

Programando WebParts – Parte 1

Los principios

Autor : Gustavo Velez
Para : www.gavd.net/servers/
Fecha : 02-02-2005
Versión : 1.0.0

Tal vez la actividad mas común cuando hablamos de programar para SharePoint, es hacer WebParts (“Elementos Web”, según la traducción de Microsoft). WebParts nos permiten añadir funcionalidad y personalizar una instalación de SharePoint/WSS de una manera fácil y rápida.

Primera parte de una serie de artículos que describirán algunas de las diferentes posibilidades en cuanto a programación de WebParts: Los principios básicos, como hacer el primer proyecto, algo de código y como implementar la WebPart.

Requisitos

- WSS o SharePoint instalados y funcionando
- Conocimientos de cómo usar WebParts in SharePoint
- Visual Studio DotNet 2003
- Plantillas para la creación de WebParts en Visual Studio. Se pueden bajar del sitio de Microsoft
(<http://www.microsoft.com/downloads/details.aspx?FamilyID=14D5D92F-C3A6-407C-AAD7-B8C41A4991BE&displaylang=en>)
- Conocimientos de programación y alguna experiencia sobre como usar Visual Studio. Todo el código en esta serie de artículos es escrito en C#, pero es fácilmente adaptable a VB.
- Conocimientos básicos del Modelo de Objetos (API) de SPS

Creando la primera WebPart

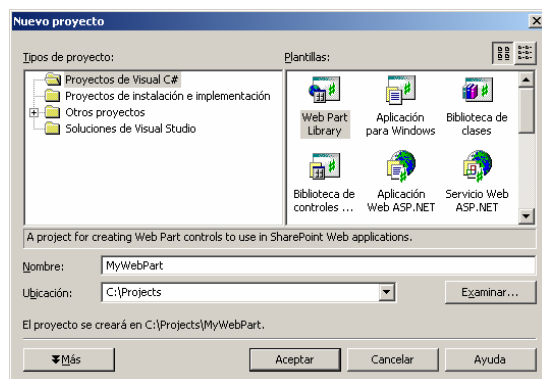
Aunque existen otras herramientas para programar WebParts, la más indicada es Visual Studio DotNet. Además, si las plantillas mencionadas en los requisitos están instaladas, el trabajo se facilita mucho más.

WebParts heredan de la clase “*Microsoft.SharePoint.WebPartPages.WebPart*”, lo que de inmediato nos proporciona todas las posibilidades básicas de un WebPart, como la interfase visual y el editor de propiedades.

Aquí se asume que el código se esta desarrollando en una maquina que tiene SPS instalado. Si no es así, haga una copia del archivo “*Microsoft.SharePoint.dll*” desde un computador que tenga SPS, y añada una referencia al ensamblado en VisualStudio.

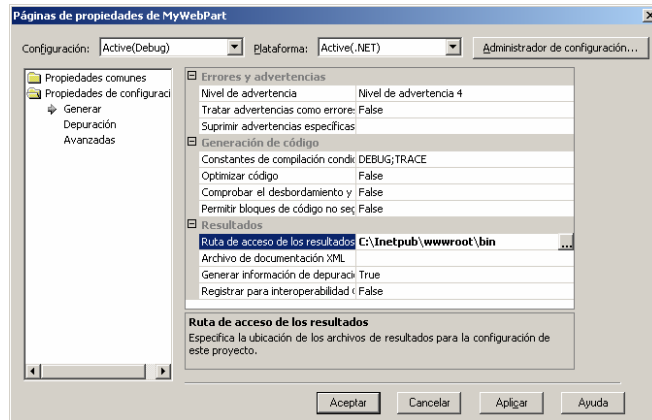
Creando e inicializando el nuevo Proyecto

1 – Inicie Visual Studio, escoja “Nuevo Proyecto”, seleccione el lenguaje de programación y la plantilla “Web Part Library” (tiene que tener las plantillas mencionadas anteriormente instaladas) y escoja un nombre



2 – Añada una referencia a “Microsoft.SharePoint” si no tiene SPS instalado en el computador. Si este no es el caso, todas las referencias necesarias ya están inicializadas.

