



Building Web Service Client for CLEAN_Address using the .NET Framework

Clients communicate with Web Services using standard web protocols, using XML-encoded messages to send and receive messages between themselves. This makes it possible for client applications running on any platform to access web services as long as they use standard web protocols, and understand the XML-encoded messages. There are three protocols that web clients can use to communicate with web services namely, HTTP GET, HTTP POST, and SOAP.

The various steps that are involved in creating a Web Service Client for CLEAN_Address using C# and the .NET Framework are as follows:

- Modify the distribution CLEAN_Address.wsdl file to correct for potential boolean to string conversion issues;
- Modify the distribution CLEAN_Address.wsdl, ensure all parameters in the "<s:complexType name='Parameters'>" section are type "string";
- Generate a Proxy for the CLNVerify Web Service;
- Modify the proxy class to change the URL to point to your clnaddrws server host and port;
- Compile the Proxy as a DLL library;
- Create a Visual C# - Console Application Project;
- Develop the Client.cs class file;
- Build the Project Files;
- Run the client.



Before you begin, download the most current source code, and documentation from:

<http://ftp.runnertech.com/clnaddr/integrations/dotnetWebserviceIntegration.pdf>

<http://ftp.runnertech.com/clnaddr/integrations/dotnetWebserviceIntegration.zip>

1. Modify the distribution CLEAN_Address.wsdl file

Copy the CLEAN_Address.wsdl file found in the CLEAN_Address installation to your build location.

Edit the new file, in the <s:complexType name="Parameters"> section, change all type="s:boolean", and "s:int" to type="s:string".

2. Generate a Proxy class for the CLNVerify Web Service

The .NET SDK simplifies the process of creating Web Service clients by providing the Web Services Description Language (wsdl.exe) utility. This utility generates proxy source code for an existing Web Service, just as IDL compilers generate DCOM proxies for DCOM components. The only difference between IDL and WSDL is that, WSDL is a language that describes the interface of a software component that is XML-based.

Let us generate the c# source code "CLNBatch.cs" for the proxy to the actual web service as shown below. In the folder where your CLEAN_Address.wsdl is located, type the command as listed below. You must have the Microsoft SDK installed for your platform.

Command Prompt

```
"C:\Program Files\Microsoft SDKs\Windows\v7.0\Bin\x64\wsdl" /I:CS /protocol:SOAP CLEAN_Address.wsdl
Microsoft (R) Web Services Description Language Utility
[Microsoft (R) .NET Framework, Version 2.0.50727.3038]
Copyright (C) Microsoft Corporation. All rights reserved.
Writing file 'C:\ASP .NET Projects\clnaddrws\CLNBatch.cs'.
```

```
C:\ASP .NET Projects\clnaddrws
```



The above command creates a proxy for the CLNVerify web service from the WSDL document obtained from the CLEAN_Address distribution media. The proxy uses SOAP as its protocol to talk to the web service and is generated as a C# source file "CLNBatch.cs", a sample of which is included with this documentation for your review.

The wsdl.exe utility references the location where the WSDL can be obtained. This C# proxy source file represents the proxy class for the CLNVerify web service that clients can compile against.

3. Modify the proxy class CLNBatch.cs to accept host parameters

Generated public method

```
/// <remarks/>
public CLNBatch() {
    this.Url = "http://webservice.clean-address.com/CLNVerify";
}
```

Add new overloaded public method

```
/// <summary>Proxy for CLEAN_Address web service</summary>
/// <param name="Protocol">The protocol to use when calling web service. Defaults to "http".</param>
/// <param name="Host">The hostname or IP address of the CLEAN_Address web service. Defaults to "localhost"</param>
/// <param name="Port">Port the CLEAN_Address web service binds to. Defaults to 8097</param>
public CLNBatch(String Protocol, String Host, String Port) {
    if (String.IsNullOrEmpty(Protocol)) {
        Protocol = "http";
    }
    if (String.IsNullOrEmpty(Host)) {
        Host = "localhost";
    }
    if (String.IsNullOrEmpty(Port)) {
        Port = "8097";
    }
    this.Url = Protocol+"://"+Host+":"+Port+"/CLNVerify";
}
```

4. Compile the Proxy class as a DLL Library

We can compile the C# source file into a dynamic link library (DLL) and then add a reference to this DLL to any project you want to create. In the folder where your c# CLNBatch.cs proxy class is located, compile the proxy class as a DLL as shown below:

