spautogensolution

SharePoint 2010 Automatically Generated Solution Demo

download archive

In this demo project I show how to generate Sandboxed Solutions by code at runtime.

home issues discussions

(Blog article: http://ikarstein.wordpress.com/2012/02/29/demo-generating-sandboxed-solutions-through-code-for-sharepoint-2010)

In this demo project I show you how to generate Sandboxed Solutions automatically at runtime.

Let’s have a look:

1. First we look into the solution gallery of “http://sharepoint.local”. It’s empty.

   image_thumb

2. I’ve created my demo solution and deployed it. This solution contains a list template and list instance. The list instance is shown in the Quick Launch as “Dynamic Menu”.

   image3_thumb

3. Now I create a new entry in this list:

   image9_thumb

   image15_thumb

4. Now we look into the solution gallery:

   image19_thumb

   There is a new Sandboxed Solution in it!!

5. In the Shared Documents libary I have a single file “My PDF”. Now I open the files context menu:

   image23_thumb

   image27_thumb

6. After creating a new item in the “Dynamic Menu” I get a new entry in the Edit Control Menu:

   image40_thumb1

   image34_thumb1

   This opens a new Outlook Window with the item URL as body.

   image31_thumb

7. For every generated solution there is an entry in the Site Collection Features list:

   image47_thumb

image50_thumb

How does it work?

1. First I’ve created a list template and list instance.

2. Then I’ve created a Event Receiver.

image_thumb28

There are two methods:

- Utilities.RemoveSolution
- Utilities.AddSolution

3. Method “Utilities.AddSolution”

This static method creates the Sandboxed Solution.

Therefore internal non-public methods of SharePoint are invoked. This is the most interesting aspect of the demo project!

```
ConstructorInfo ctor = typeof(Microsoft.SharePoint.SPSolutionExporter).Assembly.GetType("Microsoft.Share
object cabInf = ctor.Invoke(new object[] { solutionFileName });

MethodInfo[] mi = cabInf.GetType().GetMethods(BindingFlags.NonPublic | BindingFlags.Instance | BindingFl
MethodInfo mi2 = null;
foreach( var m in mi )
    if( m.Name == "CompressDirectory" && m.GetParameters().Length == 4 )
    {
       mi2 = m;
       break;
    };

mi2.Invoke(cabInf, new object[] { solutionDir.FullName, true, -1, null });
```

In the source code before this snippet I create the structure of a WSP file in a temporary folder in the file system. The shown code creates the WSP file that I upload into the solution gallery.

4. Utilities.RemoveSolution

This static method removes the solution from the solution gallery.