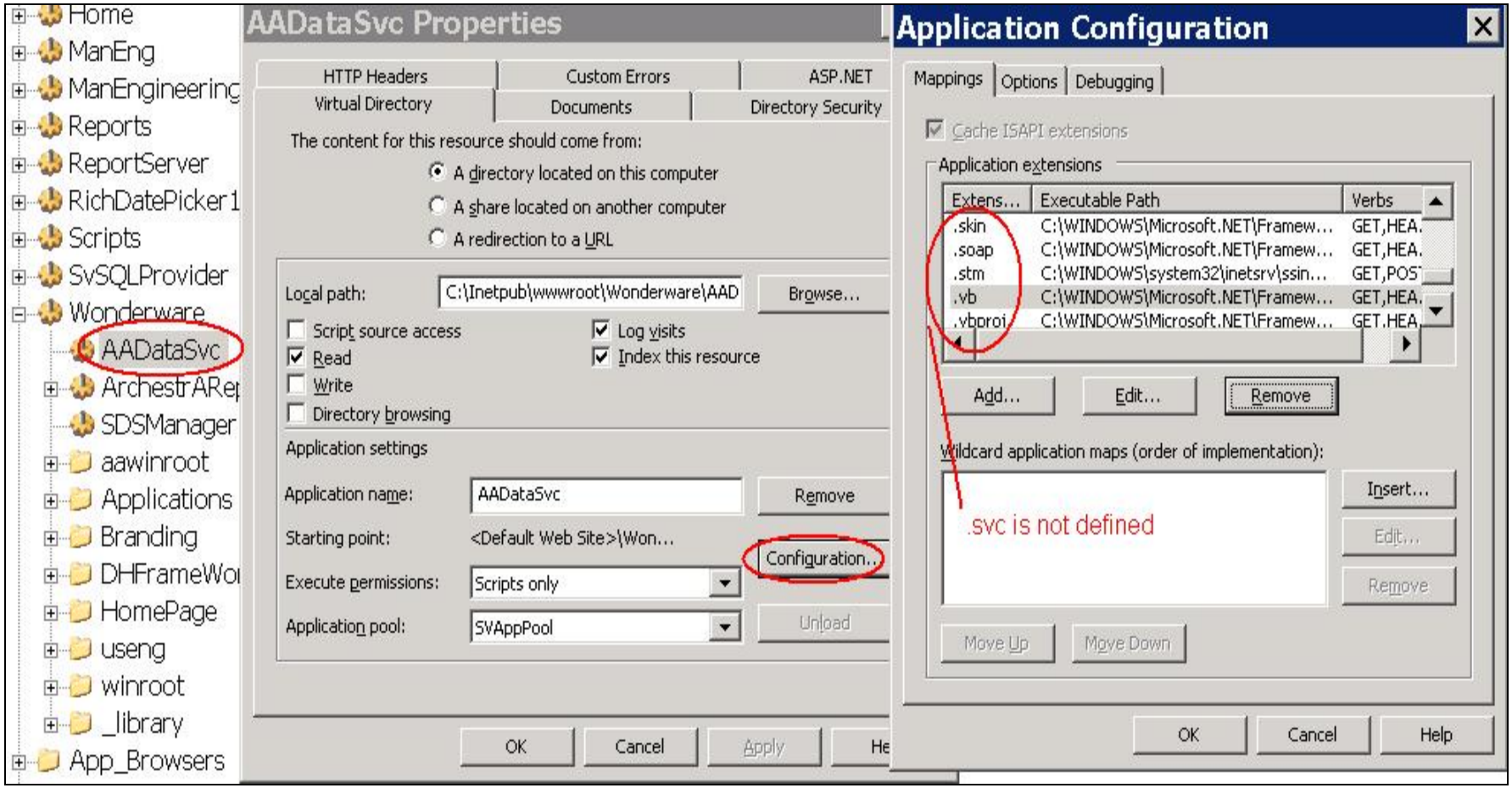

Figure 16: .svc Extension is Not In the Mappings Extensions List

4. Click **Add**.
5. Browse to the .svc Executable and configure the other options as shown in Figure 17 (below).

   In the **Executable** field, locate **[C]:\Windows\Microsoft.NET\Framework\v2.0.50727\aspnet_isapi.dll**.

