

*NOTE: This article and all content are provided on an "as is" basis without any warranties of any kind, whether express or implied, including, but not limited to the implied warranties of merchantability, fitness for a particular purpose, and non-infringement. In no event shall Wonderware NorthEast be liable for any special, indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortuous action, arising out of or in connection with the use or performance of information contained in this article.*

## Introduction

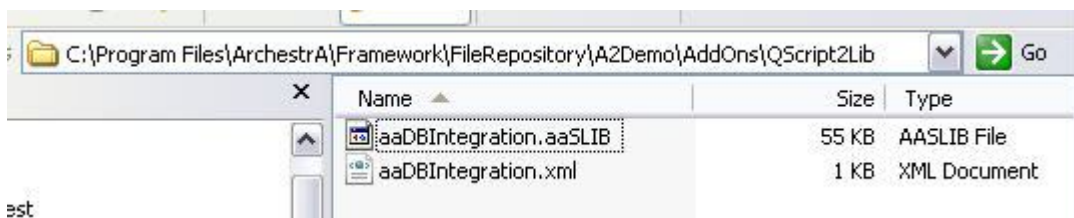
Wonderware's Industrial Application Server allows the developer to import and install custom .NET libraries developed in Microsoft Visual Studio to be used by objects in the galaxy. There may be occasions where a developer would want to remove the libraries they installed from the galaxy repository. The following article explains two ways to do this, manually, and via a SQL script.

In this example, the aaDBIntegration.aaSLIB script library was imported into the A2Demo galaxy from the SQLData Components package. However, it was decided the project did not require the library, and the customer now wants it removed from the galaxy. In order to accomplish this task, the files and database entries on the galaxy repository node must be deleted.

## The Manual Method

First, delete the following files, if they exist (where yourScriptFunctionDllName is the name of the DLL or aaSLIB, in this case, aaDBIntegration):

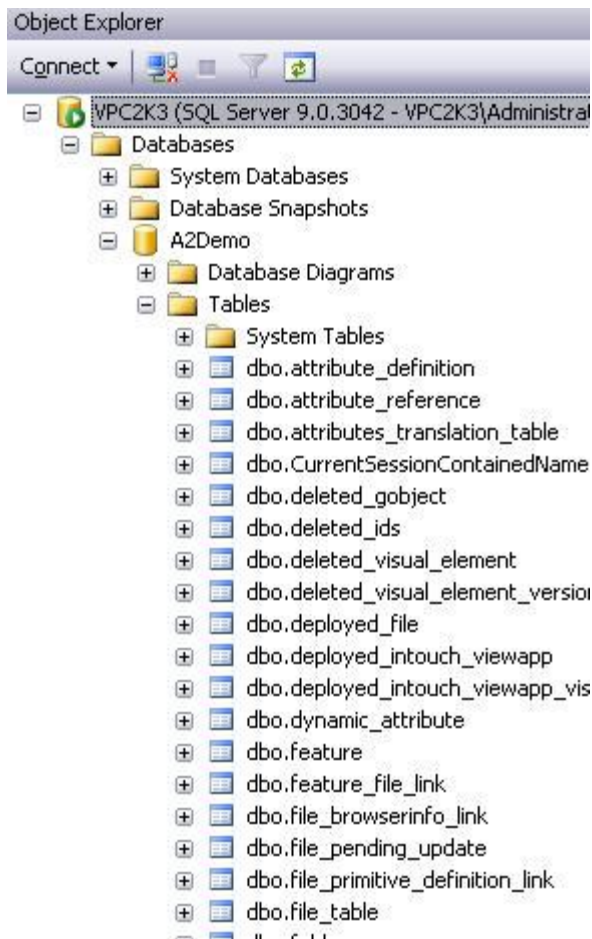
- C:\Program Files\ArchestrA\Framework\FileRepository\GalaxyName\AddOns\QScript2Lib\
  - yourScriptFunctionDllName.Net.aaSLIB
  - yourScriptFunctionDllName.Net.xml



