

Programmation web avec ASP.Net

C# .net

 Page web écrite uniquement avec des contrôles HTML simples

 Page web écrite uniquement avec des contrôles serveur

 Communication entre pages web

 Page web écrite avec BD et GridView

 cookie dans une page web

 création et consommation d'un service web

Page web écrite avec des contrôles HTML



Plan général:

Page web écrite uniquement avec des contrôles HTML simples

- 1°) Page HTML classique
- 2°) Ajouter du code élémentaire C# <% ... %> dans la page HTML
- 3°) Réponse à une requête post back de la part du client
- 4°) Utiliser Visual Studio pour générer du HTML
- 5°) Utiliser Visual Studio pour générer des appels au javascript
- 6°) Utilisez du javascript pour onmouseover et onmousemove

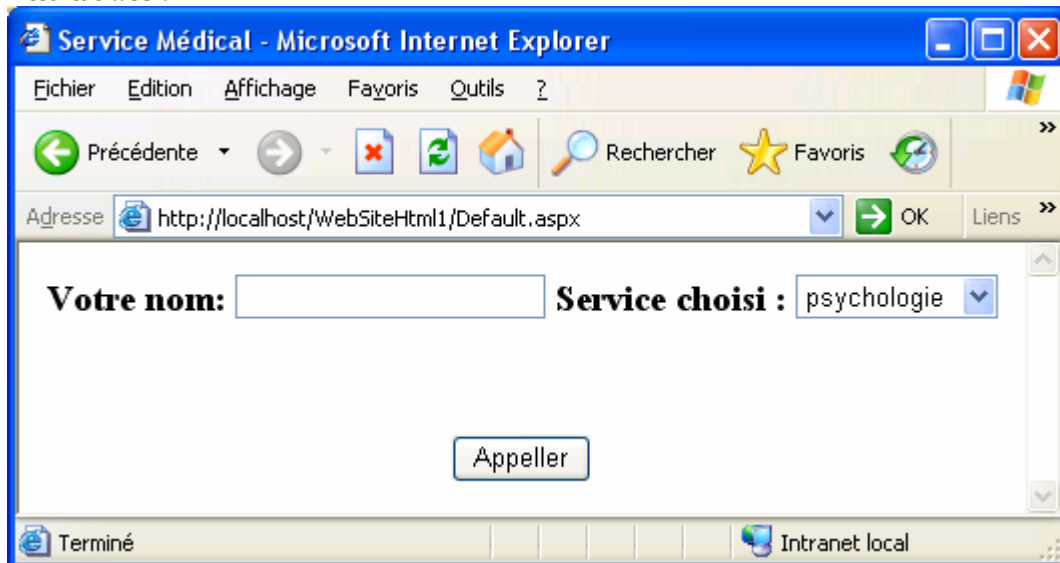
1°) Page HTML classique :

```
<html>
  <head>
    <Title>Service Médical</Title>
  </head>
  <body>
    <center>
      <form action="Default.aspx">
        <h3> Votre nom: <input id="Nom" type="text">

        Service choisi : <select id="Categorie" size=1>
          <option>psychologie</option>
          <option>pédiatrie</option>
          <option>dermatologie</option>
        </select>

      </h3>
      <br/>
      <br/>
      <input type="submit" value="Appeler">
    </form>
    </center>
  </body>
</html>
```

Résultat web :



2°) Ajouter du code élémentaire C# <% ... %> dans la page HTML

```
<%@ Page Language="C#" %>
<html>
  <head>
    <Title>Service Médical</Title>
  </head>
  <body>
    <center>
```

```

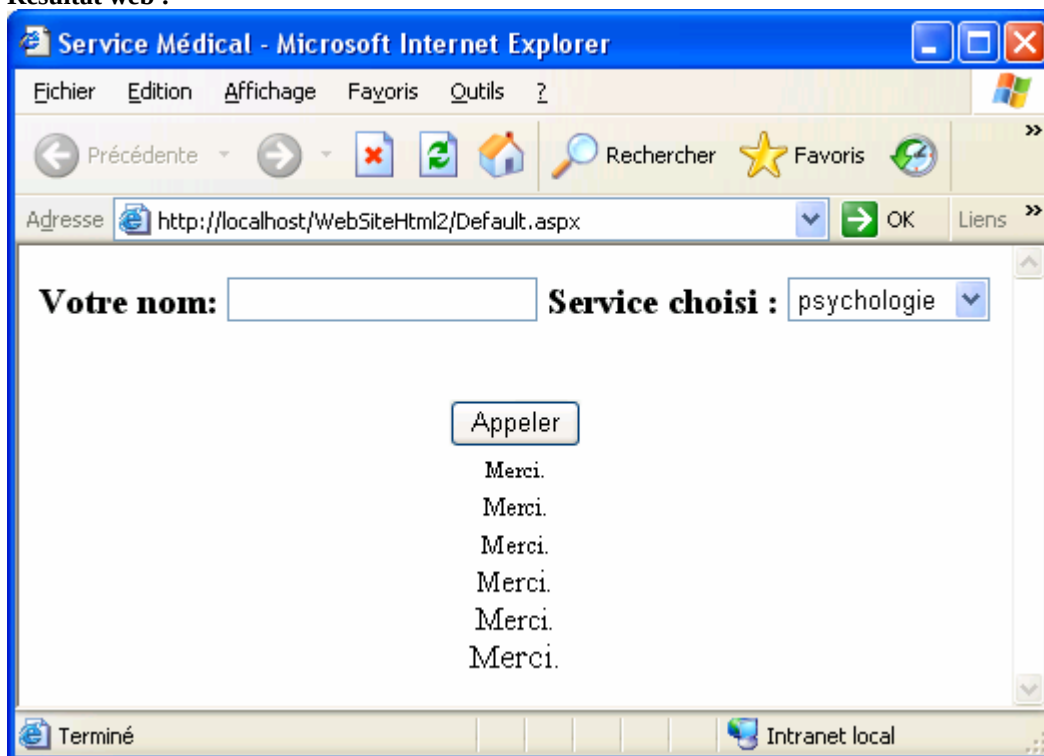
<form action="Default.aspx">
  <h3> Votre nom: <input id="Nom" type="text">

  Service choisi : <select id="Categorie"
size=1>
                    <option>psychologie</option>
                    <option>pédiatrie</option>
                    <option>dermatologie</option>
                </select>

  </h3>
  <br/>
  <input type="submit" value="Appeler">
  <br/>
  <% for (int i=8;i<14; i++) { %>
    <span style="font-size: <%=i%>pt">
Merci.</span>
    <br/>
    <% } %>
  </form>
</center>
</body>
</html>

```

Résultat web :



Important : le code inclus entre les blocs <% ... %> ne doit contenir **ni classe, ni méthode**, il est compilé effectivement sur la machine client (contrairement aux langages de scripts qui sont interprétés).

3°) Réponse à une requête post back de la part du client

```

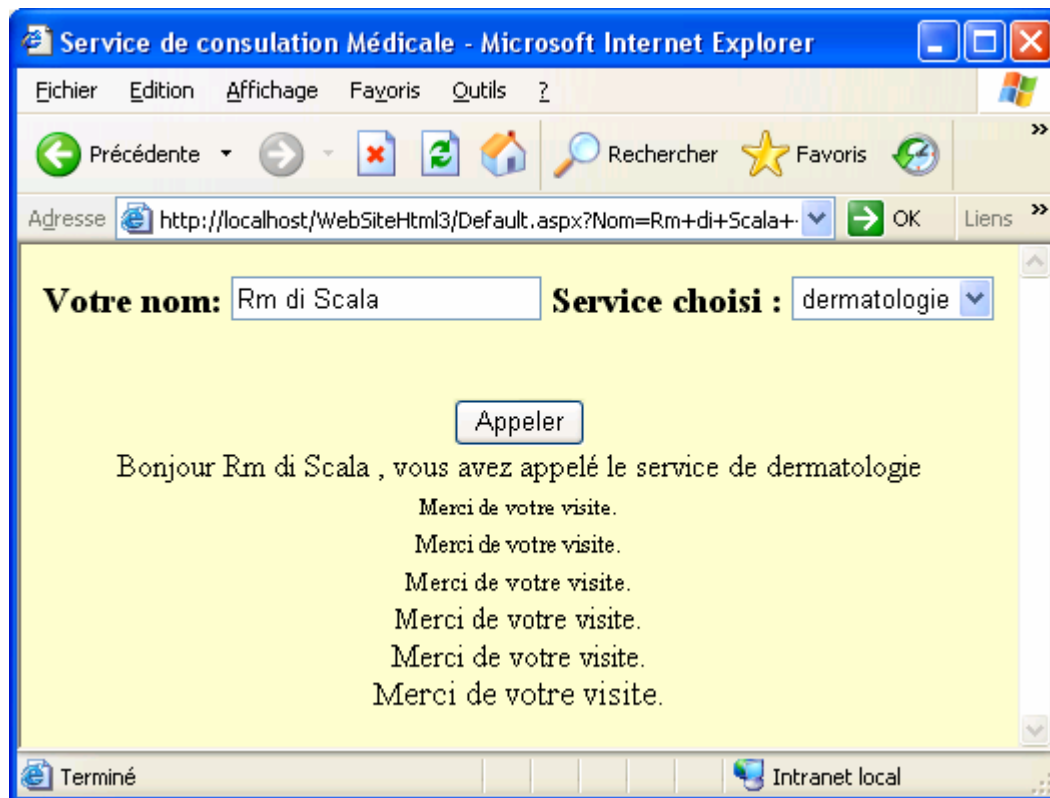
<%@ Page Language="C#" %>
<html>
  <head>
    <Title>Service de consultation Médicale</Title>
  </head>

```

```

<body style="background-color :#FFFFCC">
  <center>
    <form action="Default.aspx">
      <h3> Votre nom: <input name="Nom" type="text
        value="<%=HttpUtility.HtmlEncode(Request.QueryString["Nom"])%>
        ">
      Service choisi : <select name="service" size=1>
        <%=string[] strChoix={"psychologie","pédiatrie","dermatologie"};
        for (int i = 0; i < strChoix.Length; i++)
        {
          <%=
            <option
              <%=if (Request.QueryString["service"]==strChoix[i])
                Response.Write("selected");
              %>
              <%=strChoix[i] %>
            </option>
          <%= } %>
        </select>
      </h3>
      <br/>
      <input type="submit" name="choix" value="Appeler">
      <%= if (Request.QueryString["choix"]!=null) { %>
        <br/>
        Bonjour
        <%=HttpUtility.HtmlEncode(Request.QueryString["Nom"])%>,
        vous avez appelé le service de
        <%=HttpUtility.HtmlEncode(Request.QueryString["service"])%>
        <br/>
        <%=for (int i=8;i<14; i++) { %>
          <span style="font-size: <%=i%>pt"> Merci de votre
visite.</span>
          <br/>
          <%=} %>
        <%= } %>
      </form>
    </center>
  </body>
</html>

```



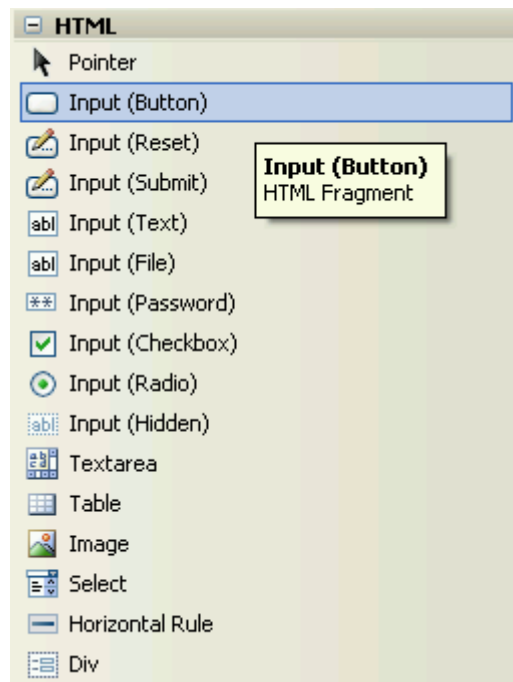
4°) Utiliser Visual Studio pour générer du HTML

L'environnement de programmation Visual Studio permet de créer visuellement votre interface HTML avec des entités visuelles générant immédiatement dans votre page source le code HTML correspondant.

La palette d'outils de Visual Studio propose 15 entités permettant de générer 15 types de balises de code HTML différentes. Ces contrôles HTML se trouvent dans le namespace System.Web.UI.HtmlControls, ils sont présents dans la palette des contrôles "toolbox" dans l'onglet HTML :

Premier exemple d'utilisation de HtmlControls : afficher un bouton "Input" dans une page web.

Onglet HTML de la Toolbox :



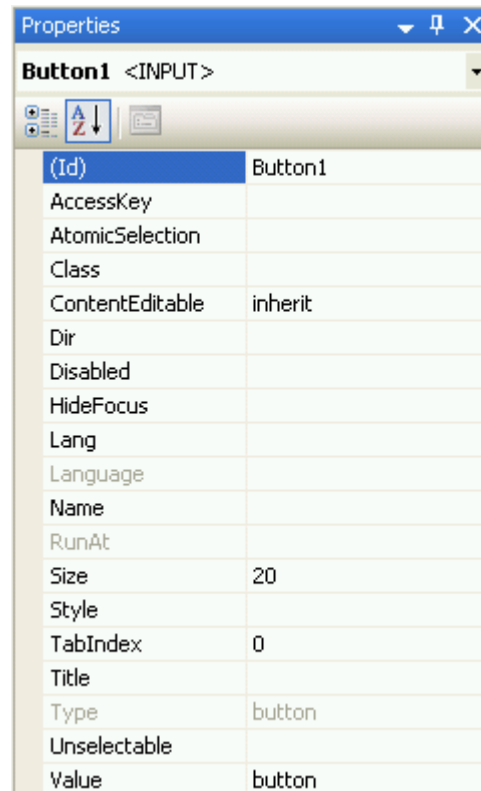
Le HtmlControls choisi dans la Toolbox est le *Input Button*.

Entité visuelle :

Code HTML engendré :

```
<input id="Button1" type="button" value="button" />
```

Propriétés paramétrables :

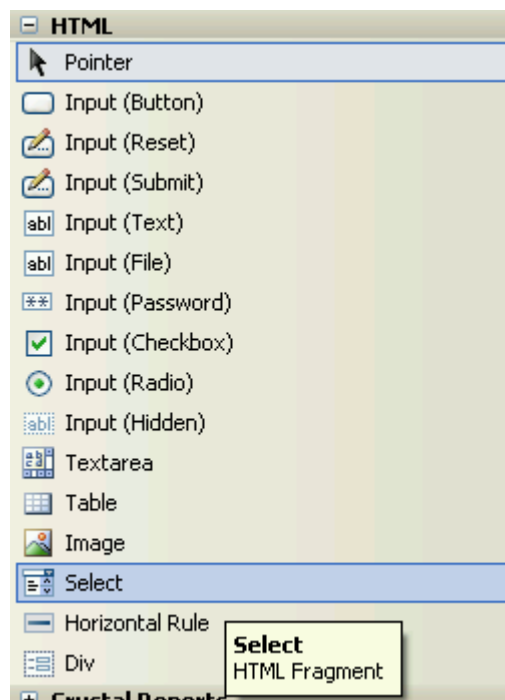


Le code source HTML obtenu est insérable dans une page :

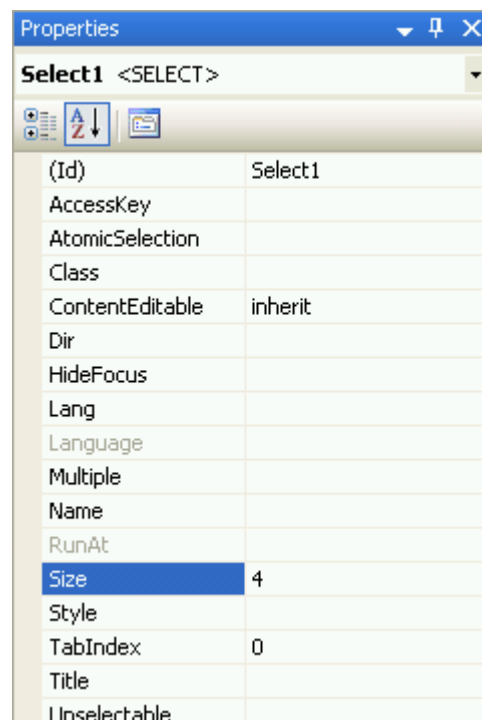
```
<html>
<head>
  <title>Untitled Page</title>
</head>
<body>
  <form id="form1">
    <div>
      <input id="Button1" type="button" value="button" />
    </div>
  </form>
</body>
</html>
```

Second exemple d'utilisation de HtmlControls : afficher une [liste de sélection](#) dans une page web.

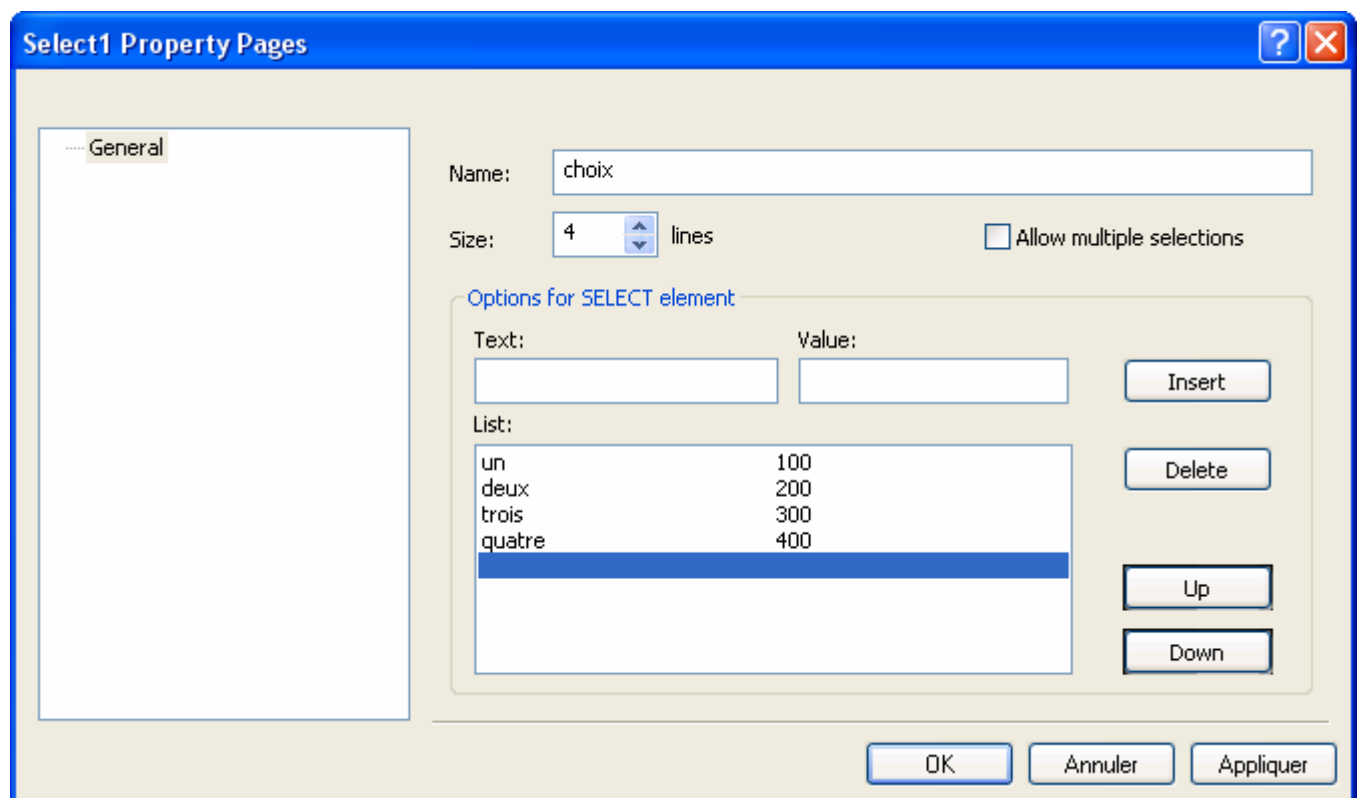
On choisi le HtmlControls **Select** dans la Toolbox et on le dépose sur la page :



On change sa propriété size à 4 (ce qui permet de définir une liste de sélection possédant 4 éléments) :



En cliquant avec le bouton droit de souris sur l'image déposée et en choisissant le champ propriétés dans le menu popup, on obtient une fenêtre permettant de modifier la liste :





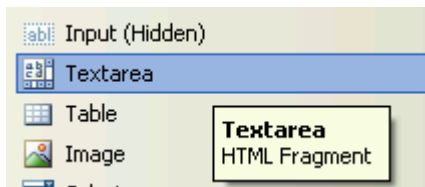
Entité visuelle :

Le code source HTML obtenu est insérable dans une page :

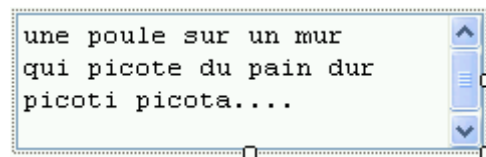
```
<html>
<head>
  <title>Untitled Page</title>
</head>
<body>
  <form id="form1">
    <div>
      <select id="Select1" name="choix" style="height: 84px" size="4">
        <option selected="selected" value="100">un</option>
        <option value="200">deux</option>
        <option value="300">trois</option>
        <option value="400">quatre</option>
      </select>
    </div>
  </form>
</body>
</html>
```

Dernier exemple d'utilisation de HtmlControls : utiliser un [TextArea](#) dans une page web.

Onglet HTML de la Toolbox :



Entité visuelle :



Le code source HTML obtenu est insérable dans une page :

```
<html>
<head>
  <title>Untitled Page</title>
</head>
<body>
  <form id="form1">
    <textarea id="TextArea1" rows="4" style="width: 227px">
une poule sur un mur
qui picote du pain dur
picoti picota....
    </textarea>
  </form>
</body>
</html>
```

5°) Utiliser Visual Studio pour générer des appels au javascript

L'environnement de programmation Visual Studio permet aussi de créer visuellement les réactions événementielles d'une entité de votre page HTML avec des fonctions javascript.

On construit une page HTML :

Contenant par exemple un <INPUT> du type "bouton" :

```
<input id="Button1" type="button" value="button" />
```



Contenant aussi un <INPUT> du type "text" :

```
<input id="Text1" type="text" />
```



On peut rendre la page HTML sensible au click de souris chez le client :



En double cliquant sur l'image de l'INPUT bouton, Visual Studio génère deux éléments :

- 1- un squelette de fonction javascript
- 2- le tag associé à l'événement onclick dans la balise INPUT

```
<html>
<head runat="server">
  <title>Untitled Page</title>
  <script language="javascript"
    type="text/javascript">
    <!--
    function Button1_onclick() {
    }
    // -->
  </script>
</head>
<body>
  <form id="form1" runat="server">
    <div>
      <input id="Button1" type="button" value="button"
        language="javascript" onclick="return
        Button1_onclick()" />
      <br />
      <br />
      <input id="Text1" type="text" />
    </div>
  </form>
</body>
</html>
```

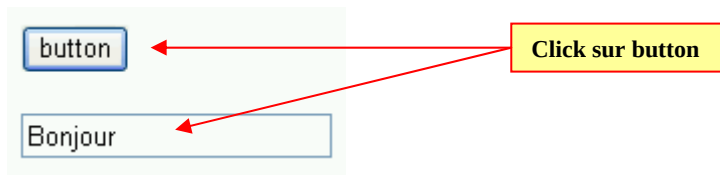
Squelette javascript

tag onclick

Action événementielle proposée : en double cliquant sur l'entité **Button1**", la fonction Button1_Click() renvoie le texte "Bonjour" dans l'entité **Text1**".

```
function Button1_onclick() {
  document.form1.Text1.value = "Bonjour";
}
```

code javascript



6°) Utilisez du javascript pour onmouseover et onmouseleave

Afin de ne pas surcharger inutilement le serveur, on programme les événements de souris (over et move en particulier) côté client avec du javascript et avec des scripts C# côté serveur :

Le code source HTML obtenu est insérable dans une page :

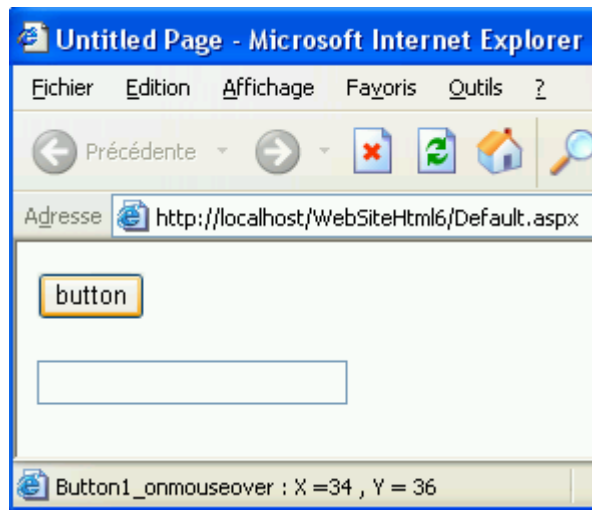
```
<html>
<head>
  <title>Untitled Page</title>

  <script language="javascript">
    <!--
      function Button1_onmouseover() {
        window.status = "Button1_onmouseover : X =" +
          window.event.x + " , Y = " + window.event.y;
      }
      function Button1_onmouseleave() {
        window.status = "Clear";
      }
      function Text1_onmousemove() {
        window.status = "Text1_onmousemove : X =" +
          window.event.x + " , Y = " + window.event.y;
      }
    // -->
  </script>

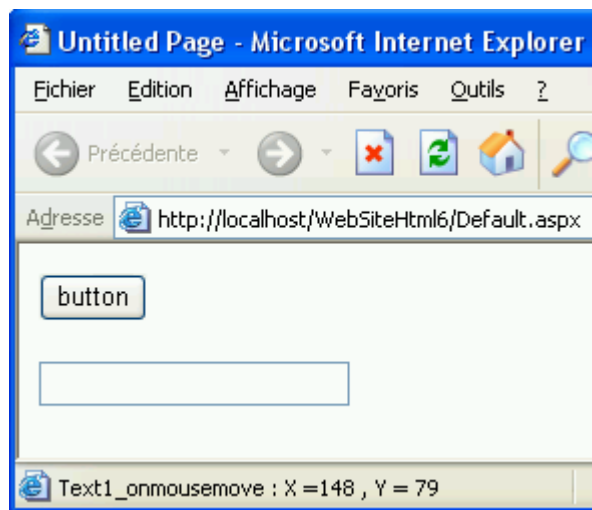
</head>
<body>
  <form id="form1" >
    <div>
      <input id="Button1" type="button" value="button"
        onmouseleave="Button1_onmouseleave()"
        onmouseover="Button1_onmouseover()" />
      <br />
      <br />
      <input id="Text1" onmousemove="Text1_onmousemove()" type="text"
    />
    </div>
  </form>
</body>
</html>
```

Les actions visualisées côtés client

`onmouseover="Button1_onmouseover()"` :



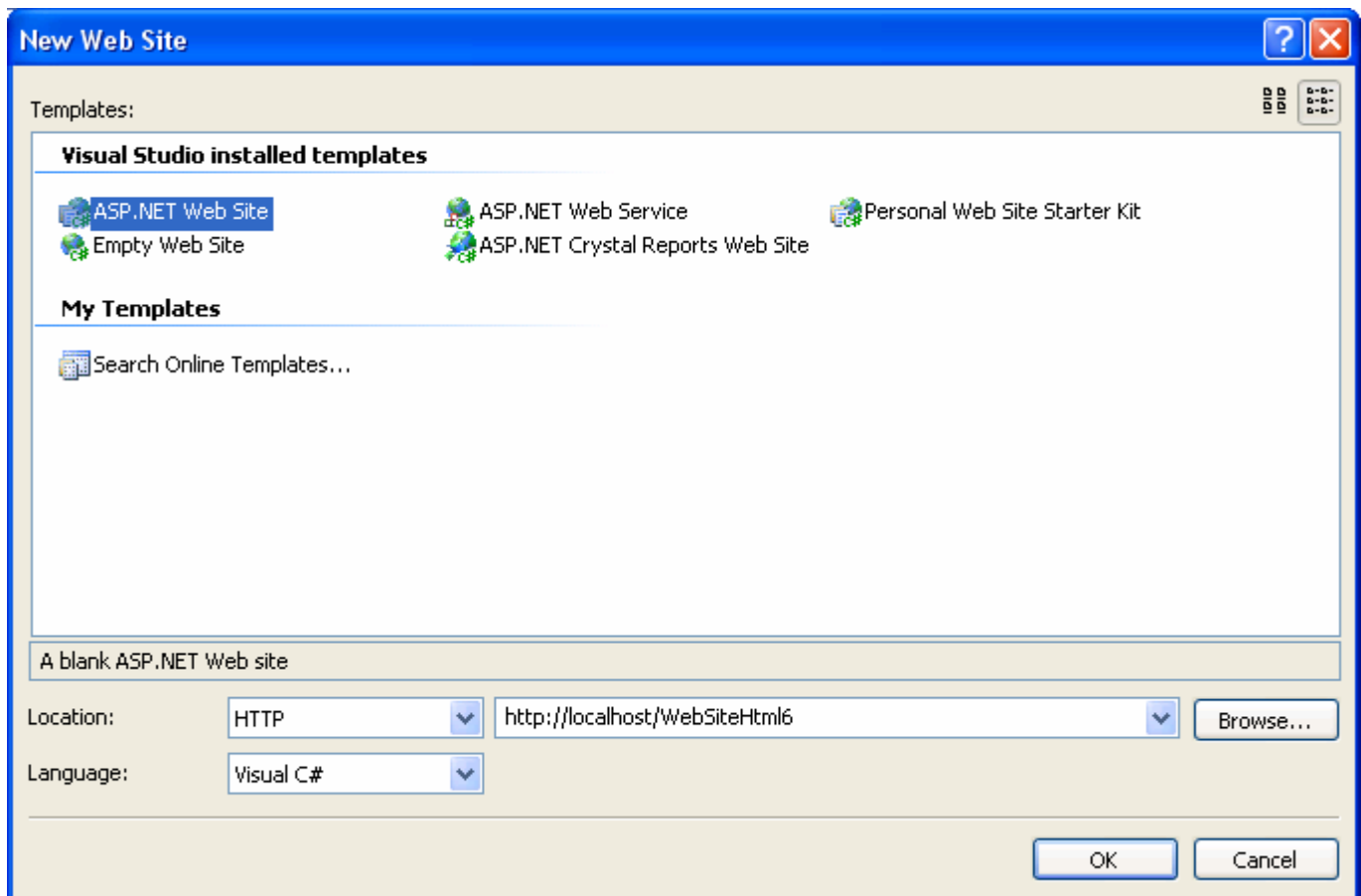
`onmousemove="Text1_onmousemove()"` :



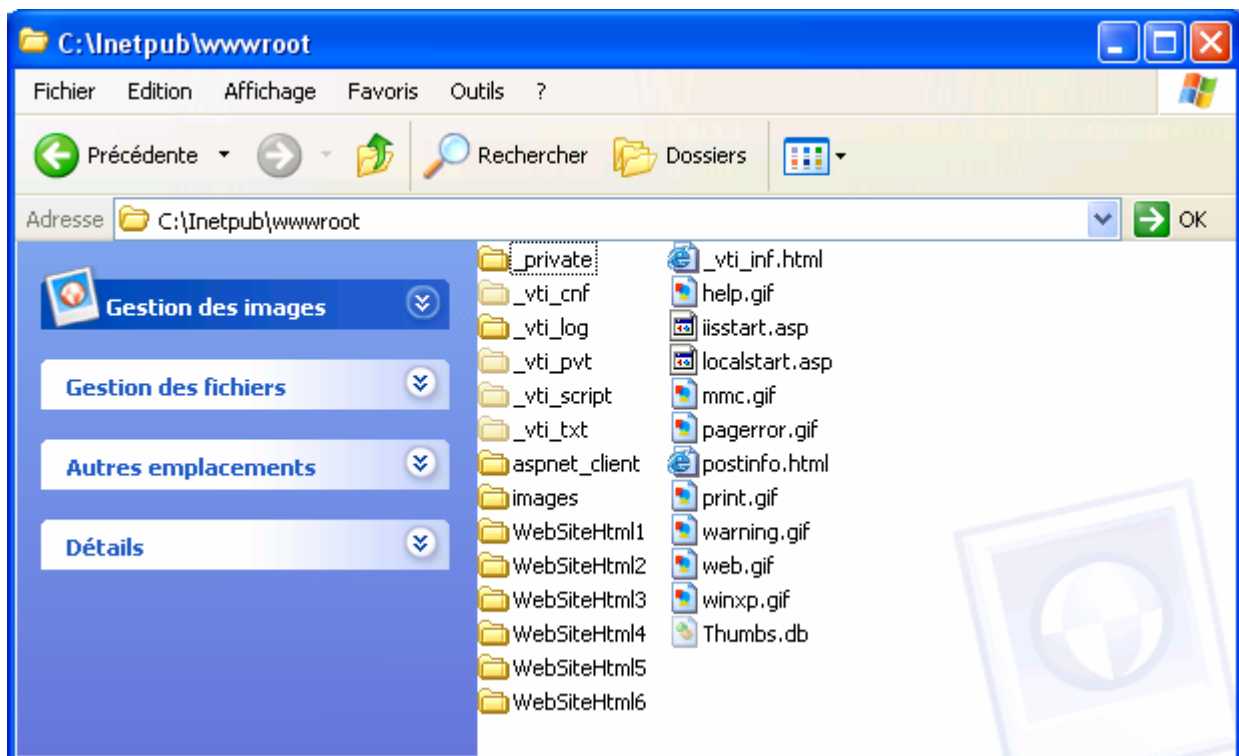
Conclusion :

Une page ASP .net 2.0 fonctionne et réagit comme une page classique HTML.

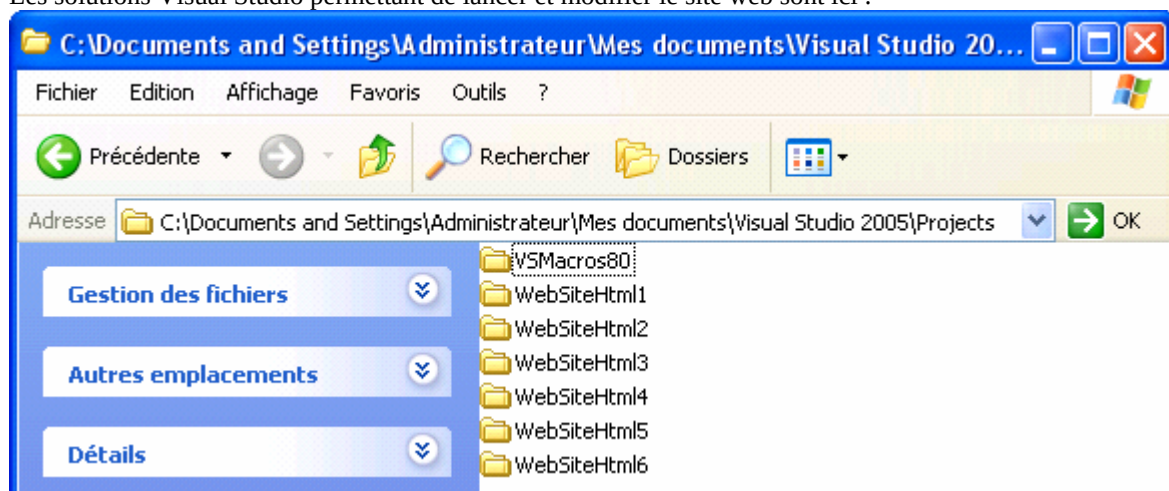
Où sont localisés les fichiers des sites web construits



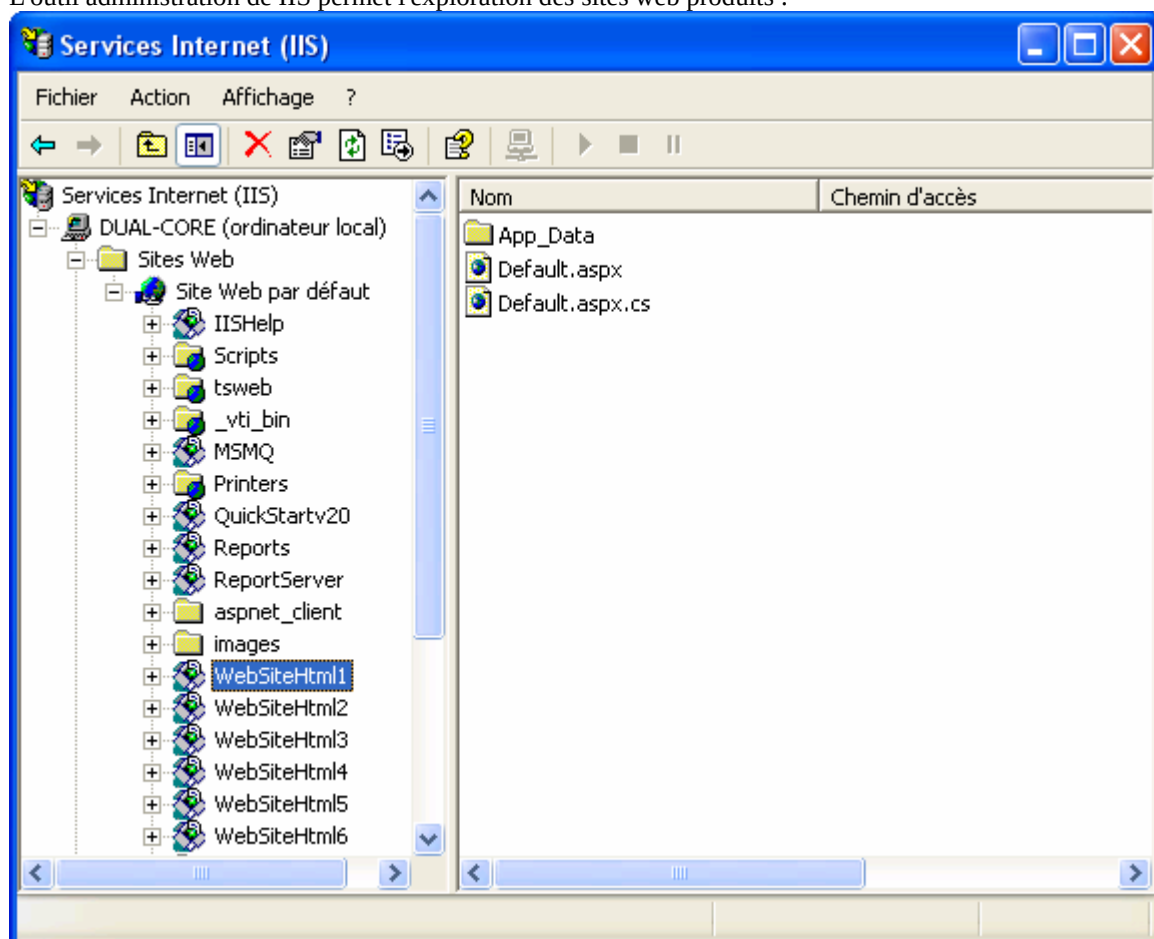
On a choisi la localisation http (sur le localhost, rangement dans *C:\Inetpub\wwwroot*)



Les solutions Visual Studio permettant de lancer et modifier le site web sont ici :



L'outil administration de IIS permet l'exploration des sites web produits :



Page web écrite avec des contrôles serveurs

C# .net

Plan général:

Page web écrite uniquement avec des contrôles serveurs

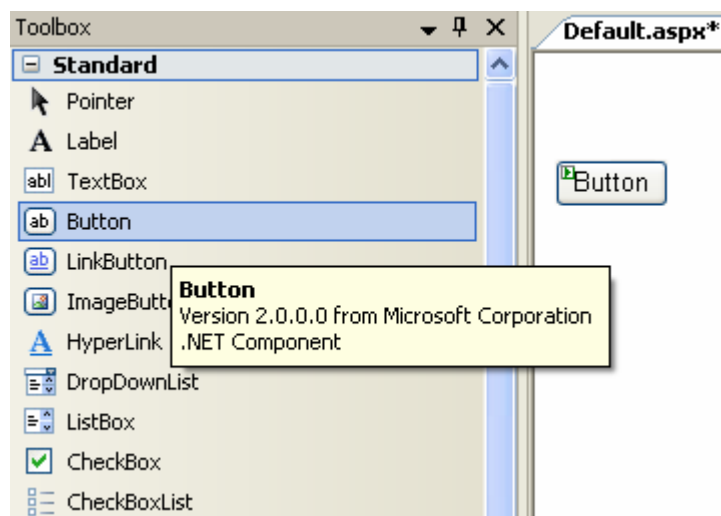
- 1°) Construire une page Asp.net côté serveur avec Visual Studio
- 2°) Une page Asp.net côté serveur peut engendrer du javascript
- 3°) Les événements gérés côté serveur sont dans une page séparée (code-behind)
- 4°) Les événements gérés côté serveur sont dans un script inclus (code-inline)
- 5°) Les événements côté serveur permettent des communications entre les contrôles

Les contrôles serveurs : ce sont des objets qui se programment et s'exécutent côté serveur. Ils servent à représenter des éléments d'IHM dans une page web (éditeurs mono et multi lignes, images, boutons, ...). Pour déclarer et utiliser de tels contrôles, il suffit d'ajouter dans le code source d'une page Asp.Net (une page Asp.Net = un fichier **XXX.aspx**) les balises HTML-like spécifiques à ces objets de contrôles serveurs.

