How to call Delphi code from scripts running in a TWebBrowser (part 4 of 6)

Calling into Delphi from JavaScript

Calling into Delphi from JavaScript

Having implemented our *external* object extension and registered it with the browser control, it is now time to look at how we call the object's methods from JavaScript. A similar approach is taken if you wish to use VBScript rather than JavaScript, but the precise details are, as they say, left as an exercise!

All we have to do to call into our Delphi code from JavaScript is to reference the relevant methods of the *external* object. For example to access external method *Foo* from JavaScript we would use the code presented in *Listing 6*.

```
external.Foo();
```

Listing 6

external object methods can be called anywhere that a JavaScript method call is valid – either from event handlers of HTML elements or from blocks of JavaScript code. For example, say we have implemented the interface shown in *Listing 7* in Delphi, and have registered it as an extension of the *external* object:

```
type
  ITest = interface(IDispatch)
    ['{XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX}']
    procedure SetLabel(const Msg: WideString);
      safecall;
    function GetEMail(const Name: WideString): WideString;
      safecall;
  end;
```

Listing 7

Assume that *SetLabel* sets the caption of a *TLabel* to the given text and *GetEMail* looks up the name of the given person in a database and returns their email address.

We can ensure the label's caption is set when a user clicks a link by using HTML such as that presented in *Listing 8*.

```
<body>
  ...
  <a href="#"
    onclick="external.SetLabel('Hello Delphi');">Click me</a>
  ...
</body>
```

Listing 8

We can also write someone's email address into the HTML document using a block of JavaScript similar to that given in *Listing 9*.

```
<body>
  ...
  <p>Fred's email address is:
  <script type="text/JavaScript">
    <--
    document.write( external.GetEMail("fred") );
    //-->
  </script>
  </p>
  ...
</body>
```

Listing 9

As can be seen, once the hard work of setting up and registering the *external* object has been done, it is very easy to call the required methods.

That concludes our discussion of the techniques involved in calling Delphi code from JavaScript. In the *next section* we will work through a case study that draws all the strands together.