Building on Windows

Building the FireBreath Plugin

See Also: FireBreath Tips: Working with Source Control

See Also: Prep Scripts

Requirements

- Visual Studio 2005, 2008, 2010, 2012 or 2013 (full or express versions)
    - To build with VC++ express editions, read this: Building with Visual Studio Express
- CMake version 2.8.7 or later (2.8.11 or later for VS 2013)
    - **Make sure to grab the Binary distribution (cmake-*ver*-win32-x86.exe).**
    - Download from here: : http://www.cmake.org/cmake/resources/software.html
    - When asked, tell the installer to *add cmake to your system path*
    - **The Cygwin version of CMake does not work, but you can use the Windows version from inside Cygwin.**
- Git (if you want to check out from source): http://git-scm.com/
    - Git Extensions is a fantastic Git GUI for windows that comes with Git binaries

Get the source

First thing is first; get the source code.

To get a copy of the source, see the download page.

32 bit vs 64 bit builds

The normal prep scripts (e.g. prep2010.cmd) build 32-bit plugins - There are also prep scripts, ending in x64, which build 64-bit plugins.
Windows doesn't support universal binaries and most windows browsers are 32 bit and *can only load 32-bit plugins*.
Unless you know that you specifically need to build a 64-bit plugin, use the normal scripts.
The normal (non-64-bit) scripts work fine on 64-bit Windows, as do the resulting 32-bit plugins.

Generate the example project files

To generate the example project files, use these batch files:

- `prep2005.cmd examples` - Generate example project files for Visual Studio 2005
- `prep2008.cmd examples` - Generate example project files for Visual Studio 2008
- `prep2010.cmd examples` - Generate example project files for Visual Studio 2010
- `prep2012.cmd examples` - Generate example project files for Visual Studio 2012
- `prep2013.cmd examples` - Generate example project files for Visual Studio 2013

The project files will all be generated into the `buildex/` directory under the project root.

Generate your own project files

If you have created your own project in the projects/ directory, you can build it by running the same command as for example projects, but (this will be a shocker) without "example".

To generate the project files, these batch files can be used:

- `prep2005.cmd` - Generate project files for Visual Studio 2005
- `prep2008.cmd` - Generate project files for Visual Studio 2008
- `prep2010.cmd` - Generate project files for Visual Studio 2010
- `prep2012.cmd` - Generate project files for Visual Studio 2012
- `prep2013.cmd` - Generate project files for Visual Studio 2013

The project files will all be generated into the `build/` directory under the project root.

Open the solution

The main solution file for the example projects is `buildex/FireBreath.sln`

Build and register the DLL

The DLL for the example project will be built in: `buildex/bin/Debug/` (replace Debug with the configuration type you use to build)

Register this DLL with regsvr32.exe:

```
regsvr32 npPluginTemplate.dll
```

Note that this installs for all browsers -- not just IE.

Open in your browser and play with it

On firefox, I recommend using FireBug for javascript testing. IE 8 has its own javascript debug tools.

Open the file `buildex/projects/FireBreathWin/gen/FBControl.htm`

Most of the links are currently broken; we're working on coming up with a good test plugin and page, but we'll get back to you on it.

Use Jash or firebug (or whatever) to make calls on the plugin. For supported calls, check out `projects/TemplatePlugin/MathAPI.cpp`.

A few JS commands to try:

```
plugin().echo("Please echo back")
"Please echo back"
Echoing: Please echo back
```

```
plugin().asString(372)
"372"
```

```
plugin().valid
true
```

Attaching a debugger

To attach a debugger, set breakpoints, etc, first open the project in Visual Studio. Select Debug : Attach to Process, and then select all instances of the browser you are using. (firefox.exe will generally only have one process; iexplore.exe could have many). IE 8 is known to launch sites in different processes sometimes, so if your breakpoint doesn't fire, check to see if a new process has shown up that you need to attach to.

Once you're attached, as soon as you go to the page with the object tag loading the plugin, you should have symbols loaded (assuming it's the debug build) and breakpoints should trigger.

For more debugging tips, see the Debugging Plugins section

MSI Distributable

If you already have WiX installed on your machine, building the project files will automatically create a re-distributable MSI (Microsoft Installer) package for your plug-in. This allows for quick installation of your plug-in on other websites without the end-user needing to know how to manually register a DLL. The MSI file itself will be located in: `build/bin/Debug/` (replace Debug with the configuration type you use to build).

For more information, see WiX Installer Help

Install WiX

If you don't have WiX already, you can download the Wix Toolset for free from CodePlex.