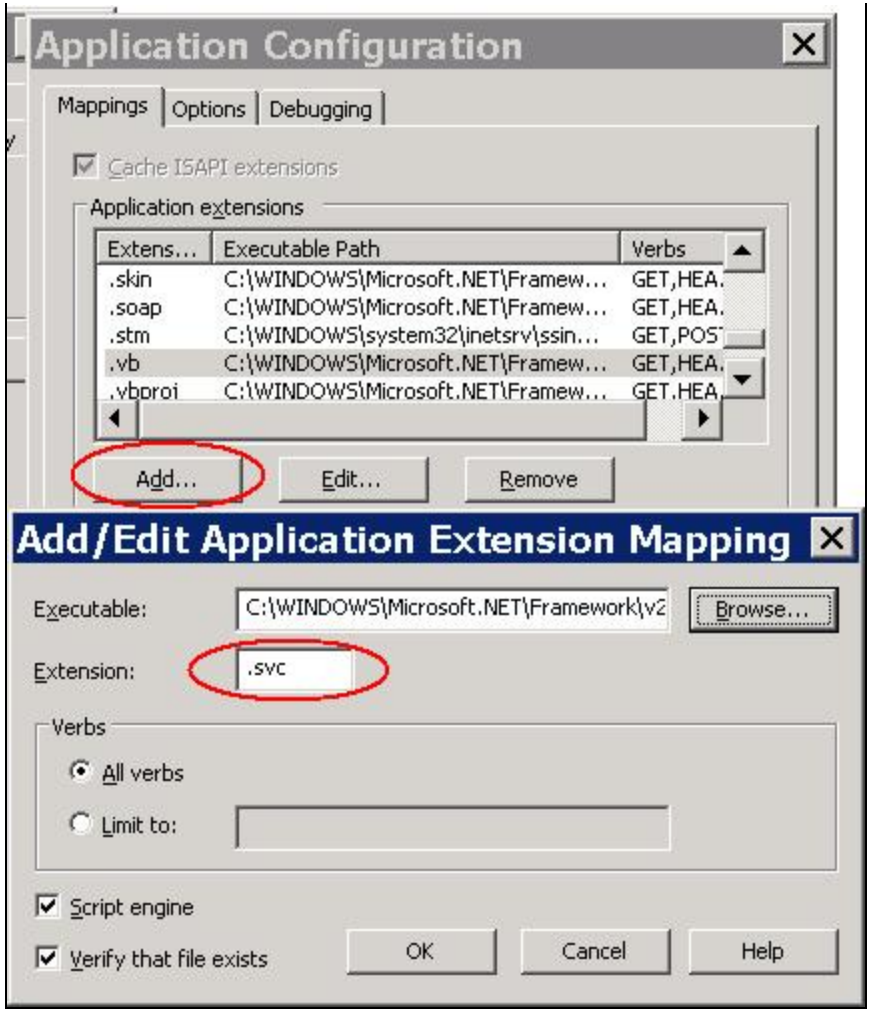

Figure 17: Application Extension Mapping Properties

For IIS 7.x

1. Double-click the **Handler Mappings** item.

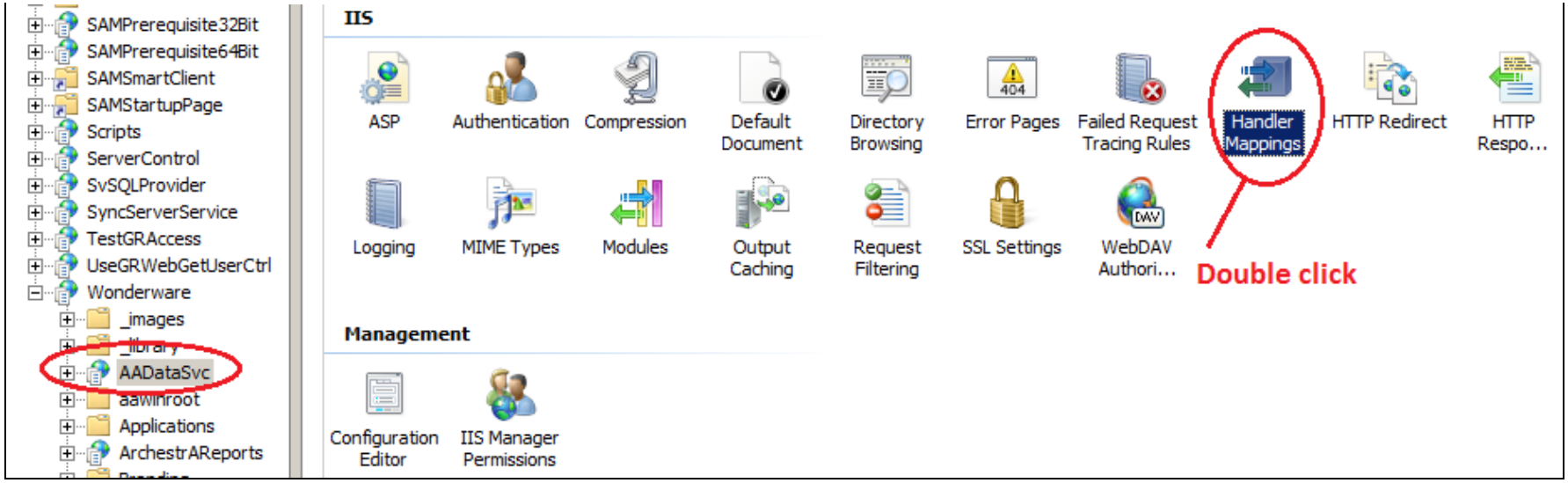

FIGURE 18: HANDLER MAPPINGS FOR AADATASVC

2. Right-click anywhere in the list view and click **Add Script Map** (Figure 19 below).

   The Handler Map for .NET Framework 2.0 and 32-bit application is missing.

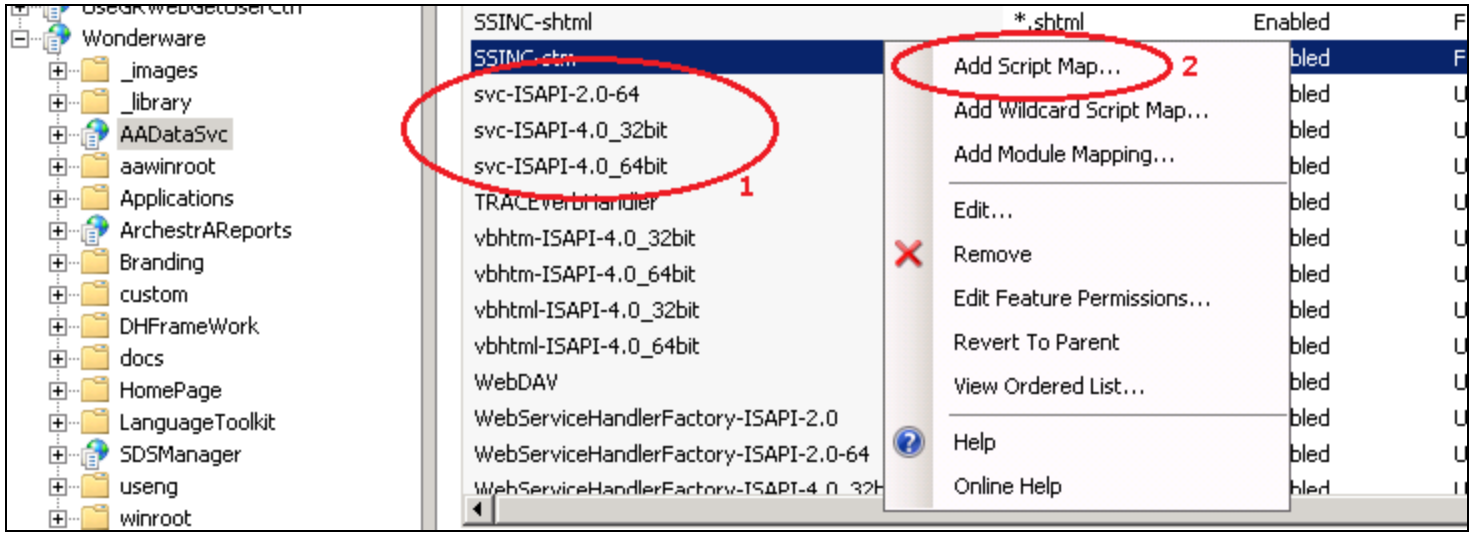

FIGURE 19: ADD SCRIPT MAP

3. Add the **Request Path**: ***.svc** (Figure 20 below).
4. Add the **Executable** path **C:\Windows\Microsoft.NET\Framework\v2.0.50727\aspnet_isapi.dll**.
5. Add the **Name**: **svc_ISAPI-2.0**.