3 – Cambie la ruta de acceso para el ensamblado a “C:\inetpub\wwwroot\bin”



Esto no es indispensable, pero si es practico de hacer. SPS espera que el ensamblado del WebPart este en el directorio “bin” de IIS, o en el GAC. Este ultimo sitio esta reservado para ensamblados que serán utilizados por diferentes aplicaciones, lo que no es el caso con el ensamblado de un WebPart. Si no sigue este paso, cada vez que haga una nueva compilación del código, tendrá que copiar manualmente el ensamblado al directorio “bin” de IIS.

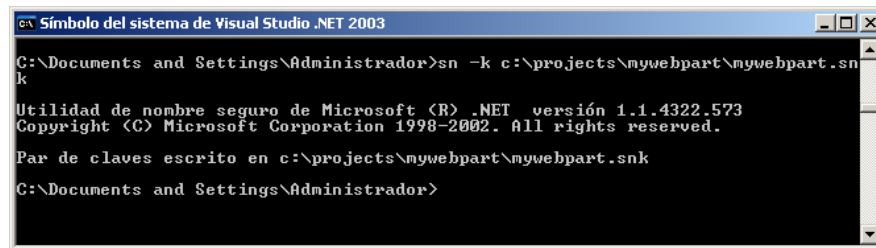
4 – Aunque tampoco es indispensable, si es recomendado cambiar la versión del ensamblado a un número fijo (SharePoint puede tener problemas de autenticación si la versión cambia después de cada compilación). Abra la pagina “AssemblyInfo.cs” en Visual Studio y cambie el renglón “[assembly: AssemblyVersion(“1.0.*”)]” en “[assembly: AssemblyVersion(“1.0.0.0”)]”. Si utiliza las plantillas de SPS, este paso no es necesario.

Un nombre seguro para la WebPart

Para que SharePoint acepte la nueva WebPart, es necesario darle un “Nombre Seguro”.

1 – Vaya a “Inicio” -> “Todos los programas” -> “Microsoft Visual Studio .NET 2003” -> “Herramientas de Visual Studio .NET” -> e inicie el “Símbolo de sistema de Visual Studio .NET 2003”

2 – Escriba el comando “sn -k c:\[directorio del proyecto]\[nombre del archivo.snk]” (en nuestro caso “sn -k c:\projects\mywebpart\mywebpart.snk”)



3 – En Visual Studio, regrese a la pagina “AssemblyInfo.cs”, vaya al renglón “[assembly: AssemblyKeyFile(“”)]” y modifiquelo en “[assembly: AssemblyKeyFile(“..\..\mywebpart.snk”)]”

Codificando la WebPart

Ahora esta el proyecto listo para empezar a introducir nuestro código. Las referencias "Using" necesarias ya las ha puesto la plantilla para nosotros, así que podemos empezar directamente.

1 – Para el código que se va a usar, es necesaria otra sección de "Using"; añada la siguiente declaración:

```
using System.Web.UI.HtmlControls;
```

2 – Precisamente debajo de la declaración de la clase ("*public class WebPart1 : ...*"), añada la siguiente declaración de variables

```
HtmlButton objButton;  
HtmlInputText objTextBox;  
HtmlInputText objLabelZoneID;
```

3 – Al final de la pagina "WebPart1.cs", precisamente bajo el método "RenderWebPart", cree un nuevo método "CreateChildControls()", con el siguiente código:

```
protected override void CreateChildControls()  
{  
    objButton = new HtmlButton();  
    objButton.InnerText = "Escriba el titulo del WebPart";  
    objButton.ServerClick += new EventHandler(objButton_click);  
    Controls.Add(objButton);  
    objTextBox = new HtmlInputText();  
    objTextBox.Value="";  
    Controls.Add(objTextBox);  
    objLabelZoneID = new HtmlInputText();  
    objLabelZoneID.Value = this.ZoneID;  
    Controls.Add(objLabelZoneID);  
}
```

Con este código se crean dos casillas de entrada de datos y un botón. Cuando se usa el botón, el evento "objButton_click" es activado.