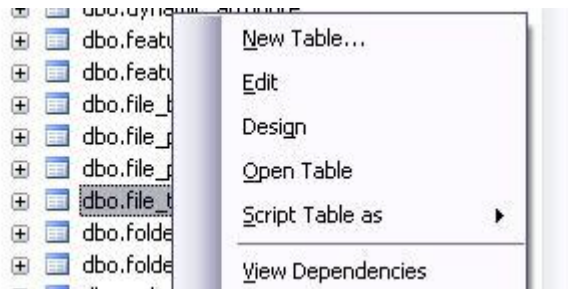
- C:\Program Files\ArchestrA\Framework\Bin\ScriptTemp
  - yourScriptFunctionDllName.idf
  - yourScriptFunctionDllName.Net.aaSLIB
  - yourScriptFunctionDllName.Net.cs
  - yourScriptFunctionDllName.Net.dll
  - yourScriptFunctionDllName.Net.xml
- C:\Program Files\ArchestrA\Framework\FileRepository\GalaxyName\Vendors\ArchestrA
  - yourScriptFunctionDllName.dll
  - yourScriptFunctionDllName.Net.dll

Next, now that the files have been removed from the repository, delete the Galaxy Database's entries of the file: (images show SQL 2005)

- Open SQL Server Enterprise Manager (SQL 2000) or SQL Server Management Studio (SQL 2005).
- In the manager, browse to Databases->|your galaxy name|->Tables.
- Open the tables by expanding the folder (SQL 2005) or double-clicking on the Tables icon (SQL2000).



- Open the "file\_table" table by right clicking on file\_table, and selecting Open table (SQL 2005) or Open Table->Return All rows (SQL 2000).



- Look for your DLL name in the “file\_name” column, and then write down the corresponding number shown in the “file\_id” column.

VPC2K3.A2Demo - dbo.file_table			
	file_id	file_name	ve
	319	aaHistClientTrendObject.dll	Inv
	320	DundasWinChart.dll	Inv
	321	aaHistClientTimeRangePicker.dll	Inv
	322	aaUniFlexLicClient.dll	Inv
	323	aaBaseFlexLicClient.dll	Inv
	324	aaFlexLicClient.dll	Inv
	325	aaHttpFlexLicClient.dll	Inv
	326	aaFlexLicClientUI.dll	Inv
	327	System.Windows.Forms.dll	Mic
▶	328	aaDBIntegration.dll	Arc
	329	aaDCM.msm	Arc

- Open the “feature\_file\_link” table. If it exists, delete the row that contains the corresponding file\_id.
- Open the “feature” table. Delete the row that contains the file name.

VPC2K3.A2Demo - dbo.feature		VPC2K3.A2Demo - dbo.file_table	
	feature_id	feature_name	feature_type
▶	1	aaDBIntegration	QuickScript2 Library
*	NULL	NULL	NULL

- Open the “file\_pending\_update” table. If it exists, delete the row that contains the corresponding file\_id.
- Open the “file\_primitive\_definition\_link” table. Delete the row that contains the corresponding file\_id. **\*NOTE:** This table can be very large. Alternatively, you can search for the file\_id by

entering the following into a query (where <your file id> is the number you found in the "file\_table" table:

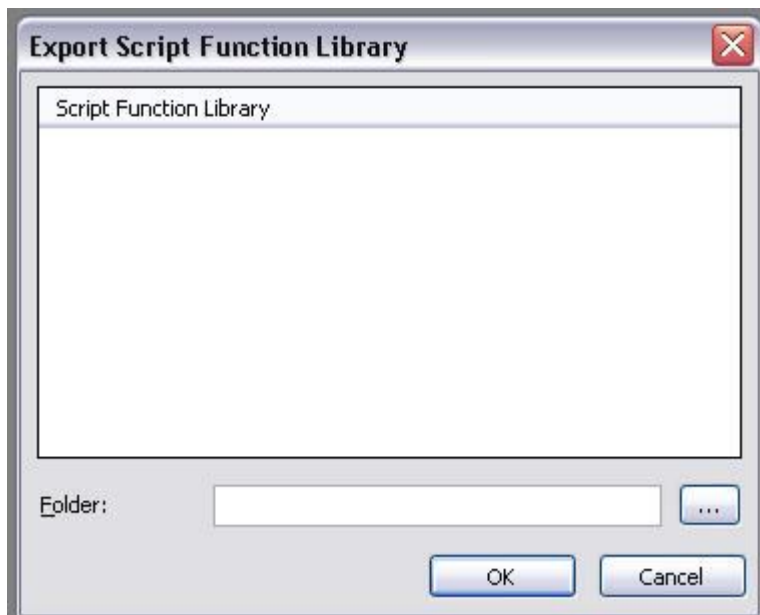
```
Select * from file_primitive_definition_link where file_id = <your file id>
```

Then, you can delete the data using the following query:

```
Delete from file_primitive_definition_link where file_id = <your file id>
```

- Finally, reopen the "file\_table" table, and delete the row that contains your DLL.