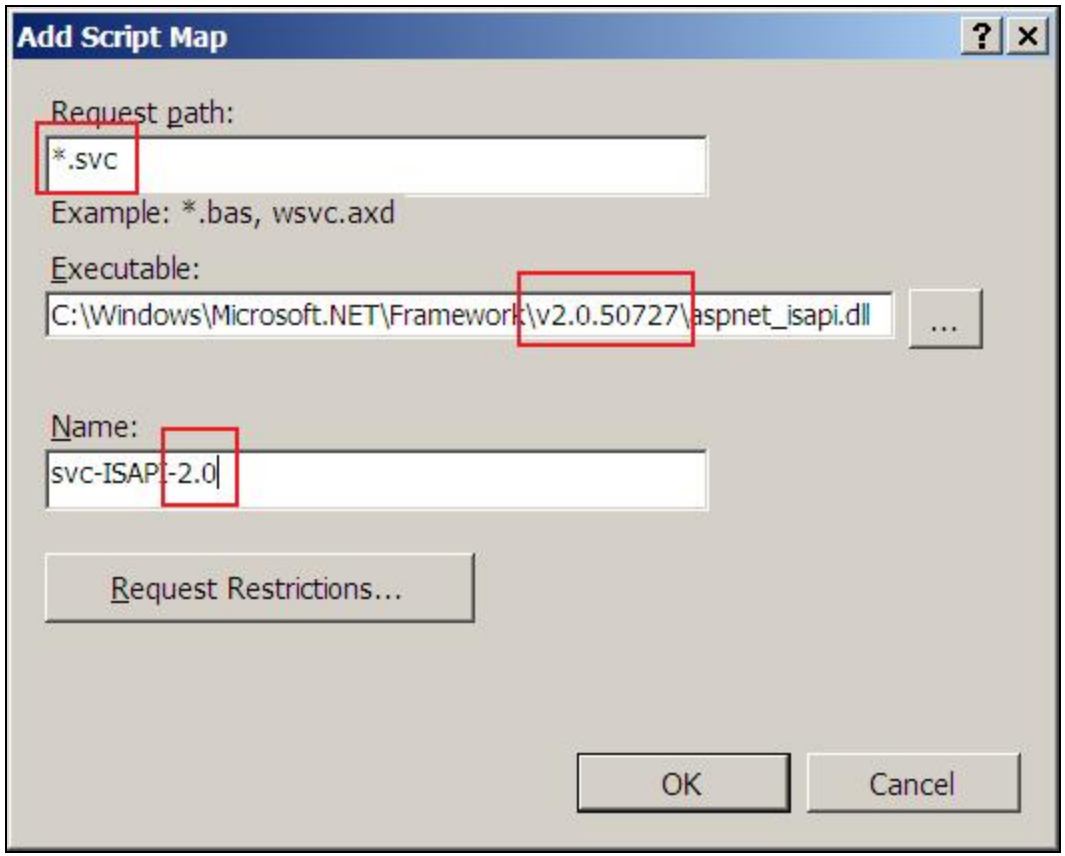

FIGURE 20: ADD SCRIPT MAP *SVC

- **Using Information server Versions 2012 R2 and later (WIS 5.0 and later)**
  1. On the WIS Server, click **Start/Run** and type **Inetmgr**.
  2. Highlight the **Wonderware** virtual directory
  3. Double-click **Handler Mappings** in the right pane.
  4. Right-click any blank area and click **Add Script Map** (Figure 21 below).

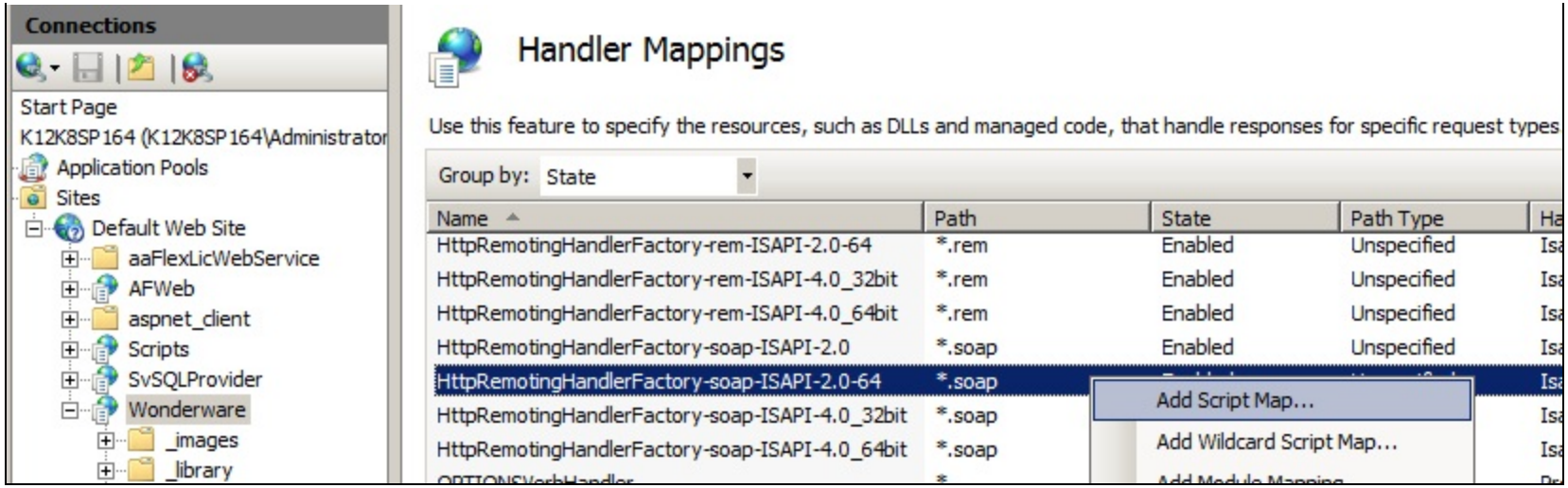

Figure 21: Add Script Map for *.ASMX (32-bit)

5. Type the following information (Figure 22 below):

   - Request Path = ***.svc**
   - Executable = **%windir%\Microsoft.NET\Framework\v4.0.30319\aspnet_isapi.dll**
   - Name = **svc-ISAPI-4.0_32bit**