4 – Bajo el anterior código, es necesario definir el evento para el botón:

```
public void objButton_click(object sender, EventArgs e)  
{  
    this.Title = objTextBox.Value;  
}
```

Este código copia el texto de la casilla en el titulo de la WebPart.

5 – Finalmente, en el método "RenderWebPart" cambie el renglón "*output.Write(SPEncode.HtmlEncode(Text));*" que la plantilla ha escrito por el siguiente código:

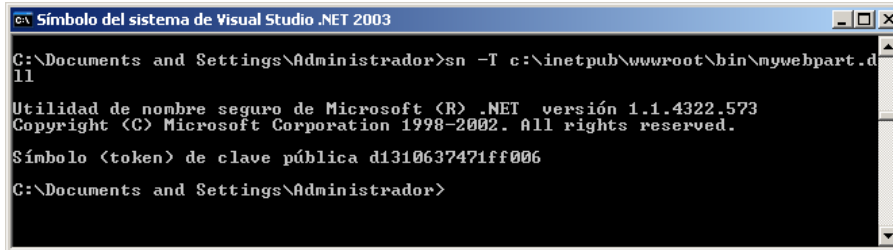
```
objTextBox.RenderControl(output);  
objButton.RenderControl(output);  
output.WriteLine("<BR>");  
objLabelZoneID.RenderControl(output);
```

6 – Compile el proyecto.

Implementación

Si la compilación no produce problemas, ahora habrá un archivo "MyWebPart.dll" en el directorio "C:\inetpub\wwwroot\bin". Siga los siguientes pasos para registrar la WebPart:

- 1 – En la pantalla "Símbolo de sistema de Visual Studio .NET 2003", escriba el comando "sn -T c:\[directorio de IIS]\bin\[nombre del ensamblado.dll]" (en nuestro caso: "sn -T c:\inetpub\wwwroot\bin\mywebpart.dll")



```
C:\Documents and Settings\Administrador>sn -T c:\inetpub\wwwroot\bin\mywebpart.dll
Utilidad de nombre seguro de Microsoft (R) .NET versión 1.1.4322.573
Copyright (C) Microsoft Corporation 1998-2002. All rights reserved.
Símbolo (token) de clave pública d1310637471ff006
C:\Documents and Settings\Administrador>
```

- 2 – Abra el archivo "C:\inetpub\wwwroot\web.config" y añada la siguiente línea en el bloque "<SafeControls>":

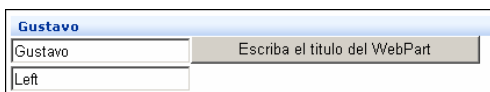
```
<SafeControl
Assembly="MyWebPart, Version=1.0.0.0, Culture=neutral,
PublicKeyToken=d1310637471ff006"
Namespace="MyWebPart"
TypeName="*" />
```

Asegurándose de copiar la clave pública conseguida en el punto 1 en el sitio "PublicKeyToken"

- 3 – Vaya a una página de SharePoint, use "Editar pág. Compartida" -> "Agregar elementos Web" -> "Importar"

- 4 – Con el botón "Examinar..." vaya al directorio del proyecto; seleccione el archivo "WebPart1.dwp" -> "Cargar"

- 5 – Posicione la WebPart en una zona de la página. Escriba algún texto en la primera casilla y utilice el botón para cambiar el nombre de la WebPart. En la segunda casilla aparecerá el nombre de la zona en la que la WebPart ha sido posicionada (el nombre de la zona es el nombre que la plantilla de la página define: puede ser que aparezca en inglés).



Gustavo	
Gustavo	Escriba el título del WebPart
Left	