- Lors de l'exécution sur le serveur, ces contrôles serveurs génèrent automatiquement un contenu HTML.
- Le code source JHTML décrivant un contrôle serveur contient, dans la balise du contrôle serveur, l'attribut **runat** positionné sur la valeur **server** (attribut **runat = server**). Cet attribut **runat** spécifie que le code inclus dans le bloc de script s'exécute sur le serveur et non chez le client (par défaut l'exécution a lieu chez le client).
- Ces contrôles serveurs sont des objets instanciés à partir de classes situés dans le **namespace** suivant : **System.Web.UI.WebControls**.
- Il est aussi possible de transformer en contrôles serveurs les contrôles HTML côté client (contrôles du namespace **System.Web.UI.HtmlControls**) d'une page Asp.Net déjà écrite.
- Un contrôle serveur est généralement décrit dans une page Asp.net grâce à une balise spécifique **<asp: ...>**.

1°) Construire une page Asp.net côté serveur avec Visual Studio :

Visual Studio permet de déposer visuellement des contrôles serveurs afin de construire son IHM, l'IDE engendre alors le code Asp.Net associé. Par exemple un bouton peut être créé à partir de l'onglet standard de la Toolbox de Visual Studio dans une page nommée **Default.aspx** :

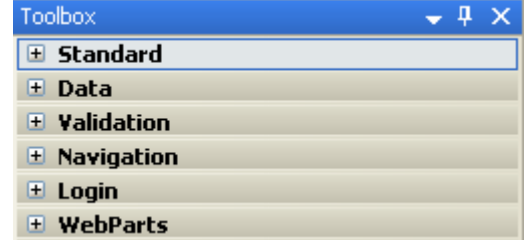


Code partiel engendré par Visual Studio pour ce bouton :

```
<form id="form1" runat="server">
    <asp:Button ID="Button1" runat="server" Text="Button" />
</form>
```

La Toolbox contient plusieurs autres onglets facilitant la création visuelle de pages Asp.Net avec de nombreux outils :

	Seul le code HTML est envoyé au navigateur du client, c'est donc Visual Studio qui transcrit les balises et le code Asp.Net
--	---

	en du code classique reconnaissable par n'importe quel navigateur Internet.
---	--

Code Asp.Net du bouton précédent :

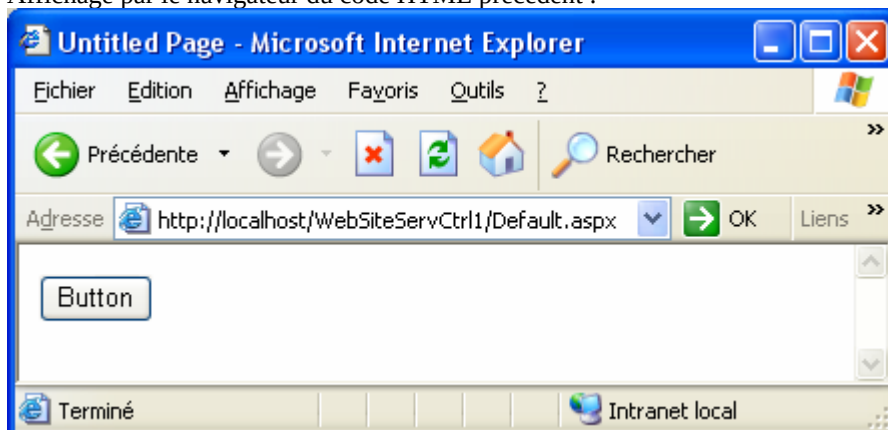
```
<form id="form1" runat="server">
  <asp:Button ID="Button1" runat="server" Text="Button" />
</form>
```



Code HTML engendré pour le bouton précédent :

```
<body>
  <form name="form1" method="post" action="Default.aspx"
id="form1">
  <div>
  <input type="hidden" name="__VIEWSTATE" id="__VIEWSTATE"
value="/wEPDwUKMTQ2OTkzNDMyMWRkNOC5wTqk2noztuYBp2ZpDkCmmE=" />
  </div>
  <div>
    <input type="submit" name="Button1" value="Button"
id="Button1" />
  </div>
  <div>
  <input type="hidden" name="__EVENTVALIDATION" id="__EVENTVALIDATION"
value="/wEWAglp8MT+DQKM54rGBvbawE34cFV6yTqH8kCIH0AvnsVn" />
  </div></form>
</body>
```

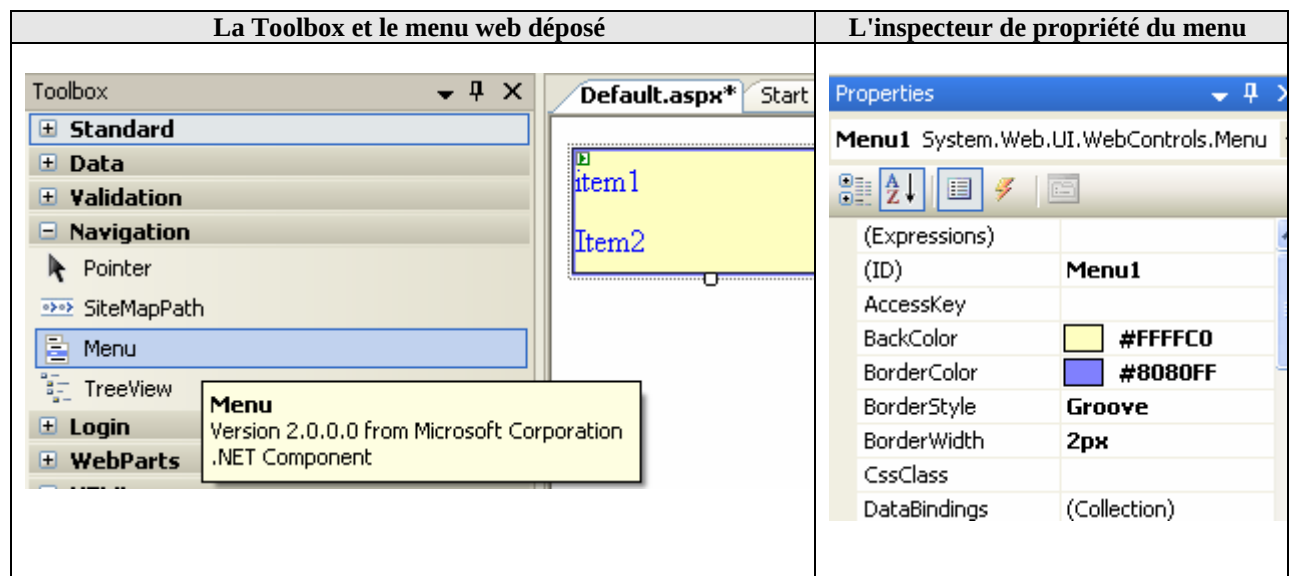
Affichage par le navigateur du code HTML précédent :



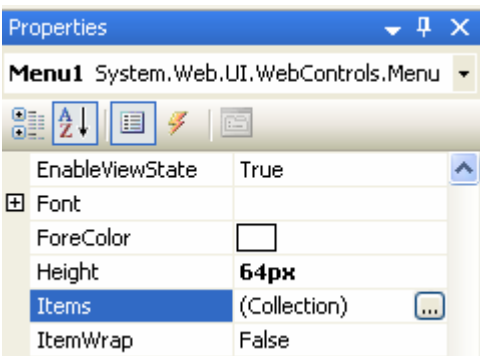
2°) Une page Asp.net côté serveur peut engendrer du javascript :

Visual Studio engendre lui-même du code javascript à partir de certains objets de classes

System.Web.UI.WebControls comme par exemple la classe **Menu** qui permet d'afficher un menu dans une page web :



Il est possible de créer visuellement une arborescence de sous-menus dans ce menu grâce à la propriété **Items** de la classe Menu :



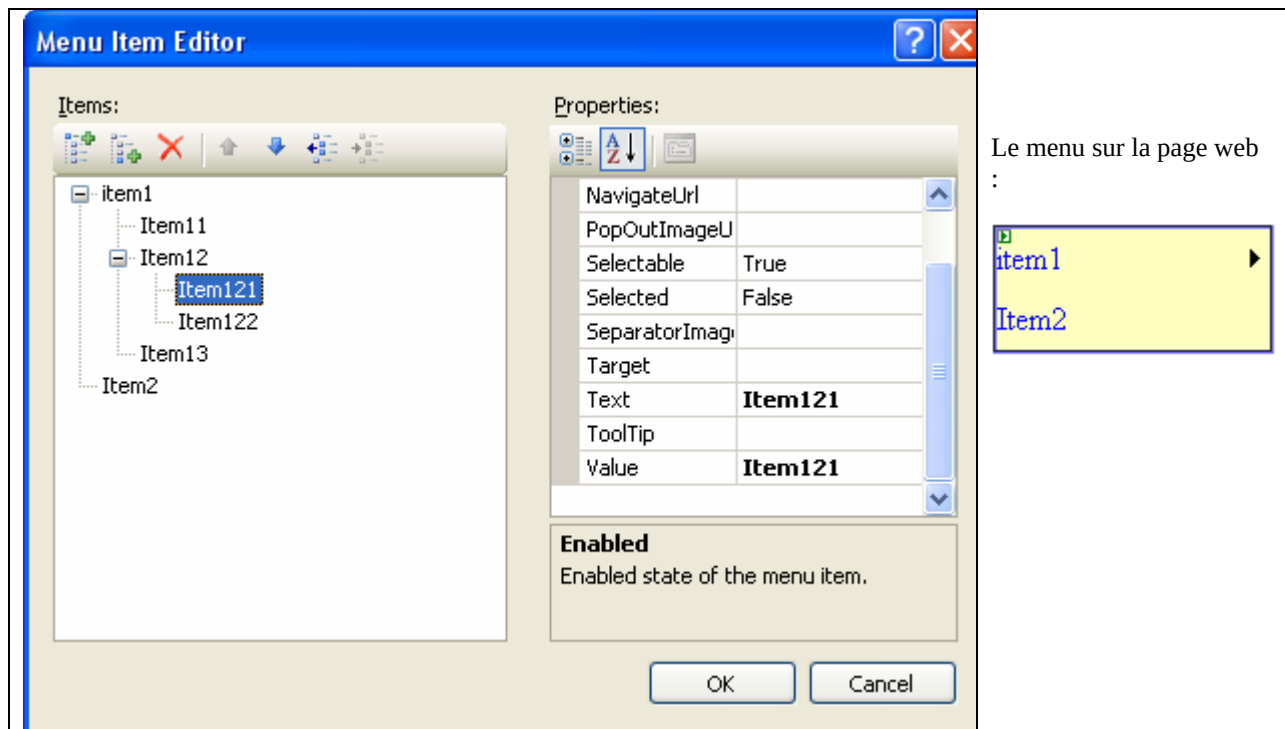
```
public MenuItemCollection Items { get; }
```

Cette propriété représente une collection d'objets de classe `MenuItem`.

public sealed class MenuItemCollection : ICollection, IEnumerable, IStateManager
 Cette classe contient un indexeur permettant d'accéder à chaque objet `MenuItem` :
public MenuItem this [**int** index] { get; }

public sealed class MenuItem : IStateManager, ICloneable
 Cette classe possède une propriété Text contenant le libellé du sous-menu.
public string Text { get; set; }

Il est possible d'utiliser un éditeur de propriété de Visual Studio afin de construire visuellement l'arborescence de sous-menu :



Code Asp.Net du Menu ainsi construit :

```
<form id="form1" runat="server">
  <div>
    <asp:Menu ID="Menu1" runat="server" BackColor="#FFFFC0"
    BorderColor="#8080FF"
    BorderStyle="Groove" BorderWidth="2px" Width="140px"
    Height="64px">
      <Items>
        <asp:MenuItem Text="item1" Value="item1">
          <asp:MenuItem Text="Item11"
Value="Item11"></asp:MenuItem>
          <asp:MenuItem Text="Item12" Value="Item12">
            <asp:MenuItem Text="Item121"
Value="Item121"></asp:MenuItem>
            <asp:MenuItem Text="Item122"
Value="Item122"></asp:MenuItem>
          </asp:MenuItem>
          <asp:MenuItem Text="Item13"
Value="Item13"></asp:MenuItem>
        </asp:MenuItem>
        <asp:MenuItem Text="Item2" Value="Item2"></asp:MenuItem>
      </Items>
      <DynamicMenuItemStyle BackColor="#FFFFC0" BorderColor="#8080FF"
        BorderStyle="Ridge"
        BorderWidth="2px" />
    </asp:Menu>
  </div>
</form>
```

Code HTML engendré pour le Menu précédent : (remarquez les appels au javascript généré)



```
<head><title>Untitled Page</title>
```

```

<style type="text/css">
    .Menu1_0 { background-color: white; visibility:hidden; display:none;
position:absolute; left:0px;top:0px; }
    .Menu1_1 { text-decoration:none; }
    .Menu1_2 { background-color:#FFFFC0;border-color:#8080FF;border-
width:2px;border-style:Groove;height:64px;width:140px; }
    .Menu1_3 { }
    .Menu1_4 { background-color:#FFFFC0;border-color:#8080FF;border-
width:2px;border-style:Ridge; }

</style></head>

<body>
<form name="form1" method="post" action="Default.aspx" id="form1">
<div>
<input type="hidden" name="__EVENTTARGET" id="__EVENTTARGET" value="" />
<input type="hidden" name="__EVENTARGUMENT" id="__EVENTARGUMENT" value="" />
<input type="hidden" name="__VIEWSTATE" id="__VIEWSTATE"
value="/wEPDwUJNTc30TkxNTEwZGSARJ/9WCLlkb/ok842mKSeYFGUyg==" />
</div>

```

```

<script type="text/javascript">
<!--
var theForm = document.forms['form1'];
if (!theForm) {
    theForm = document.form1;
}
function __doPostBack(eventTarget, eventArgument) {
    if (!theForm.onsubmit || (theForm.onsubmit() !=
false)) {
        theForm.__EVENTTARGET.value = eventTarget;
        theForm.__EVENTARGUMENT.value =
eventArgument;
        theForm.submit();
    }
}
// -->
</script>

```

```

<script
src="/WebSiteServCtrl2/WebResource.axd?d=BQ2832wAAAdlkKigxtyKV4Q2&amp;t=632738
879195937500" type="text/javascript">
</script>

```

Fichier de code javascript de gestion d'une
fiche (déjà écrit cf. annexe).

```

<script
src="/WebSiteServCtrl2/WebResource.axd?d=stc__7z_BWk1ov4p1EicdA2&amp;t=632738
879195937500" type="text/javascript">
</script>

```

Fichier de code javascript de gestion de
menus (déjà écrit cf. annexe).

Ces fichiers de code javascript nommés `WebResource.axd?d=.....` sont auto générés par Visuel Studio et se trouvent dans "`C:\Documents and Settings\Administrateur\Local Settings\Temporary Internet Files`" (ici le nom d'utilisateur est "`Administrateur`") :

C:\Documents and Settings*votreNomUtilisateur*\Local Settings\Temporary Internet Files:

Nom	Adresse Internet
Default.aspx	view-source:http://localhost/WebSiteServCtrl2/Default.aspx
WebResource.axd?d=40dWBmh5JB94j4P_hX-jzNopyRqFAQF5gdcZcekCEfk...	http://localhost/WebSiteServCtrl2/WebResource.axd?d=40dWBmh5JB94j4P_hX-jzNopyRqFAQF5gdcZcekCEfk...
WebResource.axd?d=40dWBmh5JB94j4P_hX-jzMgnL1NYlJ1X5KRooaFMVYk...	http://localhost/WebSiteServCtrl2/WebResource.axd?d=40dWBmh5JB94j4P_hX-jzMgnL1NYlJ1X5KRooaFMVYk...
WebResource.axd?d=t0BwhNHjAD7Z4AckiXLH5Wdrpiqgp9jN0a6_Au0mQVE...	http://localhost/WebSiteServCtrl2/WebResource.axd?d=t0BwhNHjAD7Z4AckiXLH5Wdrpiqgp9jN0a6_Au0mQVE...
WebResource.axd?d=JBDdLBfoNmBArrcf-A6P2g2&t=632738879195937500	http://localhost/WebSiteServCtrl2/WebResource.axd?d=JBDdLBfoNmBArrcf-A6P2g2&t=632738879195937500
WebResource.axd?d=B8o8yLHFikai-UmHjueg2_fiFIVM8HAr3u-akpSun5U1&...	http://localhost/WebSiteServCtrl2/WebResource.axd?d=B8o8yLHFikai-UmHjueg2_fiFIVM8HAr3u-akpSun5U1&...
WebResource.axd?d=LN1hR4BWAGOU64YM2MVYbNfLaEvdlaGsb2pts53rk...	http://localhost/WebSiteServCtrl2/WebResource.axd?d=LN1hR4BWAGOU64YM2MVYbNfLaEvdlaGsb2pts53rk...
WebResource.axd?d=stc__7z_BWk1ov4p1EicdA2&t=632738879195937500	http://localhost/WebSiteServCtrl2/WebResource.axd?d=stc__7z_BWk1ov4p1EicdA2&t=632738879195937500
WebResource.axd?d=u9eZty7kcs7SiKaCaDTrlg2&t=632738879195937500	http://localhost/WebSiteServCtrl2/WebResource.axd?d=u9eZty7kcs7SiKaCaDTrlg2&t=632738879195937500
Default.aspx	http://localhost/WebSiteServCtrl2/Default.aspx

```

<div>
<a href="#Menu1_SkipLink">

</a>
<table id="Menu1" class="Menu1_2" cellpadding="0" cellspacing="0" border="0">
<tr onmouseover="Menu_HoverStatic(this)" onmouseout="Menu_Unhover(this)"
onkeyup="Menu_Key(this)" id="Menu1n0">
<td><table cellpadding="0" cellspacing="0" border="0"
width="100%">
<tr>
<td style="white-space:nowrap;width:100%;"><a
class="Menu1_1"
href="javascript:__doPostBack('Menu1','item1')">item1</a></td><td
style="width:0;"></td>
</tr>
</table></td>
</tr><tr onmouseover="Menu_HoverStatic(this)"
onmouseout="Menu_Unhover(this)" onkeyup="Menu_Key(this)" id="Menu1n1">
<td><table cellpadding="0" cellspacing="0" border="0"
width="100%">
<tr>
<td style="white-space:nowrap;width:100%;"><a
class="Menu1_1"
href="javascript:__doPostBack('Menu1','Item2')">Item2</a></td>
</tr>
</table></td>
</tr>
</table><div id="Menu1n0Items" class="Menu1_0">
<table border="0" cellpadding="0" cellspacing="0">
<tr onmouseover="Menu_HoverDynamic(this)" onmouseout="Menu_Unhover(this)"
onkeyup="Menu_Key(this)" id="Menu1n2">
<td><table class="Menu1_4" cellpadding="0" cellspacing="0" border="0"
width="100%">
<tr>
<td style="white-space:nowrap;width:100%;"><a class="Menu1_1 Menu1_3"
href="javascript:__doPostBack('Menu1','item1\\Item11')">Item11</a></td>
</tr>
</table></td>
</tr><tr onmouseover="Menu_HoverDynamic(this)"
onmouseout="Menu_Unhover(this)" onkeyup="Menu_Key(this)" id="Menu1n3">

```

```

<td><table class="Menu1_4" cellpadding="0" cellspacing="0" border="0"
width="100%">
<tr>
<td style="white-space:nowrap;width:100%;"><a class="Menu1_1 Menu1_3"
href="javascript:__doPostBack('Menu1','item1\\Item12')">Item12</a></td><td
style="width:0;"></td>
</tr>
</table></td>
</tr><tr onmouseover="Menu_HoverDynamic(this)"
onmouseout="Menu_Unhover(this)" onkeyup="Menu_Key(this)" id="Menu1n4">
<td><table class="Menu1_4" cellpadding="0" cellspacing="0" border="0"
width="100%">
<tr>
<td style="white-space:nowrap;width:100%;"><a class="Menu1_1 Menu1_3"
href="javascript:__doPostBack('Menu1','item1\\Item13')">Item13</a></td>
</tr>
</table></td>
</tr>
</table><div class="Menu1_4 Menu1_0" id="Menu1n0ItemsUp"
onmouseover="PopOut_Up(this)" onmouseout="PopOut_Stop(this)" style="text-
align:center;">

</div><div class="Menu1_4 Menu1_0" id="Menu1n0ItemsDn"
onmouseover="PopOut_Down(this)" onmouseout="PopOut_Stop(this)" style="text-
align:center;">

</div>
</div><div id="Menu1n3Items" class="Menu1_0">
<table border="0" cellpadding="0" cellspacing="0">
<tr onmouseover="Menu_HoverDynamic(this)" onmouseout="Menu_Unhover(this)"
onkeyup="Menu_Key(this)" id="Menu1n5">
<td><table class="Menu1_4" cellpadding="0" cellspacing="0" border="0"
width="100%">
<tr>
<td style="white-space:nowrap;width:100%;"><a class="Menu1_1 Menu1_3"
href="javascript:__doPostBack('Menu1','item1\\Item12\\Item121')">Item121</a><
/td>
</tr>
</table></td>
</tr><tr onmouseover="Menu_HoverDynamic(this)"
onmouseout="Menu_Unhover(this)" onkeyup="Menu_Key(this)" id="Menu1n6">
<td><table class="Menu1_4" cellpadding="0" cellspacing="0" border="0"
width="100%">
<tr>
<td style="white-space:nowrap;width:100%;"><a class="Menu1_1 Menu1_3"
href="javascript:__doPostBack('Menu1','item1\\Item12\\Item122')">Item122</a><
/td>
</tr>
</table></td>
</tr>
</table><div class="Menu1_4 Menu1_0" id="Menu1n3ItemsUp"
onmouseover="PopOut_Up(this)" onmouseout="PopOut_Stop(this)" style="text-
align:center;">

```

```


</div><div class="Menu1_4 Menu1_0" id="Menu1n3ItemsDn"
onmouseover="PopOut_Down(this)" onmouseout="PopOut_Stop(this)" style="text-
align:center;">

</div>
</div><a id="Menu1_SkipLink"></a>
</div>
<div>
<input type="hidden" name="__EVENTVALIDATION" id="__EVENTVALIDATION"
value="/wEWCAKdo674AQLh6ozdAwL+4ozdAwKPrOzfBwKPrMC0DwKPrNTpBAKjqPjVDgKjq0TwCa
hFukoD2hz8YefJjmjjDVyK2FwP" />
</div>

```

```

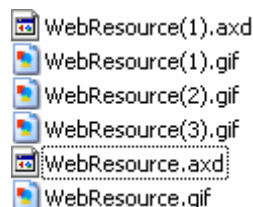
<script
type="text/javascript">
<!--
var Menu1_Data = new
Object();
Menu1_Data.disappearAfter
= 500;
Menu1_Data.horizontalOffs
et = 0;
Menu1_Data.verticalOffset
= 0;
// -->
</script>
</form>
</body>

```

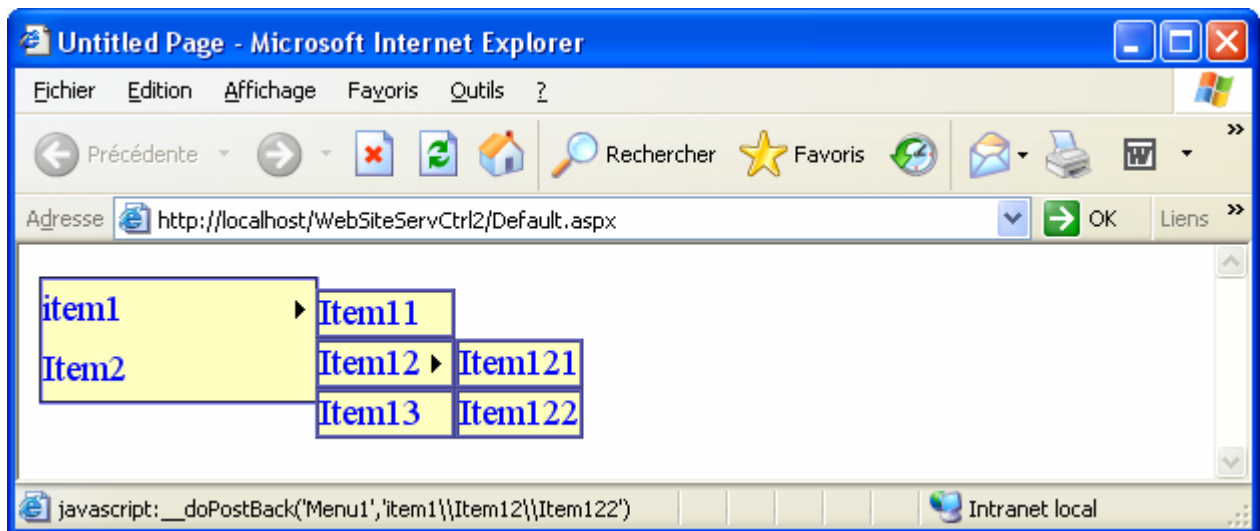
- Les fichiers contenant des fonctions javascript servant à gérer la fiche et les menus "[WebResource.axd?d=BQ2832wAAdlkKigxtyKV4Q2&t=632738879195937500](#)" et "[WebResource.axd?d=stc__7z_Bwk1ov4p1EicdA2&t=632738879195937500](#)" peuvent être trouvés dans le dossier des fichiers temporaires Internet :

C:\Documents and Settings\VotreNom\Local Settings\Temporary Internet Files

- Si l'on sauvegarde la page HTML à partir du navigateur, on récupère alors localement, une copie de ces deux fichiers (ici [WebResource\(1\).axd](#) et [WebResource.axd](#)) :

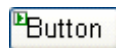


Affichage par le navigateur du code HTML précédent :

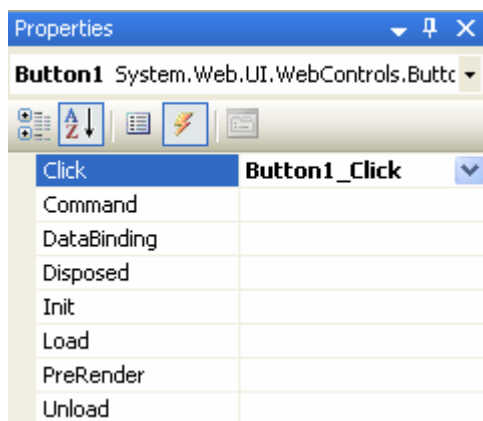


3°) Les événements gérés côté serveur sont dans une page séparée (code-behind) :

Soit un contrôle de classe **System.Web.UI.WebControls.Button** :



Ce contrôle est sensible à 8 événements, il suffit de programmer sa réaction à un des événements comme par exemple le click de l'utilisateur sur ce bouton :

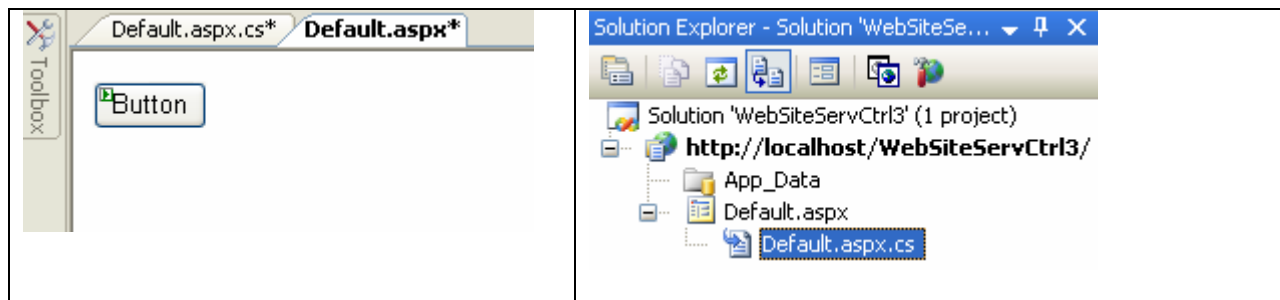


Si la forme visuelle du bouton se trouve déposée dans une page nommée **Default.aspx**, le code C# du gestionnaire de réaction à chacun des 8 événements du bouton se trouve dans une autre page associée nommée **Default.aspx.cs**.

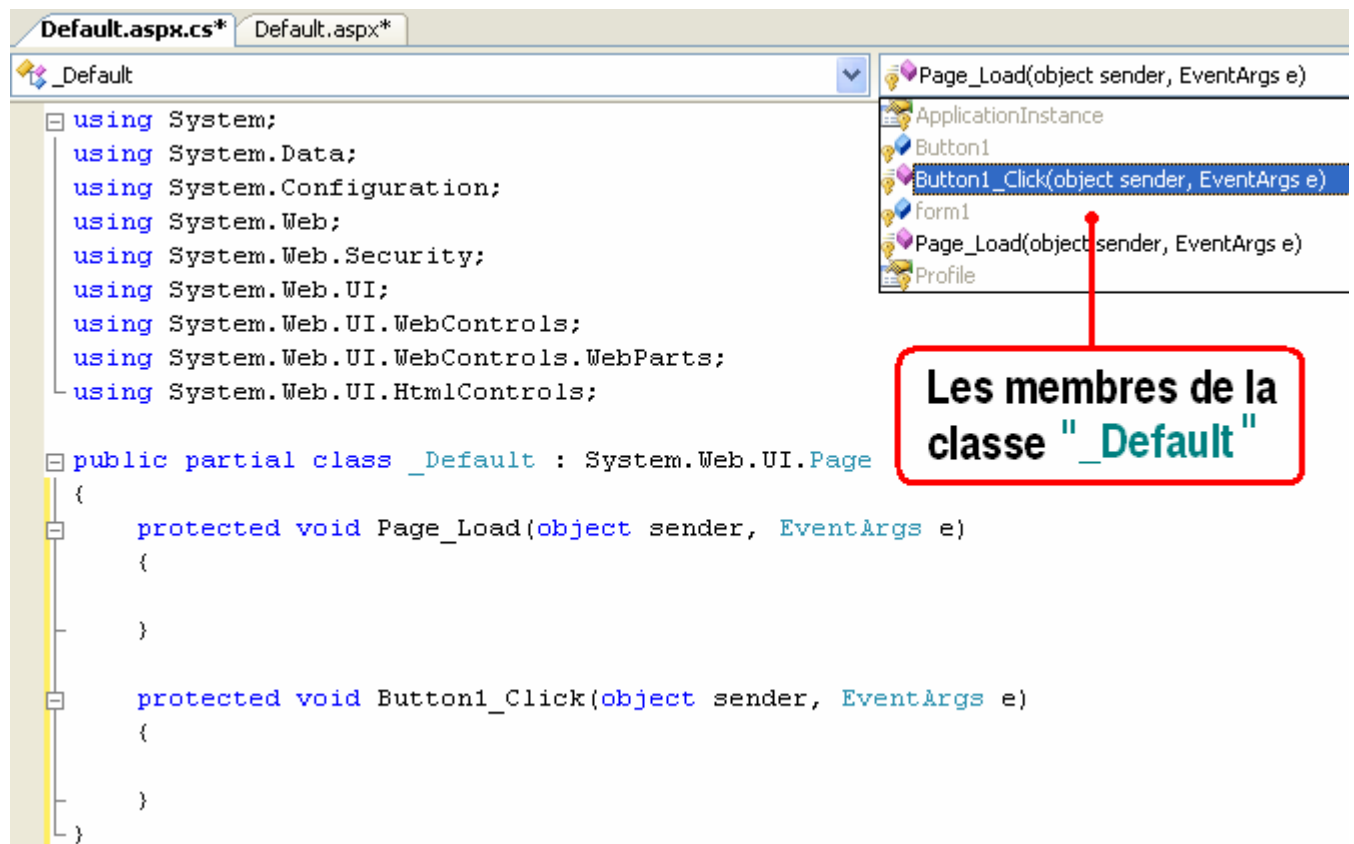
Remarque :

Cette page de code reprend le même nom que la page à laquelle elle est associée et rajoute le suffixe **cs** pour du code de réaction du type **C#**, le suffixe est **vb** (**Default.aspx.vbs**) pour du code de type **VB.Net**, **js** pour le java **J#** et ainsi de suite pour tout langage supporté par la plateforme .Net.

La page visuelle du bouton	La page de code associée dans l'explorateur



Lors de la demande de création du gestionnaire de réaction au click de souris sur le bouton, Visual Studio engendre une classe partielle nommée `_Default` qui hérite de la classe `System.Web.UI.Page` : cette technique est appelée **code-behind** (code en arrière, code en coulisse, code caché...), en outre la classe partielle permet d'écrire plusieurs fichiers de code et de les assembler en une seule classe à la compilation.



La classe `_Default` dérivée de `Page` est associée à des fichiers de suffixe **aspx**. Ces fichiers sont compilés lors de l'exécution comme des objets de type `Page` et stockés dans la mémoire cache du serveur.

Tous les RAD qui supportent la construction de pages web avec la technique du **code-behind** dans .Net doivent créer une classe héritant de la classe `Page`.

Dans la classe de notre exemple, nous trouvons deux squelettes de gestionnaires d'événements :

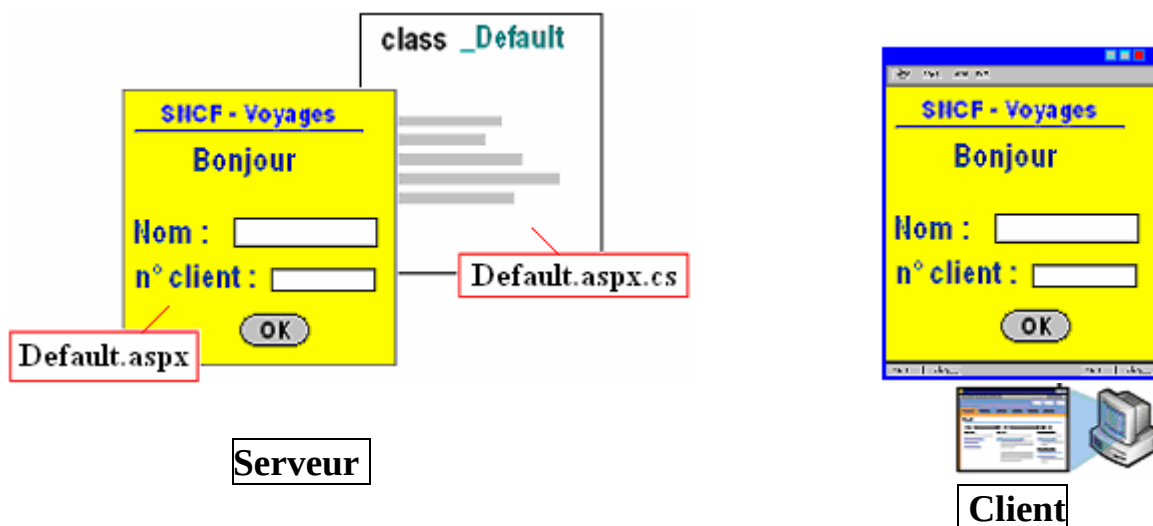
- Le premier "`protected void Page_Load(object sender, EventArgs e)`" est un gestionnaire de l'événement "Load" de la page elle-même,
- le second "`protected void Button1_Click(object sender, EventArgs e)`" est le gestionnaire de l'événement "click" du bouton sur la page.

L'événement "[Load](#)" a lieu à chaque chargement de la page à partir du serveur c'est l'un des 15 événements possibles pouvant avoir lieu sur une page web :

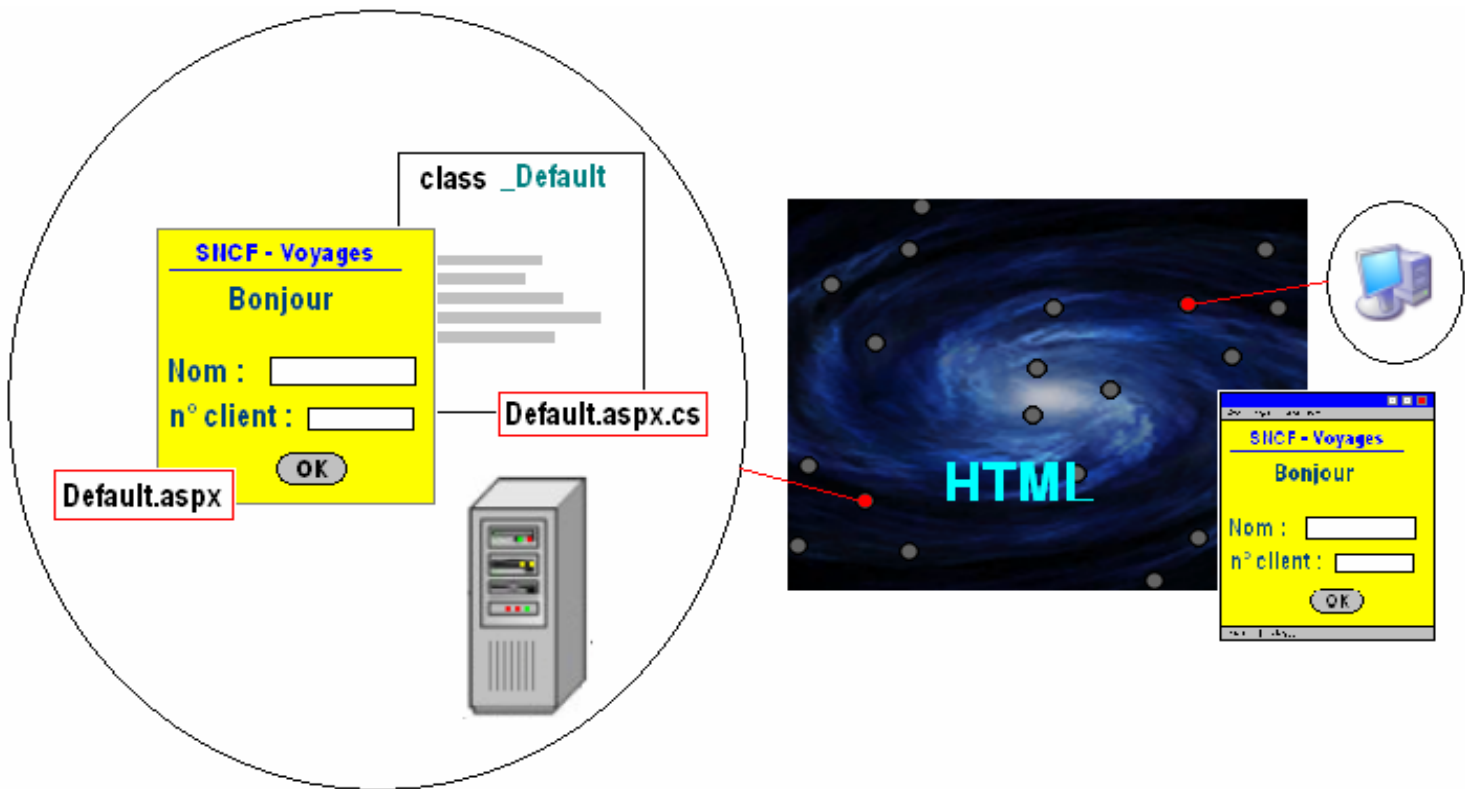
Public Events

Name	Name
 AbortTransaction	 Load
 CommitTransaction	 LoadComplete
 DataBinding	 PreInit
 Disposed	 PreLoad
 Error	 PreRender
 Init	 PreRenderComplete
 InitComplete	 SaveStateComplete
	 Unload

Le serveur utilise les deux fichiers "Default.aspx" et "Default.aspx.cs" pour engendrer le rendu de la page web du client :



Le serveur engendre le rendu de la page web du client à partir des deux fichiers "Default.aspx" et "Default.aspx.vb", il synthétise les résultats du travail dans un fichier HTML final qui est envoyé au navigateur du client via le réseau Internet :



Code Asp.Net de la page [Default.aspx](#) ainsi construite :

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default.aspx.cs"
Inherits="_Default" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" >
<head runat="server">
  <title>Untitled Page</title>
</head>
<body>
  <form id="form1" runat="server">
    <div>
      <asp:Button ID="Button1" runat="server" OnClick="Button1_Click"
Text="Button" />
    </div>
  </form>
</body>
</html>
```



Code HTML engendré pour la page précédente et envoyé au client :

```
<head>
<title> Untitled Page </title>
</head>
<body>
  <form name="form1" method="post" action="Default.aspx" id="form1">
<div>
```