6. Click **OK**.

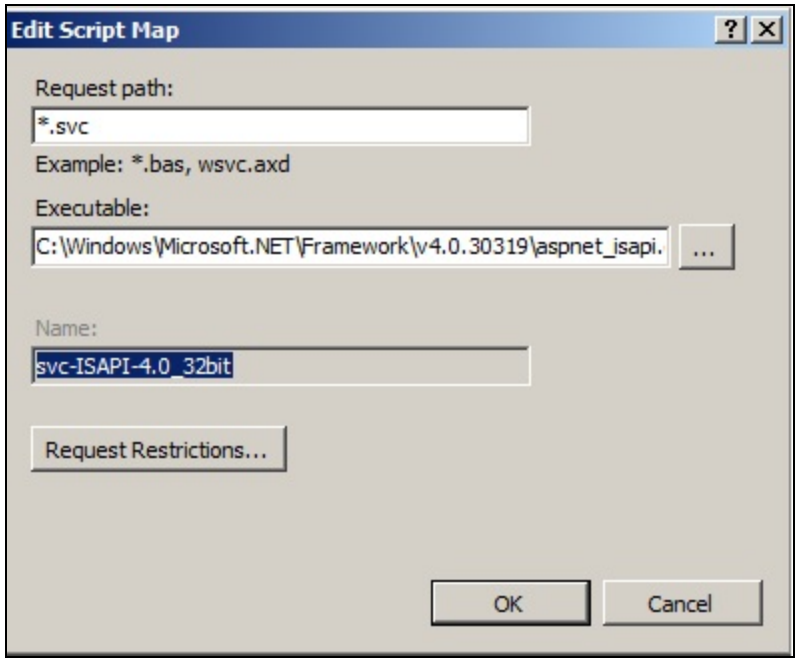

Figure 22: Add Script Map

7. Click **Yes**.

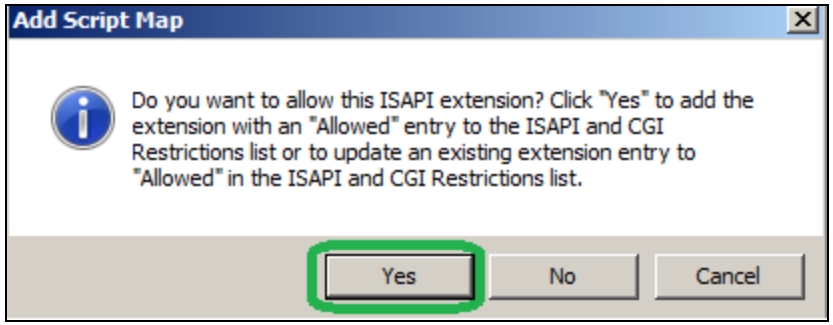

FIGURE 23: ADD ISAPI EXTENSION

8. Repeat steps 1-7 above with the following information for Step 5 (Figure 24 below).

   - Request Path = ***.svc**
   - Executable = **%windir%\Microsoft.NET\Framework64\v2.0.50727\aspnet_isapi.dll**
   - Name = **svc-ISAPI-2.0-64**

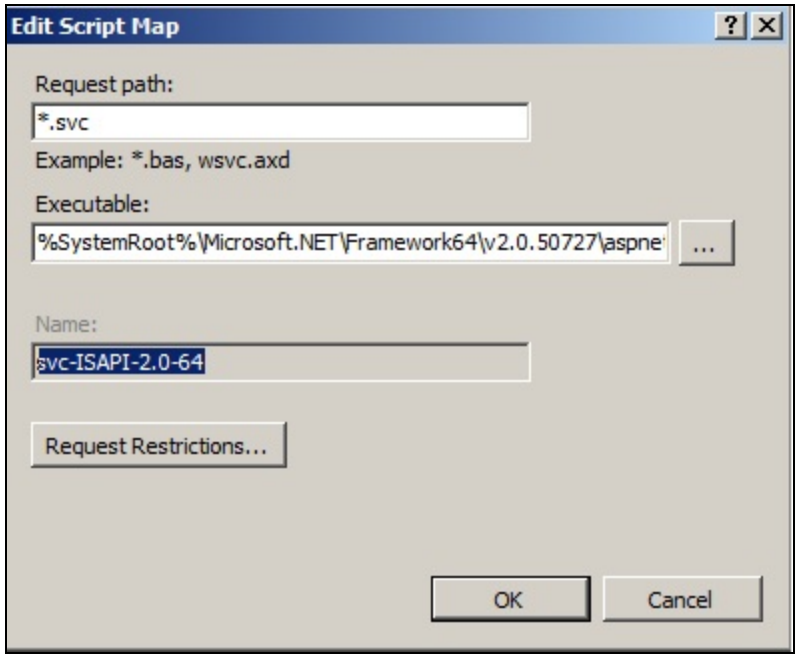

FIGURE 24: PAGEHANDLERFACTORY MAPPING

9. Repeat steps 1-7 above with the following Information for Step 5 (Figure 25 below).

   - Requested Path = ***.asmx**
   - Executable = **%windir%\Microsoft.NET\Framework\v4.0.30319\aspnet_isapi.dll**
   - Name = **WebServiceHandlerFactory-ISAPI-4.0_32bit**