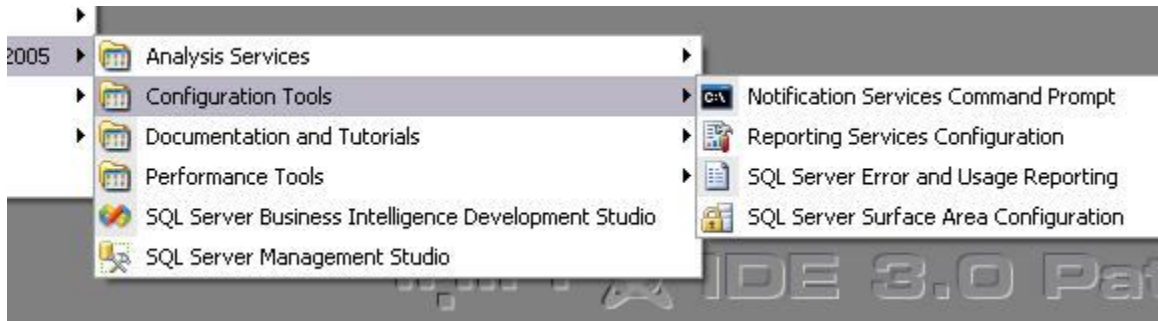
At this point, the DLL has been removed. To verify, go into the IDE and select Galaxy->Export->Script Function Library – the library should not be in the list.