```

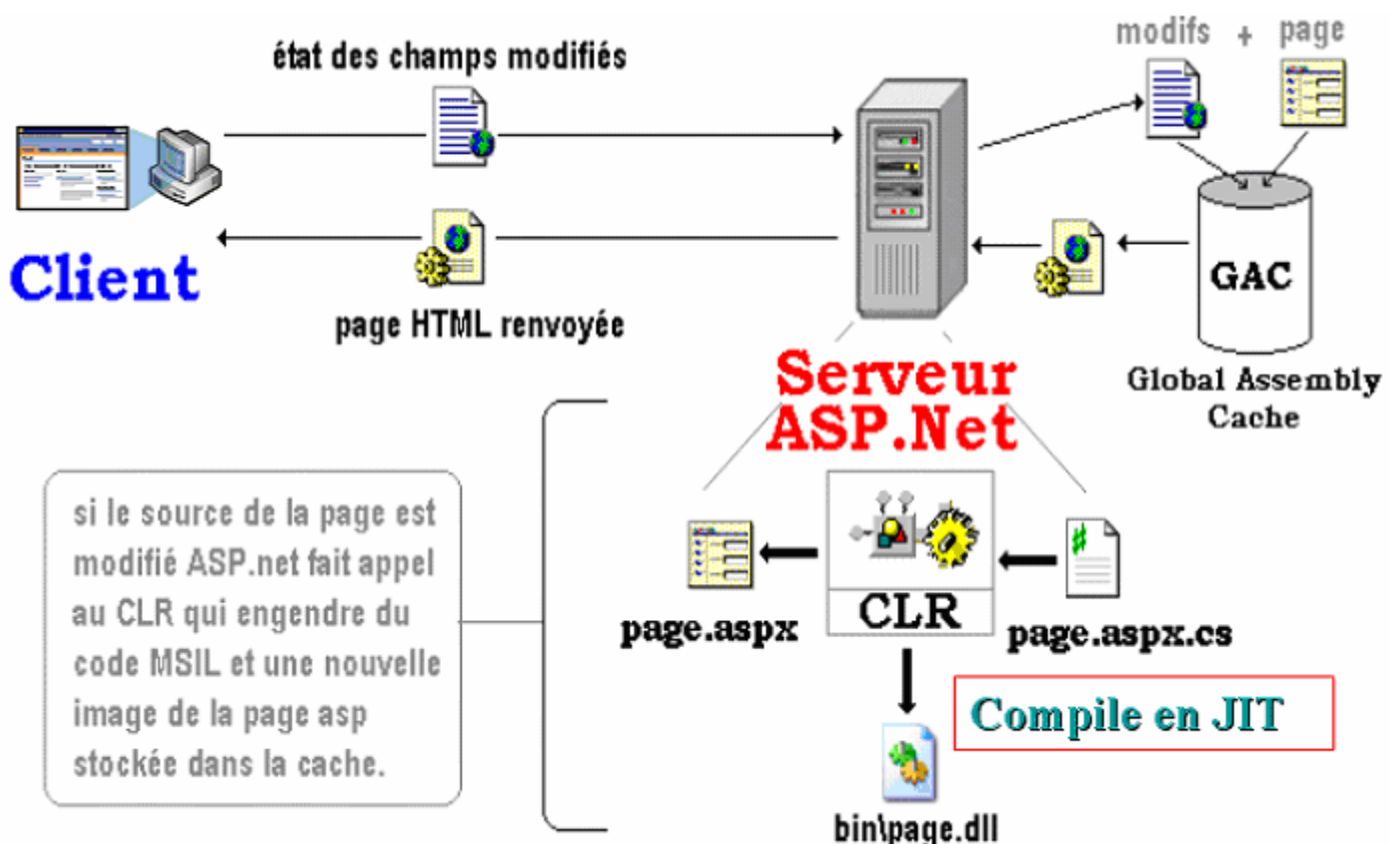
<input type="hidden" name="__VIEWSTATE" id="__VIEWSTATE"
value="/wEPDwUKMTQ2OTkzNDMyMWRkU8v8pN7KXptqdUtkKvrRf6V6R1A=" />
</div>
<div>
<input type="submit" name="Button1" value="Button" id="Button1" />
</div>
<div>
<input type="hidden" name="__EVENTVALIDATION" id="__EVENTVALIDATION"
value="/wEWAqKUGJGxCwKM54rGBkvwYpMSUXqF04d0bV7v1BzrERyr" />
</div></form>
</body>

```

Le code HTML envoyé au navigateur client est identique à celui qui a déjà été envoyé dans le cas déjà envisagé du bouton sans réaction événementielle.

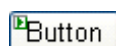
En effet, c'est le serveur ASP.Net qui maintient des fichiers de gestion des événements de la **Page** dans le fichier **Default.aspx.cs** et qui les exécute à partir de la mémoire cache du serveur selon les requêtes de l'utilisateur puis finalement renvoie une page HTML au navigateur client qui portera le nom **Default.aspx**.

Cette page HTML **Default.aspx** contient la réponse du serveur à la requête faite par le client.



4°) Les événements gérés côté serveur sont dans un script inclus (code-inline) :

Soit le même contrôle de classe **System.Web.UI.WebControls.Button** :



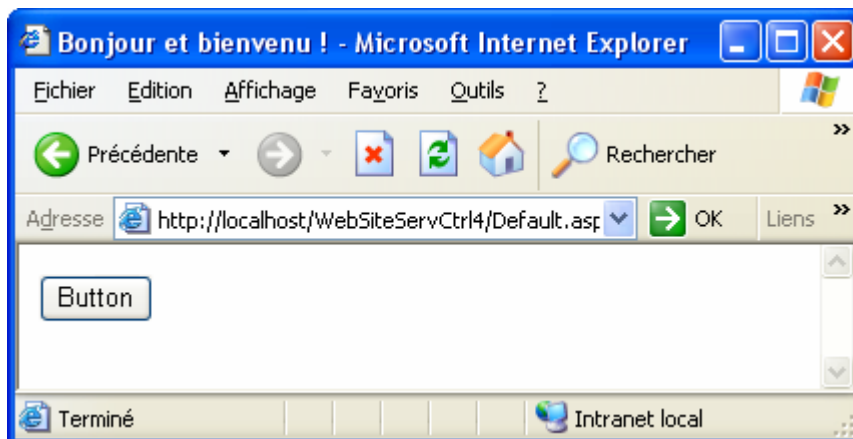
Au lieu de gérer sa réaction au click dans une page séparée, ASP.Net permet d'utiliser un deuxième modèle d'écriture le script inclus dans le code ASP.Net de la page (tous les langages supportés par .Net sont autorisés à utiliser ce modèle, alors qu'à ce jour le modèle de code behind n'est disponible que pour C# et VB à cause de la notion de classe partielle) :

```
<%@ Page Language="C#" %>
```

```
<script runat="server">
    protected void Button1_Click(object sender,
    EventArgs e)
    {
        this.Title = "Bonjour et bienvenu !" ;
    }
</script>
```

```
<html>
<head runat="server">
    <title>Untitled Page</title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <asp:Button ID="Button1" runat="server" OnClick="Button1_Click"
            Text="Button" />
        </div>
    </form>
</body>
</html>
```

Lorsque l'utilisateur clique sur le bouton, après envoi de la requête le serveur exécute le script du gestionnaire de l'événement click, comme il le fait si ce gestionnaire est situé dans une page à part (ici, la réponse consiste à modifier la barre de titre du navigateur en y mettant un message de bienvenue) :



Le panachage entre les deux modèles **code-inline** et **code-behind** est permis pour le même contrôle ou pour des contrôles différents.

Soit par exemple à programmer la réaction du bouton à l'événement click de souris grâce à un script C# et l'événement command du bouton avec du code séparé.

Code Asp.Net de la page [Default.aspx](#) construite :

```
<%@ Page Language="C#" AutoEventWireup="true"
    CodeFile="Default.aspx.cs"
    Inherits="_Default" %>
```

```

<script runat="server">
    protected void Button1_Click(object sender, EventArgs e)
    {
        this.Title = "Bonjour et bienvenu !" ;
    }
</script>

<html>
<head runat="server">
    <title>Untitled Page</title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <asp:Button ID="Button1" runat="server"
                OnClick="Button1_Click"
                OnCommand="Button1_Command" Text="Button" />
        </div>
    </form>
</body>
</html>

```

Code C# de la page [Default.aspx.cs](#) associée :

```

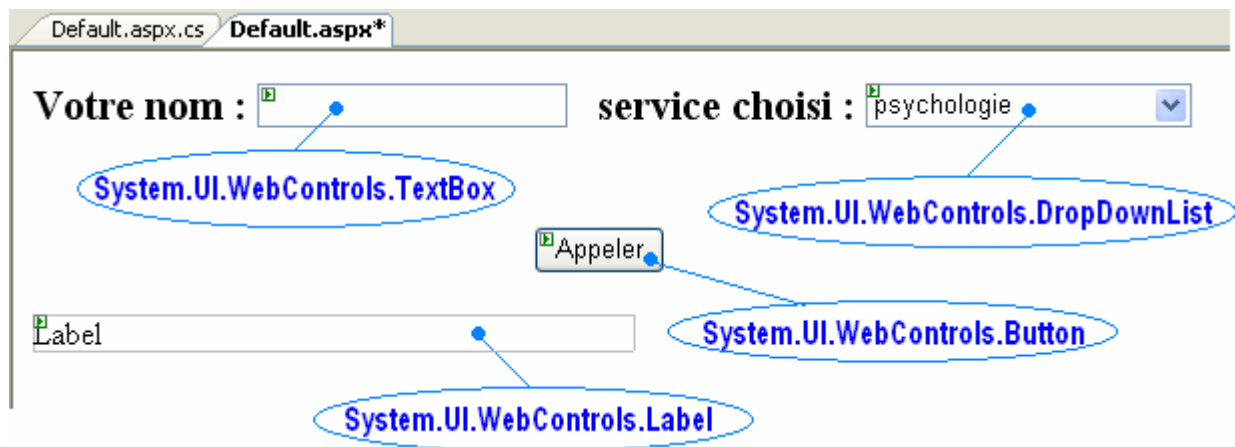
using System;
using System.Data;
using System.Configuration;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Web.UI.HtmlControls;

public partial class _Default : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
    }
    protected void Button1_Command(object sender,
    CommandEventArgs e)
    {
        Button1.Text = "Merci de votre visite";
    }
}

```

5°) Les événements côté serveur permettent des communications entre les contrôles :

Soit la page web suivante construite avec 4 contrôles web s'exécutant côté serveur :



Lorsque l'utilisateur clique dans sa page web sur le bouton [Appeler], le serveur exécute le code du gestionnaire de l'événement click programmé ci-dessous :

Code C# de la classe `_Default` dans la page `Default.aspx.cs` associée :

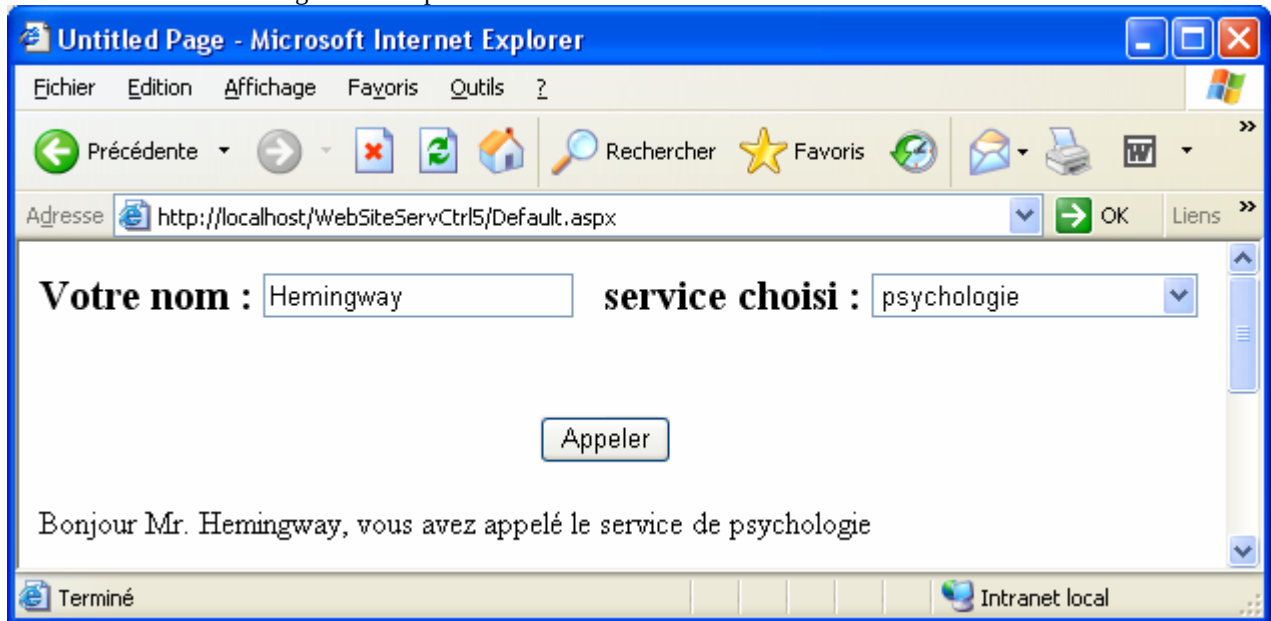
```
public partial class _Default : System.Web.UI.Page
{
    protected void Button1_Click(object sender, EventArgs e)
    {
        if (DropDownList1.SelectedValue != "" & TextBox1.Text
        != "")
        {
            Label1.Text = "Bonjour Mr. " + TextBox1.Text +
            ", vous avez appelé le service de "
+
DropDownList1.SelectedValue;
        }
        else
        {
            if (DropDownList1.SelectedValue == "")
                Label1.Text = "Choisissez un service";
            else
                if (TextBox1.Text == "")
                    Label1.Text = "Entrez votre nom";
        }
    }
}
```

Affichages dans le navigateur client :



Cas où l'utilisateur n'a pas renseigné le champ **nom** :

Cas où l'utilisateur a renseigné le champ **nom** :



Code HTML engendré pour la page précédente et envoyé au client :

[illegible]

```
<input type="hidden" name="__EVENTVALIDATION" id="__EVENTVALIDATION"
value="/wEWBgL9w6GfDQLs0bLrBgLpo+XtBwL0go0iCQLfi9rZCAKM54rGBrH1QhYvJ/0kbCf0Fj
QcFvZhLM0a" />
</div>
</form>
</body>
</html>
```

Pour le debugging on trouve les fichiers xxx.dll et xxx.cs dans :

C:\WINDOWS\Microsoft.NET\Framework\v2.0.50727\Temporary ASP.NET
Files\websiteservctrl2\f6cd9c41\1150c7b3

Communication entre pages web



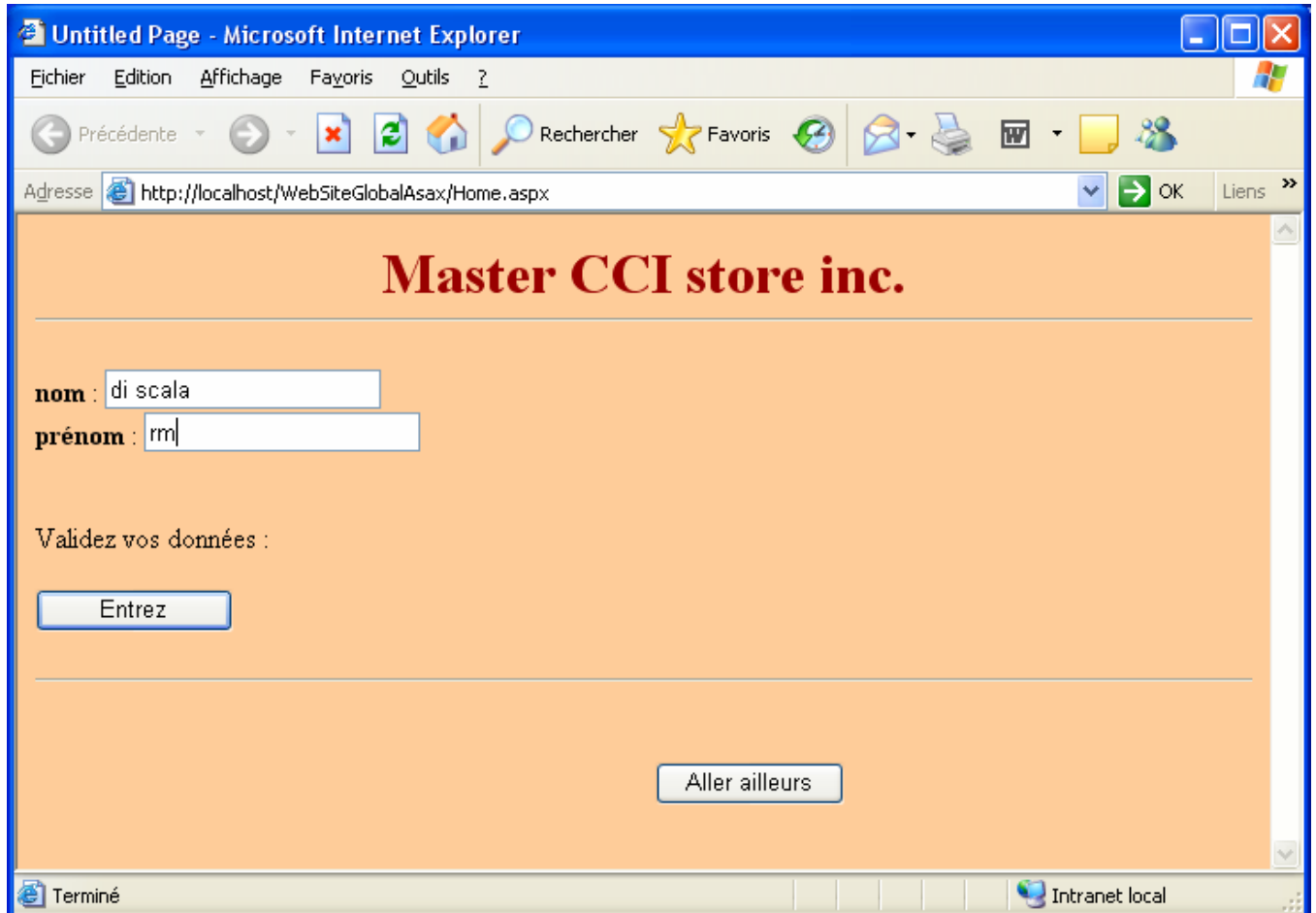
Plan général:

Objets utilisé pour la communication entre pages

- 1°) Objet Session
- 2°) Fichier Global.asax
- 3°) Répertoire App_Code

Communication entre pages web avec : l'objet de **Session**, le fichier **Global.asax** et le répertoire **App_Code**

Exemple d'utilisation avec deux pages :



Page Home.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Home.aspx.cs" Inherits="_Default" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" >
<head runat="server">
  <title>Untitled Page</title>
</head>
<body bgcolor="#ffcc99">
  <form id="form1" runat="server">
    <div style="text-align: center">
      <span style="font-size: 24pt; color: #6699ff"><strong><span style="color: #990000">Master
        CCI store inc.</span><br />
      </strong></span>
    </div>
  </form>
</body>
```



```

    {
        this.Page.Session["Nom"] = nom;
        this.Page.Session["Prenom"] = prenom;
        Response.Redirect("Client.aspx");
    }
}
protected void ButtonEndSession_Click(object sender, EventArgs e)
{
    this.Page.Session.Abandon();
    Response.Redirect("http://www.discala.net/");
}
}

```

Fichier Global.asax

```
<%@ Application Language="C#" %>
```

```
<script RunAt="server">
```

```

void Application_Start(object sender, EventArgs e)
{
    // Code that runs on application startup
}

void Application_End(object sender, EventArgs e)
{
    // Code that runs on application shutdown
}

void Application_Error(object sender, EventArgs e)
{
    // Code that runs when an unhandled error occurs
}

void Session_Start(object sender, EventArgs e)
{
    // Code that runs when a new session is started
    this.Session["montant"] = 0d;
    this.Session["Nom"] = "...";
    this.Session["Prenom"] = "...";
    Class1.LireFichier();
}

void Session_End(object sender, EventArgs e)
{
    // Code that runs when a session ends.
    // Note: The Session_End event is raised only when the sessionstate mode
    // is set to InProc in the Web.config file. If session mode is set to StateServer
    // or SQLServer, the event is not raised.
    Class1.ecrireFichier((string)Session["Nom"], (string)Session["Prenom"]);
}

</script>

```

Code dans App_Code : Class1.cs

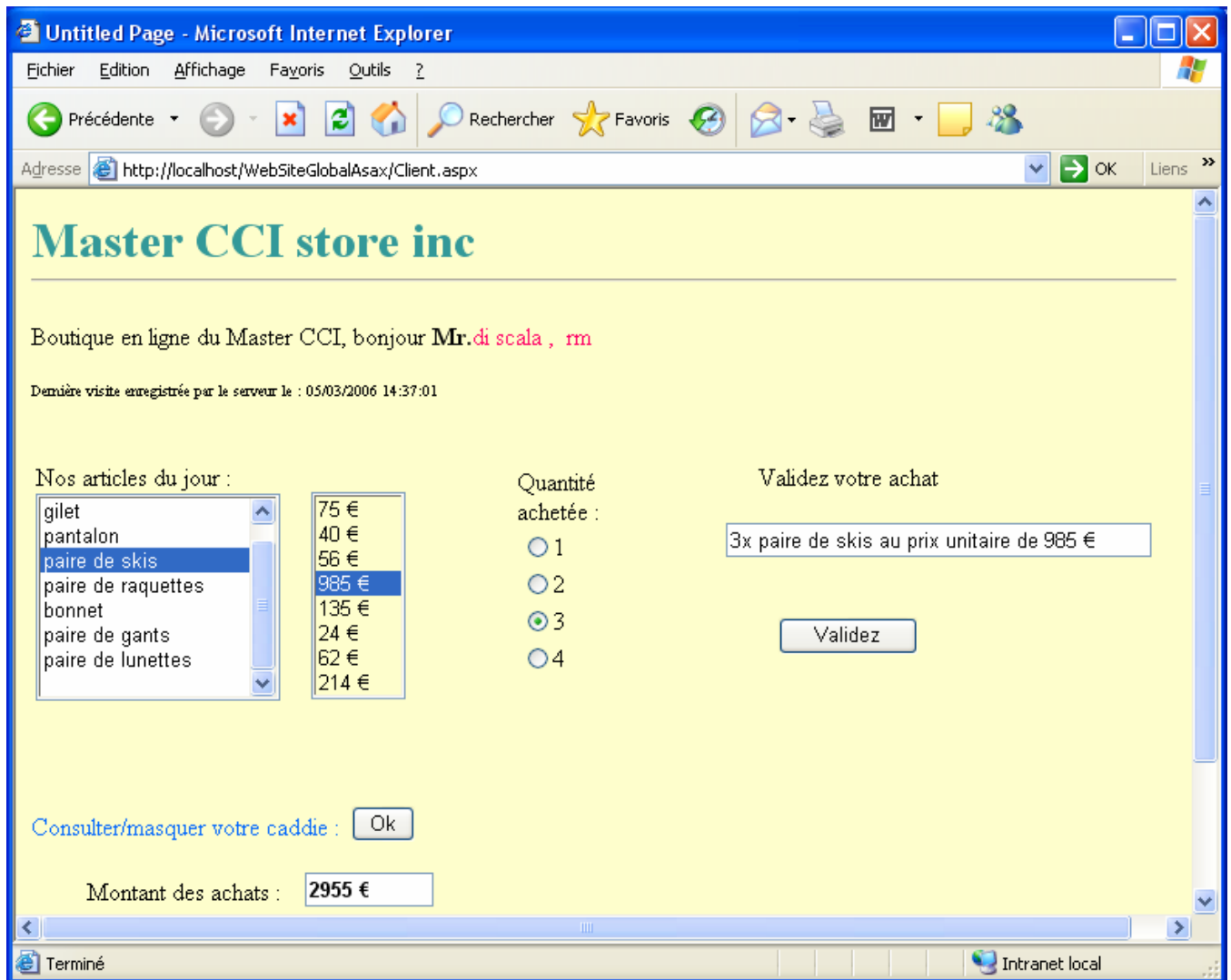
```
using System;
using System.Data;
using System.Configuration;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Web.UI.HtmlControls;
using System.Collections.Generic;
using System.IO;

/// <summary>
/// Class1 dans le dossier App_Code (version ASP.net 2.0)
/// </summary>
public class Class1
{
    public static List<string> listeClients = new List<string>();
    private static string fichierClients = @"C:\Inetpub\Datas\clients.txt";

    public Class1()
    {
        //
        // TODO: Add constructor logic here
        //
    }
    public static void LireFichier()
    {
        if (File.Exists(fichierClients))
        {
            using (StreamReader fluxRead = new StreamReader(fichierClients))
            {
                string ligne;
                listeClients.Clear();
                while ((ligne = fluxRead.ReadLine()) != null)
                {
                    listeClients.Add(ligne);
                }
            }
        }
    }
    public static void ecrireFichier(string nom, string prenom)
    {
        if (nom.Length != 0 & prenom.Length != 0)
        {
            if (File.Exists(fichierClients))
            {
                using (StreamWriter fluxWrite = new StreamWriter(fichierClients, true))
                {
                    fluxWrite.WriteLine(nom + "*" + prenom + "*" + DateTime.Now);
                }
            }
            else
            {
                using (StreamWriter fluxWrite = new StreamWriter(fichierClients))
                {
                    fluxWrite.WriteLine(nom + "*" + prenom + "*" + DateTime.Now);
                }
            }
        }
    }
}
```

Il est maintenu en accès **static** une recopie en MC du fichier des clients déjà passés sur le site "c:\inetpub\Datas\clients.txt", lors de l'ouverture d'une session d'un nouvel utilisateur : `public static List<string> listeClients` (chargement de la liste, sur l'événement `Session_Start` géré dans `Global.asax`).

Lors de validation des données d'une personne s'étant identifiée, le serveur consulte la liste des clients déjà venus et affiche éventuellement dans la nouvelle page web `Client.aspx`, un rappel de la date de la dernière visite effectuée par cette personne :



Sur cette page `Client.aspx`, l'utilisateur peut sélectionner avec la souris dans une liste, un article dont le prix est automatiquement affiché dans une autre liste. L'utilisateur dispose de la possibilité d'acheter de 1 à 4 articles du même type, le serveur affiche dès validation du choix le montant des achats en cours.

Page `Client.aspx`

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Client.aspx.cs" Inherits="Default2" %>
```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

```
<html xmlns="http://www.w3.org/1999/xhtml" >
```



```

        <br />
        <asp:Button ID="ButtonValider" runat="server" Text="Validez" OnClick="ButtonValider_Click"
                                Width="90px" /><br />
        <br />
        <br />
        <br />
        <br />
    </td>
</tr>
</table>
<br />
<span style="color: #0066ff">Consulter/masquer
votre caddie :</span>&nbsp;
<asp:Button ID="ButtonConsult" runat="server" Text="Ok" OnClick="ButtonConsult_Click" Width="41px"
Height="23px" /><br />
<br />
<asp:ListBox ID="ListBoxCaddie" runat="server" Height="142px" Width="293px" BackColor="Azure"
Visible="False"></asp:ListBox>
&nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; Montant des achats : &nbsp; &nbsp; <asp:TextBox
ID="TextBoxMontant" runat="server" Font-Bold="True" ReadOnly="True"
Width="77px"></asp:TextBox>
&nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; &nbsp; <br />
<br />
<br />
<br />
<asp:HyperLink ID="HyperLinkAccueil" runat="server" Font-Bold="True" NavigateUrl="~/Home.aspx"
Width="139px">Revenir à l'accueil</asp:HyperLink><br />
<br />
</div>
</form>
</body>
</html>

```

```
using System;
using System.Data;
using System.Configuration;
using System.Collections;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Web.UI.HtmlControls;

public partial class Default2 : System.Web.UI.Page
{
    private double montant = 0d;

    protected void Page_Load(object sender, EventArgs e)
    {
        if (!IsPostBack) // premier chargement de cette page
        {
            LabelDejaVenu.Visible = searchIdentite((string)Page.Session["Nom"], (string)Page.Session["Prenom"]);
        }
        montant = (double)this.Page.Session["montant"];
    }
}
```

```

private bool searchIdentite(string nom, string prenom)
{
    bool dejaVenu = false;
    LabelDejaVenu.Text = "";
    foreach (string s in Class1.listeClients)
        if (s.IndexOf(nom) != -1 & s.IndexOf(prenom) != -1)
        {
            LabelDejaVenu.Text = "Dernière visite enregistrée par le serveur le : " + s.Split("*")[2];
            dejaVenu = true;
        }
    return dejaVenu;
}

private void selectionShow()
{
    ListBoxCaddie.Visible = false;
    if (RadioButtonListChoix.SelectedIndex == -1)
        RadioButtonListChoix.SelectedIndex = 0;
    this.TextBoxAchat.Text = RadioButtonListChoix.SelectedItem + "x " +
        ListBoxArticle.SelectedItem + " au prix unitaire de " + ListBoxPrix.SelectedItem;
}
protected void ListBoxArticle_SelectedIndexChanged(object sender, EventArgs e)
{
    ListBoxPrix.SelectedIndex = ListBoxArticle.SelectedIndex;
    selectionShow();
}

protected void RadioButtonListChoix_SelectedIndexChanged(object sender, EventArgs e)
{
    selectionShow();
}

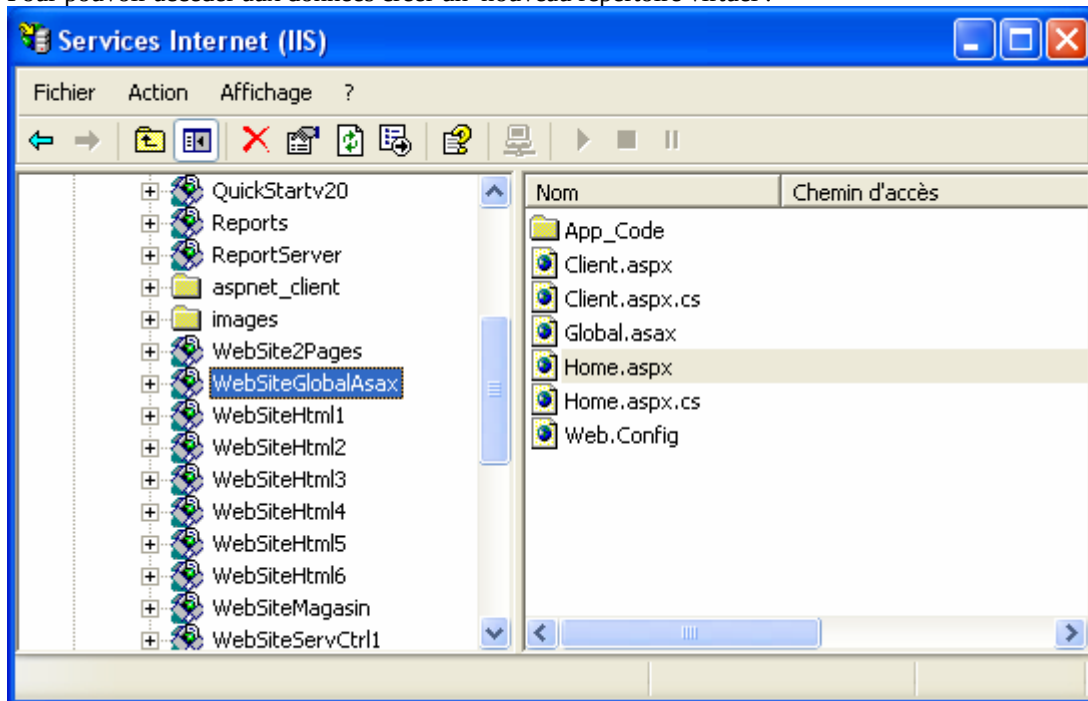
protected void ButtonValider_Click(object sender, EventArgs e)
{
    if (TextBoxAchat.Text.Length != 0)
    {
        selectionShow();
        ListBoxCaddie.Items.Add(TextBoxAchat.Text);
        montant += Convert.ToDouble(ListBoxPrix.SelectedValue) *
Convert.ToDouble(RadioButtonListChoix.SelectedValue);
        TextBoxMontant.Text = montant.ToString() + " €";
        this.Page.Session["montant"] = montant;
    }
}

protected void ButtonConsult_Click(object sender, EventArgs e)
{
    if (TextBoxAchat.Text.Length != 0)
    {
        ListBoxCaddie.Visible = !ListBoxCaddie.Visible;
    }
}
}