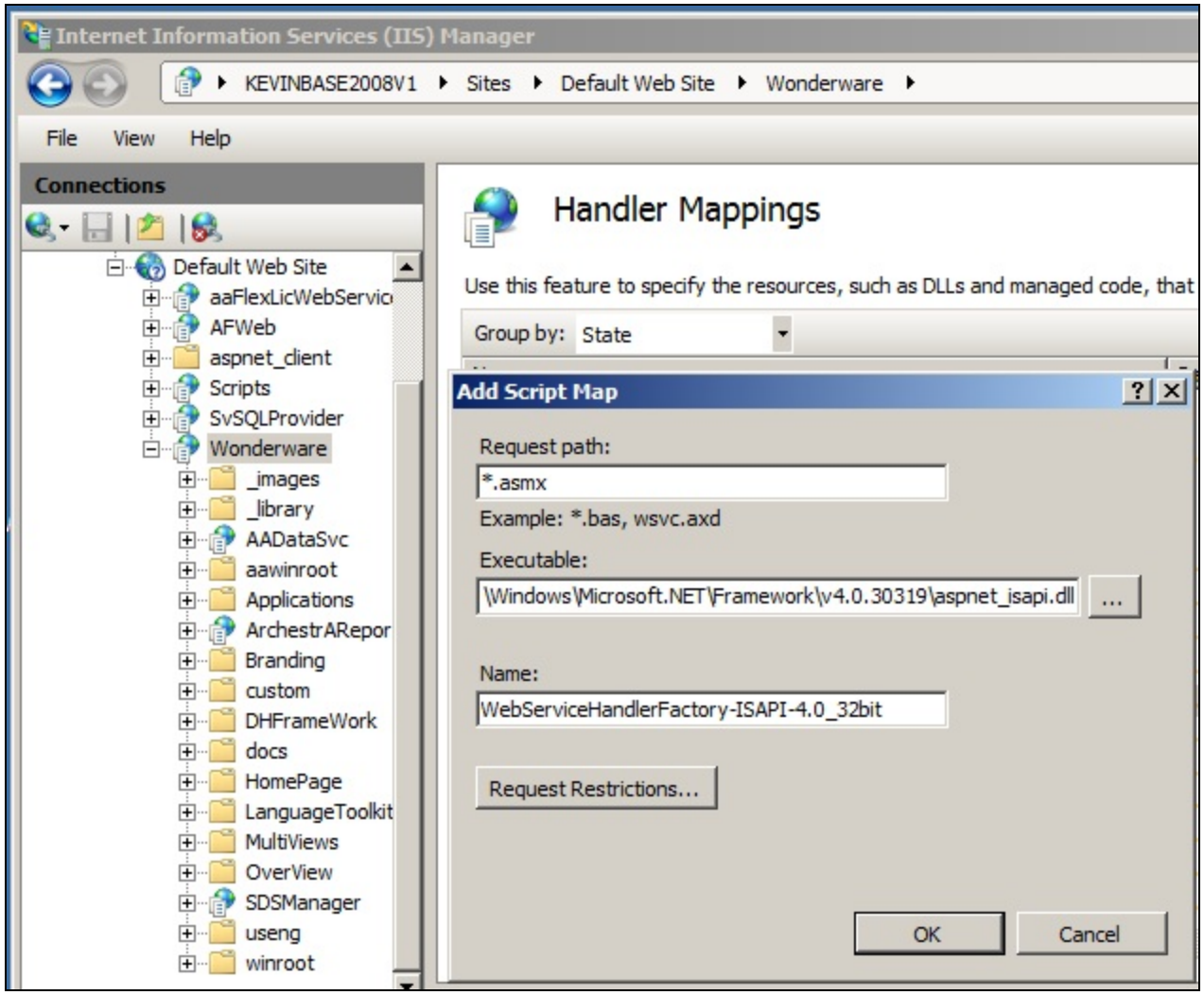

FIGURE 25: WEBSERVICEHANDLERFACTORY MAPPING – 32-BIT

10. Repeat steps 1-7 above with the following Information for Step 5 (Figure 26 below).

- Requested Path = ***.asmx**
- Executable = **%windir%\Microsoft.NET\Framework64\v4.0.30319\aspnet_isapi.dll**
- Name = **WebServiceHandlerFactory-ISAPI-4.0_64bit**

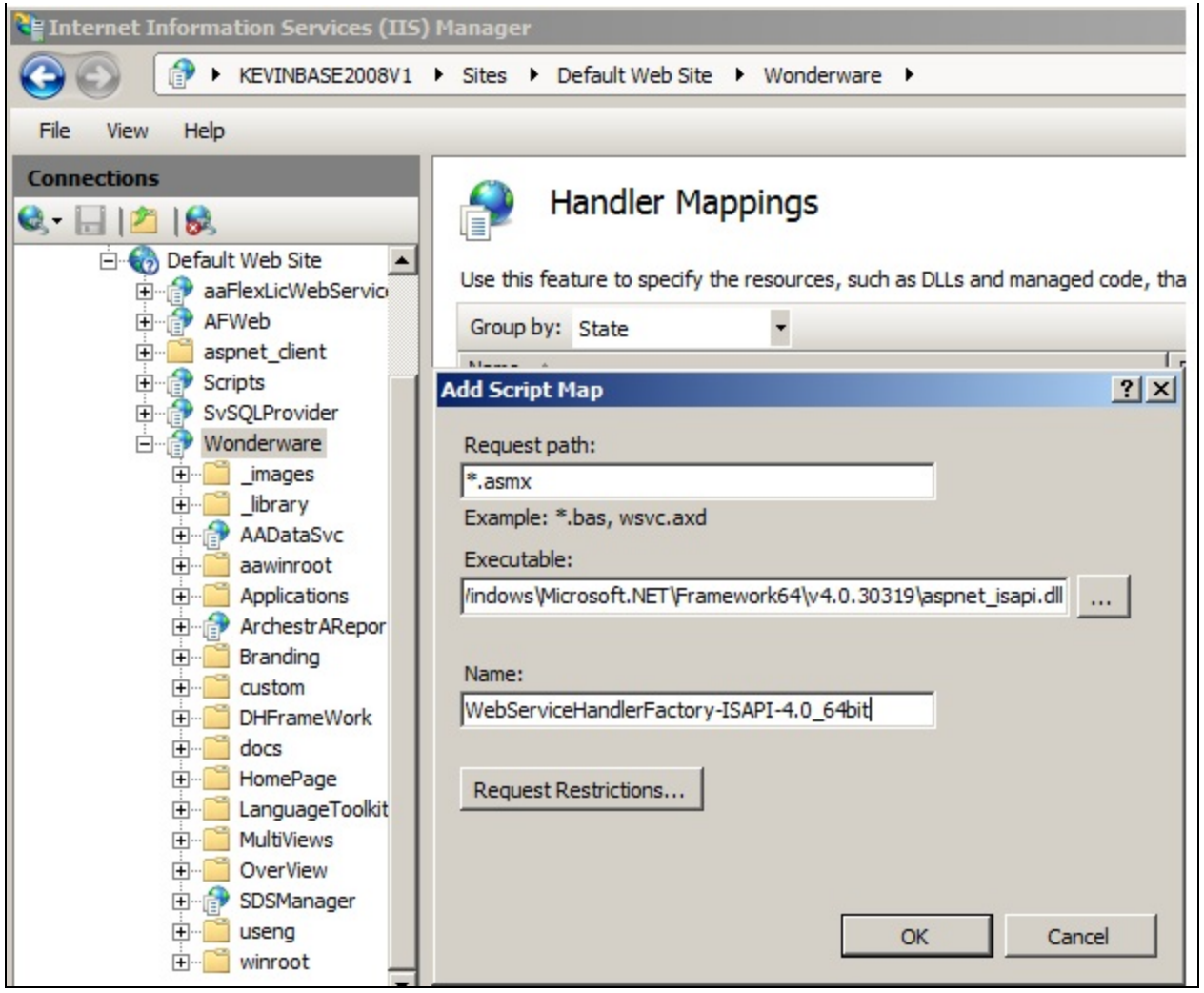

Figure 26: WebServiceHandlerFactory Mapping - 64-bit

Note: aspnet_isapi.dll is required for connecting between WCF Service and AADataSvc's ASP.Net pages. Every request such as From/To WIS WCF Service that fires ASP.NET events for Live-data must go through an extension that is registered and points at aspnet_isapi.dll. In our case, the extension is **.svc**.

Scenario: Conflicting (Multiple) Endpoints

When you browse the **SymbolDataService.svc** file in IIS, you might see the following Exception message (Figure 27 below).

