Make your javascript a Windows .exe

August 31st, 2007. Tagged: .net, JavaScript

These days an average web developer needs to have a broad matrix of skills in order to do his/her job. HTML, CSS, JavaScript, AJAX, XML, some server side language, some linux skills, some apache, some database skills, standards, accessibility, SEO, the list goes on. Parts of the list are also multiplied by (or raised to the power of?) the number of browsers you want to support. Crazy, isn't it? We're such optimists trying to make stuff work in such an environment.

There's gotta be an easier way to write code that does something meaningful! Yes, there is, it's called JavaScript. You learn JavaScript, you learn it well, and you don't need to learn anything else. Isn't that cool? JavaScript is, practically, everywhere. Learn JavaScript and you can:

- create rich and powerful web applications (the kind that runs inside the browser)
- write server-side code such as ASP scripts or for example code that is run using Rhino (A JavaScript engine written in Java)
- create rich media applications (Flash, Flex) using ActionScript which is based on ECMAScript, which is JavaScript
- write scripts that automate administrative tasks on your Windows desktop, using Windows Scripting Host
- write extensions/plugins for a plethora of desktop application such as Firefox or Dreamweaver
- create web applications that store information off-line on user's desktop, using Google Gears
- create Yahoo!, or Mac, or dunno-what-type-of widgets
- create Windows apps (those that end in .exe) and libraries (.dll)

I'm sure the list above is not even complete.

OK, it's a joke that with one programming skill only you'll be employed for life, but it's a fun thought anyway. Off to the main topic of the post.

JScript

This is Microsoft's version of JavaScript (yep, the thing that annoys us *sometimes* in IE) and can also be used to create server side pages (ASP, ASP.NET) or desktop applications. Apparently JScript is now called JScript.NET and can be compiled to create .exe files. Let's see how.

The compiler

The compiler (program that creates programs) is an exe file called jsc.exe (**J**Script**C**ompiler) and is part of the .NET framework. Good news is that you can use it without installing any MS IDE (whatever Visual Studio is called these days), free of charge. Even better, maybe it's already there, on your machine. I searched my completely normal Windows XP machine that doesn't have any special MS tools and was able to find two copies of the compiler! You can search for "jsc.exe" and in case you don't already have it, you can read how to get it here.

So once you find your jsc.exe (found one o' mine in C:\WINDOWS\Microsoft.NET\Framework\v2.0.50727), then add this path to your environment path: Right-click My Computer - Advanced - Environment Variables - System Variables - Path - Edit

Now open command prompt (Start - Run - "cmd" - OK) and type "jsc"+ENTER. You should see a list of help options for the compiler. Cool!

First .exe (in years)

The last time I created an .exe file is probably yeeears ago, when I was this Visual Basic king, writing a desktop application that takes a directory of images and creates a web gallery (example)

OK, lets create a simple application.

```
cd ..
mkdir myapps
cd myapps
```

Create a file called hello.js with the following content:

```
var d = new Date();
var n = Math.random();
print('Hello, \\ntoday is ' + d + '\\nand this is random - ' + n);
```

Now let's compile!

```
C:\\myapps>jsc hello.js
Microsoft (R) JScript Compiler version 8.00.50727
for Microsoft (R) .NET Framework version 2.0.50727
Copyright (C) Microsoft Corporation 1996-2005. All rights reserved.
```

No error messages, so we'll take that as an OK 😉 Let's check:

```
C:\\myapps>dir
 Volume in drive C has no label.
 Volume Serial Number is B96A-95DB

 Directory of C:\\myapps

08/31/2007  07:33 PM    <DIR>          .
08/31/2007  07:33 PM    <DIR>          ..
08/31/2007  07:34 PM             4,096 hello.exe
08/31/2007  07:33 PM               109 hello.js
               2 File(s)          4,205 bytes
               2 Dir(s)  40,287,092,736 bytes free
```

YES! An .exe was created! Without further ado, let's run it already!

```
C:\\myapps>hello
Hello,
today is Fri Aug 31 19:34:32 PDT 2007
and this is random - 0.5855108083158316
```

That's so cool, the compiled program works!

Making a DLL

Now, we're convinced that we have a good thing going here, so let's create a DLL, meaning create a library that other applications can use.

JScript.NET has the notion of namespaces and packages (which we usually fake on the web) and class-based objects (eww! well, it supports the prototype stuff as well). So if we simply wrap our code in a package and a class and we create a new file LibHello.js:

```
package LibHello {
    class Hello {
        function say() {
            var d = new Date();
            var n = Math.random();
            return 'Hello, \\ntoday is ' + d + '\\nand this is random -
        }
    }
}
```

Let's compile this into a library, we need the /t:library option when compiling

```
C:\\myapps>jsc /t:library LibHello.js
```

This creates hello.dll and we have a library!

Consuming the lib

Finally, let's create an app that leverages the new library we just created.

Create consumer.js with the following:

```
import LibHello;
var h = new LibHello.Hello();
print(h.say());
```

Compile and run:

```
C:\\myapps>jsc consumer.js

C:\\myapps>consumer.exe
Hello,
today is Fri Aug 31 19:53:29 PDT 2007
and this is random - 0.45013379838789525
```

Nice and easy.

So what?

I didn't have time to experiment, but I'm pretty sure you can take tools such as jsmin or jslint and easily compile them into libraries that can be consumed from windows apps, or VBA scripts in Access, Powerpoint, etc. Imagine you're writing some documentation in Word, you select some JS code you just wrote and JSlint it. That would be nice.

BTW, remember how we used /t:library option to produce a .dll and not an .exe? Well, there's also the option /t:winexe which creates a windows application I mean with the window and everything and not a console app. OK, let's give it a shot, create win.js with the following:

```
import System.Windows.Forms; // this has a MessageBox class
import LibHello;

var h = new LibHello.Hello();
MessageBox.Show(
        h.say(),
        "Dude!",
        MessageBoxButtons.OK,
        MessageBoxIcon.Exclamation
);
```

Compile:

```
C:\\myapps>jsc /t:winexe win.js
```

Double click in windows explorer and you have a nice little unquestionably useful Windows application 😉

Tell your friends about this post on Facebook and Twitter

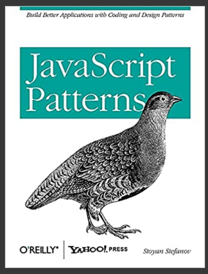
« JavaScript: Calling the unnamed

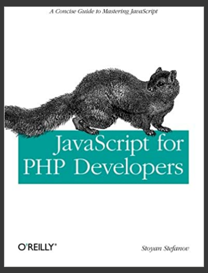
12 questions to ask your host-to-be »

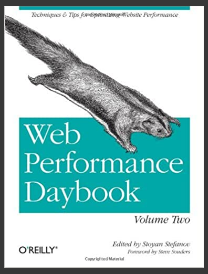
Sorry, comments disabled and hidden due to excessive spam.

Meanwhile, hit me up on twitter @stoyanstefanov

These are links to some of the books I've writ







powered by WordPress, RSS feed

Search