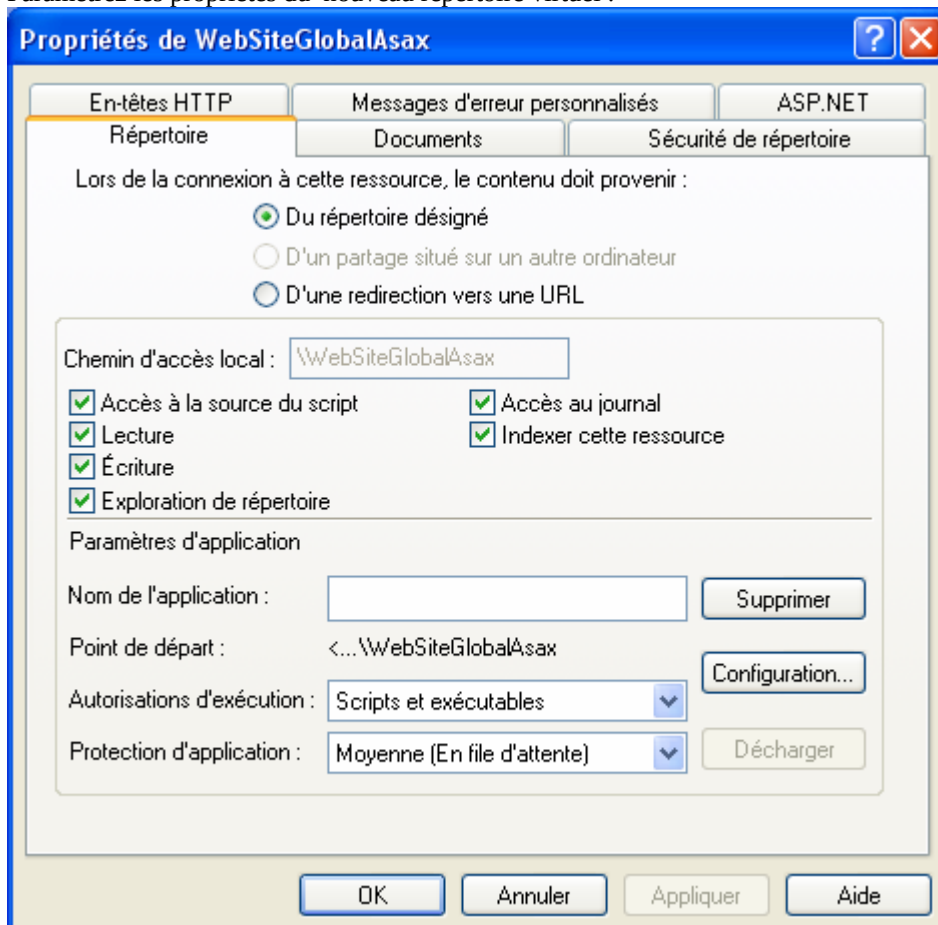
```

Serveur internet IIS

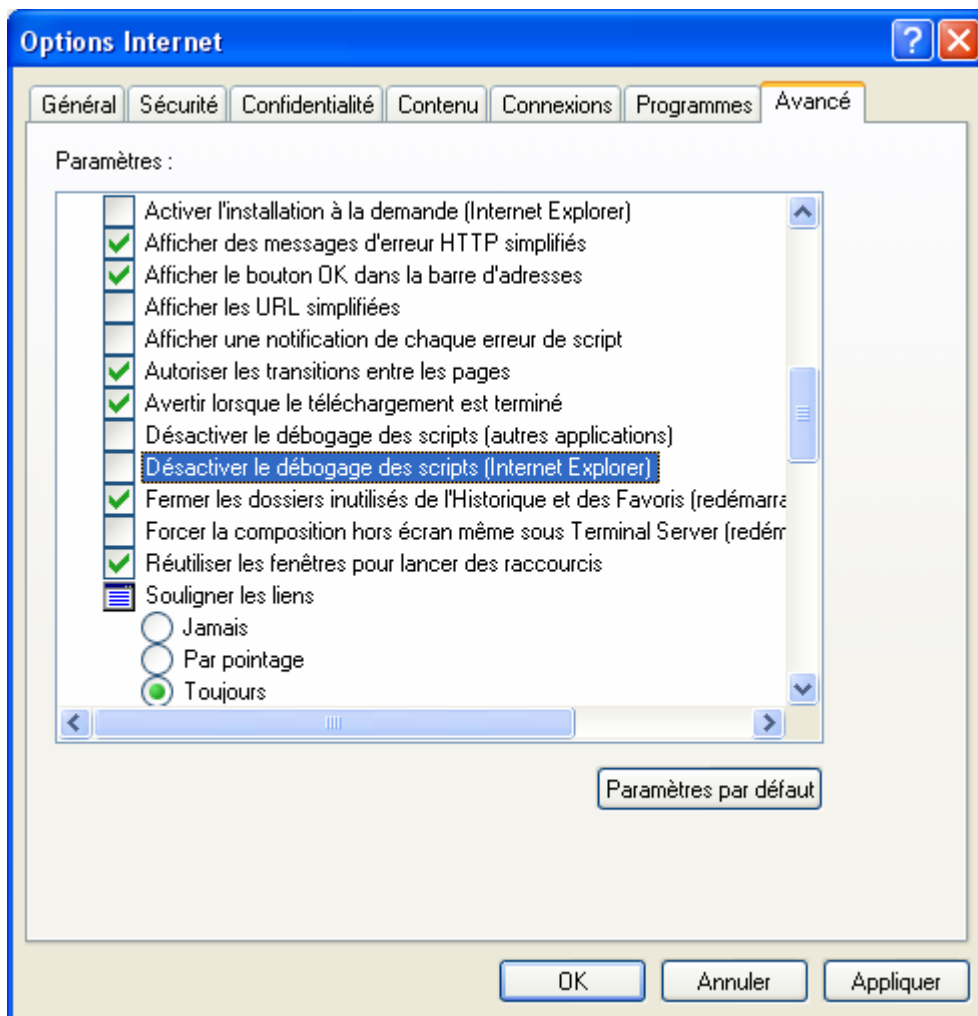
Pour pouvoir accéder aux données créer un nouveau répertoire virtuel :



Paramétrez les propriétés du nouveau répertoire virtuel :



Pour pouvoir debugger : dans le navigateur "Internet explorer IE5 et +" menu **outils\options internet**



Laisser le débogage activé si vous voulez déboguer vos script dans vos programmes ASP.Net.

Le débogage de Visual studio sur le C# fonctionne quant à lui normalement, seuls les scripts C# (ou autre langage) ont besoin du paramétrage précédent du navigateur Internet.

ASP.net etBD avec un GridView



Plan général:

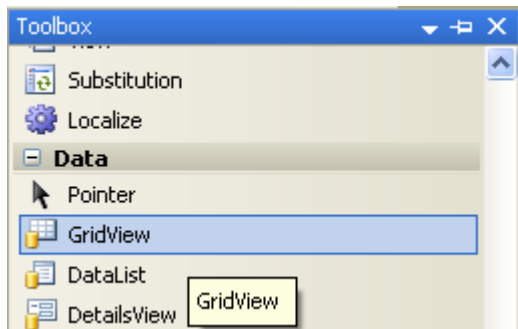
Comment utiliser un GridView pour accéder à une BD

- 1°) Accès à une BD SqlServer à travers les nouveaux composants .net
- 2°) Accès à des données hiérarchiques avec un XmlDataSource
- 3°) Exemple d'utilisation d'un cookie.

ASP.NET et BD avec GridView



Le contrôle **GridView** du namespace **System.Web.UI.WebControls** permet d'afficher, d'éditer, et d'effacer des données principalement à partir d'une base de données (Sql server, Oracle, Access, ...) et à travers le composant **SqlDataSource** :



Apparence visuelle du GridView sur la page de construction :

Column0	Column1	Column2
abc	abc	abc
abc	abc	abc
abc	abc	abc
abc	abc	abc
abc	abc	abc

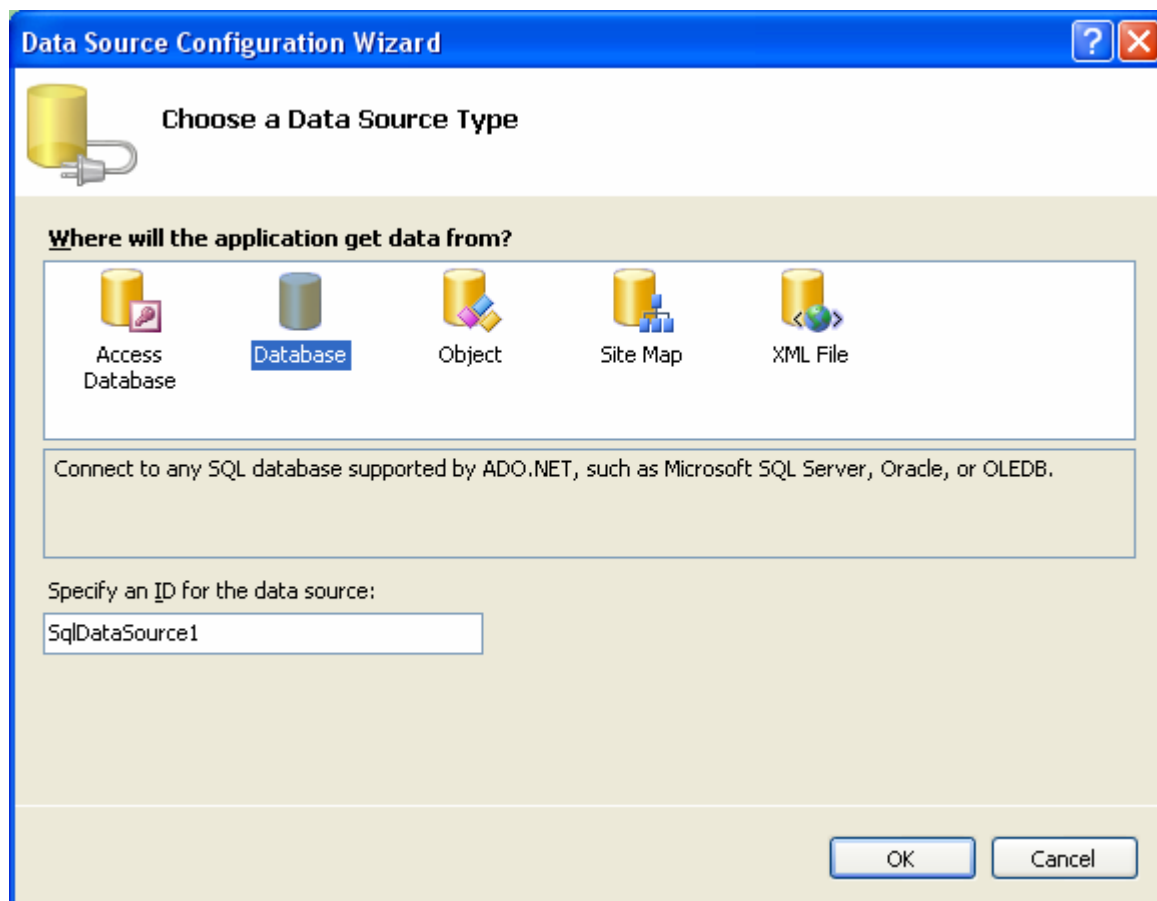
Demande de création d'une liaison avec une source de données :

Column0	Column1	Column2
abc	abc	abc
abc	abc	abc
abc	abc	abc
abc	abc	abc
abc	abc	abc

GridView Tasks
Auto Format...
Choose Data Source: (None)
Edit Columns...
Add New Column...
Edit Templates

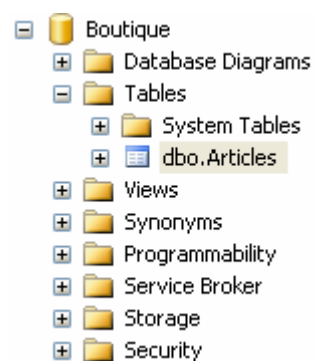
(None)
<New data source...>

Parmi les sources de données possibles, comme nous voulons visualiser une table extraite d'une base de données (Sql server, Oracle,...) il nous faut utiliser le composant **SqlDataSource** que l'on sélectionne par l'icône Database dans l'expert de configuration :

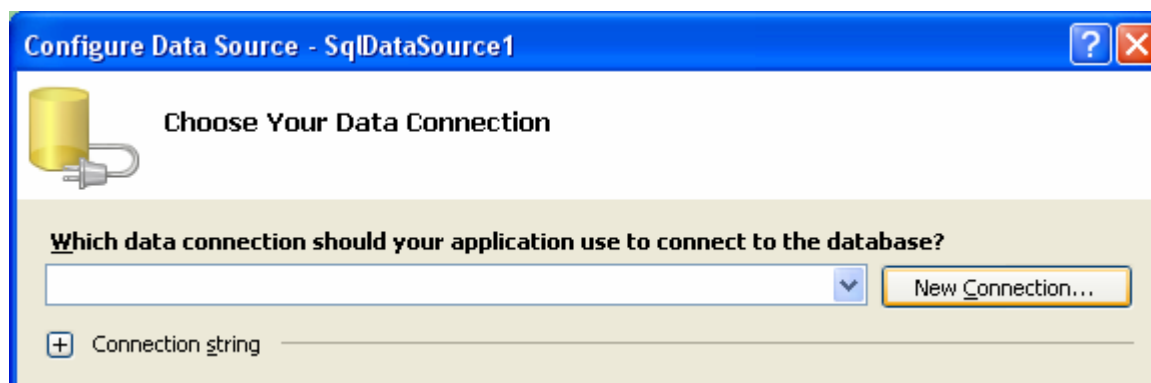


Ci-dessous la création de la base de données "[Boutique.mdf](#)" et de la table "[Articles](#)" avec Sql server management studio:

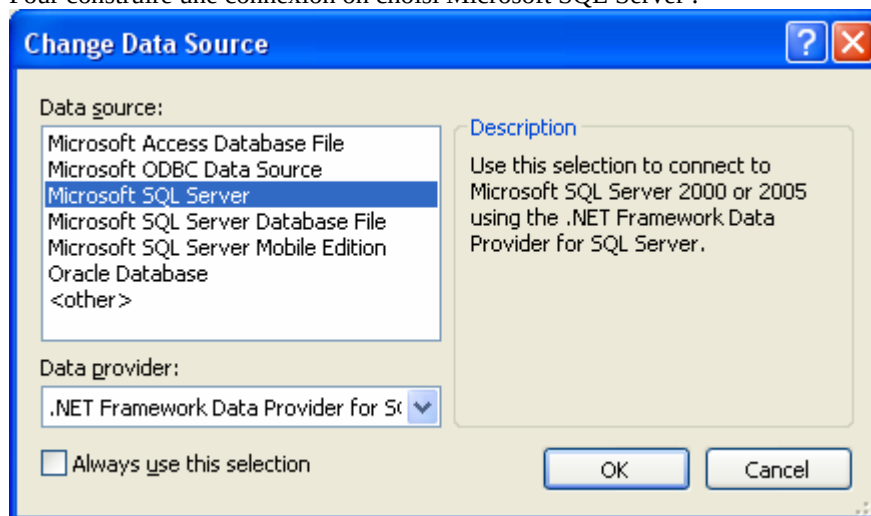
Table - dbo.Articles*			
Summary			
	Column Name	Data Type	Allow Nulls
	idArticle	int	☐
	genreArticle	nvarchar(50)	☒
	quantité	int	☒
	prix	money	☒
			☐



Les étapes de connexion de l'objet SqlDataSource à Sql Server 2005 :



Pour construire une connexion on choisi Microsoft SQL Server :



Il faut nommer le serveur ici (DUAL-CORE) et la BD (ici Boutique) :

Add Connection

Enter information to connect to the selected data source or click "Change" to choose a different data source and/or provider.

Data source:
 Microsoft SQL Server (SqlClient) Change...

Server name:
 DUAL-CORE Refresh

Log on to the server

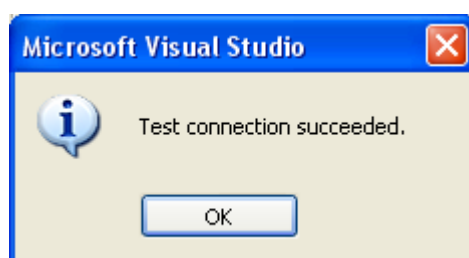
☒ Use Windows Authentication
☐ Use SQL Server Authentication

User name:
 Password:
☐ Save my password

Connect to a database

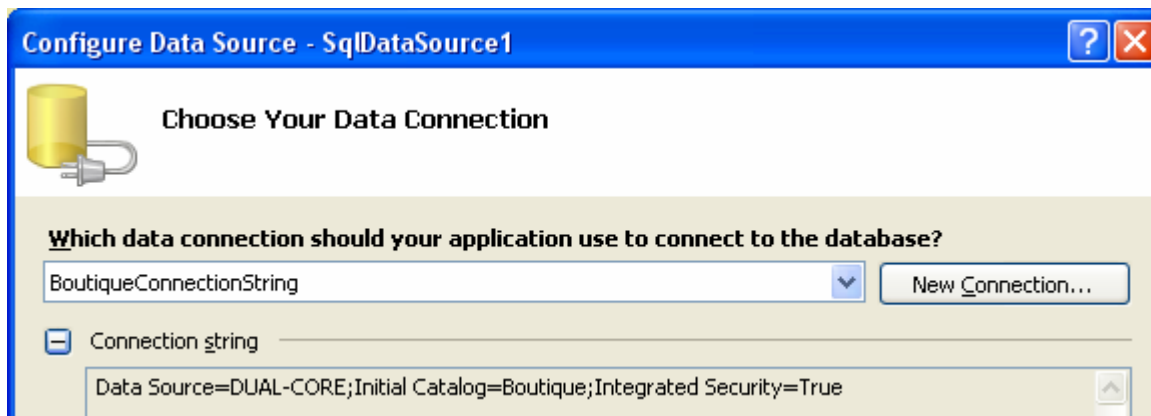
☒ Select or enter a database name:
 Boutique
☐ AdventureWorks
 AdventureWorksDW
 Boutique
 CCI_Store
 master
 model
 msdb
 ReportServer Advanced...

Test Connection OK Cancel



(pour mémoire la BD "Boutique.mdf" a été rangée lors de sa création, dans MSSQL.1\MSSQL\Data\)

Une string de connexion est créée à partir des renseignements qui viennent d'être fournis :

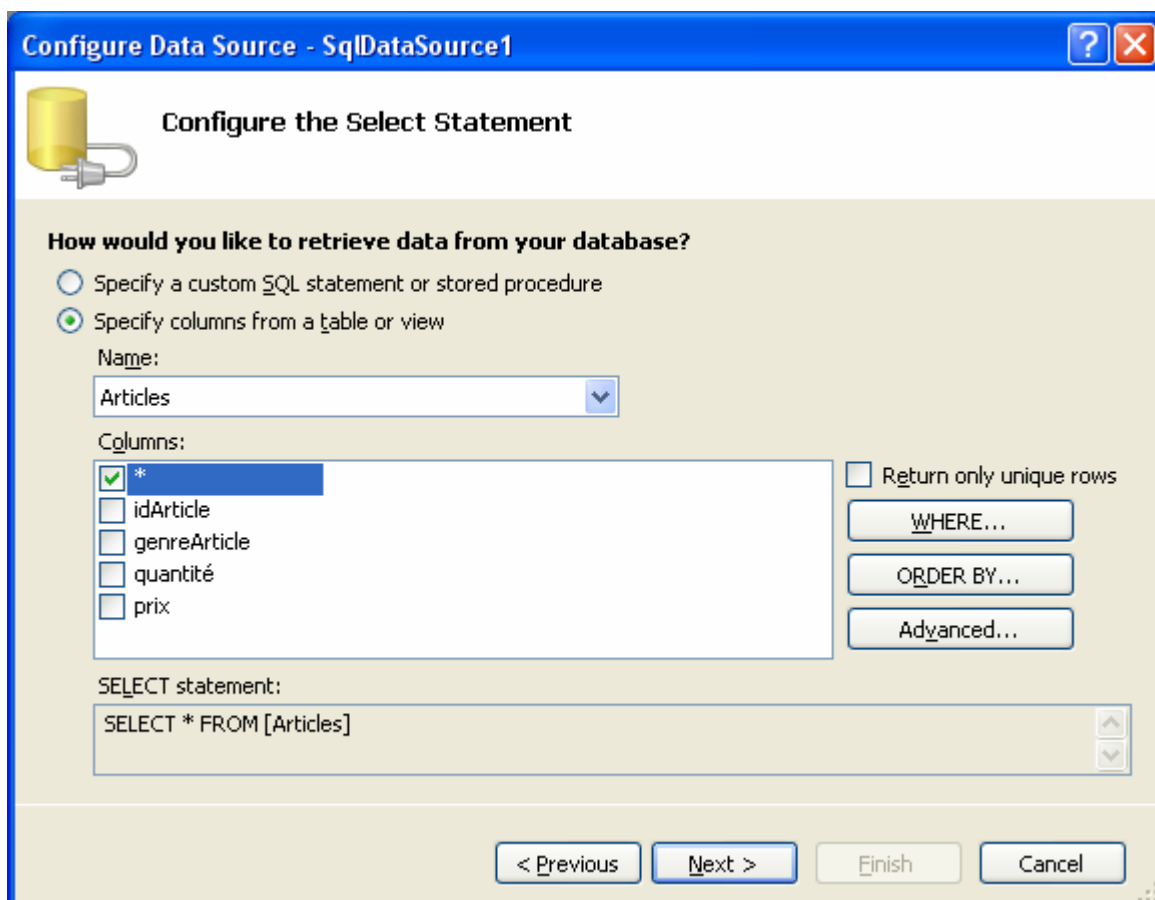


Cette chaîne de connexion est stockée dans le fichier **Web.config** sous le nom BoutiqueConnectionString

Dans Web.config :

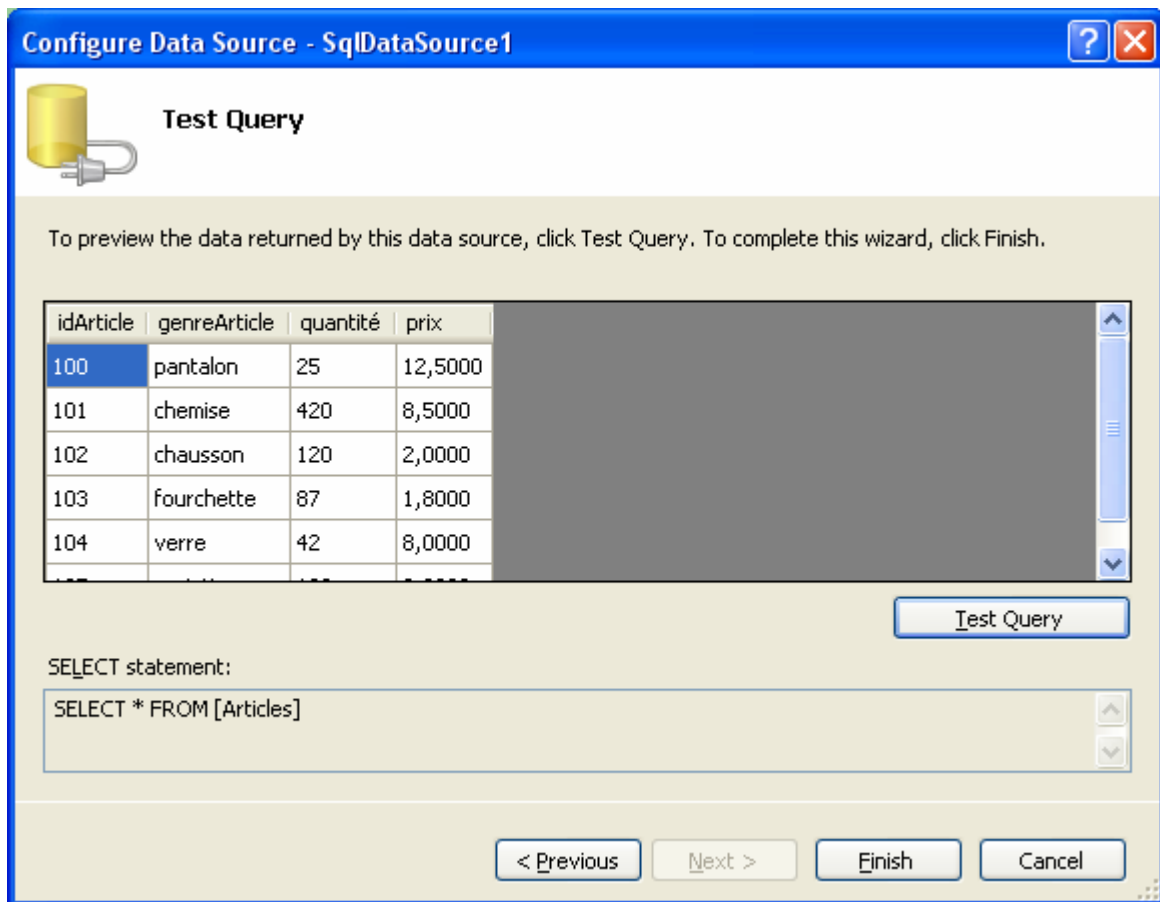
```
<connectionStrings>
  <add name="BoutiqueConnectionString"
        connectionString="Data Source=DUAL-CORE;
                          Initial Catalog=Boutique;
                          Integrated Security=True"
        providerName="System.Data.SqlClient" />
</connectionStrings>
```

L'expert de construction permet ensuite de configurer l'envoi de la commande SELECT à la BD :

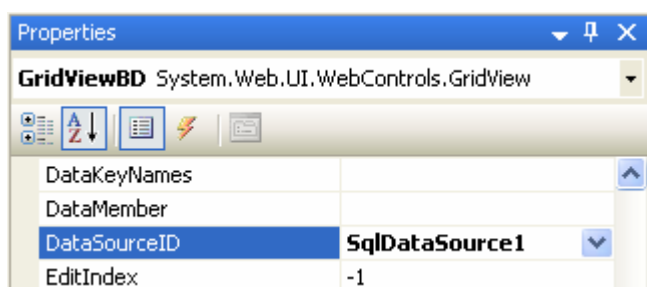


Ici cette commande est : `SELECT * FROM [Articles]`

L'expert de construction permet de visualiser à des fins de correction, le résultat de la commande choisie :



Le GridView est lié au SqlDataSource par sa propriété DataSourceID :



voici le code ASP.net engendré :

```
<asp:GridView ID="GridViewBD" runat="server"
AutoGenerateColumns="False"
    BackColor="LightGoldenrodYellow"
    BorderColor="Tan" BorderWidth="1px" CellPadding="2"
    DataSourceID="SqlDataSource1" ForeColor="Black"
GridLines="None" Width="371px" RowHeaderColumn="idArticle" ShowFooter="True">
    <FooterStyle BackColor="Tan" />
    <SelectedRowStyle BackColor="DarkSlateBlue"
ForeColor="GhostWhite" />
    <PagerStyle BackColor="PaleGoldenrod" ForeColor="DarkSlateBlue"
```

```

HorizontalAlign="Center" />
    <HeaderStyle BackColor="Tan" Font-Bold="True" />
    <AlternatingRowStyle BackColor="PaleGoldenrod" />
</asp:GridView>

<asp:SqlDataSource ID="SqlDataSource1" runat="server"
    ConnectionString="<%"$ ConnectionStrings:BoutiqueConnectionString
%>"
    ProviderName="<%"$
ConnectionStrings:BoutiqueConnectionString.ProviderName %>"
    SelectCommand="SELECT * FROM [Articles]">
</asp:SqlDataSource>

```

Si l'on sélectionne le bouton Advanced generation options dans l'expert de construction :

How would you like to retrieve data from your database?

☐ Specify a custom SQL statement or stored procedure

☒ Specify columns from a table or view

Name:

Articles

Columns:

- ☒ *
- ☐ idArticle
- ☐ genreArticle
- ☐ quantité
- ☐ prix

☐ Return only unique rows

WHERE...

ORDER BY...

Advanced...

on peut choisir de générer automatiquement les instructions INSERT, UPDATE et DELETE :

Advanced SQL Generation Options

Additional INSERT, UPDATE, and DELETE statements can be generated to update the data source.

☒ **Generate INSERT, UPDATE, and DELETE statements**

Generates INSERT, UPDATE, and DELETE statements based on your SELECT statement. You must have all primary key fields selected for this option to be enabled.

☒ **Use optimistic concurrency**

Modifies UPDATE and DELETE statements to detect whether the database has changed since the record was loaded into the DataSet. This helps prevent concurrency conflicts.

OK Cancel

voici le code ASP.net engendré :

```

<asp:SqlDataSource ID="SqlDataSource1" runat="server"
    ConnectionString="<%"$ ConnectionStrings:BoutiqueConnectionString
%>"
    ProviderName="<%"$
ConnectionStrings:BoutiqueConnectionString.ProviderName %>"
    SelectCommand="SELECT * FROM [Articles]"
    ConflictDetection="CompareAllValues"

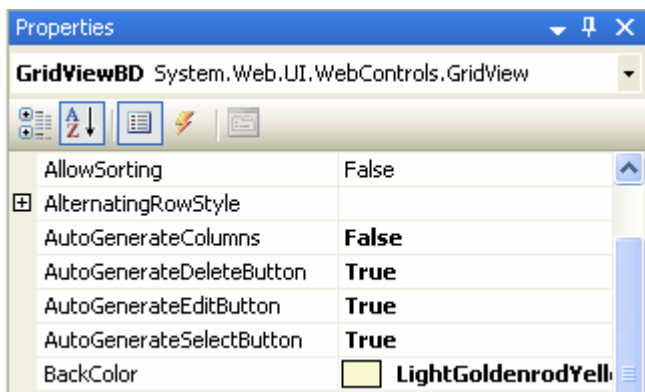
```

```

DeleteCommand="DELETE FROM [Articles] WHERE [idArticle] =
@original_idArticle AND [genreArticle] = @original_genreArticle AND
[quantité] = @original_quantité AND [prix] = @original_prix"
InsertCommand="INSERT INTO [Articles] ([idArticle],
[genreArticle], [quantité], [prix]) VALUES (@idArticle, @genreArticle,
@quantité, @prix)"
OldValuesParameterFormatString="original_{0}"
UpdateCommand="UPDATE [Articles] SET [genreArticle] =
@genreArticle, [quantité] = @quantité, [prix] = @prix WHERE [idArticle] =
@original_idArticle AND [genreArticle] = @original_genreArticle AND
[quantité] = @original_quantité AND [prix] = @original_prix">
<DeleteParameters>
  <asp:Parameter Name="original_idArticle" Type="Int32" />
  <asp:Parameter Name="original_genreArticle" Type="String" />
  <asp:Parameter Name="original_quantité" Type="Int32" />
  <asp:Parameter Name="original_prix" Type="Decimal" />
</DeleteParameters>
<UpdateParameters>
  <asp:Parameter Name="genreArticle" Type="String" />
  <asp:Parameter Name="quantité" Type="Int32" />
  <asp:Parameter Name="prix" Type="Decimal" />
  <asp:Parameter Name="original_idArticle" Type="Int32" />
  <asp:Parameter Name="original_genreArticle" Type="String" />
  <asp:Parameter Name="original_quantité" Type="Int32" />
  <asp:Parameter Name="original_prix" Type="Decimal" />
</UpdateParameters>
<InsertParameters>
  <asp:Parameter Name="idArticle" Type="Int32" />
  <asp:Parameter Name="genreArticle" Type="String" />
  <asp:Parameter Name="quantité" Type="Int32" />
  <asp:Parameter Name="prix" Type="Decimal" />
</InsertParameters>
</asp:SqlDataSource>

```

Paramétrer les boutons d'édition, de suppression et de sélection dans le GridView :



	idArticle	genreArticle	quantité	prix
Edit Delete Select	0	abc	0	0
Edit Delete Select	1	abc	1	0,1
Edit Delete Select	2	abc	2	0,2
Edit Delete Select	3	abc	3	0,3
Edit Delete Select	4	abc	4	0,4

SqlDataSource - SqlDataSource1

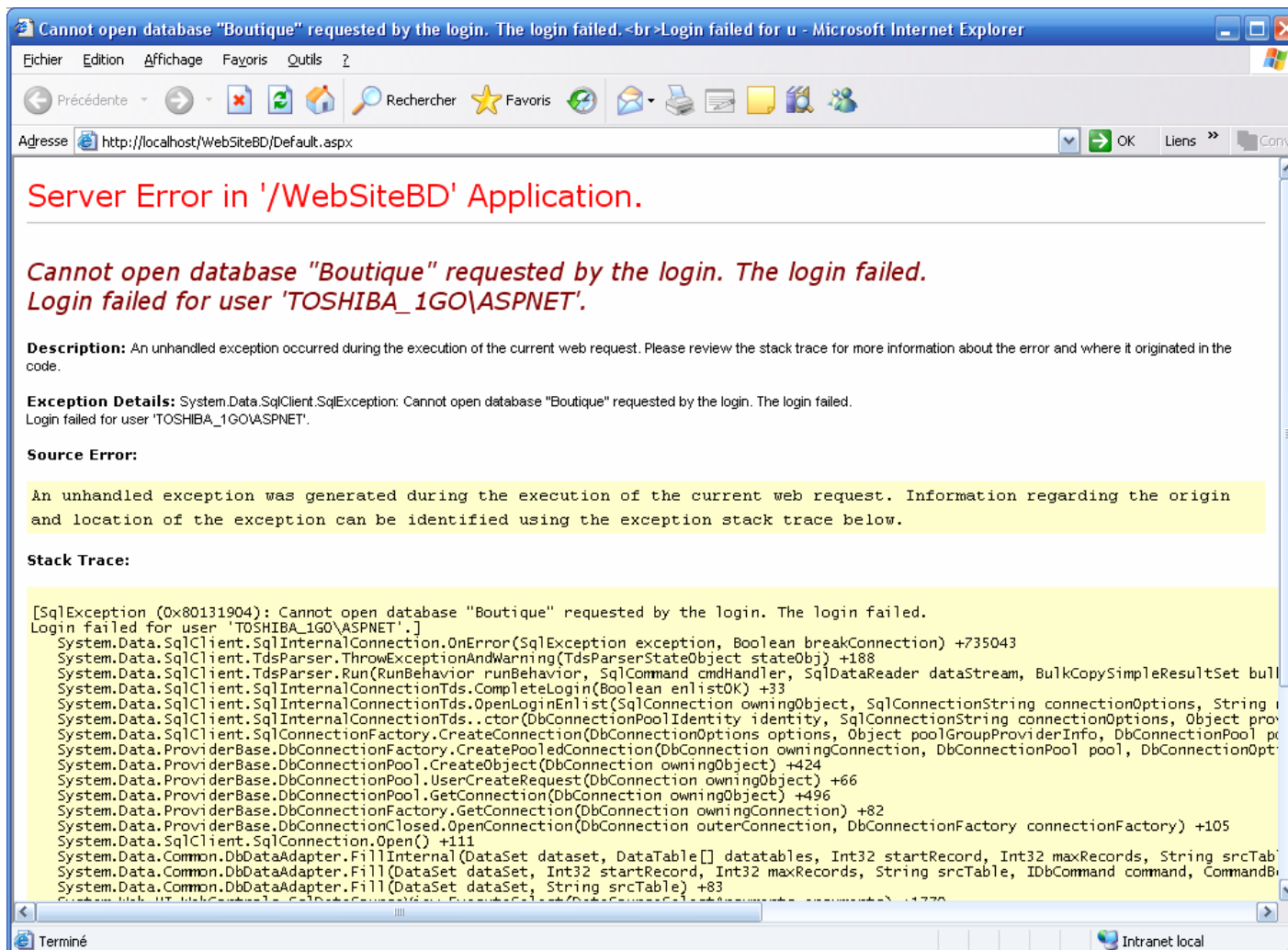
```

    <asp:GridView ID="GridViewBD" runat="server"
AutoGenerateColumns="False"
    BackColor="LightGoldenrodYellow"
    BorderColor="Tan" BorderWidth="1px" CellPadding="2"
    DataSourceID="SqlDataSource1" ForeColor="Black" GridLines="None"
    Width="402px" RowHeaderColumn="idArticle" ShowFooter="True"
    AutoGenerateDeleteButton="True"
    AutoGenerateEditButton="True"
    AutoGenerateSelectButton="True">
        . . .
    </asp:GridView>

```

Si le serveur SQL n'a pas les permissions adéquates pour ASP.NET

Voici le genre de message envoyé dans le navigateur (ici le serveur SQL se nomme **TOSHIBA_1GO**) :

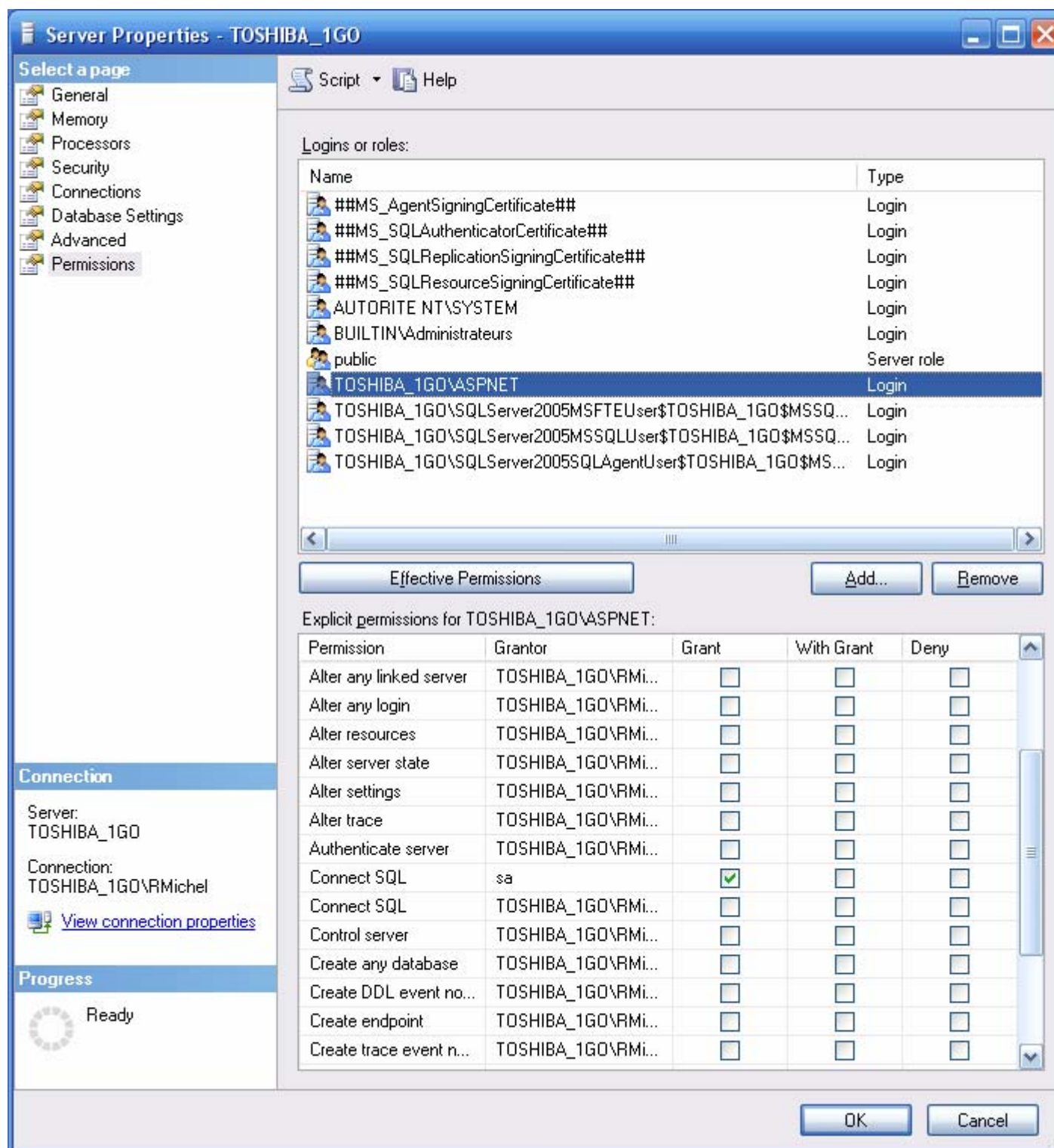


Par défaut nous allons donner au serveur AspNet (dénommé : TOSHIBA_1GO\ASPNET) toutes les permissions pour pouvoir utiliser la BD avec Asp.Net :

- 1°) sélectionner propriétés de la BD (ici Boutique.mdf)
- 2°) sélectionner page Permissions, puis le lien à cliquer est : [View server permissions](#).

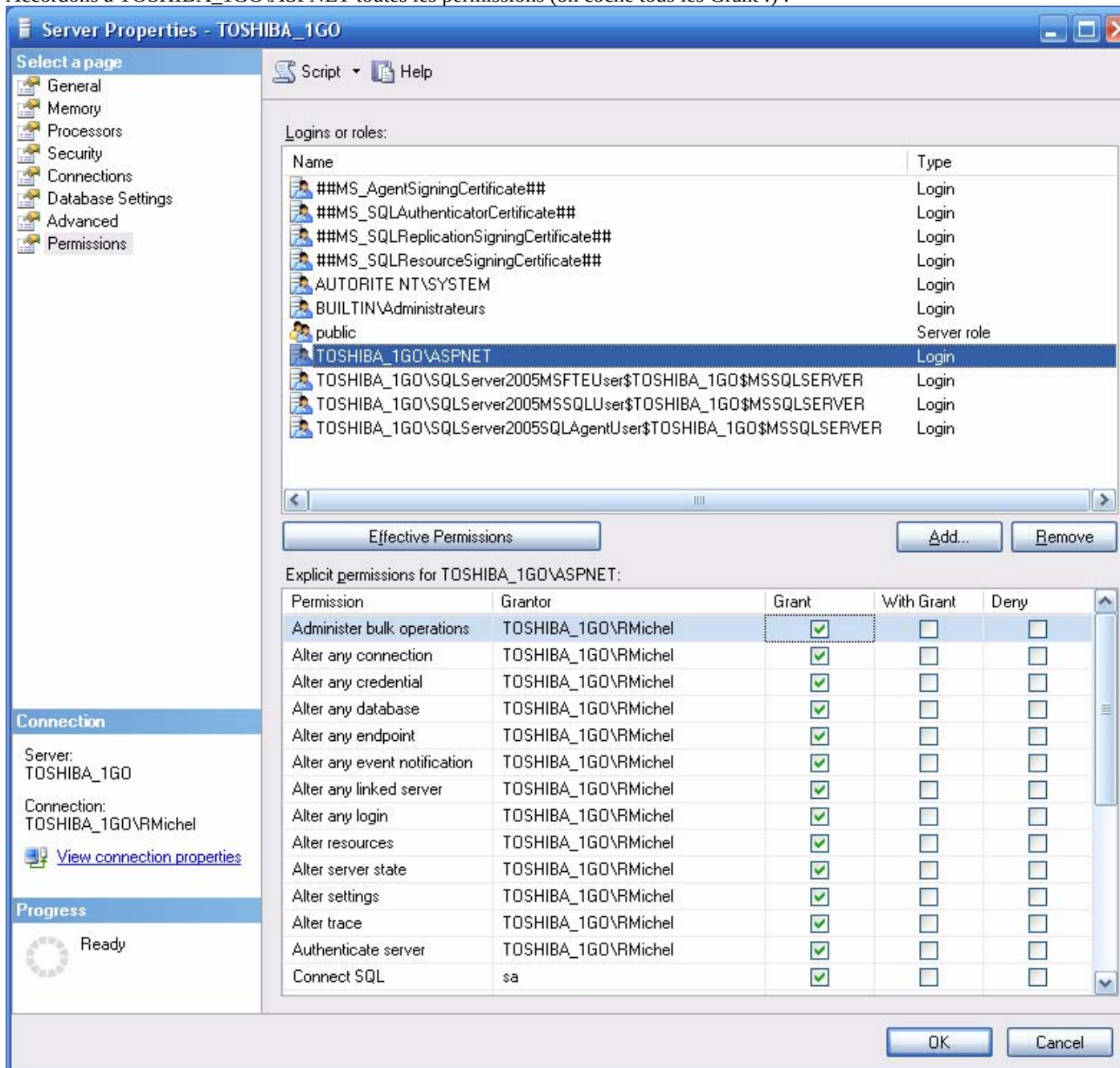


Les informations de permissions des serveurs sont affichées :



On note que par défaut, le serveur ASPNET ici nommé TOSHIBA_1GO\ASPNET n'a pour l'instant qu'une seule permission accordée **Connect SQL (Grant)**.

Accordons à TOSHIBA_1GO\ASPNET toutes les permissions (on coche tous les Grant !) :



En relançant le programme Asp.Net, le navigateur affiche maintenant :

Boutique

Test de GridView et BD Sql Server 2005 avec ASP.net 2.0

	idArticle	genreArticle	quantité	prix
Edit Delete Select	100	cahier	106	1,2000
Edit Delete Select	101	fourchette	57	0,9000
Edit Delete Select	102	cahier	120	2,0000
Edit Delete Select	103	serviette	28	1,4000
Edit Delete Select	104	verre	42	8,0000
Edit Delete Select	105	assiette	108	2,0000
Edit Delete Select	106	chemise	420	8,5000
Edit Delete Select	107	couteau	83	1,1000
Edit Delete Select	108	pantalon	27	45,0000
Edit Delete Select	109	ski	25	12,5000
Edit Delete Select	110	craie	1249	0,0400
Edit Delete Select	111	xcxxc	1000	100,0000

Rendre Delete actif

ligne sélectionnée = 9

Select

...

Autre vue de la même BD dans la page web :

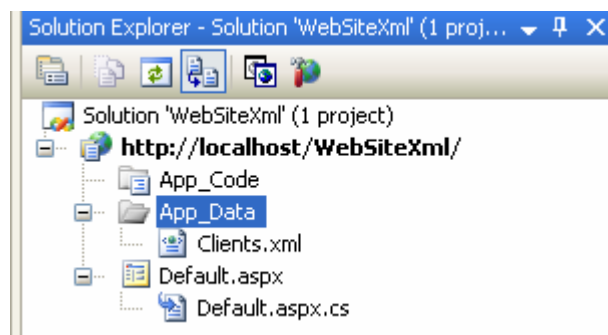
	idArticle	genreArticle	quantité	prix
Edit Select	100	cahier	106	1,2000
Edit Select	101	fourchette	57	0,9000
Edit Select	102	cahier	120	2,0000
Edit Select	103	serviette	28	1,4000
Edit Select	104	verre	42	8,0000
Edit Select	105	assiette	108	2,0000

Des données hiérarchiques au format XML avec un XmlDataSource

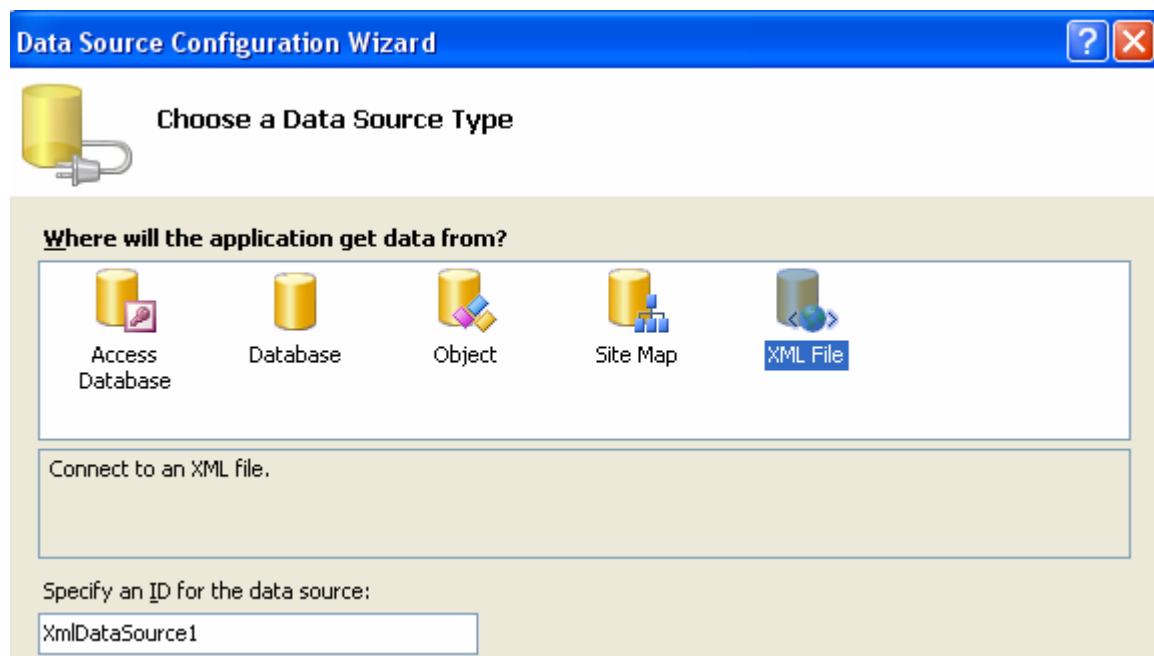
Créer (ou utiliser) un fichier xml au format ci-dessous :