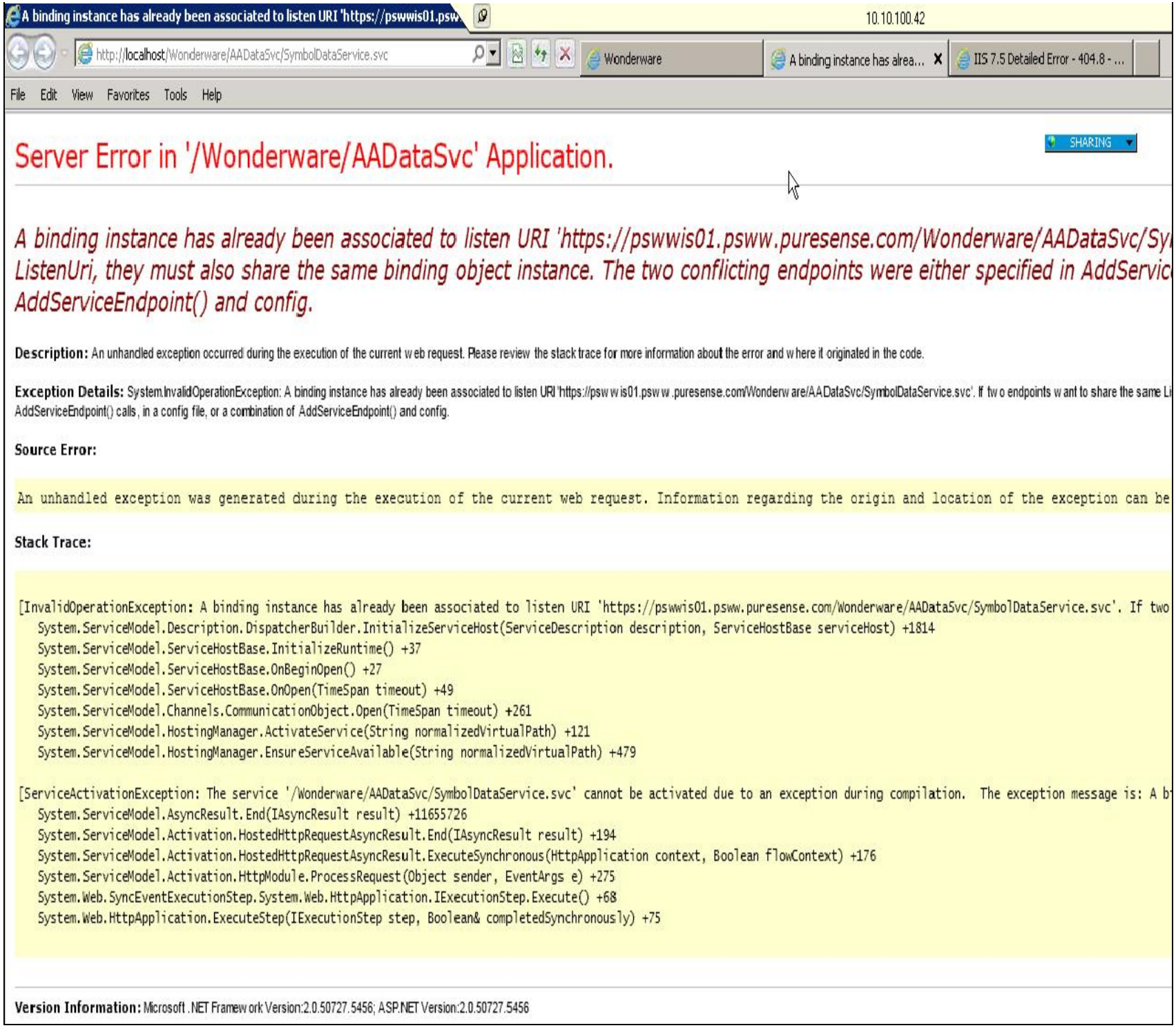
A binding instance has already been associated to listen URI 'https://pswwis01.psw
10.10.100.42
http://localhost/Wonderware/AADataSvc/SymbolDataService.svc
Wonderware
A binding instance has alrea...
IIS 7.5 Detailed Error - 404.8 - ...
File Edit View Favorites Tools Help
SHARING
Server Error in '/Wonderware/AADataSvc' Application.
A binding instance has already been associated to listen URI 'https://pswwis01.psww.puresense.com/Wonderware/AADataSvc/Sy
ListenUri, they must also share the same binding object instance. The two conflicting endpoints were either specified in AddServic
AddServiceEndpoint() and config.
Description: An unhandled exception occurred during the execution of the current web request. Please review the stack trace for more information about the error and where it originated in the code.
Exception Details: System.InvalidOperationException: A binding instance has already been associated to listen URI 'https://pswwis01.psww.puresense.com/Wonderware/AADataSvc/SymbolDataService.svc'. If two endpoints want to share the same Li
AddServiceEndpoint() calls, in a config file, or a combination of AddServiceEndpoint() and config.
Source Error:
An unhandled exception was generated during the execution of the current web request. Information regarding the origin and location of the exception can be
Stack Trace:
[InvalidOperationException: A binding instance has already been associated to listen URI 'https://pswwis01.psww.puresense.com/Wonderware/AADataSvc/SymbolDataService.svc'. If two
System.ServiceModel.Description.DispatcherBuilder.InitializeServiceHost(ServiceDescription description, ServiceHostBase serviceHost) +1814
System.ServiceModel.ServiceHostBase.InitializeRuntime() +37
System.ServiceModel.ServiceHostBase.OnBeginOpen() +27
System.ServiceModel.ServiceHostBase.OnOpen(TimeSpan timeout) +49
System.ServiceModel.Channels.CommunicationObject.Open(TimeSpan timeout) +261
System.ServiceModel.HostingManager.ActivateService(String normalizedVirtualPath) +121
System.ServiceModel.HostingManager.EnsureServiceAvailable(String normalizedVirtualPath) +479
[ServiceActivationException: The service '/Wonderware/AADataSvc/SymbolDataService.svc' cannot be activated due to an exception during compilation. The exception message is: A b
System.ServiceModel.AsyncResult.End(IAsyncResult result) +11655726
System.ServiceModel.Activation.HostedHttpRequestAsyncResult.End(IAsyncResult result) +194
System.ServiceModel.Activation.HostedHttpRequestAsyncResult.ExecuteSynchronous(HttpApplication context, Boolean flowContext) +176
System.ServiceModel.Activation.HttpModule.ProcessRequest(Object sender, EventArgs e) +275
System.Web.SyncEventExecutionStep.System.Web.HttpApplication.IExecutionStep.Execute() +68
System.Web.HttpApplication.ExecuteStep(IExecutionStep step, Boolean& completedSynchronously) +75
Version Information: Microsoft .NET Framework Version:2.0.50727.5456; ASP.NET Version:2.0.50727.5456

Figure 27: Binding Exception Message

This message indicates that something is wrong in the web.config under [C]:\inetpub\wwwroot\Wonderware\AADataSvc\ directory.

Web.config configured by WIS's Configurator has the following default section on <endpoint address>. If your web.config file has multiple sections, try to comment out the rest of the <endpoint address> element(s). Call Wonderware Technical Support if the problem still exists afterwards.

```
<client>
  <!-- <endpoint address="net.tcp://localhost:5666/LMXPublisher" binding="netTcpBinding" bindingConfiguration="pipeTcpBinding"
  contract="ArchestrA.Visualization.LMXPublisher.ILMXPublisherSvc" name="LMXEndpoint"/> -->
  <endpoint address="net.pipe://localhost/LMXPublisher" binding="netNamedPipeBinding" bindingConfiguration=""
    contract="ArchestrA.Visualization.LMXPublisher.ILMXPublisherSvc" name="LMXEndpoint" />
</client>
```

Figure 28: Multiple Endoint Address Elements

Scenario: AADataSvc Points to Wrong .NET Framework

- **When using Information Server Versions prior to 2012 R2 (WIS 5.0)**

WIS versions prior to version 5 do not support .NET Framework 4.0.