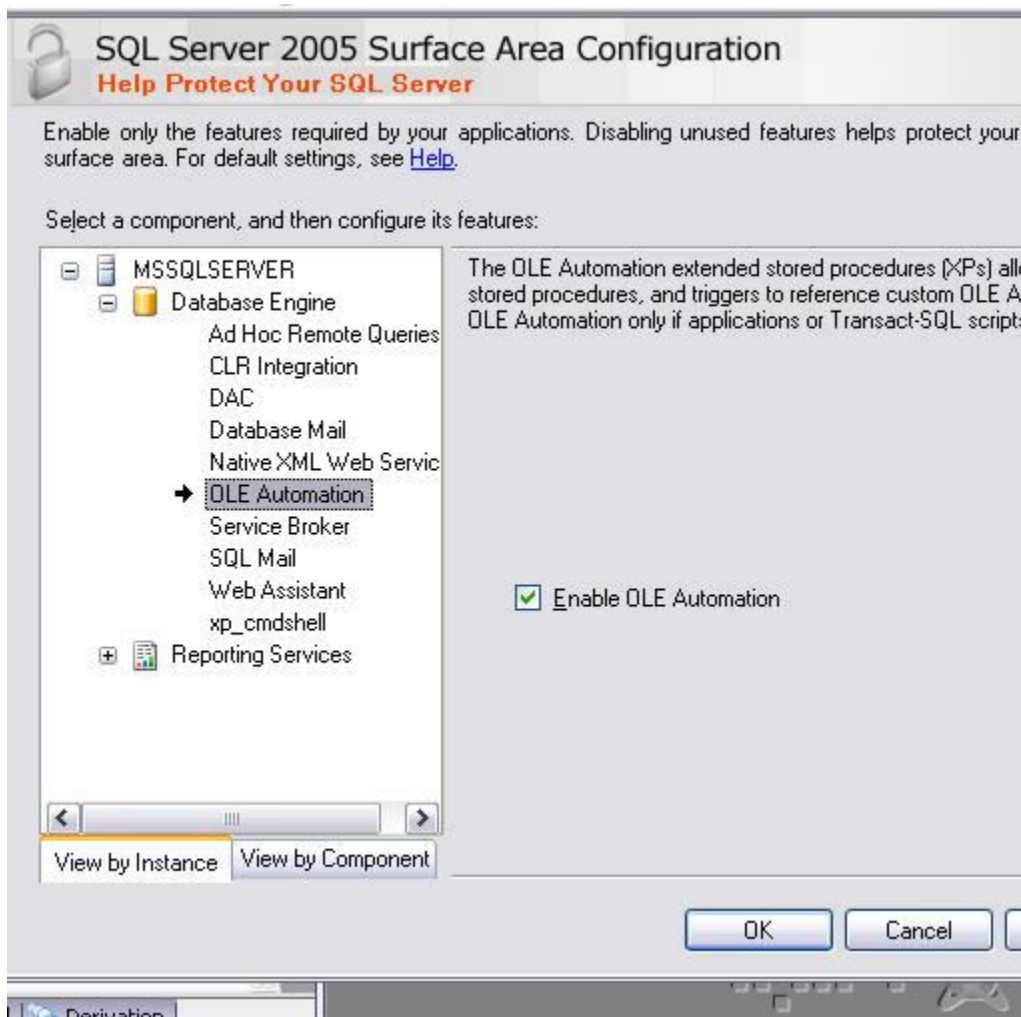
## The SQL Method

While the above method will work, it does require that every one of the entries above must be removed; if one is missed, then the library still technically exists in the galaxy. In order to simplify this, a stored procedure can be written that uses OLE automation to delete the files, as well as then delete the appropriate rows from the galaxy database. In order to use a procedure like this, OLE automation must be enabled. This can be enabled in the SQL Server Surface Area Configuration (Start->Programs->SQL Server 2005->Configuration Tools->SQL Server Surface Area Configuration).

**TechTip: Removing Custom .NET libraries from Industrial Application Server 3.0**



Once in the Surface Area Configuration, Select Surface Area Configuration for Features, then select the option for OLE Automation, and check to enable. Hit apply, and then OK to close out the window.



Then, install the stored procedure written at the end of this article into the galaxy database (database named your galaxy name) by running the SQL script at the end of this article in a query window. **NOTE:** The stored

## **TechTip: Removing Custom .NET libraries from Industrial Application Server 3.0**

procedure provided has been written for Application Server 3.0 and SQL 2005 and may not work for earlier versions.

After the stored procedure has been installed, to execute the stored procedure, open a query window and execute the following code:

```
declare @result integer

execute @result = dbo.RemoveMyDll 'xxxx '
```

Where xxxx is the name of the DLL you want to remove. At this point, the stored procedure will execute all the actions written in the manual method above. To verify the DLL has been removed, again, go into the IDE and select Galaxy->Export->Script Function Library – the library should not be in the list.

### **Stored Procedure: RemoveMyDll.sql**

(Thanks to Rich Liddell for providing a majority of the stored procedure code)

```
if exists (select * from dbo.sysobjects where id = object_id(N'[dbo].[RemoveMyDll]') and OBJECTPROPERTY(id, N'IsProcedure') = 1)
drop procedure [dbo].[RemoveMyDll]
GO
```

```
SET QUOTED_IDENTIFIER OFF
GO
SET ANSI_NULLS OFF
GO
```

```
CREATE PROCEDURE RemoveMyDll @dllname nvarchar(256) AS
```

```
-- The following files will be deleted
```

```
--C:\Program Files\Archestra\Framework\FileRepository\GalaxyName\AddOns\QScript2Lib\
--yourScriptFunctionDllName.Net.aaSLIB
--yourScriptFunctionDllName.Net.xml
```

```
--C:\Program Files\Archestra\Framework\Bin\ScriptTemp
--yourScriptFunctionDllName.idf
--yourScriptFunctionDllName.Net.aaSLIB
--yourScriptFunctionDllName.Net.cs
--yourScriptFunctionDllName.Net.dll
--yourScriptFunctionDllName.Net.xml
```