```
Start Page App_Data/Clients.xml Default.aspx.cs* Default.aspx
<?xml version="1.0" encoding="utf-8" ?>
<tableClient>
  <tclient idClient="1001" Nom="duval" Article="brosse" Montant="60.5"/>
  <tclient idClient="1002" Nom="dupont" Article="savon" Montant="140"/>
  <tclient idClient="1003" Nom="durand" Article="gant" Montant="24.8"/>
  <tclient idClient="1004" Nom="duchenaud" Article="serviette" Montant="35.7"/>
  <tclient idClient="1005" Nom="dumartin" Article="tapis" Montant="46"/>
</tableClient>
```

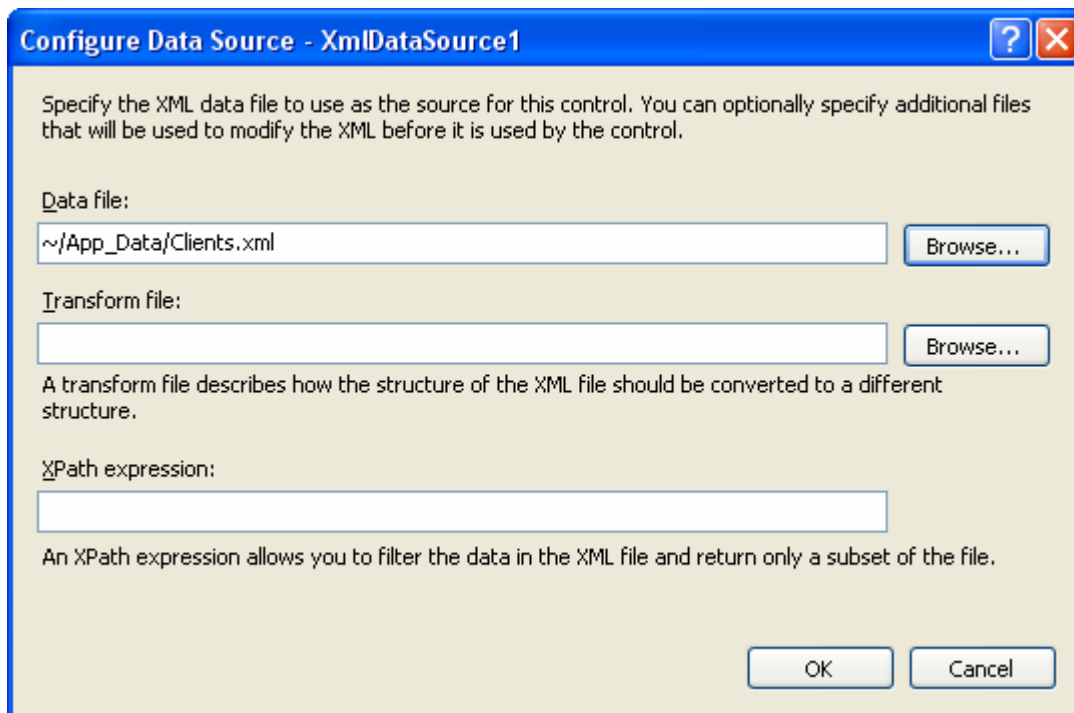
Ajouter ce fichier nommé par exemple "Clients.xml", au dossier **App_Data** :



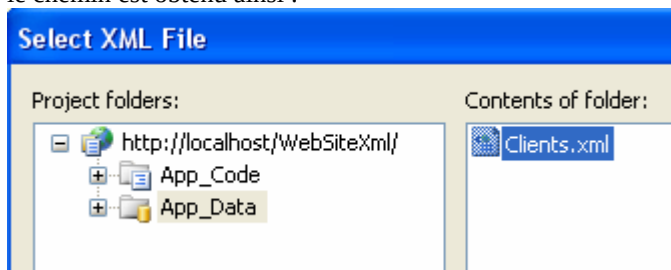
Ajouter un composant XmlDataSource à partir de la ToolBox :



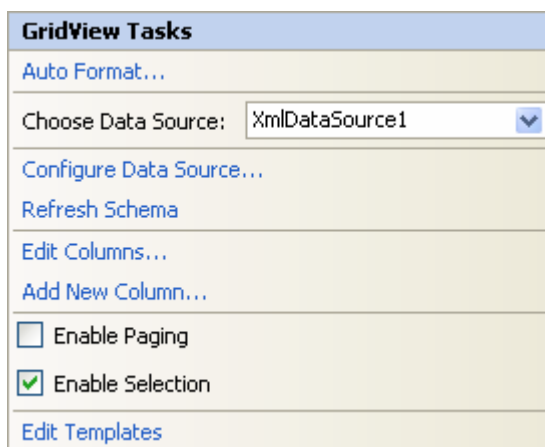
Configurer le XmlDataSource à partir du fichier Clients.xml :



le chemin est obtenu ainsi :



Il faut lier le GridView au XmlDataSource :



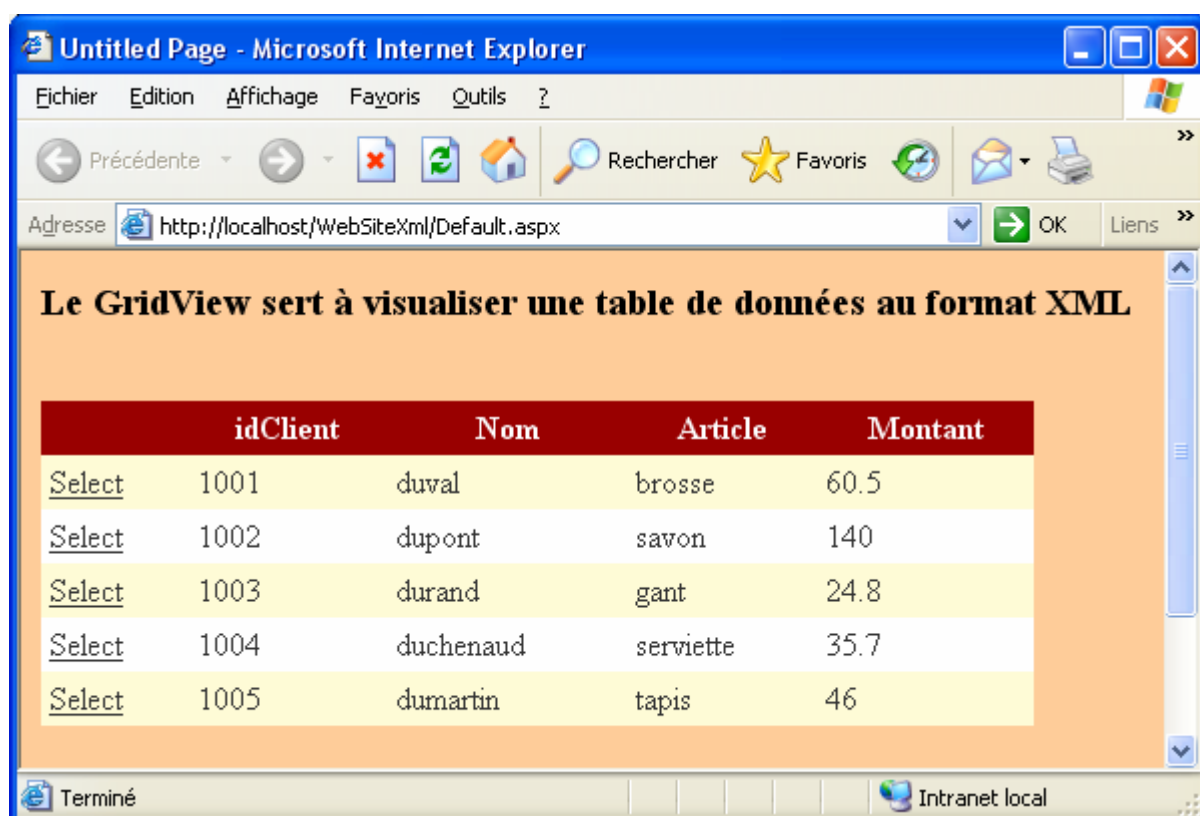
On obtient dès lors en mode construction l'affichage tabulaire du fichier Clients.xml :

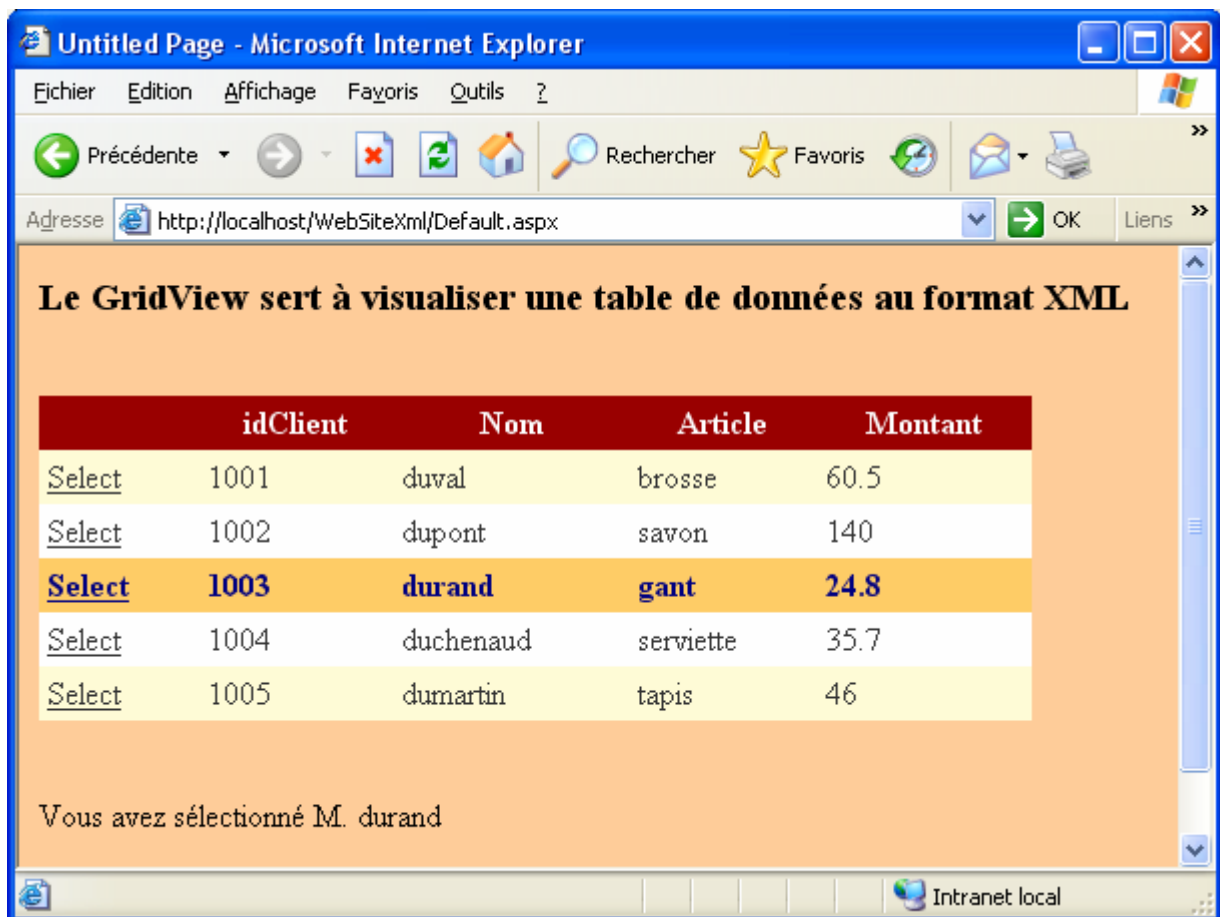
Start Page	App_Data/Clients.xml	Default.aspx.cs*	Default.aspx
------------	----------------------	------------------	--------------

Le GridView sert à visualiser une table de données au format XML

	idClient	Nom	Article	Montant
Select	1001	duval	brosse	60.5
Select	1002	dupont	savon	140
Select	1003	durand	gant	24.8
Select	1004	duchenaud	serviette	35.7
Select	1005	dumartin	tapis	46

En exécutant le programme voici ce qui apparaît dans le navigateur utilisateur :





```
using System;
using System.Data;
using System.Configuration;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Web.UI.HtmlControls;

public partial class _Default : System.Web.UI.Page
{
    private int SelectedIndex;

    protected void Page_Load(object sender, EventArgs e)
    {
        if (IsPostBack)
        {
            SelectedIndex = (int)Page.Session["SelectedIndex"];
        }
        else
        {
            Page.Session["SelectedIndex"] = -1;
        }
    }

    protected void GridView1_SelectedIndexChanged(object sender, EventArgs e)
    {
        Page.Session["SelectedIndex"] = GridView1.SelectedIndex;
        LabelNom.Text = GridView1.SelectedRow.Cells[2].Text;
    }
}
```

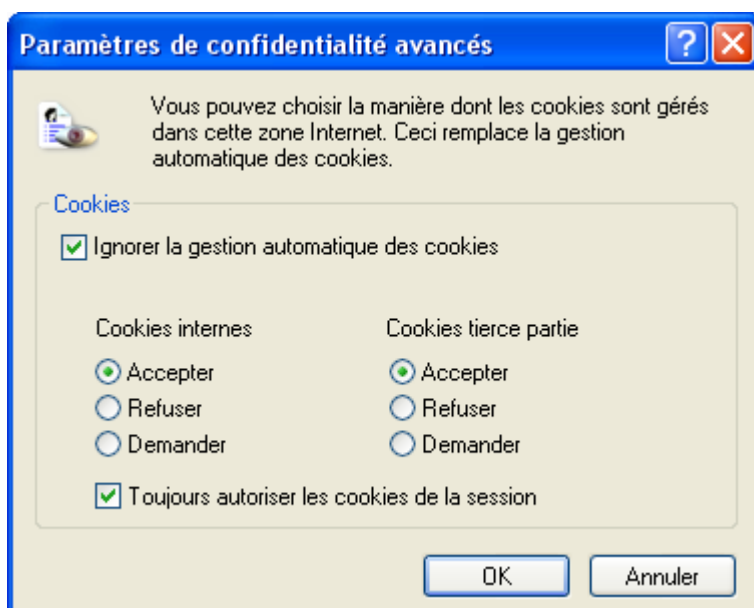

ASP.NET et Cookie

C#.net

Les principes de sécurité d'Internet interdisent à tout programme d'accéder directement aux disques du client pour y stocker des fichiers ou bien même les lire, les seuls fichiers autorisés s'appellent des cookies.

Un **cookie** est obligatoirement un fichier texte ASCII dont la taille ne dépasse pas les 4 Ko et à durée de vie limitée, il est le seul type de fichier qu'une application web peut écrire et lire sur le disque dur de l'utilisateur, si l'utilisateur le permet par paramétrage de son navigateur.

Exemple avec Internet explorer (l'utilisateur paramètre pour accepter les cookies) :



Dans cette éventualité, nous montrons une page web de compte bancaire qui récolte les données entrées (nom, prénom, caractéristiques du client) et qui les stocke dans un "cookie".

Soit une connexion à notre site web d'ouverture d'un compte en vue d'un prêt pour habitation de Mr Durand dalle âgé de 51 ans ingénieur de profession et pourvu d'un salaire mensuel de 5000 euros, le site web lui envoie sous forme d'une page html le formulaire de saisie ci-dessous :

Flouze Bank - service des prêts - Microsoft Internet Explorer

Fichier Edition Affichage Favoris Outils ?

Précédente Recherche Favoris OK Liens »

Adresse <http://localhost/WebsiteCookies/Default.aspx>

Prêts habitation sur du long terme à des taux très bas

(les infos clients sont mémorisées dans un Cookie et utilisées lors du chargement)

Renseignements personnels :

Nom : Prénom :

Age	Profession	Salaire mensuel actuel
mineur (inférieur à 18 ans)	Avocat	☐ Moins de 1000\$
entre 18 et 25 ans	Ingénieur	☐ Entre 1000\$ et 3000\$
entre 25 et 50 ans	Enseignant	☒ Plus de 3000\$
entre 50 et 65 ans	Président	
plus de 65 ans	CPE	
	Ministre	

Terminé Intranet local

Après avoir saisi les renseignements demandés, Mr Durand dalle clique sur le bouton "envoyer vos informations" et le site web de la banque lui renvoie une nouvelle page html avec la réponse à sa demande :

Ci-après l'analyse de votre demande :

Votre nom : **Durand**

Votre prénom : **dalle**

Votre profession : **Ingénieur**

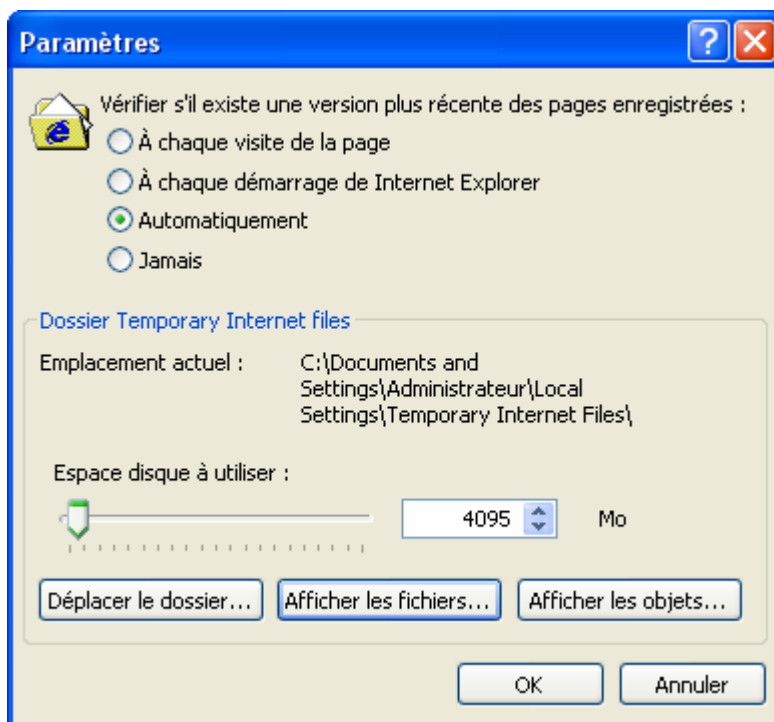
Salaire déclaré : **plus de 3000\$**

Notre réponse : plus de 3000\$!!, satisfaisant

Cher Mr. Durand, nous sommes heureux de pouvoir disposer de votre argent, notre banque accueille très favorablement dans ses comptes tous les Ingénieurs tels que vous.

A cet instant le site web a écrit et déposé sur le disque dur de Mr Durand un fichier texte (cookie) contenant les informations sur les 4 champs précédents (nom, prénom,...).

Par défaut les cookies sont automatiquement stockés dans le dossier "Temporary Internet Files" de la machine utilisateur sous son profil (dans l'exemple ci-dessous l'utilisateur se nomme "Administrateur") :



Localisation du cookie sur la machine

C:\Documents and Settings\Administrateur\Local Settings\Temporary Internet Files :

Nom	Adresse Internet	Type	Taille	Date d'expiration	Dernière modificatio,
 Cookie:administrateur@localhost/	Cookie:administrateur@localhost/	Document texte	1 Ko	29/08/2006 08:25	28/08/2006 08:25
 Default.aspx	http://localhost/WebSiteCookies...	HTML Document	6 Ko	Aucun	Aucun
 ShowMe.bmp	ms-itss:C:\PROGRA~1\ARCHIT~...	Image bitmap	2 Ko	Aucun	Aucun
 copycode.gif	ms-itss:C:\Program Files\MSDN\...	Image GIF	1 Ko	Aucun	Aucun
 Footer.gif	ms-itss:C:\Program Files\MSDN\...	Image GIF	1 Ko	Aucun	Aucun
 copycodeHighlight.gif	ms-itss:C:\Program Files\MSDN\...	Image GIF	1 Ko	Aucun	Aucun
 drpdwn.gif	ms-itss:C:\Program Files\MSDN\...	Image GIF	1 Ko	Aucun	Aucun
 drpdwn_orange.gif	ms-itss:C:\Program Files\MSDN\...	Image GIF	1 Ko	Aucun	Aucun
 script.js	ms-itss:C:\Program Files\MSDN\...	JScript Script ...	25 Ko	Aucun	Aucun

Dans l'exemple, le fichier a été déposé le 28.08.2006 à 8h 25 sur la machine client, sa durée de vie est limitée à un jour, il expire le 29.08.2006 à 8h 25, ce qui veut dire qu'il sera effacé par le système à cette heure là le 29.08.2006.

Un double permanent (sauf si l'utilisateur décide de l'effacer) du cookie, se trouve dans le dossier Cookies sous le profil de l'utilisateur :



Contenu du fichier cookie sur la machine

```

identite
nom=Durand&prenom=dalle&profession=Ingénieur&salaire=plus de 3000$
localhost/
1024
4099591552
29805363
3395668048
29805162
*
```

Mr Durand coupe sa connexion au site de la banque et l'abandonne le cookie rste présent sur son disque. Si Mr Durand revient plus tard à 8h 38 sur le site de cette même banque, alors le programme web de la banque peut aller consulter le cookie présent sur son disque dur et récupérer les informations déjà entrées afin par exemple de les afficher dans la page HTML du navigateur de Mr Durand pour une éventuelle modification :

Nom : Prénom :

Age	Profession	Salaire mensuel actuel
mineur (inférieur à 18 ans)	Avocat	☐ Moins de 1000\$
entre 18 et 25 ans	Ingénieur	☒ Entre 1000\$ et 3000\$
entre 25 et 50 ans	Enseignant	☐ Plus de 3000\$
entre 50 et 65 ans	Président	
plus de 65 ans	CPE	
	Ministre	

Ci-après l'analyse de votre demande :

Votre nom : **Durand**

Votre prénom : **dalle**

Votre profession : **Ingénieur**

Salaire déclaré : **plus de 3000\$**

Dans C:\Documents and Settings\Administrateur\Local Settings\Temporary Internet Files :

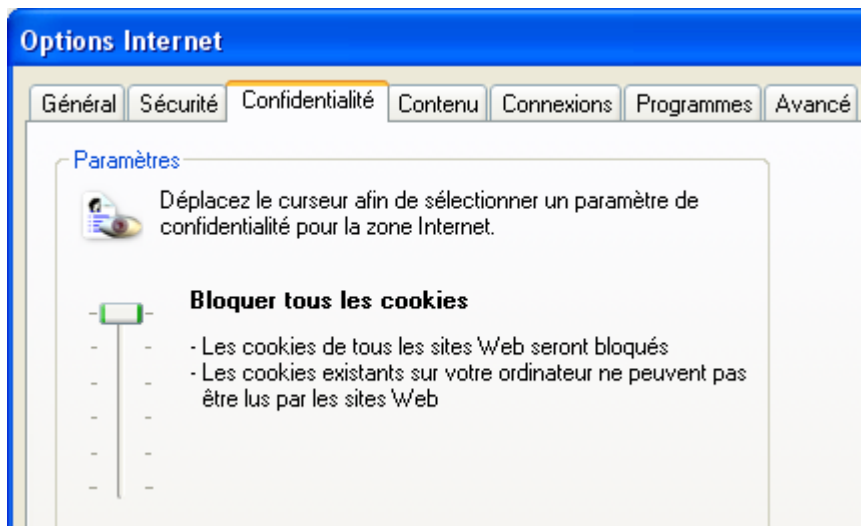
Nom	Date d'expiration	Dernière modificatio...	Dernier accès le
Default.aspx	Aucun	Aucun	28/08/2006 08:38
Cookie:administrateur@localhost/	29/08/2006 08:25	28/08/2006 08:25	28/08/2006 08:38
ShowMe.bmp	Aucun	Aucun	18/08/2006 20:36
copycode.gif	Aucun	Aucun	08/08/2006 08:57
footer.gif	Aucun	Aucun	08/08/2006 08:57
copycodeHighlight.gif	Aucun	Aucun	08/08/2006 08:57
drpdown.gif	Aucun	Aucun	08/08/2006 08:57
drpdown_orange.gif	Aucun	Aucun	08/08/2006 08:57
script.js	Aucun	Aucun	08/08/2006 08:57

Le système a bien marqué le cookie comme ayant été accédé la dernière fois à 8h 38.

Attention

Une application web ne doit pas être dépendante des cookies, car l'utilisateur peut refuser les cookies et l'application risque de ne pas fonctionner correctement :

Exemple avec Internet explorer de blocage de tous les cookies :



Comment programmer et utiliser le cookie précédent

Code Asp.Net de la page [Default.aspx](#) construite :

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default.aspx.cs" Inherits="_Default" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" >
<head runat="server">
  <title>Flouze Bank - service des prêts</title>
</head>
<body bgcolor="#cfffcc">
  <form id="form1" runat="server">
    <div>
      <h1>
<asp:listbox id="Select2" style="Z-INDEX: 105; LEFT: 51px; POSITION: absolute; TOP: 349px"
runat="server" Width="162px" Height="102px">
<asp:ListItem Value="0">mineur (inf&#233;rieur &#226; 18 ans)</asp:ListItem>
<asp:ListItem Value="1" Selected="True">entre 18 et 25 ans</asp:ListItem>
<asp:ListItem Value="2">entre 25 et 50 ans</asp:ListItem>
<asp:ListItem Value="3">entre 50 et 65 ans</asp:ListItem>
<asp:ListItem Value="4">plus de 65 ans</asp:ListItem>
</asp:listbox>
<asp:textbox id="TextSalaire" style="Z-INDEX: 112; LEFT: 126px; POSITION: absolute; TOP: 704px"
runat="server" Width="101px" Height="18px" BackColor="MediumSpringGreen" ReadOnly="True"
BorderWidth="0px" ForeColor="#804040">
</asp:textbox>
<asp:textbox id="TextProfession" style="Z-INDEX: 111; LEFT: 138px; POSITION: absolute; TOP: 664px"
runat="server" Width="154px" Height="19px" BackColor="MediumSpringGreen" ReadOnly="True"
BorderWidth="0px" ForeColor="#804040">
</asp:textbox>
<asp:textbox id="TextPrenom" style="Z-INDEX: 110; LEFT: 121px; POSITION: absolute; TOP: 626px"
runat="server" Width="166px" Height="20px" BackColor="MediumSpringGreen" ReadOnly="True"
BorderWidth="0px" ForeColor="#804040">
</asp:textbox>
<asp:textbox id="TextNom" style="Z-INDEX: 109; LEFT: 97px; POSITION: absolute; TOP: 589px"
runat="server" Width="168px" Height="20px" BackColor="MediumSpringGreen" ReadOnly="True"
BorderWidth="0px" ForeColor="#804040" Font-Bold="True">
```



```
<br>
<br>
<br>
<br>
<br>
<br>
<br>
<br>
<br>
<br>
<br>
<br>
<br>
<br>
<br>
<br>
<h2 id="H11">&nbsp;</h2>
&nbsp;
<h2>Ci-après l'analyse de votre demande :</h2>
    <b><font color="blue">Votre nom :</font></b>
        <br>
        <br>
    <b><font color="blue">Votre prénom :</font></b>
        <br>
        <br>
    <b><font color="blue">Votre profession :</font></b>
        <br>
        <br>
    <b><font color="blue">Salaire déclaré :</font></b>
        <br>
        <br>
    <i>Notre réponse</i> :
    <%= texteRep %>

</div>
</form>
</body>
</html>
```

C'est un objet de la classe `HttpCookie` que nous nommons `CookId`, qui prend en charge dans `.Net` la gestion des cookies, syntaxiquement un tel objet se programme comme un objet `Session` déjà vu à travers un adressage indirect.

Lorsque l'utilisateur Mr Durand dalle clique sur le bouton "envoyer vos informations", le serveur exécute le code C# ci-après qui permet de créer le fichier de cookie sur la machine de Mr Durand :

```
//-- Persistance des données : (un seul cookie pour un site et pas plus de 4Ko de données)
string Reponse = "";
.....
HttpCookie CookId = new HttpCookie("identite");
CookId["nom"] = VotreNom; // création du champ "nom" dans le cookie avec sa donnée
CookId["prenom"] = VotrePreNom; // création du champ "prenom" dans le cookie avec sa donnée
CookId["profession"] = votreprofession; // création du champ "profession" dans le cookie avec sa donnée
CookId["salaire"] = salaire; // création du champ "salaire" dans le cookie avec sa donnée
this.Session["Reponse"] = Reponse;
CookId.Expires = DateTime.Now.AddDays(1); // date d'expiration réglée sur jour actuel +1

//-- validation et stockage :
bool isChampValid = VotreNom.Length * VotrePreNom.Length != 0;
if (isChampValid)//si tous les champs sont remplis
    this.Response.Cookies.Add(CookId);// on envoie la réponse http au client : ajoutez le cookie
```



Lors du chargement de la page sur le serveur, le programme teste la présence du cookie sur la machine web, s'il n'y a pas de cookie présentement associé à l'application, le serveur ne renseigne que le champ TextNom avec le texte "nouveau client" :

Affichage dans le navigateur client :

The screenshot shows a web application interface. At the top, there is a button labeled "Envoyer vos informations". Below it, the text "Ci-après l'analyse de votre demande :". Then, there are four labels with corresponding input fields: "Votre nom : nouveau client", "Votre prénom :", "Votre profession :", and "Salaire déclaré :". The input fields are represented by redacted areas.

Si lors du chargement de la page sur le serveur, le programme teste la présence du cookie sur la machine web et que celui-ci existe, le serveur renseigne tous les champs TextNom, TextPrenom, TextProfession, TextSalaire à partir des données lues dans le cookie :

```
protected void Page_Load(object sender, System.EventArgs e)
{
    HttpCookie CookId = this.Request.Cookies["identite"];
    if (CookId != null)
    {
        TextNom.Text = this.Request.Cookies["identite"]["nom"];
        TextPrenom.Text = this.Request.Cookies["identite"]["prenom"];
        TextProfession.Text = this.Request.Cookies["identite"]["profession"];
        TextSalaire.Text = this.Request.Cookies["identite"]["salaire"];
        texteRep = Convert.ToString(this.Session["Reponse"]);
    }
    else TextNom.Text = "nouveau client";
}
```

Affichage dans le navigateur client :



Envoyer vos informations

Ci-après l'analyse de votre demande :

Votre nom : Durand

Votre prénom : dalle

Votre profession : Ingénieur

Salaire déclaré : plus de 3000\$

Renseignements obtenus à partir du cookie

Code C# de la page [Default.aspx.cs](#) associée :

```
using System;
using System.Data;
using System.Configuration;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Web.UI.HtmlControls;

public partial class _Default : System.Web.UI.Page
{
    private String[] tpprofession = { "Avocat", "Ingénieur", "Enseignant", "Président", "CPE", "Ministre" };
    private String[] tage = { "age < 18", "18 < age < 25", "25 < age < 50", "50 < age < 65", "age > 65" };
    protected string texteRep;

    protected void Page_Load(object sender, System.EventArgs e)
    {
        HttpCookie CookId = this.Request.Cookies["identite"];
        if (CookId != null)
        {
            TextNom.Text = this.Request.Cookies["identite"]["nom"];
            TextPrenom.Text = this.Request.Cookies["identite"]["prenom"];
            TextProfession.Text = this.Request.Cookies["identite"]["profession"];
            TextSalaire.Text = this.Request.Cookies["identite"]["salaire"];
            texteRep = Convert.ToString(this.Session["Reponse"]);
        }
        else TextNom.Text = "nouveau client";
    }

    protected void Button1_Click(object sender, EventArgs e)
    {
        string VotreNom = "";
        string VotrePreNom = "";
        string votreprofession = "";
        string votreAge = "";
        string salaire = "";
        string Reponse = "";
        if (Text1.Text != null && Text1.Text.Length != 0)
    }
```

```

{
    VotreNom = Text1.Text;
    if (Text2.Text != null && Text2.Text.Length != 0)
    {
        VotrePreNom = Text2.Text;
        votreAge = tage[Select2.SelectedIndex];
        votreprofession = tprofession[Select1.SelectedIndex];

        switch (RadioButtonList1.SelectedIndex)
        {
            case 0: Reponse = "<br>moins de 1000$, insuffisant !! Dossier rejeté !";
                salaire = "moins de 1000$"; break;
            case 1: Reponse = "<br>moins de 3000$ ?, à peine suffisant, nous conservons votre dossier pour étude
approfondie.";
                salaire = "moins de 3000$"; break;
            case 2: Reponse = "plus de 3000$ !!, satisfaisant";
                if (votreprofession != "CPE")
                    Reponse += "<br><b>Cher Mr. " + VotreNom + "</b>, nous sommes heureux de pouvoir disposer
de votre argent,<br>    notre banque accueille très favorablement dans ses comptes tous les " + votreprofession +
"s tels que vous.";
                salaire = "plus de 3000$";
                break;
        }
        if (votreprofession == "CPE")
            if (RadioButtonList1.SelectedIndex == 2)
                Reponse += "<br><b>Cher Mr. " + VotreNom + "</b>, nous sommes heureux de pouvoir disposer
de votre argent,<br>    notre banque accueille malgré tout dans ses comptes les " + votreprofession + " tels que
vous."
                    + " <br>Toutefois votre situation précaire ne nous permet " +
                    "d'envisager un prêt a l'habitat que sur 3 mois maximum et à un taux majoré de 200% pour risque de
solvabilité." +
                    "<br>Avec nos conditions vous pourrez ainsi acheter un magnifique carton renforcé, dans lequel il vous
sera possible de faire la manche pour nous rembourser.";
            else
                if (RadioButtonList1.SelectedIndex == 1)
                    Reponse += " <br>Toutefois vue votre situation précaire votre dossier vient d'être
malencontreusement jeté dans notre poubelle." +
                        "<br>Revenez nous voir avec un CDI.";
                else
                    if (RadioButtonList1.SelectedIndex == 0)
                        Reponse += " <br>Avec votre situation précaire, vous osez en plus vous payer un logement !";
        }
        else
        {
            Reponse = "<br><i><font color=#ff0066>Votre prénom est incorrectement entré !</font></i>";
        }
    }
    else
    {
        Reponse = "<br><i><font color=#ff0066>Vous n'avez pas saisi correctement votre nom !!</font></i>";
    }
}
TextNom.Text = VotreNom;
TextPrenom.Text = VotrePreNom;
TextProfession.Text = votreprofession;
TextSalaire.Text = salaire;
texteRep = Reponse;
//-- Persistance des données :
// (un seul cookie pour un site et pas plus de 4Ko de données)
HttpCookie CookId = new HttpCookie("identite");
CookId["nom"] = VotreNom;

```

```

CookId["prenom"] = VotrePreNom;
CookId["profession"] = votreprofession;
CookId["salaire"] = salaire;
this.Session["Reponse"] = Reponse;
CookId.Expires = DateTime.Now.AddDays(1);

/-- validation et stockage :
bool isChampValid = VotreNom.Length * VotrePreNom.Length != 0;
if (isChampValid)//si tous les champs sont remplis
    this.Response.Cookies.Add(CookId);// on envoie la réponse http au client
}

```

ANNEXE javascript

Sources des fichiers [WebResource.axd?d=.....](#) auto générés et se trouvent dans "C:\Documents and Settings\Administrateur\Local Settings\Temporary Internet Files" :

Famille de fonctions javascript de gestion de menus auto-générée

```
var __rootMenuItem;
var __menuInterval;
var __scrollPanel;
var __disappearAfter = 500;
function Menu_ClearInterval() {
    if (__menuInterval) {
        window.clearInterval(__menuInterval);
    }
}
function Menu_Collapse(item) {
    Menu_SetRoot(item);
    if (__rootMenuItem) {
        Menu_ClearInterval();
        if (__disappearAfter >= 0) {
            __menuInterval = window.setInterval("Menu_HideItems()", __disappearAfter);
        }
    }
}
function Menu_Expand(item, horizontalOffset, verticalOffset, hideScrollers) {
    Menu_ClearInterval();
    var tr = item.parentNode.parentNode.parentNode.parentNode;
    var horizontal = true;
    if (!tr.id) {
        horizontal = false;
        tr = tr.parentNode;
    }
    var child = Menu_FindSubMenu(item);
    if (child) {
        var data = Menu_GetData(item);
        if (!data) {
            return null;
        }
        child.rel = tr.id;
        child.x = horizontalOffset;
        child.y = verticalOffset;
        if (horizontal) child.pos = "bottom";
        PopOut_Show(child.id, hideScrollers, data);
    }
    Menu_SetRoot(item);
    if (child) {
        if (!document.body.__oldOnClick && document.body.onclick) {
            document.body.__oldOnClick = document.body.onclick;
        }
        if (__rootMenuItem) {
            document.body.onclick = Menu_HideItems;
        }
    }
    Menu_ResetSiblings(tr);
    return child;
}
```

```

function Menu_FindMenu(item) {
    if (item && item.menu) return item.menu;
    var tr = item.parentNode.parentNode.parentNode.parentNode;
    if (!tr.id) {
        tr = tr.parentNode;
    }
    for (var i = tr.id.length - 1; i >= 0; i--) {
        if (tr.id.charAt(i) < '0' || tr.id.charAt(i) > '9') {
            var menu = WebForm_GetElementById(tr.id.substr(0, i));
            if (menu) {
                item.menu = menu;
                return menu;
            }
        }
    }
    return null;
}

function Menu_FindNext(item) {
    var a = WebForm_GetElementByTagName(item, "A");
    var parent = Menu_FindParentContainer(item);
    var first = null;
    if (parent) {
        var links = WebForm_GetElementsByTagName(parent, "A");
        var match = false;
        for (var i = 0; i < links.length; i++) {
            var link = links[i];
            if (Menu_IsSelectable(link)) {
                if (Menu_FindParentContainer(link) == parent) {
                    if (match) {
                        return link;
                    }
                    else if (!first) {
                        first = link;
                    }
                }
                if (!match && link == a) {
                    match = true;
                }
            }
        }
    }
    return first;
}

function Menu_FindParentContainer(item) {
    if (item.menu_ParentContainerCache) return item.menu_ParentContainerCache;
    var a = (item.tagName.toLowerCase() == "a") ? item : WebForm_GetElementByTagName(item, "A");
    var menu = Menu_FindMenu(a);
    if (menu) {
        var parent = item;
        while (parent && parent.tagName &&
            parent.id != menu.id &&
            parent.tagName.toLowerCase() != "div") {
            parent = parent.parentNode;
        }
        item.menu_ParentContainerCache = parent;
        return parent;
    }
}

function Menu_FindParentItem(item) {
    var parentContainer = Menu_FindParentContainer(item);

```

```

var parentContainerID = parentContainer.id;
var len = parentContainerID.length;
if (parentContainerID && parentContainerID.substr(len - 5) == "Items") {
    var parentItemID = parentContainerID.substr(0, len - 5);
    return WebForm_GetElementById(parentItemID);
}
return null;
}
function Menu_FindPrevious(item) {
    var a = WebForm_GetElementByTagName(item, "A");
    var parent = Menu_FindParentContainer(item);
    var last = null;
    if (parent) {
        var links = WebForm_GetElementsByTagName(parent, "A");
        for (var i = 0; i < links.length; i++) {
            var link = links[i];
            if (Menu_IsSelectable(link)) {
                if (link == a && last) {
                    return last;
                }
                if (Menu_FindParentContainer(link) == parent) {
                    last = link;
                }
            }
        }
    }
    return last;
}
function Menu_FindSubMenu(item) {
    var tr = item.parentNode.parentNode.parentNode.parentNode;
    if (!tr.id) {
        tr=tr.parentNode;
    }
    return WebForm_GetElementById(tr.id + "Items");
}
function Menu_Focus(item) {
    if (item && item.focus) {
        var pos = WebForm_GetElementPosition(item);
        var parentContainer = Menu_FindParentContainer(item);
        if (!parentContainer.offset) {
            parentContainer.offset = 0;
        }
        var posParent = WebForm_GetElementPosition(parentContainer);
        var delta;
        if (pos.y + pos.height > posParent.y + parentContainer.offset + parentContainer.clippedHeight) {
            delta = pos.y + pos.height - posParent.y - parentContainer.offset - parentContainer.clippedHeight;
            PopOut_Scroll(parentContainer, delta);
        }
        else if (pos.y < posParent.y + parentContainer.offset) {
            delta = posParent.y + parentContainer.offset - pos.y;
            PopOut_Scroll(parentContainer, -delta);
        }
        PopOut_HideScrollers(parentContainer);
        item.focus();
    }
}
function Menu_GetData(item) {
    if (!item.data) {
        var a = (item.tagName.toLowerCase() == "a" ? item : WebForm_GetElementByTagName(item, "a"));
        var menu = Menu_FindMenu(a);
    }
}

```

```

    try {
        item.data = eval(menu.id + "_Data");
    }
    catch(e) {}
}
return item.data;
}
function Menu_HideItems(items) {
    if (document.body.__oldOnClick) {
        document.body.onclick = document.body.__oldOnClick;
        document.body.__oldOnClick = null;
    }
    Menu_ClearInterval();
    if (!items || ((typeof(items.tagName) == "undefined") && (items instanceof Event))) {
        items = __rootMenuItem;
    }
    var table = items;
    if ((typeof(table) == "undefined") || (table == null) || !table.tagName || (table.tagName.toLowerCase() !=
"table")) {
        table = WebForm_GetElementByTagName(table, "TABLE");
    }
    if ((typeof(table) == "undefined") || (table == null) || !table.tagName || (table.tagName.toLowerCase() !=
"table")) {
        return;
    }
    var rows = table.rows ? table.rows : table.firstChild.rows;
    var isVertical = false;
    for (var r = 0; r < rows.length; r++) {
        if (rows[r].id) {
            isVertical = true;
            break;
        }
    }
    var i, child, nextLevel;
    if (isVertical) {
        for(i = 0; i < rows.length; i++) {
            if (rows[i].id) {
                child = WebForm_GetElementById(rows[i].id + "Items");
                if (child) {
                    Menu_HideItems(child);
                }
            }
            else if (rows[i].cells[0]) {
                nextLevel = WebForm_GetElementByTagName(rows[i].cells[0], "TABLE");
                if (nextLevel) {
                    Menu_HideItems(nextLevel);
                }
            }
        }
    }
    else {
        for(i = 0; i < rows[0].cells.length; i++) {
            if (rows[0].cells[i].id) {
                child = WebForm_GetElementById(rows[0].cells[i].id + "Items");
                if (child) {
                    Menu_HideItems(child);
                }
            }
            else {
                nextLevel = WebForm_GetElementByTagName(rows[0].cells[i], "TABLE");
            }
        }
    }
}

```

```

        if (nextLevel) {
            Menu_HideItems(rows[0].cells[i].firstChild);
        }
    }
}
}
if (items && items.id) {
    PopOut_Hide(items.id);
}
}
function Menu_HoverDisabled(item) {
    var node = (item.tagName.toLowerCase() == "td") ?
        item:
        item.cells[0];
    var data = Menu_GetData(item);
    if (!data) return;
    node = WebForm_GetElementByTagName(node, "table").rows[0].cells[0].childNodes[0];
    if (data.disappearAfter >= 200) {
        __disappearAfter = data.disappearAfter;
    }
    Menu_Expand(node, data.horizontalOffset, data.verticalOffset);
}
function Menu_HoverDynamic(item) {
    var node = (item.tagName.toLowerCase() == "td") ?
        item:
        item.cells[0];
    var data = Menu_GetData(item);
    if (!data) return;
    var nodeTable = WebForm_GetElementByTagName(node, "table");
    if (data.hoverClass) {
        nodeTable.hoverClass = data.hoverClass;
        WebForm_AppendToClassName(nodeTable, data.hoverClass);
    }
    node = nodeTable.rows[0].cells[0].childNodes[0];
    if (data.hoverHyperLinkClass) {
        node.hoverHyperLinkClass = data.hoverHyperLinkClass;
        WebForm_AppendToClassName(node, data.hoverHyperLinkClass);
    }
    if (data.disappearAfter >= 200) {
        __disappearAfter = data.disappearAfter;
    }
    Menu_Expand(node, data.horizontalOffset, data.verticalOffset);
}
function Menu_HoverRoot(item) {
    var node = (item.tagName.toLowerCase() == "td") ?
        item:
        item.cells[0];
    var data = Menu_GetData(item);
    if (!data) {
        return null;
    }
    var nodeTable = WebForm_GetElementByTagName(node, "table");
    if (data.staticHoverClass) {
        nodeTable.hoverClass = data.staticHoverClass;
        WebForm_AppendToClassName(nodeTable, data.staticHoverClass);
    }
    node = nodeTable.rows[0].cells[0].childNodes[0];
    if (data.staticHoverHyperLinkClass) {
        node.hoverHyperLinkClass = data.staticHoverHyperLinkClass;
        WebForm_AppendToClassName(node, data.staticHoverHyperLinkClass);
    }
}