If there is no Live-data and you see the following Handler Mappings message when you click on Handler-Mappings for AADataSvc level:

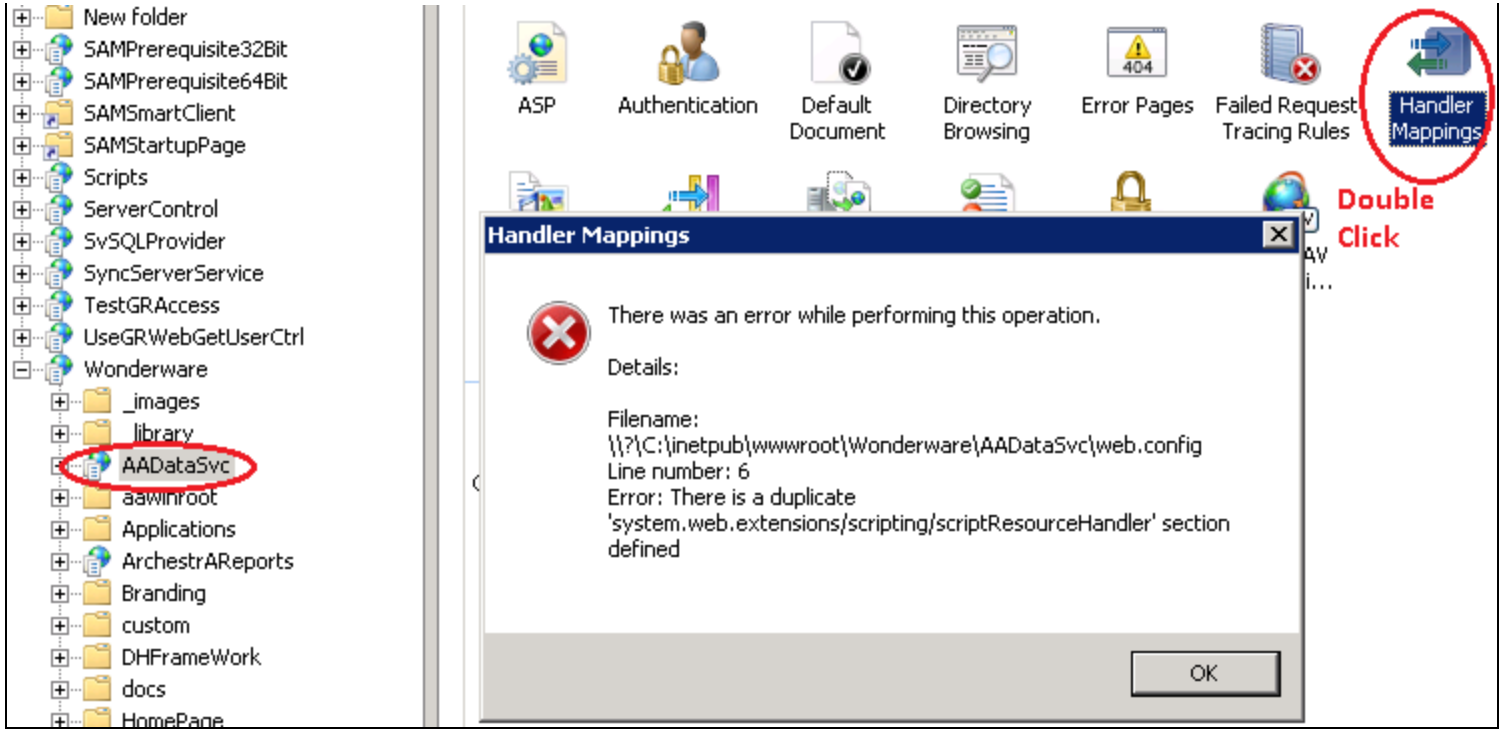

FIGURE 29: HANDLER-MAPPINGS ERROR MESSAGE

Recall that that **AADataSsc's** Application Pool is called **SVAppPool**.

- Go to the **SVAppPool** Advanced Settings. The (General) list shows that the .NET Framework Version is 4.0.