Command Prompt

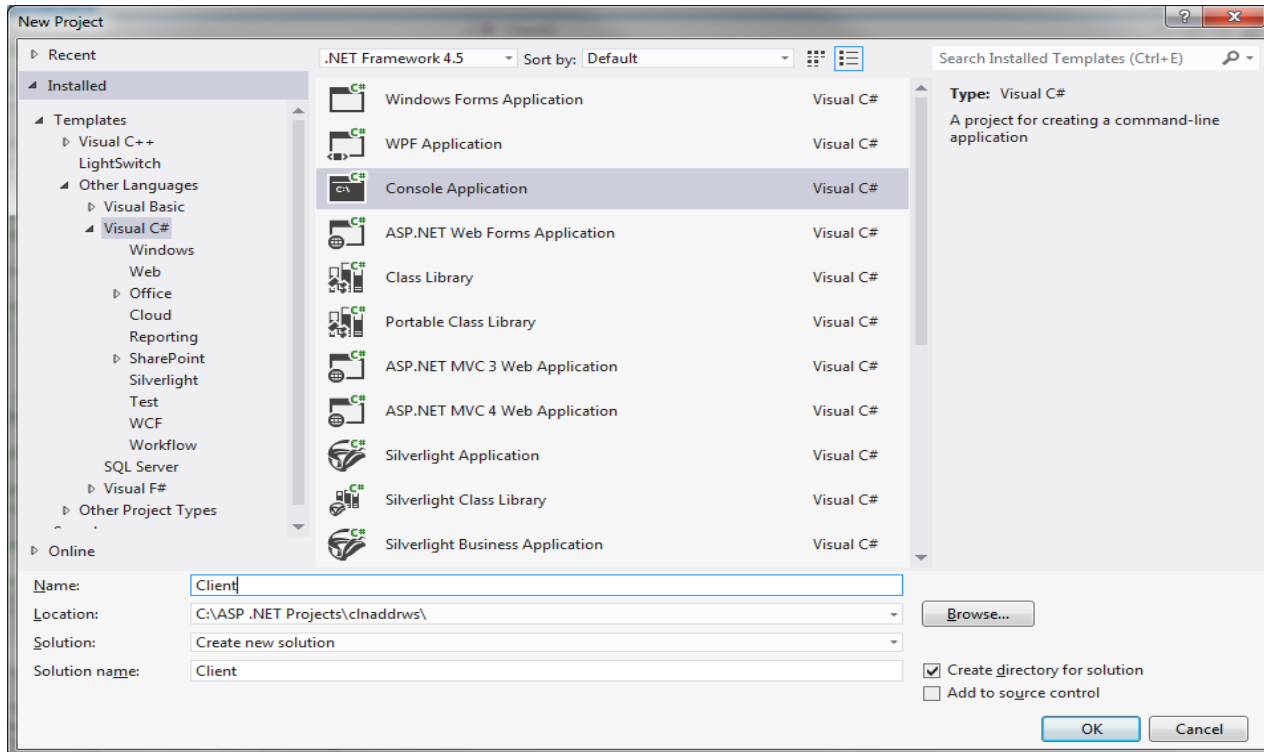
```
C:\ASP.NET Projects\clnaddrws>C:\Windows\Microsoft.NET\Framework64\v4.0.30319\csc /t:library /r:System.Web.Services.dll /r:System.Xml.dll CLNBatch.cs
Microsoft (R) Visual C# Compiler version 4.0.30319.17929
for Microsoft (R) .NET Framework 4.5
Copyright (C) Microsoft Corporation. All rights reserved.
```

```
C:\ASP.NET Projects\clnaddrws
```

5. Create a Visual C# - ASP.NET Web Service project

Create a new Visual C# ASP.NET Web Service project. Create a Reference to the CLNBatch.dll library.

To add reference, in the solution explorer right mouse click on references, select add references, and browse to the CLNBatch.dll then select ok.



6. Develop the Client.cs class file

A sample client "Program.cs" is available in the distribution for your review.