**TechTip: Removing Custom .NET libraries from Industrial Application Server 3.0**

```
--C:\Program Files\Archestra\Framework\FileRepository\GalaxyName\Vendors\Archestra
--yourScriptFunctionDllName.dll
--yourScriptFunctionDllName.Net.dll
```

```
--The Galaxy database will have rows deleted from the following tables
--delete from dbo.file_primitive_definition_link where file_ID in (select file_id from file_table where file_name
like @dllname + '%')
--delete from feature_file_link where file_ID in (select file_id from file_table where file_name like @dllname +
'%')
--delete from dbo.file_pending_update where file_ID in (select file_id from file_table where file_name like
@dllname + '%')
--delete from dbo.feature where feature_name like @dllname + '%'
--delete from dbo.file_table where file_name like @dllname + '%'
```

```
Declare @id as integer
Declare @Archestra_Dir VARCHAR(256)
Declare @ScriptTempPath VARCHAR(256)
Declare @ArchestraPath VARCHAR(256)
Declare @QScript2LibPath VARCHAR(256)
Declare @galaxyName VARCHAR(256)
```

```
-- Get the Install Path from the registry
EXECUTE master..xp_regread
'HKEY_LOCAL_MACHINE',
'SOFTWARE\Archestra\Framework',
'InstallPath',
@Archestra_Dir OUTPUT
```

```
--Get the Galaxy Name
select @galaxyName = tag_name from dbo.internal_list_objects_view where hierarchical_name = 'Galaxy_001'
```

```
--Get all the file paths
set @QScript2LibPath = @Archestra_Dir + 'FileRepository\' + @galaxyName + '\AddOns\QScript2Lib\' +
@dllname + '*.*'
set @ScriptTempPath = @Archestra_Dir + 'Bin\ScriptTemp\' + @dllname + '*.*'
set @ArchestraPath = @Archestra_Dir + 'FileRepository\' + @galaxyName + '\Vendors\Archestra\' + @dllname
+ '*.*'
```

```
DECLARE @hr int
DECLARE @hr1 int
DECLARE @ole_FileSystem int
DECLARE @Success int
DECLARE @Testfile varchar(250)
```

425 Caredean Drive, Horsham, PA 19044

Tel: 215.675.5800

[www.wonderwareNE.com](http://www.wonderwareNE.com)

## TechTip: Removing Custom .NET libraries from Industrial Application Server 3.0

```
set @Success = 0
```

```
EXEC @hr = sp_OACreate 'Scripting.FileSystemObject',  
@ole_FileSystem OUT
```

```
set @QScript2LibPath = @Archestra_Dir + 'FileRepository\' + @galaxyName + '\AddOns\QScript2Lib\' +  
@dllname + '.net.xml'
```

```
print @QScript2LibPath
```

--If the DLL does not exist in the \AddOns\QScript2Lib directory you probably have entered the wrong name or it has already been deleted

--Some times lib is added to the DLL name so I test for it too

```
EXEC @hr = sp_OAMethod @ole_FileSystem, 'FileExists', @Success out, @QScript2LibPath--'C:\Program  
Files\Archestra\framework\FileRepository\Bike\AddOns\QScript2Lib\servicecontrol.Net.xml'
```

if @Success = 0 --Now test if Lib was added to the name

```
Begin
```

```
set @QScript2LibPath = @Archestra_Dir + 'FileRepository\' + @galaxyName + '\AddOns\QScript2Lib\' +  
@dllname + 'lib.net.xml'
```

```
print @QScript2LibPath
```

```
EXEC @hr = sp_OAMethod @ole_FileSystem, 'FileExists', @Success out, @QScript2LibPath--'C:\Program  
Files\Archestra\framework\FileRepository\Bike\AddOns\QScript2Lib\servicecontrol.Net.xml'
```

```
End
```

if @Success = 0 --Now test for lib with no .net

```
Begin
```

```
set @QScript2LibPath = @Archestra_Dir + 'FileRepository\' + @galaxyName + '\AddOns\QScript2Lib\' +  
@dllname + 'lib.xml'
```

```
print @QScript2LibPath
```

```
EXEC @hr = sp_OAMethod @ole_FileSystem, 'FileExists', @Success out, @QScript2LibPath--'C:\Program  
Files\Archestra\framework\FileRepository\Bike\AddOns\QScript2Lib\servicecontrol.Net.xml'
```

```
End
```