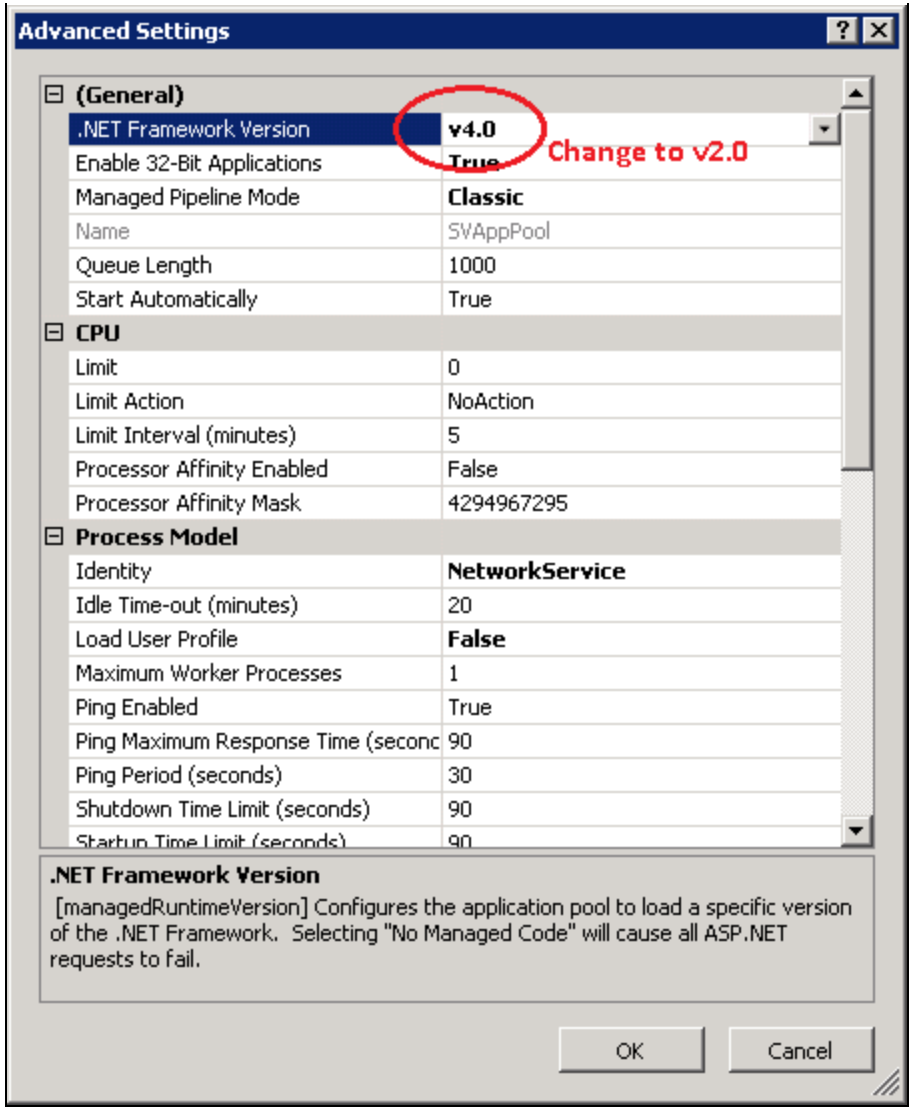

Figure 30: Change .NET Framework Version to v2.0

This is the problem. Change it back to **2.0**, and the above Error message will no longer appear.

Note: No Action is required if you are using WIS version 5.0 and later (System Platform 2012 R2)

Scenario: Applying HotFix Causes No Live-data Update

When applying a HotFix, you may see an instruction from the HotFix's Readme file similar to:

Backup the existing file and copy the HotFix file into the directory.

If the above instruction is for a .NET DLL, it is not detailed enough, and will cause a problem. The following example is derived from a support case.

The customer applied HotFix **L0018926** for fixing the High CPU problem when running WIS ArchestrA Graphics.

However, after applying the HotFix, there is no Live-data.

Root Cause

The cause of this problem is that two of the same .NET DLLs are in the *same* directory after you have renamed the original .NET DLL.

In the runtime flow, the WIS ArchestrA Graphics subsystem is confused when it needs to load the .NET DLL because there are two DLLs with same assembly name and version.

This scenario causes a Common Problem when applying a HotFix that contains .NET DLL(s).

Note: Renaming a .NET DLL does *not* change its original Assembly name and version number.

Resolution

Cut/paste the original .NET DLL(s) to another folder/location. The original DLL must be removed from its working directory, and the replacement put in.

Scenario: SMC Log Flogs for Live-data

Sometimes, the no Live-data error can still exist after completing these scenarios. In order to troubleshoot further, the WIS ArchestrA Graphics Subsystem defines three SMC Log flags for helping to collect Runtime information. The flags are:

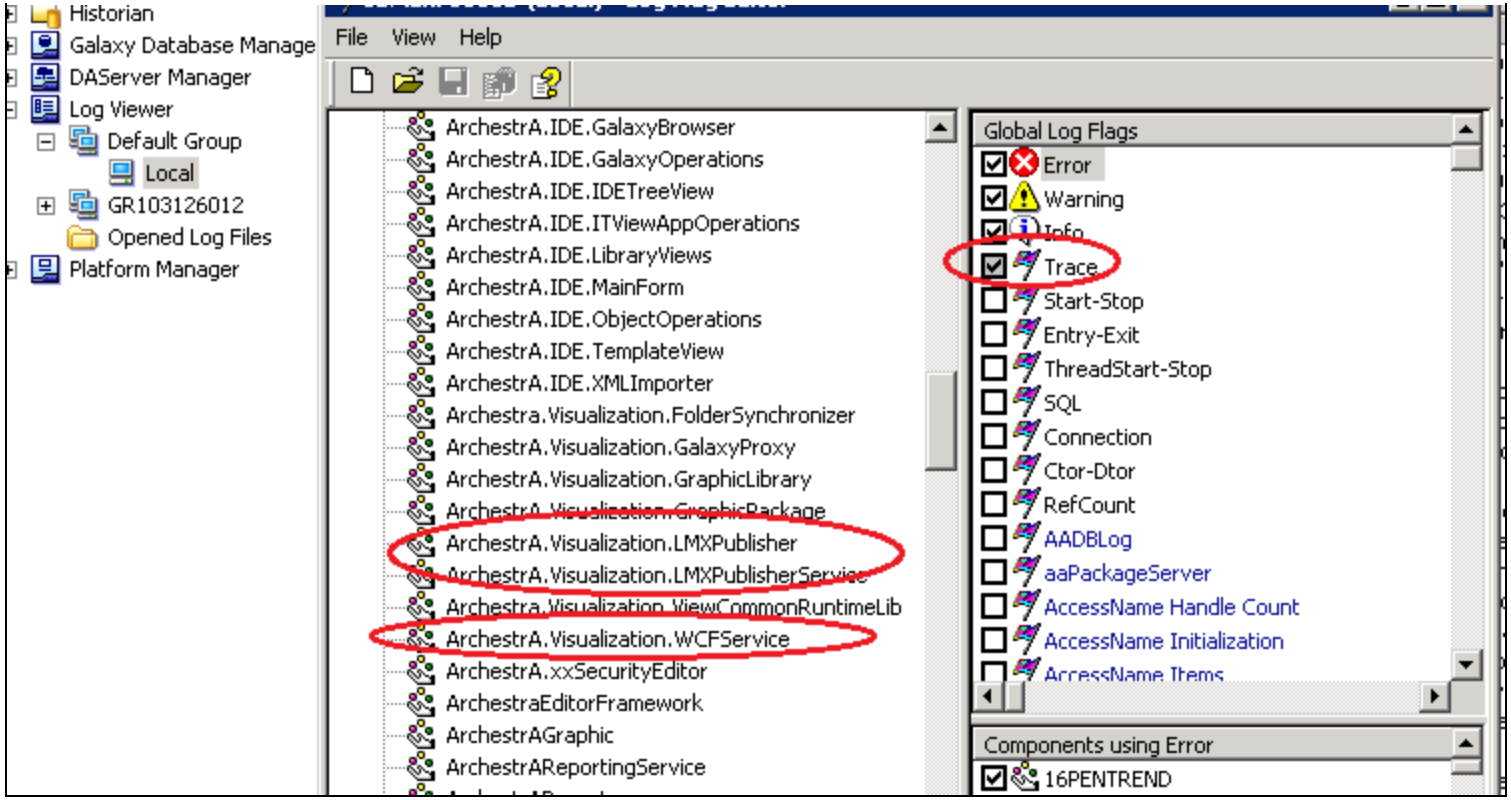

Figure 31: Turn on Trace for the Above Flags

The most important flag is **ArchestrA.Visualization.WCFService**. After you enable **Trace**, open the page that has no Live-data. You should see some log entries for this flag.

Reference

- **Tech Note 984 Adding IIS Handler Mappings: Checklist**

E. Xu, K. Nourbakhsh

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