```



```

    }
    return node;
}
function Menu_HoverStatic(item) {
    var node = Menu_HoverRoot(item);
    var data = Menu_GetData(item);
    if (!data) return;
    __disappearAfter = data.disappearAfter;
    Menu_Expand(node, data.horizontalOffset, data.verticalOffset);
}
function Menu_IsHorizontal(item) {
    if (item) {
        var
            a = ((item.tagName && (item.tagName.toLowerCase == "a")) ? item :
WebForm_GetElementByTagName(item, "A"));
        if (!a) {
            return false;
        }
        var td = a.parentNode.parentNode.parentNode.parentNode.parentNode;
        if (td.id) {
            return true;
        }
    }
    return false;
}
function Menu_IsSelectable(link) {
    return (link && link.href)
}
function Menu_Key(item) {
    var event;
    if (window.event) {
        event = window.event;
    }
    else {
        event = item;
        item = event.currentTarget;
    }
    var key = (event ? event.keyCode : -1);
    var data = Menu_GetData(item);
    if (!data) return;
    var horizontal = Menu_IsHorizontal(item);
    var a = WebForm_GetElementByTagName(item, "A");
    var nextItem, parentItem, previousItem;
    if ((!horizontal && key == 38) || (horizontal && key == 37)) {
        previousItem = Menu_FindPrevious(item);
        while (previousItem && previousItem.disabled) {
            previousItem = Menu_FindPrevious(previousItem);
        }
        if (previousItem) {
            Menu_Focus(previousItem);
            Menu_Expand(previousItem, data.horizontalOffset, data.verticalOffset, true);
            event.cancelBubble = true;
            if (event.stopPropagation) event.stopPropagation();
            return;
        }
    }
    if ((!horizontal && key == 40) || (horizontal && key == 39)) {
        if (horizontal) {
            var subMenu = Menu_FindSubMenu(a);
            if (subMenu && subMenu.style && subMenu.style.visibility &&

```

```

        subMenu.style.visibility.toLowerCase() == "hidden") {
            Menu_Expand(a, data.horizontalOffset, data.verticalOffset, true);
            event.cancelBubble = true;
            if (event.stopPropagation) event.stopPropagation();
            return;
        }
    }
    nextItem = Menu_FindNext(item);
    while (nextItem && nextItem.disabled) {
        nextItem = Menu_FindNext(nextItem);
    }
    if (nextItem) {
        Menu_Focus(nextItem);
        Menu_Expand(nextItem, data.horizontalOffset, data.verticalOffset, true);
        event.cancelBubble = true;
        if (event.stopPropagation) event.stopPropagation();
        return;
    }
}
if ((!horizontal && key == 39) || (horizontal && key == 40)) {
    var children = Menu_Expand(a, data.horizontalOffset, data.verticalOffset, true);
    if (children) {
        var firstChild;
        children = WebForm_GetElementsByTagName(children, "A");
        for (var i = 0; i < children.length; i++) {
            if (!children[i].disabled && Menu_IsSelectable(children[i])) {
                firstChild = children[i];
                break;
            }
        }
        if (firstChild) {
            Menu_Focus(firstChild);
            Menu_Expand(firstChild, data.horizontalOffset, data.verticalOffset, true);
            event.cancelBubble = true;
            if (event.stopPropagation) event.stopPropagation();
            return;
        }
    }
    else {
        parentItem = Menu_FindParentItem(item);
        while (parentItem && !Menu_IsHorizontal(parentItem)) {
            parentItem = Menu_FindParentItem(parentItem);
        }
        if (parentItem) {
            nextItem = Menu_FindNext(parentItem);
            while (nextItem && nextItem.disabled) {
                nextItem = Menu_FindNext(nextItem);
            }
            if (nextItem) {
                Menu_Focus(nextItem);
                Menu_Expand(nextItem, data.horizontalOffset, data.verticalOffset, true);
                event.cancelBubble = true;
                if (event.stopPropagation) event.stopPropagation();
                return;
            }
        }
    }
}
if ((!horizontal && key == 37) || (horizontal && key == 38)) {
    parentItem = Menu_FindParentItem(item);

```

```

    if (parentItem) {
        if (Menu_IsHorizontal(parentItem)) {
            previousItem = Menu_FindPrevious(parentItem);
            while (previousItem && previousItem.disabled) {
                previousItem = Menu_FindPrevious(previousItem);
            }
            if (previousItem) {
                Menu_Focus(previousItem);
                Menu_Expand(previousItem, data.horizontalOffset, data.verticalOffset, true);
                event.cancelBubble = true;
                if (event.stopPropagation) event.stopPropagation();
                return;
            }
        }
        var parentA = WebForm_GetElementByTagName(parentItem, "A");
        if (parentA) {
            Menu_Focus(parentA);
        }
        Menu_ResetSiblings(parentItem);
        event.cancelBubble = true;
        if (event.stopPropagation) event.stopPropagation();
        return;
    }
}
if (key == 27) {
    Menu_HideItems();
    event.cancelBubble = true;
    if (event.stopPropagation) event.stopPropagation();
    return;
}
}
function Menu_ResetSiblings(item) {
    var table = (item.tagName.toLowerCase() == "td") ?
        item.parentNode.parentNode.parentNode :
        item.parentNode.parentNode;
    var isVertical = false;
    for (var r = 0; r < table.rows.length; r++) {
        if (table.rows[r].id) {
            isVertical = true;
            break;
        }
    }
    var i, child, childNode;
    if (isVertical) {
        for (i = 0; i < table.rows.length; i++) {
            childNode = table.rows[i];
            if (childNode != item) {
                child = WebForm_GetElementById(childNode.id + "Items");
                if (child) {
                    Menu_HideItems(child);
                }
            }
        }
    }
    else {
        for (i = 0; i < table.rows[0].cells.length; i++) {
            childNode = table.rows[0].cells[i];
            if (childNode != item) {
                child = WebForm_GetElementById(childNode.id + "Items");
                if (child) {

```

```

        Menu_HideItems(child);
    }
}
}
Menu_ResetTopMenus(table, table, 0, true);
}
function Menu_ResetTopMenus(table, doNotReset, level, up) {
    var i, child, childNode;
    if (up && table.id == "") {
        var parentTable = table.parentNode.parentNode.parentNode;
        if (parentTable.tagName.toLowerCase() == "table") {
            Menu_ResetTopMenus(parentTable, doNotReset, level + 1, true);
        }
    }
    else {
        if (level == 0 && table != doNotReset) {
            if (table.rows[0].id) {
                for(i = 0; i < table.rows.length; i++) {
                    childNode = table.rows[i];
                    child = WebForm_GetElementById(childNode.id + "Items");
                    if (child) {
                        Menu_HideItems(child);
                    }
                }
            }
            else {
                for(i = 0; i < table.rows[0].cells.length; i++) {
                    childNode = table.rows[0].cells[i];
                    child = WebForm_GetElementById(childNode.id + "Items");
                    if (child) {
                        Menu_HideItems(child);
                    }
                }
            }
        }
        else if (level > 0) {
            for (i = 0; i < table.rows.length; i++) {
                for (var j = 0; j < table.rows[i].cells.length; j++) {
                    var subTable = table.rows[i].cells[j].firstChild;
                    if (subTable && subTable.tagName.toLowerCase() == "table") {
                        Menu_ResetTopMenus(subTable, doNotReset, level - 1, false);
                    }
                }
            }
        }
    }
}
function Menu_RestoreInterval() {
    if (__menuInterval && __rootMenuItem) {
        Menu_ClearInterval();
        __menuInterval = window.setInterval("Menu_HideItems()", __disappearAfter);
    }
}
function Menu_SetRoot(item) {
    var newRoot = Menu_FindMenu(item);
    if (newRoot) {
        if (__rootMenuItem && __rootMenuItem != newRoot) {
            Menu_HideItems();
        }
    }
}

```

```

    __rootMenuItem = newRoot;
}
}
function Menu_Unhover(item) {
    var node = (item.tagName.toLowerCase() == "td") ?
        item:
        item.cells[0];
    var nodeTable = WebForm_GetElementByTagName(node, "table");
    if (nodeTable.hoverClass) {
        WebForm_RemoveClassName(nodeTable, nodeTable.hoverClass);
    }
    node = nodeTable.rows[0].cells[0].childNodes[0];
    if (node.hoverHyperLinkClass) {
        WebForm_RemoveClassName(node, node.hoverHyperLinkClass);
    }
    Menu_Collapse(node);
}
function PopOut_Clip(element, y, height) {
    if (element && element.style) {
        element.style.clip = "rect(" + y + "px auto " + (y + height) + "px auto)";
        element.style.overflow = "hidden";
    }
}
function PopOut_Down(roller) {
    Menu_ClearInterval();
    var panel;
    if (roller) {
        panel = roller.parentNode
    }
    else {
        panel = __scrollPanel;
    }
    if (panel && ((panel.offset + panel.clippedHeight) < panel.physicalHeight)) {
        PopOut_Scroll(panel, 2)
        __scrollPanel = panel;
        PopOut_ShowScrollers(panel);
        PopOut_Stop();
        __scrollPanel.interval = window.setInterval("PopOut_Down()", 8);
    }
    else {
        PopOut_ShowScrollers(panel);
    }
}
function PopOut_Hide(panelId) {
    var panel = WebForm_GetElementById(panelId);
    if (panel && panel.tagName.toLowerCase() == "div") {
        panel.style.visibility = "hidden";
        panel.style.display = "none";
        panel.offset = 0;
        panel.scrollTop = 0;
        var table = WebForm_GetElementByTagName(panel, "TABLE");
        if (table) {
            WebForm_SetElementY(table, 0);
        }
        if (window.navigator && window.navigator.appName == "Microsoft Internet Explorer" &&
            !window.opera) {
            var childFrameId = panel.id + "_MenuIFrame";
            var childFrame = WebForm_GetElementById(childFrameId);
            if (childFrame) {
                childFrame.style.display = "none";
            }
        }
    }
}

```

```

    }
  }
}
function PopOut_HideScrollers(panel) {
  if (panel && panel.style) {
    var up = WebForm_GetElementById(panel.id + "Up");
    var dn = WebForm_GetElementById(panel.id + "Dn");
    if (up) {
      up.style.visibility = "hidden";
      up.style.display = "none";
    }
    if (dn) {
      dn.style.visibility = "hidden";
      dn.style.display = "none";
    }
  }
}
function PopOut_Position(panel, hideScrollers) {
  if (window.opera) {
    panel.parentNode.removeChild(panel);
    document.forms[0].appendChild(panel);
  }
  var rel = WebForm_GetElementById(panel.rel);
  var relTable = WebForm_GetElementByTagName(rel, "TABLE");
  var relCoordinates = WebForm_GetElementPosition(relTable ? relTable : rel);
  var panelCoordinates = WebForm_GetElementPosition(panel);
  var panelHeight = ((typeof(panel.physicalHeight) != "undefined") && (panel.physicalHeight != null)) ?
    panel.physicalHeight :
    panelCoordinates.height;
  panel.physicalHeight = panelHeight;
  var panelParentCoordinates;
  if (panel.offsetParent) {
    panelParentCoordinates = WebForm_GetElementPosition(panel.offsetParent);
  }
  else {
    panelParentCoordinates = new Object();
    panelParentCoordinates.x = 0;
    panelParentCoordinates.y = 0;
  }
  var overflowElement = WebForm_GetElementById("__overflowElement");
  if (!overflowElement) {
    overflowElement = document.createElement("img");
    overflowElement.id = "__overflowElement";
    WebForm_SetElementWidth(overflowElement, 1);
    document.body.appendChild(overflowElement);
  }
  WebForm_SetElementHeight(overflowElement, panelHeight + relCoordinates.y + parseInt(panel.y ? panel.y
: 0));
  overflowElement.style.visibility = "visible";
  overflowElement.style.display = "inline";
  var clientHeight = 0;
  var clientWidth = 0;
  if (window.innerHeight) {
    clientHeight = window.innerHeight;
    clientWidth = window.innerWidth;
  }
  else if (document.documentElement && document.documentElement.clientHeight) {
    clientHeight = document.documentElement.clientHeight;
    clientWidth = document.documentElement.clientWidth;
  }

```

```

}
else if (document.body && document.body.clientHeight) {
    clientHeight = document.body.clientHeight;
    clientWidth = document.body.clientWidth;
}
var scrollTop = 0;
var scrollLeft = 0;
if (typeof(window.pageYOffset) != "undefined") {
    scrollTop = window.pageYOffset;
    scrollLeft = window.pageXOffset;
}
else if (document.documentElement && (typeof(document.documentElement.scrollTop) != "undefined")) {
    scrollTop = document.documentElement.scrollTop;
    scrollLeft = document.documentElement.scrollLeft;
}
else if (document.body && (typeof(document.body.scrollTop) != "undefined")) {
    scrollTop = document.body.scrollTop;
    scrollLeft = document.body.scrollLeft;
}
overflowElement.style.visibility = "hidden";
overflowElement.style.display = "none";
var bottomWindowBorder = clientHeight + scrollTop;
var rightWindowBorder = clientWidth + scrollLeft;
var position = panel.pos;
if ((typeof(position) == "undefined") || (position == null) || (position == "")) {
    position = (WebForm_GetElementDir(rel) == "rtl" ? "middleleft" : "middleright");
}
position = position.toLowerCase();
var y = relCoordinates.y + parseInt(panel.y ? panel.y : 0) - panelParentCoordinates.y;
var borderParent = (rel && rel.parentNode && rel.parentNode.parentNode &&
rel.parentNode.parentNode.parentNode
&& rel.parentNode.parentNode.parentNode.tagName.toLowerCase() == "div") ?
rel.parentNode.parentNode.parentNode : null;
WebForm_SetElementY(panel, y);
PopOut_SetPanelHeight(panel, panelHeight, true);
var clip = false;
var overflow;
if (position.indexOf("top") != -1) {
    y -= panelHeight;
    WebForm_SetElementY(panel, y);
    if (y < -panelParentCoordinates.y) {
        y = -panelParentCoordinates.y;
        WebForm_SetElementY(panel, y);
        if (panelHeight > clientHeight - 2) {
            clip = true;
            PopOut_SetPanelHeight(panel, clientHeight - 2);
        }
    }
}
else {
    if (position.indexOf("bottom") != -1) {
        y += relCoordinates.height;
        WebForm_SetElementY(panel, y);
    }
    overflow = y + panelParentCoordinates.y + panelHeight - bottomWindowBorder;
    if (overflow > 0) {
        y -= overflow;
        WebForm_SetElementY(panel, y);
        if (y < -panelParentCoordinates.y) {
            y = 2 - panelParentCoordinates.y + scrollTop;

```

```

        WebForm_SetElementY(panel, y);
        clip = true;
        PopOut_SetPanelHeight(panel, clientHeight - 2);
    }
}
}
if (!clip) {
    PopOut_SetPanelHeight(panel, panel.clippedHeight, true);
}
var panelParentOffsetY = 0;
if (panel.offsetParent) {
    panelParentOffsetY = WebForm_GetElementPosition(panel.offsetParent).y;
}
var panelY = ((typeof(panel.originY) != "undefined") && (panel.originY != null)) ?
    panel.originY :
    y - panelParentOffsetY;
panel.originY = panelY;
if (!hideScrollers) {
    PopOut_ShowScrollers(panel);
}
else {
    PopOut_HideScrollers(panel);
}
var x = relCoordinates.x + parseInt(panel.x ? panel.x : 0) - panelParentCoordinates.x;
if (borderParent && borderParent.clientLeft) {
    x += 2 * borderParent.clientLeft;
}
WebForm_SetElementX(panel, x);
if (position.indexOf("left") != -1) {
    x -= panelCoordinates.width;
    WebForm_SetElementX(panel, x);
    if (x < -panelParentCoordinates.x) {
        WebForm_SetElementX(panel, -panelParentCoordinates.x);
    }
}
else {
    if (position.indexOf("right") != -1) {
        x += relCoordinates.width;
        WebForm_SetElementX(panel, x);
    }
    overflow = x + panelParentCoordinates.x + panelCoordinates.width - rightWindowBorder;
    if (overflow > 0) {
        if (position.indexOf("bottom") == -1 && relCoordinates.x > panelCoordinates.width) {
            x -= relCoordinates.width + panelCoordinates.width;
        }
        else {
            x -= overflow;
        }
        WebForm_SetElementX(panel, x);
        if (x < -panelParentCoordinates.x) {
            WebForm_SetElementX(panel, -panelParentCoordinates.x);
        }
    }
}
}
}
function PopOut_Scroll(panel, offsetDelta) {
    var table = WebForm_GetElementByTagName(panel, "TABLE");
    if (!table) return;
    table.style.position = "relative";
    var tableY = (table.style.top ? parseInt(table.style.top) : 0);

```



```

panel.offset += offsetDelta;
WebForm_SetElementY(table, tableY - offsetDelta);
}
function PopOut_SetPanelHeight(element, height, doNotClip) {
    if (element && element.style) {
        var size = WebForm_GetElementPosition(element);
        element.physicalWidth = size.width;
        element.clippedHeight = height;
        WebForm_SetElementHeight(element, height - (element.clientTop ? (2 * element.clientTop) : 0));
        if (doNotClip && element.style) {
            element.style.clip = "rect(auto auto auto auto)";
        }
        else {
            PopOut_Clip(element, 0, height);
        }
    }
}
function PopOut_Show(panelId, hideScrollers, data) {
    var panel = WebForm_GetElementById(panelId);
    if (panel && panel.tagName.toLowerCase() == "div") {
        panel.style.visibility = "visible";
        panel.style.display = "inline";
        if (!panel.offset || hideScrollers) {
            panel.scrollTop = 0;
            panel.offset = 0;
            var table = WebForm_GetElementByTagName(panel, "TABLE");
            if (table) {
                WebForm_SetElementY(table, 0);
            }
        }
        PopOut_Position(panel, hideScrollers);
        var z = 1;
        var isIE = window.navigator && window.navigator.appName == "Microsoft Internet Explorer" &&
!window.opera;
        if (isIE && data) {
            var childFrameId = panel.id + "_MenuIFrame";
            var childFrame = WebForm_GetElementById(childFrameId);
            var parent = panel.offsetParent;
            if (!childFrame) {
                childFrame = document.createElement("iframe");
                childFrame.id = childFrameId;
                childFrame.src = (data.iframeUrl ? data.iframeUrl : "about:blank");
                childFrame.style.position = "absolute";
                childFrame.style.display = "none";
                childFrame.scrolling = "no";
                childFrame.frameBorder = "0";
                if (parent.tagName.toLowerCase() == "html") {
                    document.body.appendChild(childFrame);
                }
                else {
                    parent.appendChild(childFrame);
                }
            }
            var pos = WebForm_GetElementPosition(panel);
            var parentPos = WebForm_GetElementPosition(parent);
            WebForm_SetElementX(childFrame, pos.x - parentPos.x);
            WebForm_SetElementY(childFrame, pos.y - parentPos.y);
            WebForm_SetElementWidth(childFrame, pos.width);
            WebForm_SetElementHeight(childFrame, pos.height);
            childFrame.style.display = "block";

```

```

        if (panel.currentStyle && panel.currentStyle.zIndex) {
            z = panel.currentStyle.zIndex;
        }
        else if (panel.style.zIndex) {
            z = panel.style.zIndex;
        }
    }
    panel.style.zIndex = z;
}
}
function PopOut_ShowScrollers(panel) {
    if (panel && panel.style) {
        var up = WebForm_GetElementById(panel.id + "Up");
        var dn = WebForm_GetElementById(panel.id + "Dn");
        var cnt = 0;
        if (up && dn) {
            if (panel.offset && panel.offset > 0) {
                up.style.visibility = "visible";
                up.style.display = "inline";
                cnt++;
                if (panel.clientWidth) {
                    WebForm_SetElementWidth(up, panel.clientWidth
                        - (up.clientLeft ? (2 * up.clientLeft) : 0));
                }
                WebForm_SetElementY(up, 0);
            }
            else {
                up.style.visibility = "hidden";
                up.style.display = "none";
            }
            if (panel.offset + panel.clippedHeight + 2 <= panel.physicalHeight) {
                dn.style.visibility = "visible";
                dn.style.display = "inline";
                cnt++;
                if (panel.clientWidth) {
                    WebForm_SetElementWidth(dn, panel.clientWidth
                        - (dn.clientLeft ? (2 * dn.clientLeft) : 0));
                }
                WebForm_SetElementY(dn, panel.clippedHeight - WebForm_GetElementPosition(dn).height
                    - (panel.clientTop ? (2 * panel.clientTop) : 0));
            }
            else {
                dn.style.visibility = "hidden";
                dn.style.display = "none";
            }
            if (cnt == 0) {
                panel.style.clip = "rect(auto auto auto auto)";
            }
        }
    }
}
function PopOut_Stop() {
    if (__scrollPanel && __scrollPanel.interval) {
        window.clearInterval(__scrollPanel.interval);
    }
    Menu_RestoreInterval();
}
function PopOut_Up(scroller) {
    Menu_ClearInterval();
    var panel;

```

```

if (scroller) {
    panel = scroller.parentNode
}
else {
    panel = __scrollPanel;
}
if (panel && panel.offset && panel.offset > 0) {
    PopOut_Scroll(panel, -2);
    __scrollPanel = panel;
    PopOut_ShowScrollers(panel);
    PopOut_Stop();
    __scrollPanel.interval = window.setInterval("PopOut_Up()", 8);
}
}
}

```

Famille de fonctions javascript de gestion d'une fiche

```

function WebForm_PostBackOptions(eventTarget, eventArgument, validation, validationGroup, actionUrl,
trackFocus, clientSubmit) {
    this.eventTarget = eventTarget;
    this.eventArgument = eventArgument;
    this.validation = validation;
    this.validationGroup = validationGroup;
    this.actionUrl = actionUrl;
    this.trackFocus = trackFocus;
    this.clientSubmit = clientSubmit;
}
function WebForm_DoPostBackWithOptions(options) {
    var validationResult = true;
    if (options.validation) {
        if (typeof(Page_ClientValidate) == 'function') {
            validationResult = Page_ClientValidate(options.validationGroup);
        }
    }
    if (validationResult) {
        if ((typeof(options.actionUrl) != "undefined") && (options.actionUrl != null) && (options.actionUrl.length >
0)) {
            theForm.action = options.actionUrl;
        }
        if (options.trackFocus) {
            var lastFocus = theForm.elements["__LASTFOCUS"];
            if ((typeof(lastFocus) != "undefined") && (lastFocus != null)) {
                if (typeof(document.activeElement) == "undefined") {
                    lastFocus.value = options.eventTarget;
                }
                else {
                    var active = document.activeElement;
                    if ((typeof(active) != "undefined") && (active != null)) {
                        if ((typeof(active.id) != "undefined") && (active.id != null) && (active.id.length > 0)) {
                            lastFocus.value = active.id;
                        }
                        else if (typeof(active.name) != "undefined") {
                            lastFocus.value = active.name;
                        }
                    }
                }
            }
        }
    }
}
}

```

```

    }
    if (options.clientSubmit) {
        __doPostBack(options.eventTarget, options.eventArgument);
    }
}
var __pendingCallbacks = new Array();
var __synchronousCallBackIndex = -1;
function WebForm_DoCallback(eventTarget, eventArgument, eventCallback, context, errorCallback, useAsync) {
    var postData = __theFormPostData +
        "__CALLBACKID=" + WebForm_EncodeCallback(eventTarget) +
        "&__CALLBACKPARAM=" + WebForm_EncodeCallback(eventArgument);
    if (theForm["__EVENTVALIDATION"]) {
        postData += "&__EVENTVALIDATION=" +
WebForm_EncodeCallback(theForm["__EVENTVALIDATION"].value);
    }
    var xmlRequest,e;
    try {
        xmlRequest = new XMLHttpRequest();
    }
    catch(e) {
        try {
            xmlRequest = new ActiveXObject("Microsoft.XMLHTTP");
        }
        catch(e) {
        }
    }
    var setRequestHeaderMethodExists = true;
    try {
        setRequestHeaderMethodExists = (xmlRequest && xmlRequest.setRequestHeader);
    }
    catch(e) {}
    var callback = new Object();
    callback.eventCallback = eventCallback;
    callback.context = context;
    callback.errorCallback = errorCallback;
    callback.async = useAsync;
    var callbackIndex = WebForm_FillFirstAvailableSlot(__pendingCallbacks, callback);
    if (!useAsync) {
        if (__synchronousCallBackIndex != -1) {
            __pendingCallbacks[__synchronousCallBackIndex] = null;
        }
        __synchronousCallBackIndex = callbackIndex;
    }
    if (setRequestHeaderMethodExists) {
        xmlRequest.onreadystatechange = WebForm_CallbackComplete;
        callback.xmlRequest = xmlRequest;
        xmlRequest.open("POST", theForm.action, true);
        xmlRequest.setRequestHeader("Content-Type", "application/x-www-form-urlencoded");
        xmlRequest.send(postData);
        return;
    }
    callback.xmlRequest = new Object();
    var callbackFrameID = "__CALLBACKFRAME" + callbackIndex;
    var xmlRequestFrame = document.frames[callbackFrameID];
    if (!xmlRequestFrame) {
        xmlRequestFrame = document.createElement("IFRAME");
        xmlRequestFrame.width = "1";
        xmlRequestFrame.height = "1";
        xmlRequestFrame.frameBorder = "0";
        xmlRequestFrame.id = callbackFrameID;
    }
}

```

```

xmlRequestFrame.name = callbackFrameID;
xmlRequestFrame.style.position = "absolute";
xmlRequestFrame.style.top = "-100px";
xmlRequestFrame.style.left = "-100px";
try {
    if (callBackFrameUrl) {
        xmlRequestFrame.src = callBackFrameUrl;
    }
}
catch(e) {}
document.body.appendChild(xmlRequestFrame);
}
var interval = window.setInterval(function() {
    xmlRequestFrame = document.frames[callbackFrameID];
    if (xmlRequestFrame && xmlRequestFrame.document) {
        window.clearInterval(interval);
        xmlRequestFrame.document.write("");
        xmlRequestFrame.document.close();
        xmlRequestFrame.document.write('<html><body><form method="post"><input type="hidden"
name="__CALLBACKLOADSCRIPT" value="t"></form></body></html>');
        xmlRequestFrame.document.close();
        xmlRequestFrame.document.forms[0].action = theForm.action;
        var count = __theFormPostCollection.length;
        var element;
        for (var i = 0; i < count; i++) {
            element = __theFormPostCollection[i];
            if (element) {
                var fieldElement = xmlRequestFrame.document.createElement("INPUT");
                fieldElement.type = "hidden";
                fieldElement.name = element.name;
                fieldElement.value = element.value;
                xmlRequestFrame.document.forms[0].appendChild(fieldElement);
            }
        }
        var callbackIdFieldElement = xmlRequestFrame.document.createElement("INPUT");
        callbackIdFieldElement.type = "hidden";
        callbackIdFieldElement.name = "__CALLBACKID";
        callbackIdFieldElement.value = eventTarget;
        xmlRequestFrame.document.forms[0].appendChild(callbackIdFieldElement);
        var callbackParamFieldElement = xmlRequestFrame.document.createElement("INPUT");
        callbackParamFieldElement.type = "hidden";
        callbackParamFieldElement.name = "__CALLBACKPARAM";
        callbackParamFieldElement.value = eventArgument;
        xmlRequestFrame.document.forms[0].appendChild(callbackParamFieldElement);
        if (theForm["__EVENTVALIDATION"]) {
            var callbackValidationFieldElement = xmlRequestFrame.document.createElement("INPUT");
            callbackValidationFieldElement.type = "hidden";
            callbackValidationFieldElement.name = "__EVENTVALIDATION";
            callbackValidationFieldElement.value = theForm["__EVENTVALIDATION"].value;
            xmlRequestFrame.document.forms[0].appendChild(callbackValidationFieldElement);
        }
        var callbackIndexFieldElement = xmlRequestFrame.document.createElement("INPUT");
        callbackIndexFieldElement.type = "hidden";
        callbackIndexFieldElement.name = "__CALLBACKINDEX";
        callbackIndexFieldElement.value = callbackIndex;
        xmlRequestFrame.document.forms[0].appendChild(callbackIndexFieldElement);
        xmlRequestFrame.document.forms[0].submit();
    }
}, 10);
}

```

```

function WebForm_CallbackComplete() {
    for (i = 0; i < __pendingCallbacks.length; i++) {
        callbackObject = __pendingCallbacks[i];
        if (callbackObject && callbackObject.xmlRequest && (callbackObject.xmlRequest.readyState == 4)) {
            WebForm_ExecuteCallback(callbackObject);
            if (!__pendingCallbacks[i].async) {
                __synchronousCallBackIndex = -1;
            }
            __pendingCallbacks[i] = null;
            var callbackFrameID = "__CALLBACKFRAME" + i;
            var xmlRequestFrame = document.getElementById(callbackFrameID);
            if (xmlRequestFrame) {
                xmlRequestFrame.parentNode.removeChild(xmlRequestFrame);
            }
        }
    }
}

function WebForm_ExecuteCallback(callbackObject) {
    var response = callbackObject.xmlRequest.responseText;
    if (response.charAt(0) == "s") {
        if ((typeof(callbackObject.eventCallback) != "undefined") && (callbackObject.eventCallback != null)) {
            callbackObject.eventCallback(response.substring(1), callbackObject.context);
        }
    }
    else if (response.charAt(0) == "e") {
        if ((typeof(callbackObject.errorCallback) != "undefined") && (callbackObject.errorCallback != null)) {
            callbackObject.errorCallback(response.substring(1), callbackObject.context);
        }
    }
    else {
        var separatorIndex = response.indexOf("|");
        if (separatorIndex != -1) {
            var validationFieldLength = parseInt(response.substring(0, separatorIndex));
            if (!isNaN(validationFieldLength)) {
                var validationField = response.substring(separatorIndex + 1, separatorIndex + validationFieldLength + 1);
                if (validationField != "") {
                    var validationFieldElement = theForm["__EVENTVALIDATION"];
                    if (!validationFieldElement) {
                        validationFieldElement = document.createElement("INPUT");
                        validationFieldElement.type = "hidden";
                        validationFieldElement.name = "__EVENTVALIDATION";
                        theForm.appendChild(validationFieldElement);
                    }
                    validationFieldElement.value = validationField;
                }
                if ((typeof(callbackObject.eventCallback) != "undefined") && (callbackObject.eventCallback != null)) {
                    callbackObject.eventCallback(response.substring(separatorIndex + validationFieldLength + 1),
callbackObject.context);
                }
            }
        }
    }
}

function WebForm_FillFirstAvailableSlot(array, element) {
    var i;
    for (i = 0; i < array.length; i++) {
        if (!array[i]) break;
    }
    array[i] = element;
    return i;
}

```

```

}
var __nonMSDOMBrowser = (window.navigator.appName.toLowerCase().indexOf('explorer') == -1);
var __theFormPostData = "";
var __theFormPostCollection = new Array();
function WebForm_InitCallback() {
    var count = theForm.elements.length;
    var element;
    for (var i = 0; i < count; i++) {
        element = theForm.elements[i];
        var tagName = element.tagName.toLowerCase();
        if (tagName == "input") {
            var type = element.type;
            if ((type == "text" || type == "hidden" || type == "password" ||
                (type == "checkbox" || type == "radio") && element.checked)) &&
                (element.id != "__EVENTVALIDATION")) {
                WebForm_InitCallbackAddField(element.name, element.value);
            }
        }
        else if (tagName == "select") {
            var selectCount = element.options.length;
            for (var j = 0; j < selectCount; j++) {
                var selectChild = element.options[j];
                if (selectChild.selected == true) {
                    WebForm_InitCallbackAddField(element.name, element.value);
                }
            }
        }
        else if (tagName == "textarea") {
            WebForm_InitCallbackAddField(element.name, element.value);
        }
    }
}
function WebForm_InitCallbackAddField(name, value) {
    var nameValue = new Object();
    nameValue.name = name;
    nameValue.value = value;
    __theFormPostCollection[__theFormPostCollection.length] = nameValue;
    __theFormPostData += name + "=" + WebForm_EncodeCallback(value) + "&";
}
function WebForm_EncodeCallback(parameter) {
    if (encodeURIComponent) {
        return encodeURIComponent(parameter);
    }
    else {
        return escape(parameter);
    }
}
var __disabledControlArray = new Array();
function WebForm_ReEnableControls() {
    if (typeof(__enabledControlArray) == 'undefined') {
        return false;
    }
    var disabledIndex = 0;
    for (var i = 0; i < __enabledControlArray.length; i++) {
        var c;
        if (__nonMSDOMBrowser) {
            c = document.getElementById(__enabledControlArray[i]);
        }
        else {
            c = document.all[__enabledControlArray[i]];
        }
    }
}

```

```

    }
    if ((typeof(c) != "undefined") && (c != null) && (c.disabled == true)) {
        c.disabled = false;
        __disabledControlArray[disabledIndex++] = c;
    }
}
setTimeout("WebForm_ReDisableControls()", 0);
return true;
}
function WebForm_ReDisableControls() {
    for (var i = 0; i < __disabledControlArray.length; i++) {
        __disabledControlArray[i].disabled = true;
    }
}
var __defaultFired = false;
function WebForm_FireDefaultButton(event, target) {
    if (!__defaultFired && event.keyCode == 13 && !(event.srcElement &&
(event.srcElement.tagName.toLowerCase() == "textarea"))) {
        var defaultButton;
        if (__nonMSDOMBrowser) {
            defaultButton = document.getElementById(target);
        }
        else {
            defaultButton = document.all[target];
        }
        if (defaultButton && typeof(defaultButton.click) != "undefined") {
            __defaultFired = true;
            defaultButton.click();
            event.cancelBubble = true;
            if (event.stopPropagation) event.stopPropagation();
            return false;
        }
    }
    return true;
}
function WebForm_GetScrollX() {
    if (__nonMSDOMBrowser) {
        return window.pageXOffset;
    }
    else {
        if (document.documentElement && document.documentElement.scrollLeft) {
            return document.documentElement.scrollLeft;
        }
        else if (document.body) {
            return document.body.scrollLeft;
        }
    }
    return 0;
}
function WebForm_GetScrollY() {
    if (__nonMSDOMBrowser) {
        return window.pageYOffset;
    }
    else {
        if (document.documentElement && document.documentElement.scrollTop) {
            return document.documentElement.scrollTop;
        }
        else if (document.body) {
            return document.body.scrollTop;
        }
    }
}

```



```

    }
    return 0;
}
function WebForm_SaveScrollPositionSubmit() {
    if (__nonMSDOMBrowser) {
        theForm.elements['__SCROLLPOSITIONY'].value = window.pageYOffset;
        theForm.elements['__SCROLLPOSITIONX'].value = window.pageXOffset;
    }
    else {
        theForm.__SCROLLPOSITIONX.value = WebForm_GetScrollX();
        theForm.__SCROLLPOSITIONY.value = WebForm_GetScrollY();
    }
    if ((typeof(this.oldSubmit) != "undefined") && (this.oldSubmit != null)) {
        return this.oldSubmit();
    }
    return true;
}
function WebForm_SaveScrollPositionOnSubmit() {
    theForm.__SCROLLPOSITIONX.value = WebForm_GetScrollX();
    theForm.__SCROLLPOSITIONY.value = WebForm_GetScrollY();
    if ((typeof(this.oldOnSubmit) != "undefined") && (this.oldOnSubmit != null)) {
        return this.oldOnSubmit();
    }
    return true;
}
function WebForm_RestoreScrollPosition() {
    if (__nonMSDOMBrowser) {
        window.scrollTo(theForm.elements['__SCROLLPOSITIONX'].value,
theForm.elements['__SCROLLPOSITIONY'].value);
    }
    else {
        window.scrollTo(theForm.__SCROLLPOSITIONX.value, theForm.__SCROLLPOSITIONY.value);
    }
    if ((typeof(theForm.oldOnLoad) != "undefined") && (theForm.oldOnLoad != null)) {
        return theForm.oldOnLoad();
    }
    return true;
}
function WebForm_TextBoxKeyHandler(event) {
    if (event.keyCode == 13) {
        var target;
        if (__nonMSDOMBrowser) {
            target = event.target;
        }
        else {
            target = event.srcElement;
        }
        if ((typeof(target) != "undefined") && (target != null)) {
            if (typeof(target.onChange) != "undefined") {
                target.onChange();
                event.cancelBubble = true;
                if (event.stopPropagation) event.stopPropagation();
                return false;
            }
        }
    }
    return true;
}
function WebForm_AppendToClassName(element, className) {
    var current = element.className;

```

```

if (current) {
    if (current.charAt(current.length - 1) != ' ') {
        current += ' ';
    }
    current += className;
}
else {
    current = className;
}
element.className = current;
}

function WebForm_RemoveClassName(element, className) {
    var current = element.className;
    if (current) {
        if (current.substring(current.length - className.length - 1, current.length) == ' ' + className) {
            element.className = current.substring(0, current.length - className.length - 1);
            return;
        }
        if (current == className) {
            element.className = '';
            return;
        }
        var index = current.indexOf(' ' + className + ' ');
        if (index != -1) {
            element.className = current.substring(0, index) + current.substring(index + className.length + 2,
current.length);
            return;
        }
        if (current.substring(0, className.length) == className + ' ') {
            element.className = current.substring(className.length + 1, current.length);
        }
    }
}

function WebForm_GetElementById(elementId) {
    if (document.getElementById) {
        return document.getElementById(elementId);
    }
    else if (document.all) {
        return document.all[elementId];
    }
    else return null;
}

function WebForm_GetElementByTagName(element, tagName) {
    var elements = WebForm_GetElementsByTagName(element, tagName);
    if (elements && elements.length > 0) {
        return elements[0];
    }
    else return null;
}

function WebForm_GetElementsByTagName(element, tagName) {
    if (element && tagName) {
        if (element.getElementsByTagName) {
            return element.getElementsByTagName(tagName);
        }
        if (element.all && element.all.tags) {
            return element.all.tags(tagName);
        }
    }
    return null;
}

```

```

function WebForm_GetElementDir(element) {
    if (element) {
        if (element.dir) {
            return element.dir;
        }
        return WebForm_GetElementDir(element.parentNode);
    }
    return "ltr";
}
function WebForm_GetElementPosition(element) {
    var result = new Object();
    result.x = 0;
    result.y = 0;
    result.width = 0;
    result.height = 0;
    if (element.offsetParent) {
        result.x = element.offsetLeft;
        result.y = element.offsetTop;
        var parent = element.offsetParent;
        while (parent) {
            result.x += parent.offsetLeft;
            result.y += parent.offsetTop;
            var parentTagName = parent.tagName.toLowerCase();
            if (parentTagName != "table" &&
                parentTagName != "body" &&
                parentTagName != "html" &&
                parentTagName != "div" &&
                parent.clientTop &&
                parent.clientLeft) {
                result.x += parent.clientLeft;
                result.y += parent.clientTop;
            }
            parent = parent.offsetParent;
        }
    }
    else if (element.left && element.top) {
        result.x = element.left;
        result.y = element.top;
    }
    else {
        if (element.x) {
            result.x = element.x;
        }
        if (element.y) {
            result.y = element.y;
        }
    }
    if (element.offsetWidth && element.offsetHeight) {
        result.width = element.offsetWidth;
        result.height = element.offsetHeight;
    }
    else if (element.style && element.style.pixelWidth && element.style.pixelHeight) {
        result.width = element.style.pixelWidth;
        result.height = element.style.pixelHeight;
    }
    return result;
}
function WebForm_GetParentByTagName(element, tagName) {
    var parent = element.parentNode;
    var upperTagName = tagName.toUpperCase();

```

```

while (parent && (parent.tagName.toUpperCase() != upperTagName)) {
    parent = parent.parentNode ? parent.parentNode : parent.parentElement;
}
return parent;
}
function WebForm_SetElementHeight(element, height) {
    if (element && element.style) {
        element.style.height = height + "px";
    }
}
function WebForm_SetElementWidth(element, width) {
    if (element && element.style) {
        element.style.width = width + "px";
    }
}
function WebForm_SetElementX(element, x) {
    if (element && element.style) {
        element.style.left = x + "px";
    }
}
function WebForm_SetElementY(element, y) {
    if (element && element.style) {
        element.style.top = y + "px";
    }
}

```

ASP.net et WebServices



Comment créer et consommer un webservice

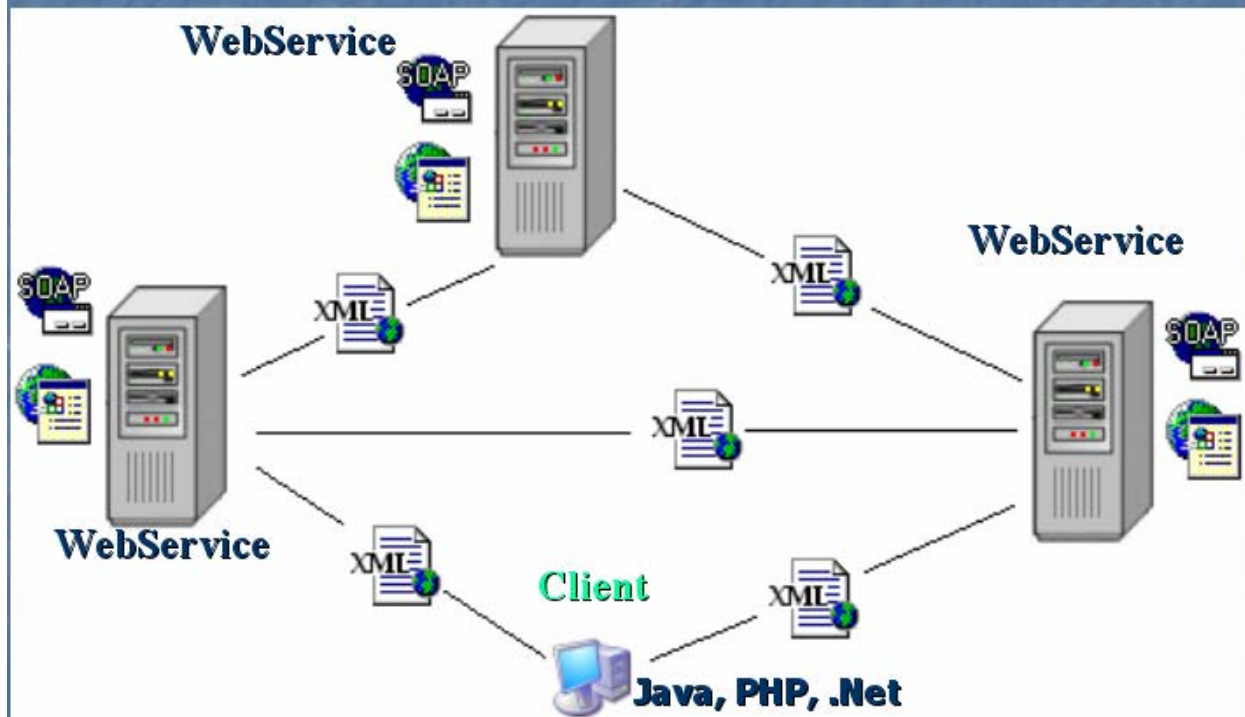
avec ASP.net

En 53 diapositives

WebServices services d'applications distribuées - .Net/Java

- Interconnexion de plateformes hétérogènes et incompatibles : .Net / Java.
- Intégration d'applications existantes de l'entreprise dans de nouvelles applications .Net / Java.
- Développement d'application Client / serveur sur Internet.