if @Success = 0 --Now test for no .net

```
Begin
```

```
set @QScript2LibPath = @Archestra_Dir + 'FileRepository\' + @galaxyName + '\AddOns\QScript2Lib\' +  
@dllname + '.xml'
```

```
print @QScript2LibPath
```

```
EXEC @hr = sp_OAMethod @ole_FileSystem, 'FileExists', @Success out, @QScript2LibPath--'C:\Program  
Files\Archestra\framework\FileRepository\Bike\AddOns\QScript2Lib\servicecontrol.Net.xml'
```

```
End
```

if @Success = 0 --Now test for no .net and no .DLL

```
Begin
```

425 Caredean Drive, Horsham, PA 19044

Tel: 215.675.5800

[www.wonderwareNE.com](http://www.wonderwareNE.com)

## TechTip: Removing Custom .NET libraries from Industrial Application Server 3.0

```
set @QScript2LibPath = @Archestra_Dir + 'FileRepository\' + @galaxyName + '\AddOns\QScript2Lib\' +
left(@dllname,len(@dllname)-4) + '.xml'
print @QScript2LibPath
EXEC @hr = sp_OAMethod @ole_FileSystem, 'FileExists', @Success out, @QScript2LibPath--'C:\Program
Files\Archestra\Framework\FileRepository\Bike\AddOns\QScript2Lib\servicecontrol.Net.xml'
End
```

```
if @Success = 0
Begin
    Print 'Error! Check your DLL name,'" + @dllname + "' is not a ScriptFunction DLL or it has already been
removed'
    Return
End
```

```
--Set the Path back to normal with the wild cards
set @QScript2LibPath = @Archestra_Dir + 'FileRepository\' + @galaxyName + '\AddOns\QScript2Lib\' +
left(@dllname,len(@dllname)-4) + '*.*'
```

```
EXEC @hr = sp_OAMethod @ole_FileSystem, 'DeleteFile',
NULL, @ArchestraPath
```

-- If the object is deployed it will not always delete the files in this directory so I test it first and do not continue the stored procedure unless it is successful

--I never noticed any problems with the other directories

```
IF @hr =0x800a0046
```

```
BEGIN
```

```
    Print 'Error! ' + @dllname + '*.*' + ' File locked could not be deleted'
```

```
    Print @ArchestraPath
```

```
    Print '*** Undeploy Objects using this DLL and close the IDE ***'
```

```
    Return
```

```
END
```

```
EXEC @hr = sp_OAMethod @ole_FileSystem, 'DeleteFile',
NULL, @QScript2LibPath
```

```
IF @hr = 0x800a0046
```

```
BEGIN
```

```
    Print 'Error! ' + @dllname + '*.*' + ' File locked could not be deleted'
```

```
    Print @QScript2LibPath
```

```
    Print '*** Undeploy Objects using this DLL and close the IDE ***'
```

```
    Return
```

```
END
```

**TechTip: Removing Custom .NET libraries from Industrial Application Server 3.0**

---

```
EXEC @hr = sp_OAMethod @ole_FileSystem, 'DeleteFile',  
    NULL, @ScriptTempPath
```

```
IF @hr = 0x800a0046  
BEGIN  
    Print 'Error! ' + @dllname + '.*.*' + ' File locked could not be deleted'  
    Print @ScriptTempPath  
    Print '*** Undeploy Objects using this DLL and close the IDE ***'  
    Return  
END
```

```
-- Just use this for a simple success message  
if @hr = 0  
Begin  
    set @Success = 1  
End
```

```
EXEC @hr = sp_OADestroy @ole_FileSystem
```

```
delete from file_primitive_definition_link where file_ID in (select file_id from file_table where file_name like  
@dllname + '%')  
delete from feature_file_link where file_ID in (select file_id from file_table where file_name like @dllname + '%')  
delete from file_pending_update where file_ID in (select file_id from file_table where file_name like @dllname  
+ '%')  
delete from feature where feature_name like @dllname + '%'  
delete from file_table where file_name like @dllname + '%'
```

```
If @Success = 1  
Begin  
    print "*** The Script Funtion has been removed ***"  
End  
GO  
SET QUOTED_IDENTIFIER OFF  
GO  
SET ANSI_NULLS ON  
GO
```

```
GRANT EXECUTE ON [dbo].[RemoveMyDII] TO [public]  
GO
```