7. Build the Project Files

Build the files that make up the project.

```
1>----- Rebuild All started: Project: Client, Configuration: Debug Any CPU -----
1>Build started 5/16/2013 1:13:42 PM.
1>CoreClean:
1> Deleting file "C:\ASP .NET Projects\clnaddrws\Client\Client\obj\Debug\Client.csprojResolveAssemblyReference.cache".
1>GenerateTargetFrameworkMonikerAttribute:
1>Skipping target "GenerateTargetFrameworkMonikerAttribute" because all output files are up-to-date with respect to the input files.
1>CoreCompile:
1> C:\Windows\Microsoft.NET\Framework\v4.0.30319\Csc.exe /noconfig /nowarn:1701,1702,2008 /nostdlib+ /platform:anycpu32bitpreferred /errorreport:prompt /warn:4
/define:DEBUG;TRACE /errorendlocation /preferreduilang:en-US /highentropyva+ /reference:"C:\ASP .NET Projects\clnaddrws\CLNBatch.dll" /reference:"C:\Program Files (x86)\Reference
Assemblies\Microsoft\Framework\.NETFramework\v4.5\Microsoft.CSharp.dll" /reference:"C:\Program Files (x86)\Reference
Assemblies\Microsoft\Framework\.NETFramework\v4.5\mscorlib.dll" /reference:"C:\Program Files (x86)\Reference
Assemblies\Microsoft\Framework\.NETFramework\v4.5\System.Core.dll" /reference:"C:\Program Files (x86)\Reference
Assemblies\Microsoft\Framework\.NETFramework\v4.5\System.Data.DataSetExtensions.dll" /reference:"C:\Program Files (x86)\Reference
Assemblies\Microsoft\Framework\.NETFramework\v4.5\System.Data.dll" /reference:"C:\Program Files (x86)\Reference Assemblies\Microsoft\Framework\.NETFramework\v4.5\System.dll"
```



```
/reference:"C:\Program Files (x86)\Reference Assemblies\Microsoft\Framework\NETFramework\v4.5\System.Web.Services.dll" /reference:"C:\Program Files (x86)\Reference
Assemblies\Microsoft\Framework\NETFramework\v4.5\System.Xml.dll" /reference:"C:\Program Files (x86)\Reference
Assemblies\Microsoft\Framework\NETFramework\v4.5\System.Xml.Linq.dll" /debug+ /debug:full /filealign:512 /optimize- /out:obj\Debug\Client.exe /subsystemversion:6.00 /target:exe
/utf8output Program.cs Properties\AssemblyInfo.cs "C:\Users\ablake\AppData\Local\Temp\NETFramework,Version=v4.5.AssemblyAttributes.cs"
obj\Debug\TemporaryGeneratedFile_E7A71F73-0F8D-4B9B-B56E-8E70B10BC5D3.cs obj\Debug\TemporaryGeneratedFile_036C0B5B-1481-4323-8D20-8F5ADC823D92.cs
obj\Debug\TemporaryGeneratedFile_5937a670-0e60-4077-877b-f7221da3dda1.cs
1>_CopyFilesMarkedCopyLocal:
1> Copying file from "C:\ASP.NET Projects\clnaddrws\CLNBatch.dll" to "bin\Debug\CLNBatch.dll".
1>_CopyAppConfigFile:
1> Copying file from "App.config" to "bin\Debug\Client.exe.config".
1>CopyFilesToOutputDirectory:
1> Copying file from "obj\Debug\Client.exe" to "bin\Debug\Client.exe".
1> Client -> C:\ASP.NET Projects\clnaddrws\Client\Client\bin\Debug\Client.exe
1> Copying file from "obj\Debug\Client.pdb" to "bin\Debug\Client.pdb".
1>
1>Build succeeded.
1>
1>Time Elapsed 00:00:00.18
===== Rebuild All: 1 succeeded, 0 failed, 0 skipped =====
```

8. Run the Client

Command Prompt

```
C:\ASP.NET Projects\clnaddrws\Client\Client\bin\Debug>client
```

Proxy Request Return Code: 0

Address Line 1: 6530 W Rogers Circle

Address Line 2:

Address Line 3:

City: Boca Raton

County: Palm Beach

State: FL

Zip/Postal Code: 33487-2753

Country: United States

Error Message:

Error String: Suite/Apt Missing

DPV Status: N1

Suggestion List: Company: Number: Pre Dir: Street: Suffix: Post Dir: Suite: ;6530:W:Rogers: Cir::;6530:W:Rogers: Cir::Ste [28-33]