WebServices 3 standards : XML-SOAP-HTTP



WebService - principes

La pile des services :

Annuaire : publication et recherche de services	UDDI
Inspection : recherche de services sur un serveur	DISCO
Description des services (Web Service Description Language)	WSDL
Description des échanges (Simple Object Access Protocol)	SOAP
Format des données	XML
Communication	Réseau (Internet)

WebService - techniques



SOAP - format d'un message

- Un document XML utilisant le schéma SOAP :

```
<?xml version="1.0"?>
<soap:Envelope ...>
  <soap:Header ...>
    ...
  </soap:Header>
  <soap:Body>
    <Add xmlns="http://tempuri.org/">
      <n1>12</n1>
      <n2>10</n2>
    </Add>
  </soap:Body>
</soap:Envelope>
```

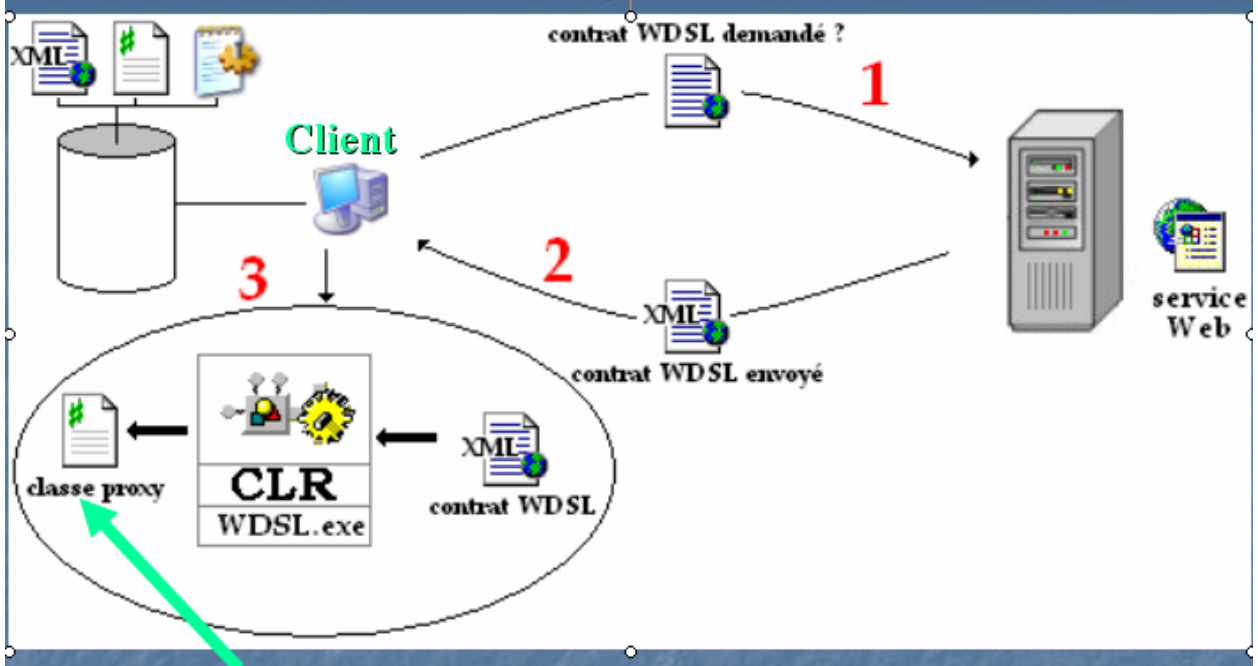
SOAP - format de la réponse

- Message de réponse du serveur de Webservice :

```
HTTP/1.1 200 OK
...
Content-Type:text/xml
Content-Length: 391

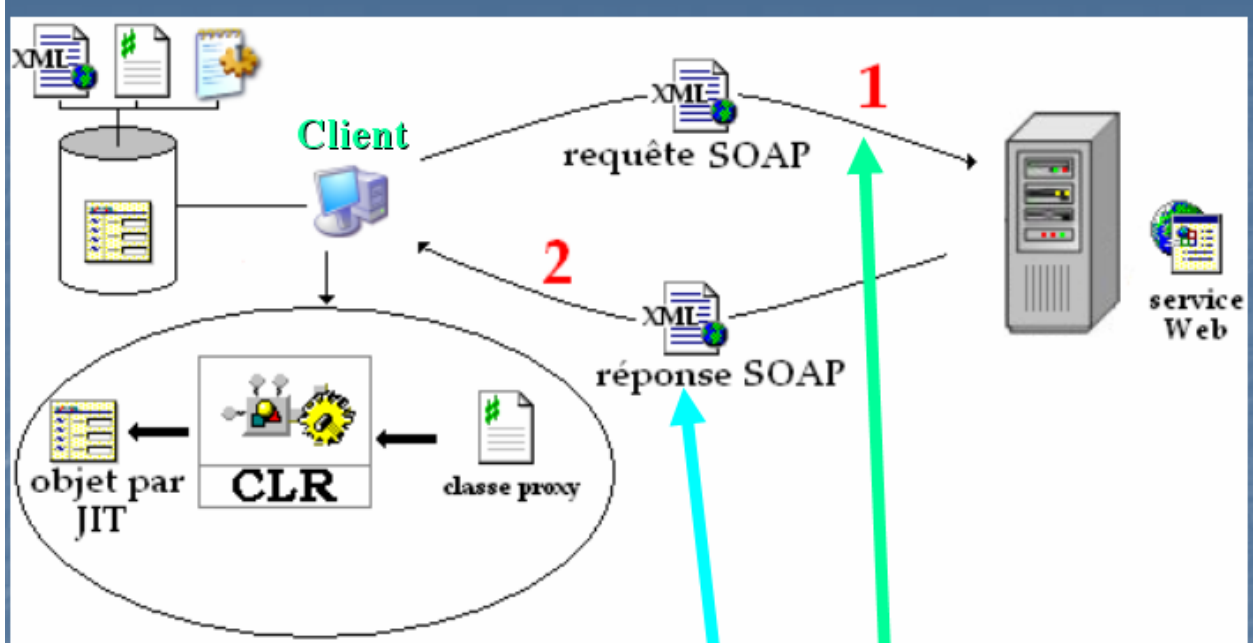
<?xml version="1.0"?>
<soap:Envelope ...>
  <soap:Body>
    <AddResult xmlns="http://tempuri.org/">
      <result>120</result>
    </AddResult>
  </soap:Body>
</soap:Envelope>
```


WebService – acquisition sous .Net



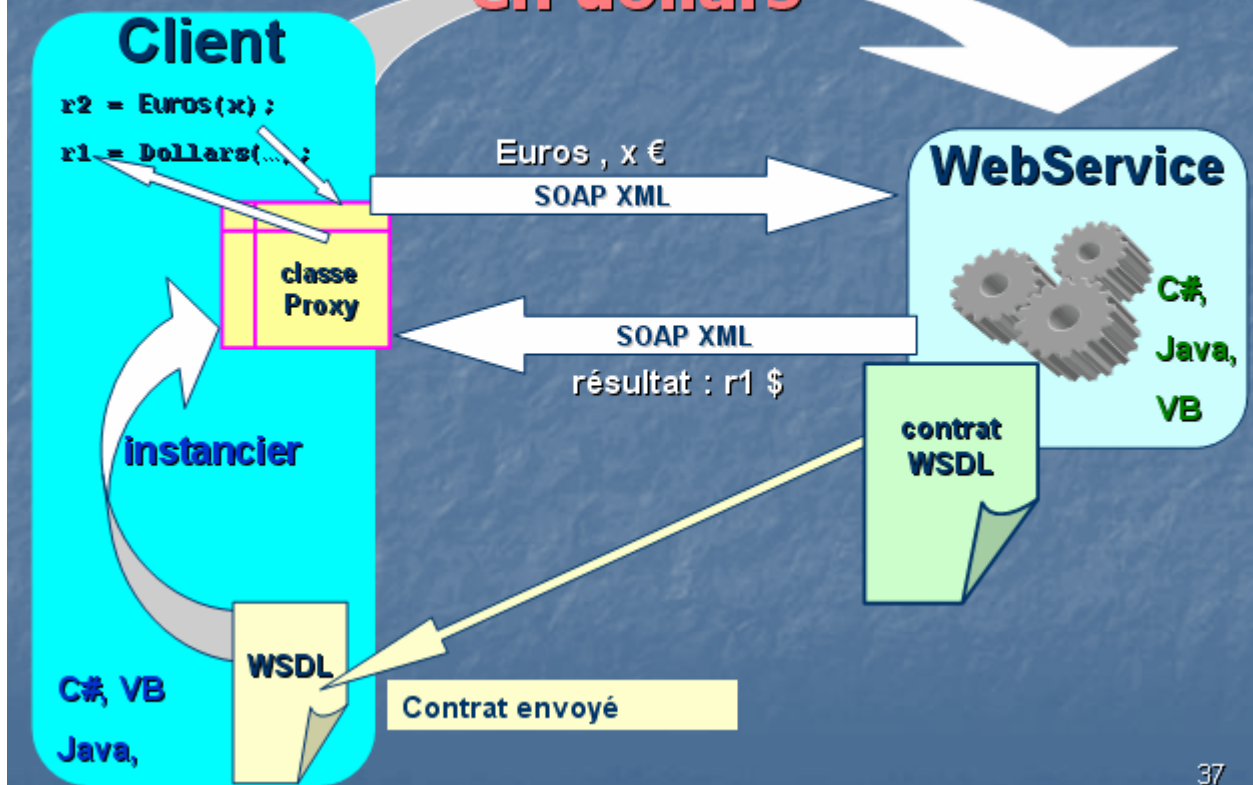
Création du code source d'une classe **proxy** à partir du contrat WDSL

WebService – consommation sous .Net



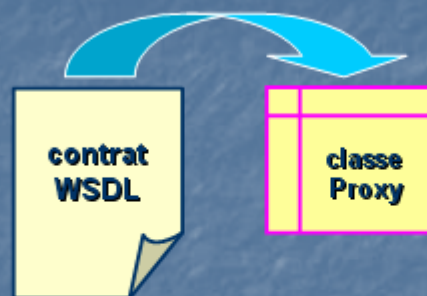
Utilisation des méthodes de la classe **proxy** et retour des réponses par le WebService.

WebService : changer des euros en dollars



37

WebService – classe proxy générée



Le proxy est une classe contenant uniquement la logique d'invocation du webservice (attributs et méthodes publiques).

Cette classe est utilisée dans un programme comme une classe ordinaire.

WebService – conditions d'interopérabilité

Ne pas se lier au langage.

Ne pas se lier à l'implémentation du langage.

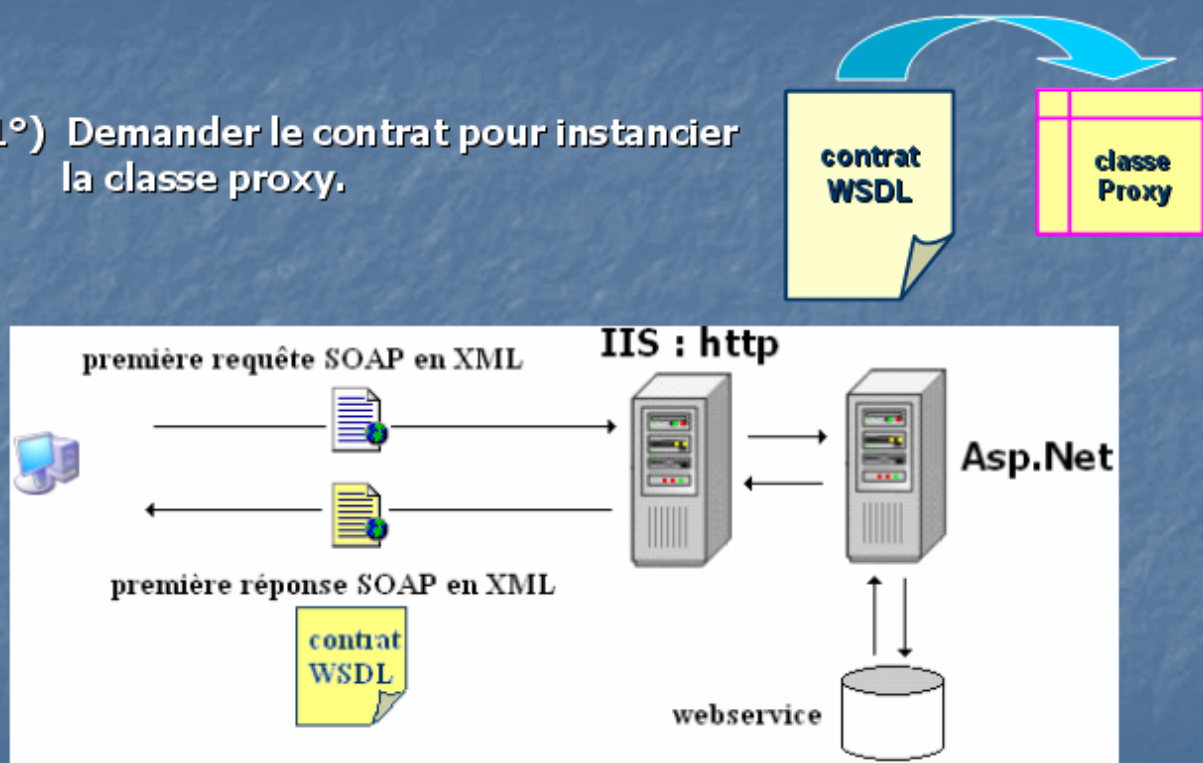
Utilisez les types suivants :

- Types simples : int, float, string, ...
- Classes avec propriétés (ou javaBean enJava)
- Tableaux typés
- Collections génériques

N'utilisez pas la classe Object de .Net ou de Java.

ASP.Net et webservice

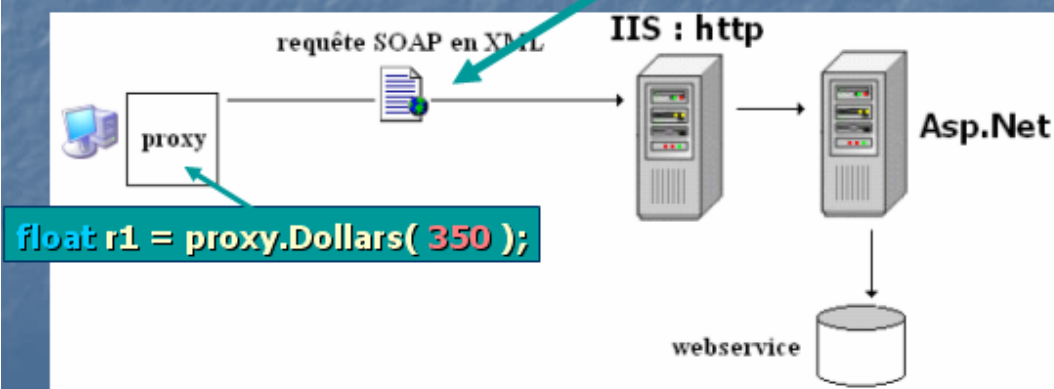
1°) Demander le contrat pour instancier la classe proxy.



ASP.Net et webservice

- 2°) Envoyer une requête SOAP pour convertir **350 €** en dollars avec la classe proxy.

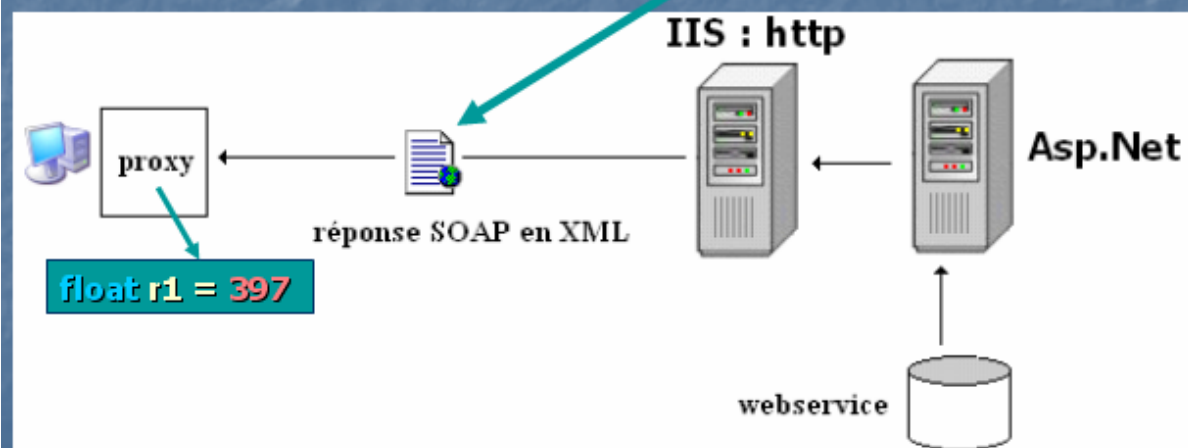
```
<Soap : body >  
  <Dollars>  
    <euros>350</euros>  
  </Dollars>  
</Soap : body >
```



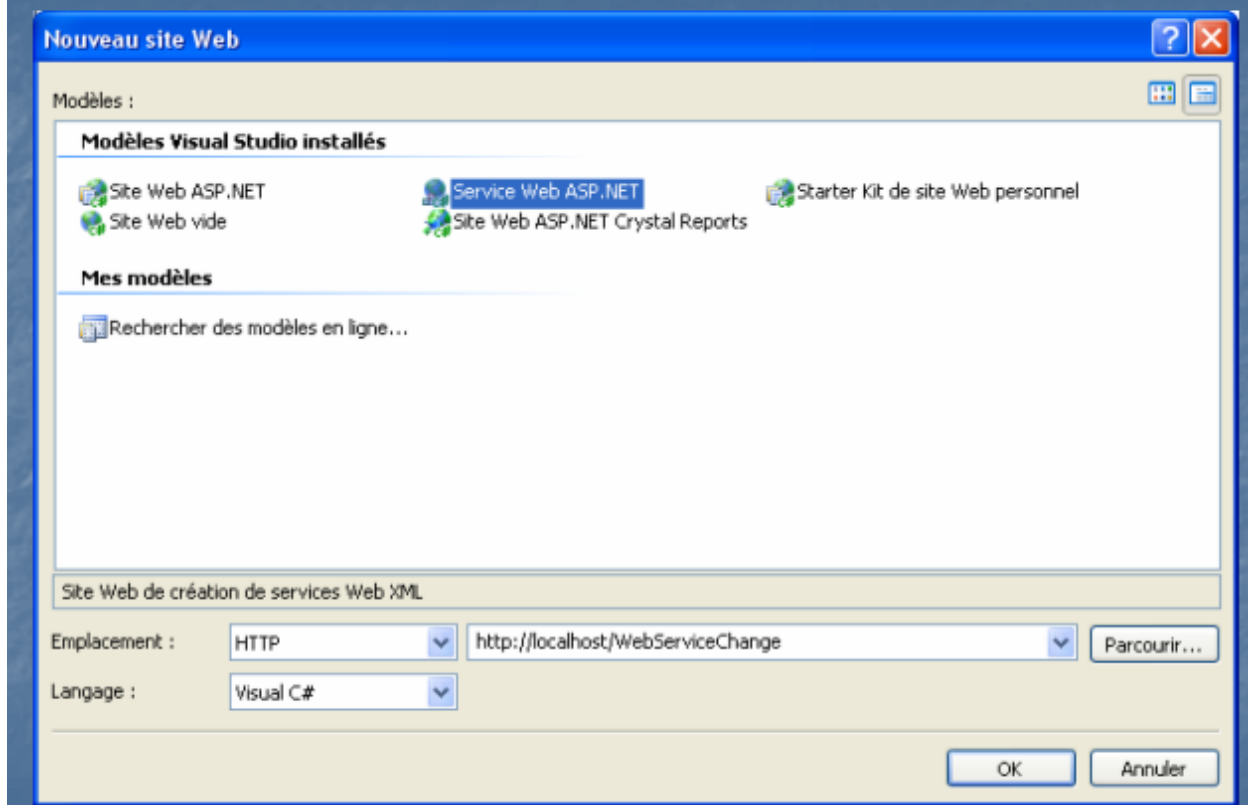
ASP.Net et webservice

- 3°) le webservice renvoie la réponse au format SOAP à destination de la classe proxy.

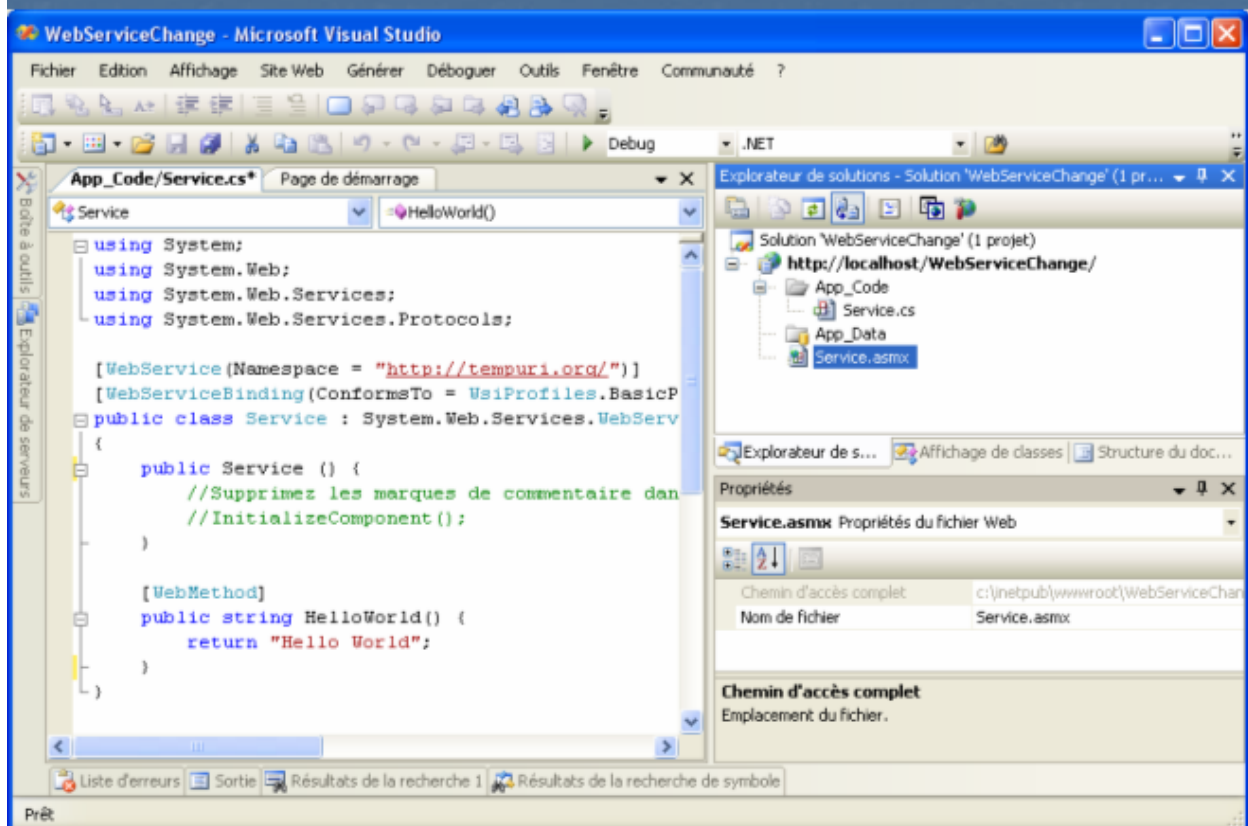
```
<Soap : body >  
  <Dollars>  
    <return>397</return>  
  </Dollars>  
</Soap : body >
```



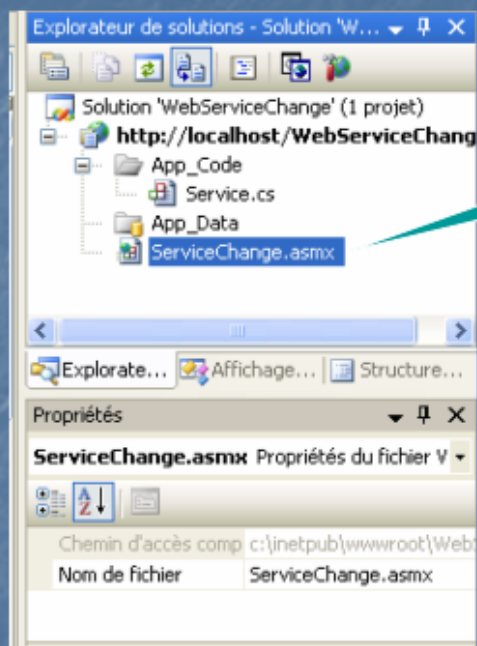
ASP.Net et webservice de change



ASP.Net et webservice de change



ASP.Net et webservice de change



Renommer
Le fichier **.asmx** du
webservice

```
Page de démarrage ServiceChange.aspx* App_Code/Ser
<%@ WebService Language="C#"
CodeBehind="~/App_Code/Service.cs"
Class="Service"
%>
```

Le contenu du
fichier **.asmx**

ASP.Net et webservice de change

```
using System;
using System.Web;
using System.Web.Services;
using System.Web.Services.Protocols;

[WebService(Namespace = "http://tempuri.org/")]
[WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
public class ServiceDeChange : System.Web.Services.WebService
{
    public ServiceDeChange () {
        //Supprimez les marques de commentaire dans la ligne sui
        //Initialia Component();
    }

    [WebMethod]
```

Le code-behind du
fichier **.asmx**

Renommer
La classe

Microsoft Visual Studio



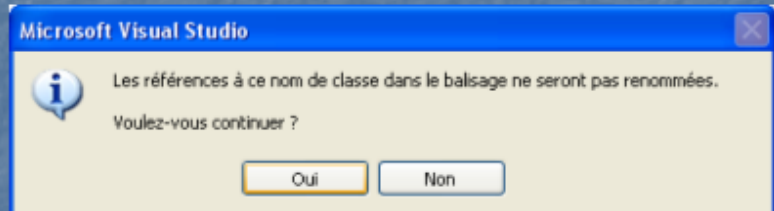
Les références à ce nom de classe dans le balisage ne seront pas renommées.
Voulez-vous continuer ?

Oui

Non

ASP.Net et webservice de change

Il faut renommer manuellement le nom de la classe dans la balise du fichier .asmx.



```
Page de démarrage ServiceChange.asmx* App_Code/
<%@ WebService Language="C#"
CodeBehind="~/App_Code/Service.cs"
Class="ServiceDeChange" %>
```

ASP.Net et webservice de change

Programmer les méthodes de conversion adéquates :

```
[WebService(Namespace = "http://tempuri.org/")]
[WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
public class ServiceDeChange : System.Web.Services.WebService
{
    public ServiceDeChange () {
    }

    private float convertEuroDollar(float somme)
    {
        return (float) (somme * 1.137);
    }

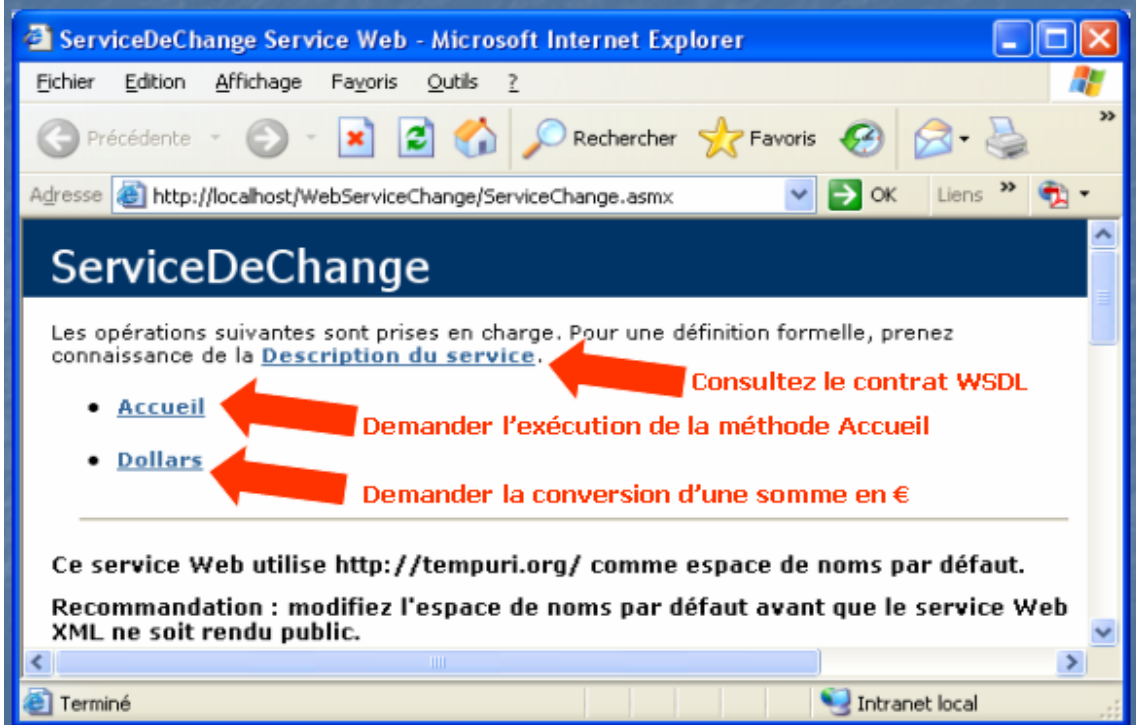
    [WebMethod]
    public string Accueil()
    {
        return "France-Change bonjour , nous sommes le :" + DateTime.Now.ToLongDate();
    }

    [WebMethod]
    public float Dollars(float euros)
    {
        float commission = 1.05f;
        return convertEuroDollar(euros) * commission;
    }
}
```

Modifier le code-
behind du fichier
.asmx

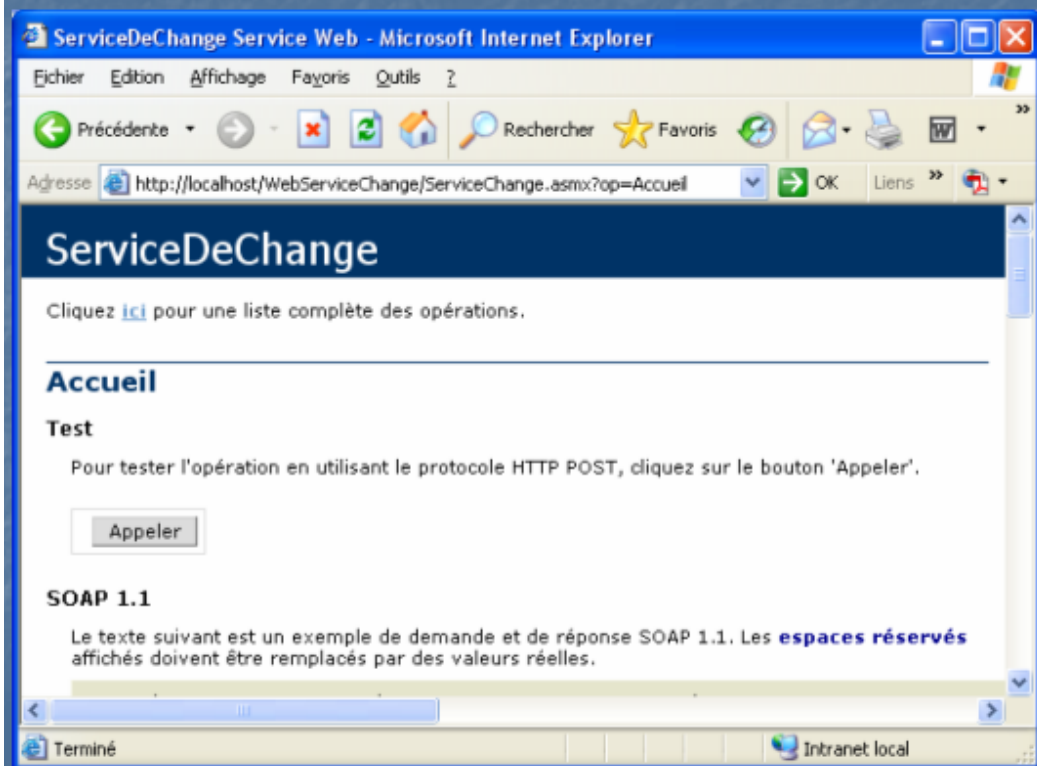
ASP.Net et webservice de change

Lancer le test du webservice :



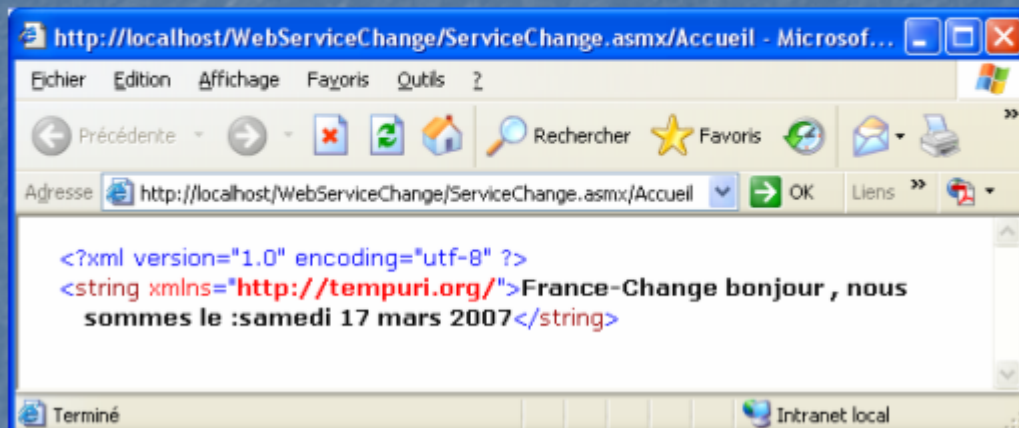
ASP.Net et webservice de change

Lancer le test de la méthode Accueil du webservice :



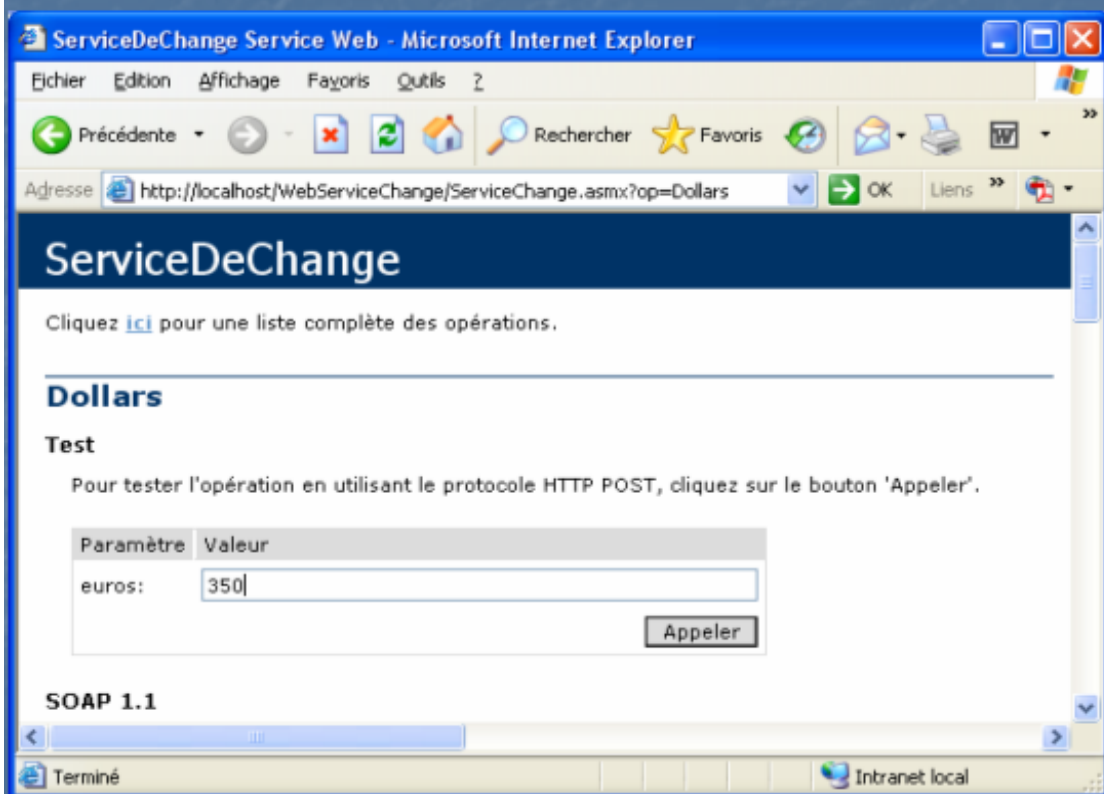
ASP.Net et webservice de change

Résultat du test de la méthode Accueil du webservice :



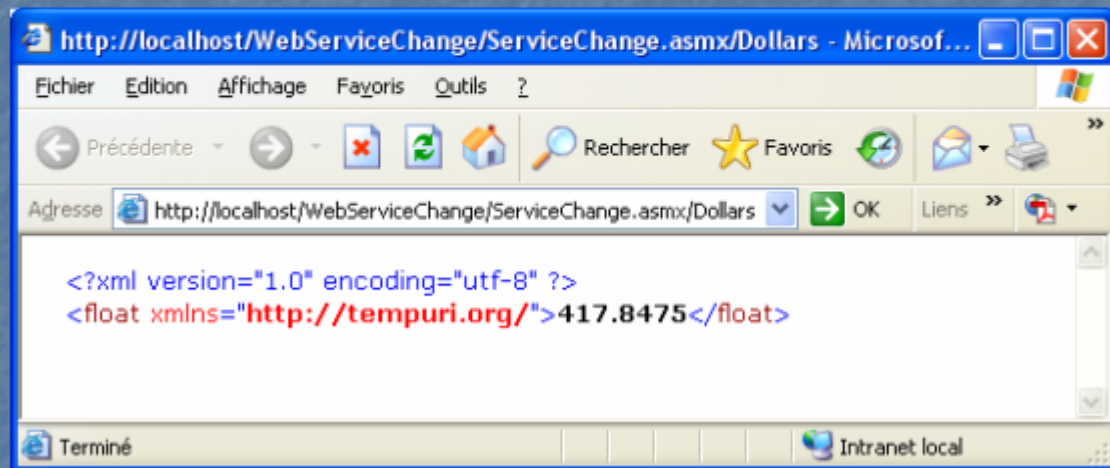
ASP.Net et webservice de change

Lancer le test de la méthode Dollars du webservice :

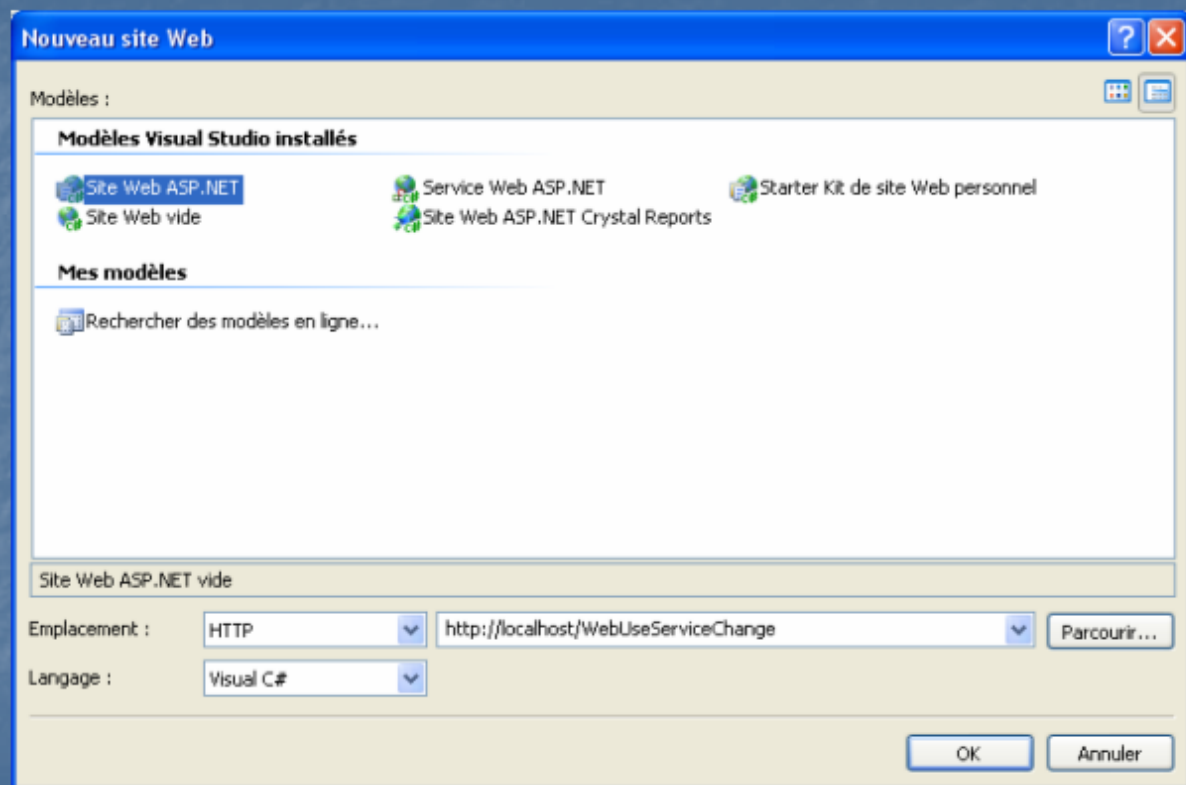


ASP.Net et webservice de change

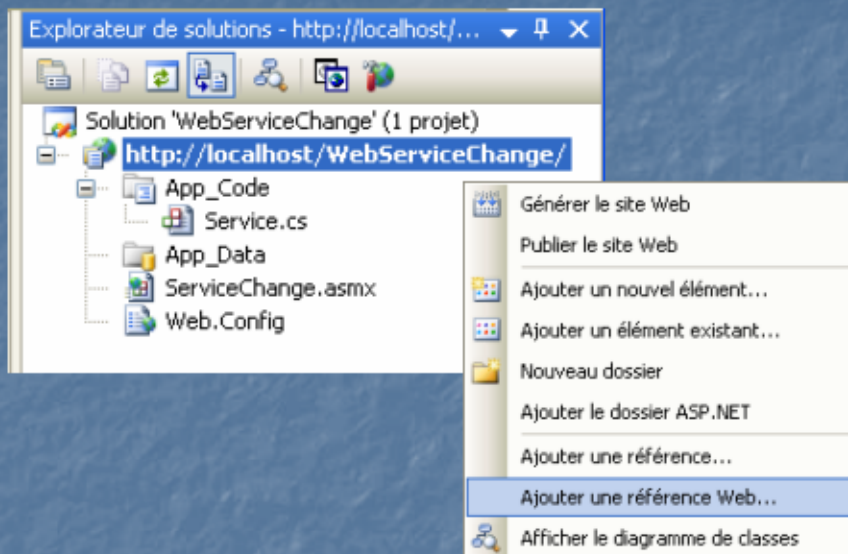
Résultat du test de la méthode Dollars du webservice :



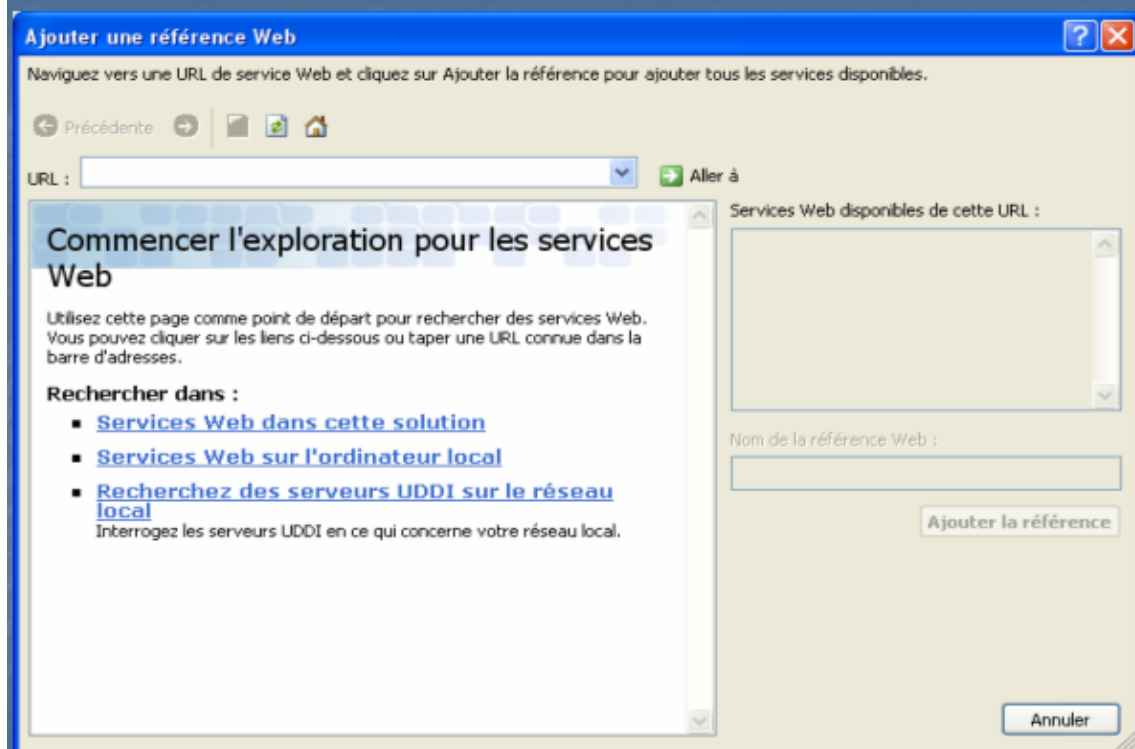
webservice de change : consommation par site web



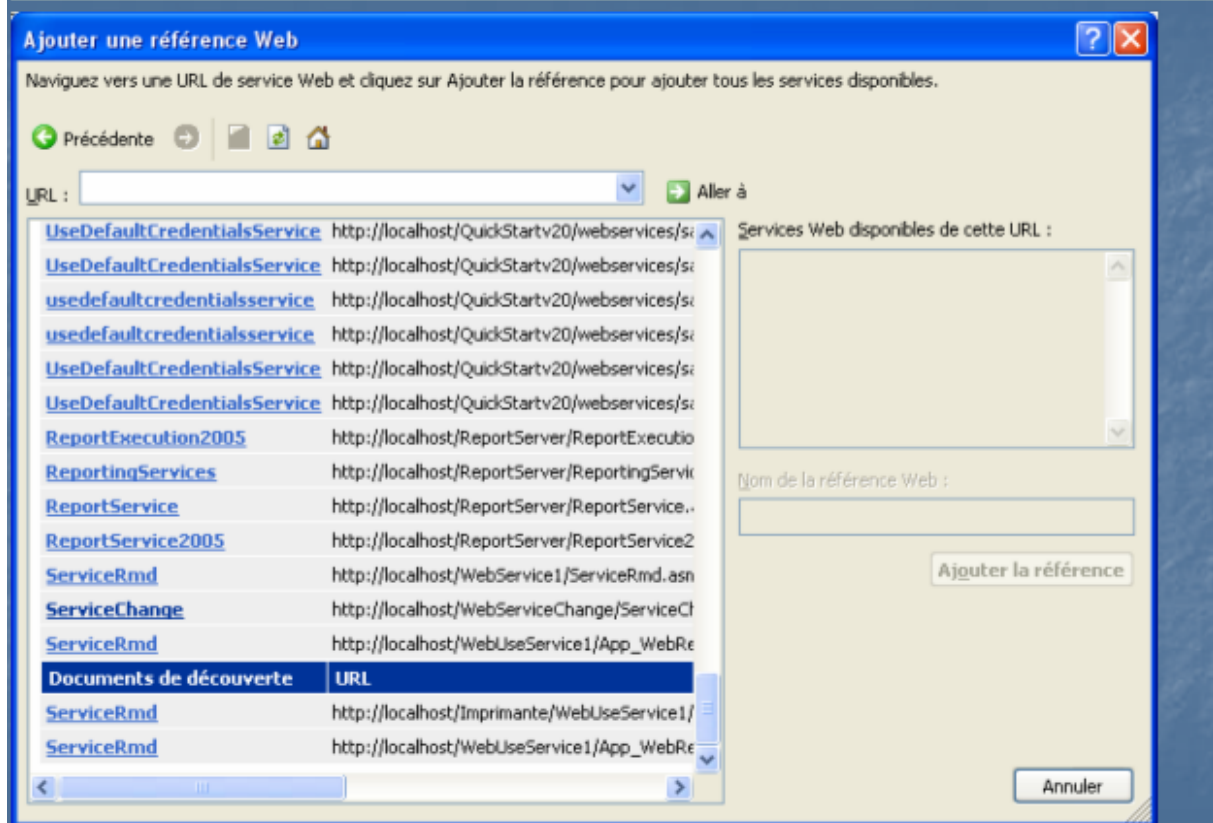
webservice de change : consommation par site web



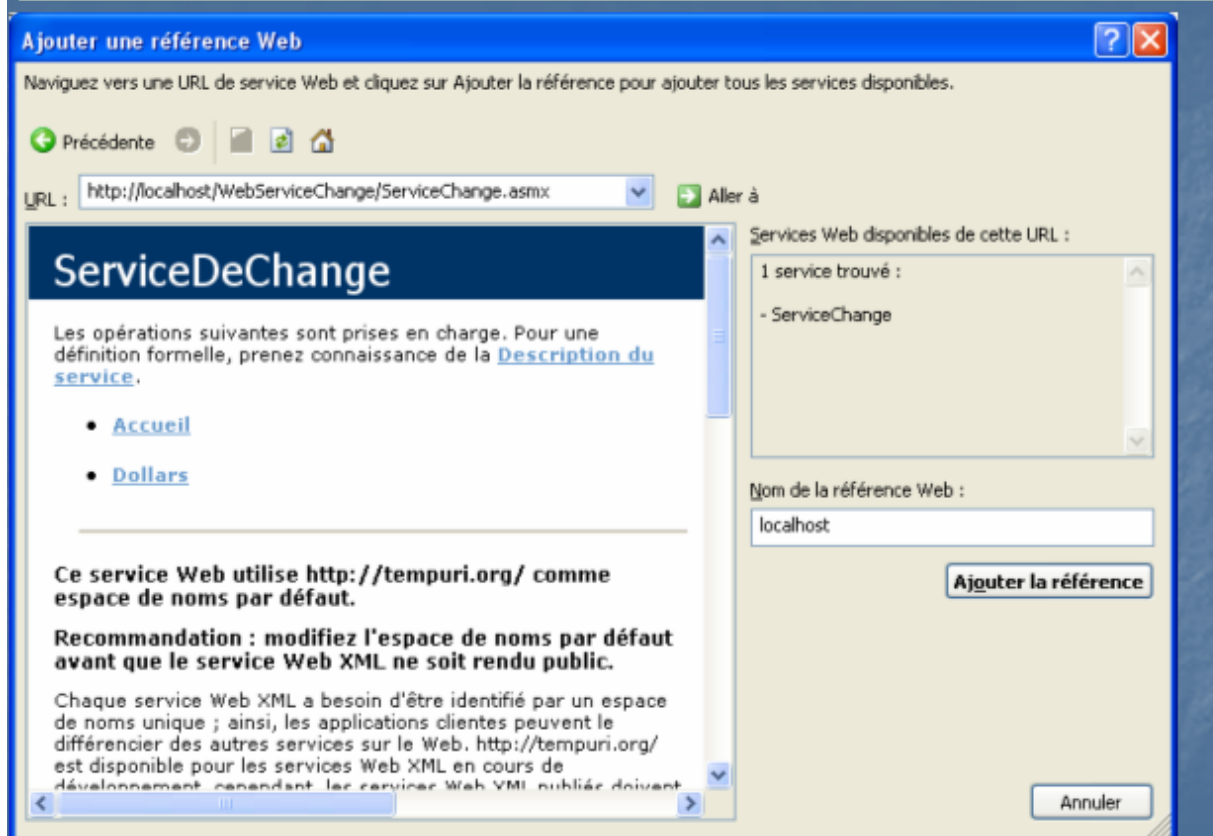
webservice de change : consommation par site web



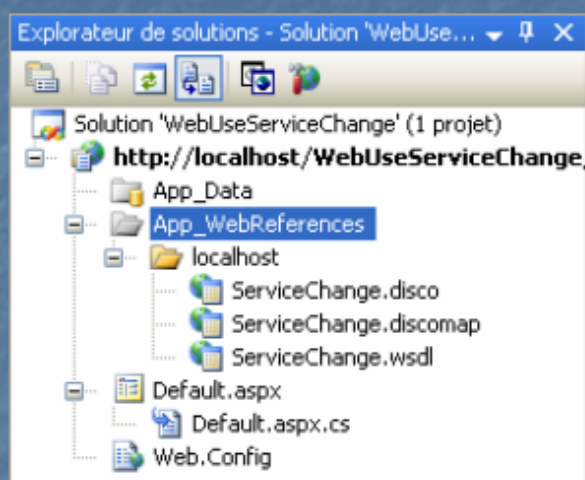
webservice de change : consommation par site web



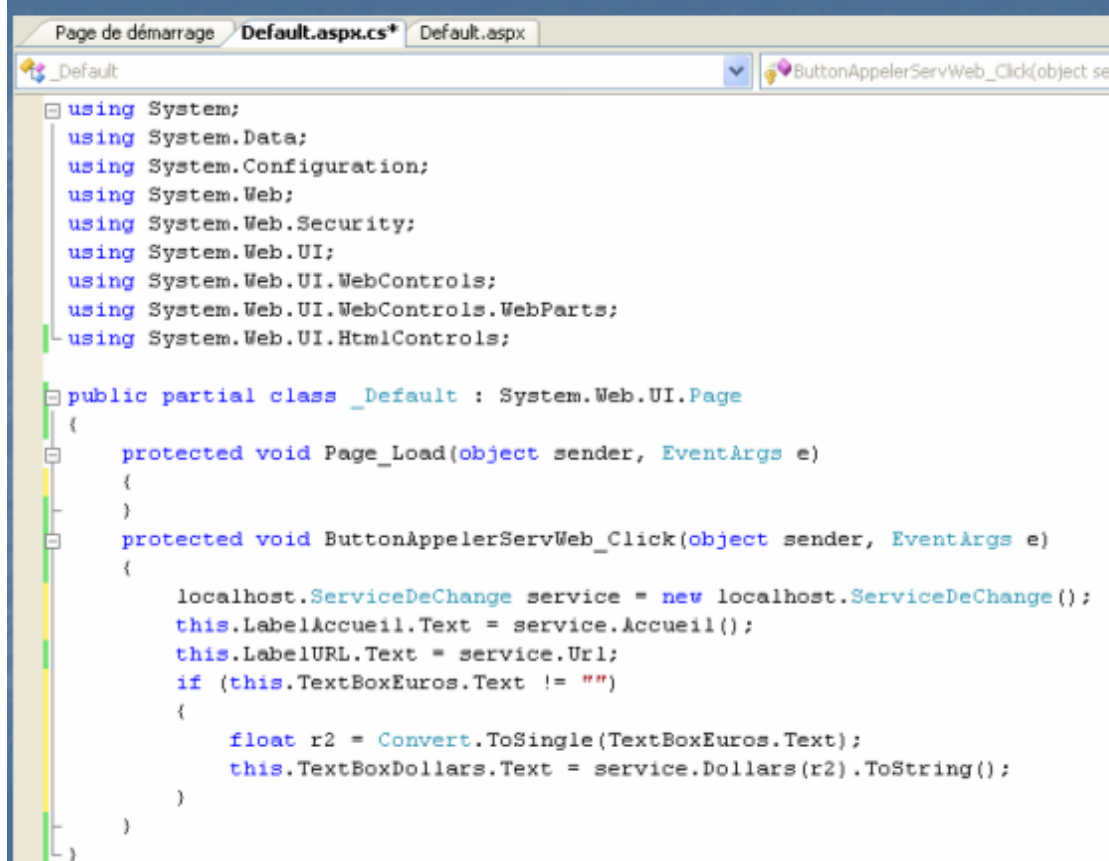
webservice de change : consommation par site web



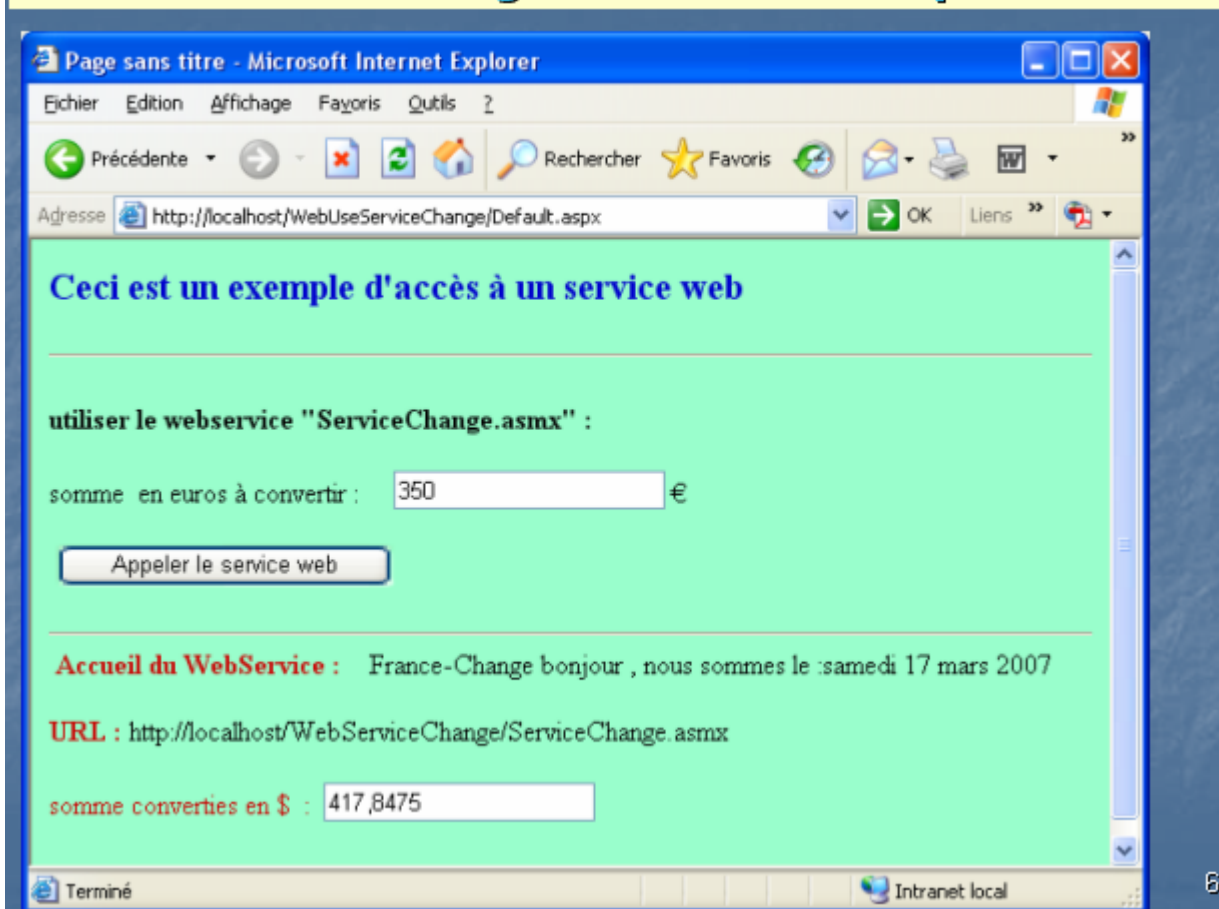
webservice de change : consommation par site web



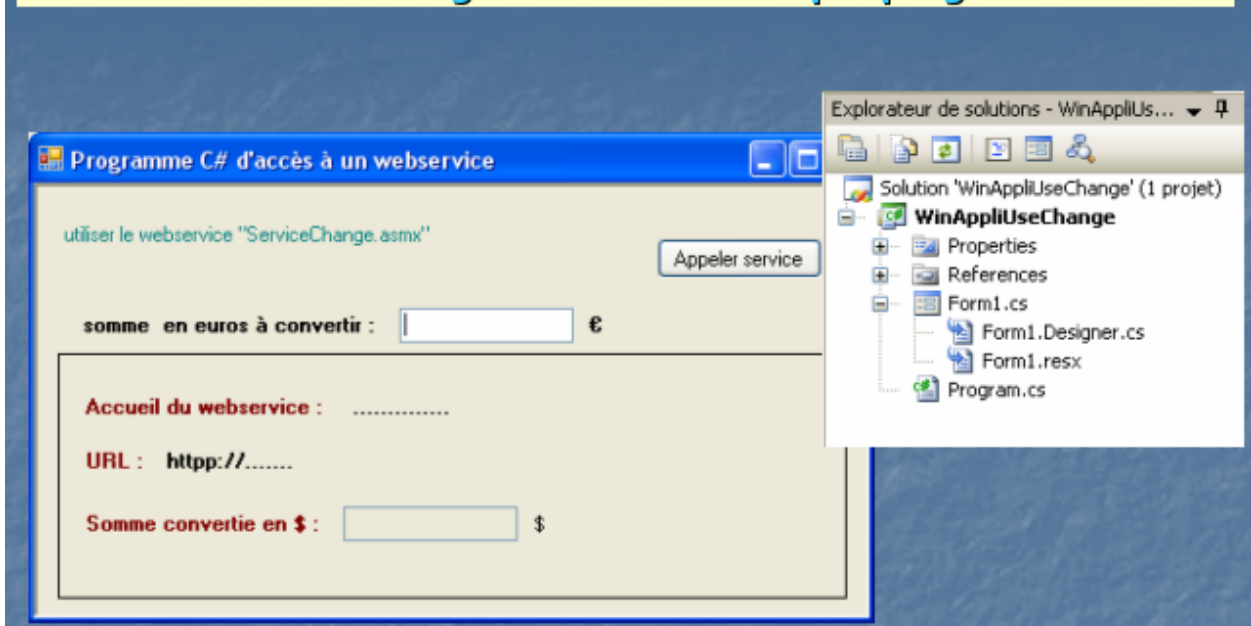
webservice de change : consommation par site web



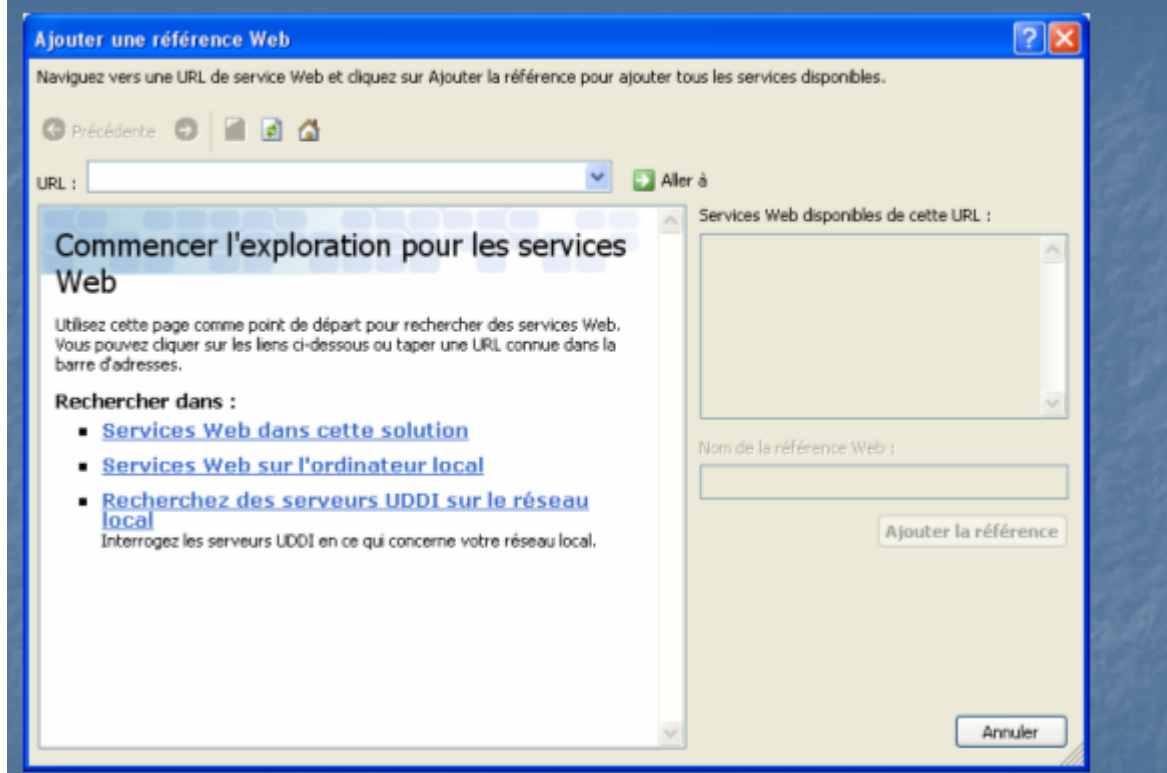
webservice de change : consommation par site web



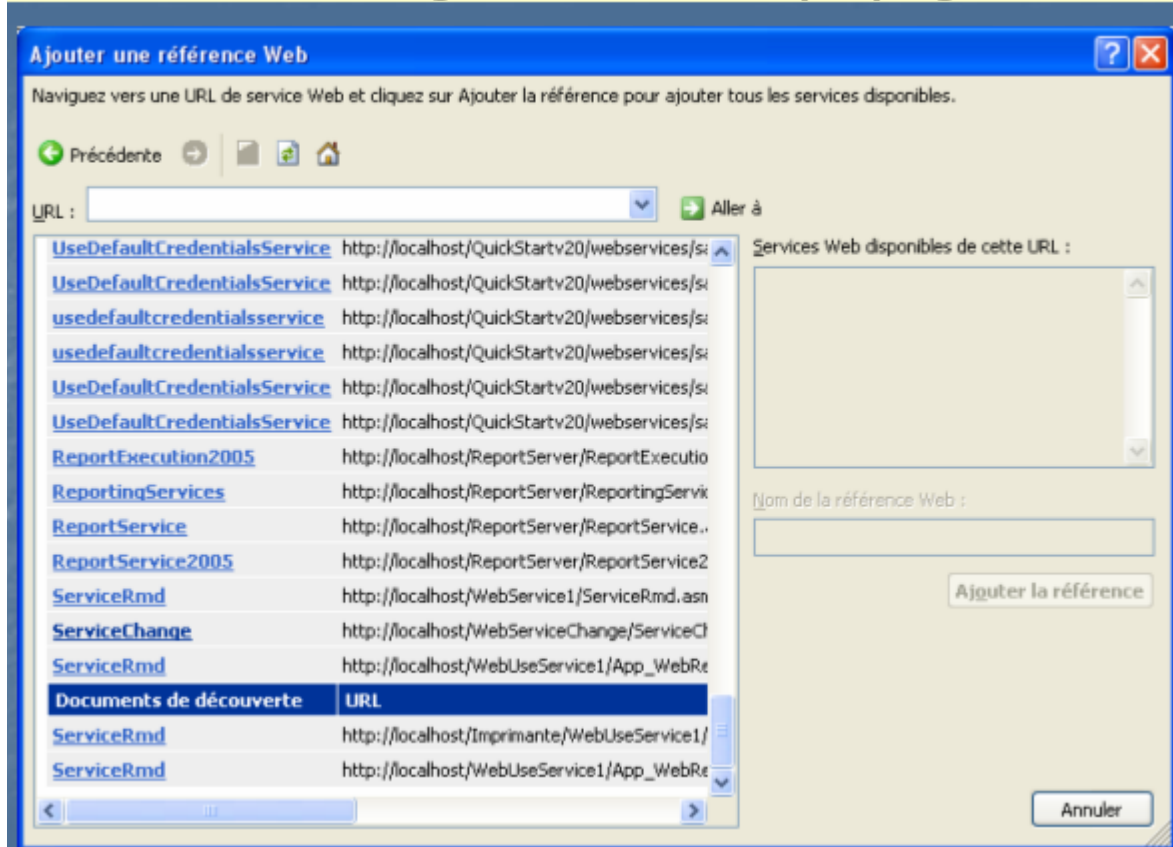
webservice de change : consommation par programme C#



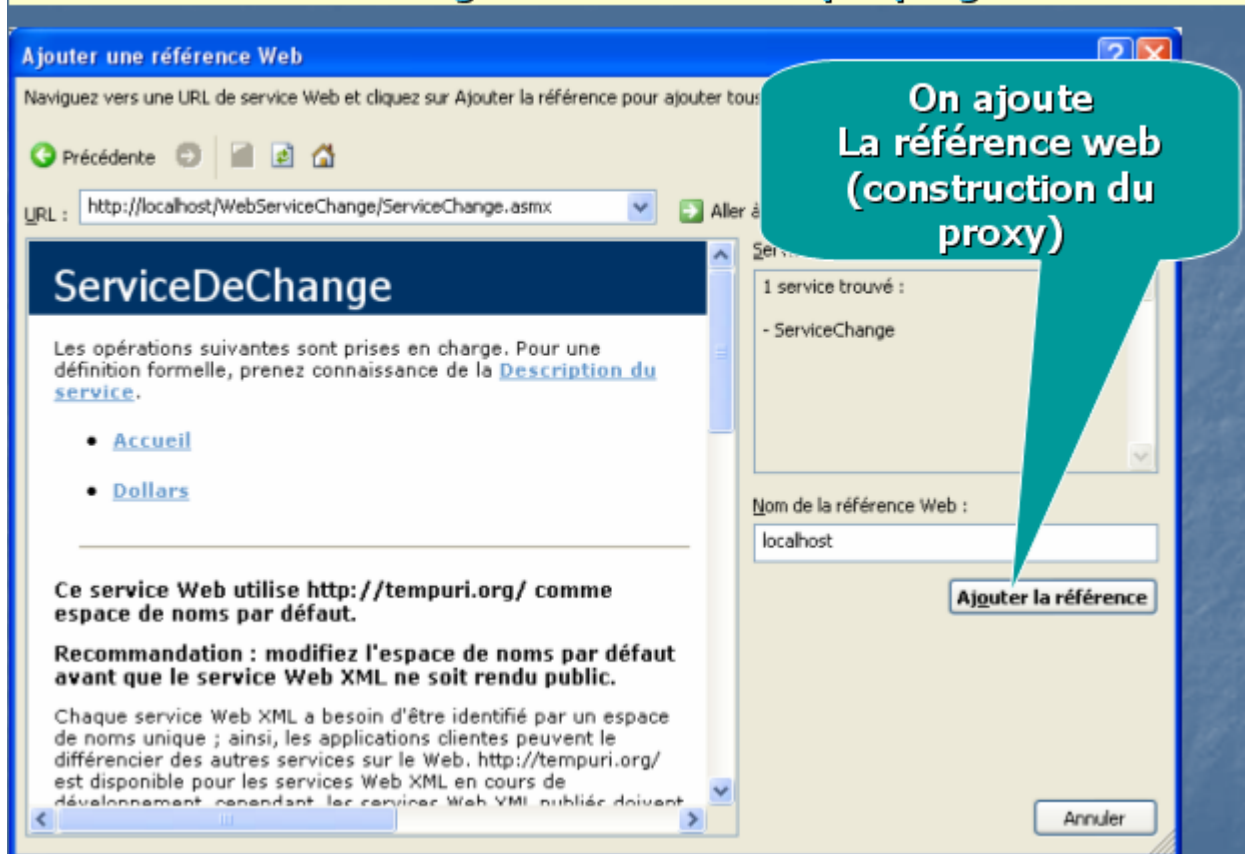
webservice de change : consommation par programme C#



webservice de change : consommation par programme C#



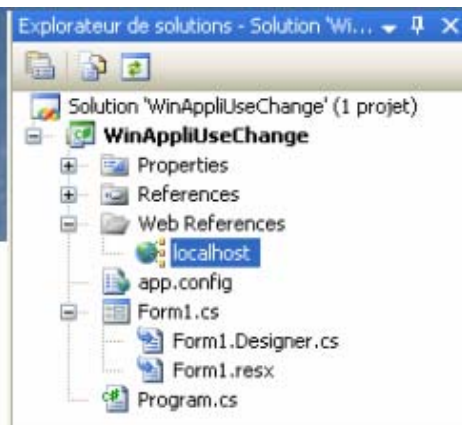
webservice de change : consommation par programme C#



La classe Form1 d'IHM du programme

```
public partial class Form1 : Form
{
    public Form1()
    {
        InitializeComponent();
    }

    private void buttonAppelerServWeb_Click(object sender, EventArgs e)
    {
        localhost.ServiceDeChange service = new localhost.ServiceDeChange();
        this.LabelAccueil.Text = service.Accueil();
        this.LabelURL.Text = service.Url;
        if (this.TextBoxEuros.Text != "")
        {
            float r2 = Convert.ToSingle(TextBoxEuros.Text);
            this.TextBoxDollars.Text = service.Dollars(r2).ToString();
        }
    }
}
```

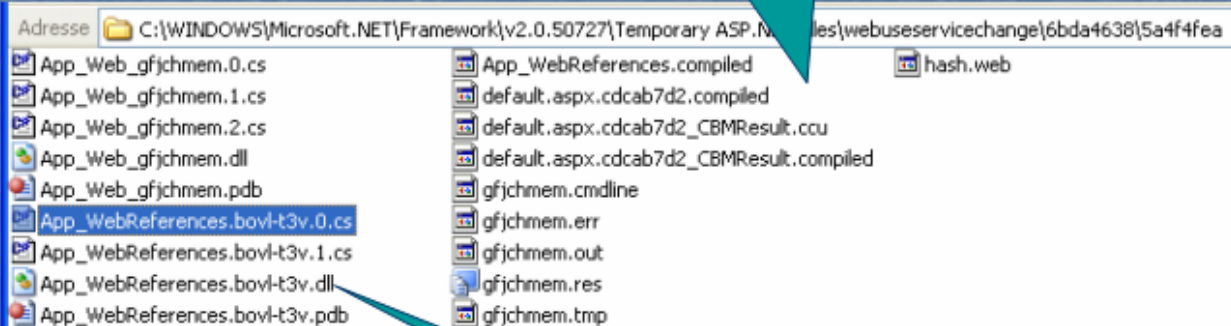


proxy

Web reference

Web reference

Fichiers de codes
temporaires engendrés
(~ servlets en Java)



Assembly final du
webservice

Web reference

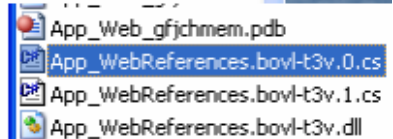
signature de la classe proxy
engendrée

```
using System;
using System.CodeDom.Compiler;
using System.ComponentModel;
using System.Diagnostics;
using System.Web.Services;
using System.Web.Services.Protocols;

namespace localhost
{
    [WebServiceBinding(Name = "ServiceDeChangeSoap", Namespace = "http://tempuri.org/")]
    [GeneratedCode("System.Web.Services", "2.0.50727.42")]
    [DebuggerStepThrough]
    [DesignerCategory("code")]
    public class ServiceDeChange : SoapHttpClientProtocol
    {
        public ServiceDeChange();

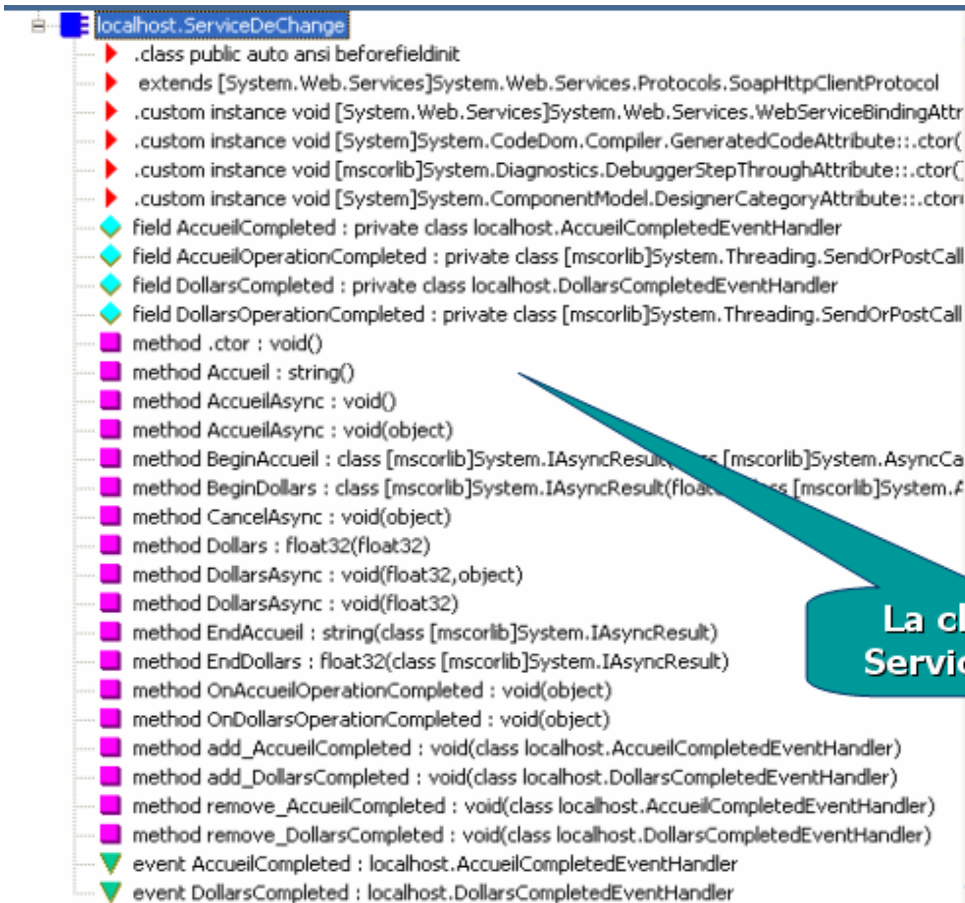
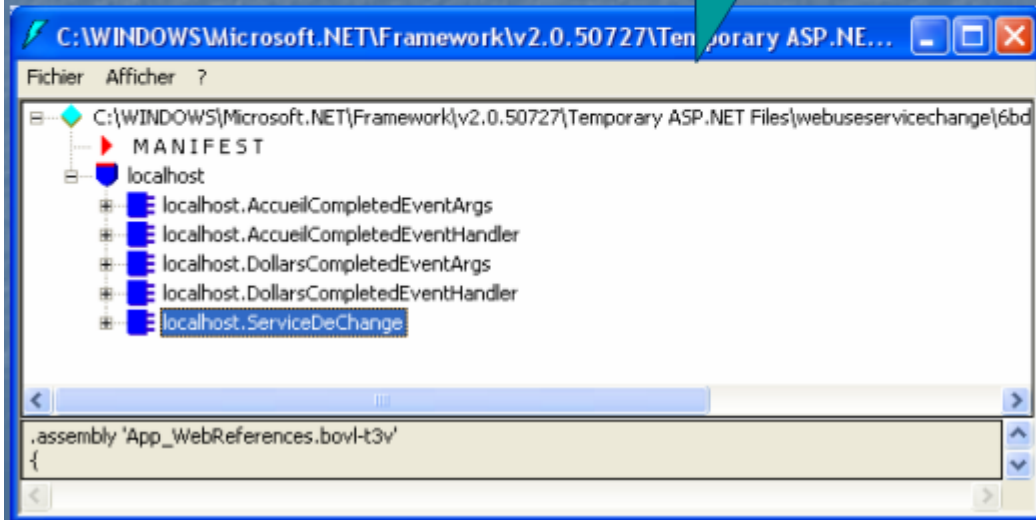
        public event AccueilCompletedEventHandler AccueilCompleted;
        public event DollarsCompletedEventHandler DollarsCompleted;

        public string Accueil();
        public void AccueilAsync();
        public void AccueilAsync(object userState);
        public IAsyncResult BeginAccueil(AsyncCallback callback, object asyncState);
        public IAsyncResult BeginDollars(float euros, AsyncCallback callback, object asyncState);
        public void CancelAsync(object userState);
        public float Dollars(float euros);
        public void DollarsAsync(float euros);
        public void DollarsAsync(float euros, object userState);
        public string EndAccueil(IAsyncResult asyncResult);
        public float EndDollars(IAsyncResult asyncResult);
    }
}
```



Inspection des métadonnées de l'Assembly final du webservice

App_WebReferences.bovl-t3v.0.cs
App_WebReferences.bovl-t3v.1.cs
App_WebReferences.bovl-t3v.dll



La classe proxy ServiceDeChange

70

Code partiel de la classe proxy engendrée

App_Web_gfjchmem.pdb
App_WebReferences.bowl-t3v.0.cs
App_WebReferences.bowl-t3v.1.cs
App_WebReferences.bowl-t3v.dll

method Dollars : float32(float32)
method DollarsAsync : void(float32,object)
method DollarsAsync : void(float32)

```
[System.Web.Services.Protocols.SoapDocumentMethodAttribute("http://tempuri.org/Dollars",  
RequestNamespace="http://tempuri.org/",  
ResponseNamespace="http://tempuri.org/",  
Use=System.Web.Services.Description.SoapBindingUse.Literal,  
ParameterStyle=System.Web.Services.Protocols.SoapParameterStyle.Wrapped)]
```

```
public float Dollars(float euros) {  
    object[] results = this.Invoke("Dollars", new object[] { euros});  
    return ((float)(results[0]));  
}  
public void DollarsAsync(float euros) {  
    this.DollarsAsync(euros, null);  
}  
public void DollarsAsync(float euros, object userState) {  
    if ((this.DollarsOperationCompleted == null)) {  
        this.DollarsOperationCompleted =  
            new System.Threading.SendOrPostCallback(this.OnDollarsOperationCompleted);  
    }  
    this.InvokeAsync("Dollars", new object[] { euros},  
        this.DollarsOperationCompleted, userState);  
}
```

Appel synchrone

délégué

71