



Create RESTful WCF Service API: Step By Step Guide

Licence [CPOL](#)
First Posted **30 Aug 2010**
Views **143,485**
Downloads **7,304**
Bookmarked **148 times**

See Also

- [More like this](#)
- [More by this author](#)

By [ashutosh k. shukla](#) | 30 Aug 2010 | [Article](#)

[.NET3.5](#) [C#](#) [WCF](#) [C#4.0](#) [.NET4](#)

Step by Step Guide to create Restful WCF service API in ASP.NET and C#

[Article](#) [Browse Code](#) [Stats](#) [Revisions \(4\)](#) [Alternatives](#)  4.93 (74 votes)



 [Download source - 9.29 KB](#)

Introduction

Windows Communication Foundation (WCF) is an SDK for developing and deploying services on Windows. WCF provides a runtime environment for your services, enabling you to expose CLR types as services, and to consume other services as CLR types. In this article, I am going to explain how to implement restful service API using WCF 4.0 . The Created API returns XML and JSON data using WCF attributes.

What is REST?

Based on the Roy Fielding theory "Representational State Transfer (REST), attempts to codify the architectural style and design constraints that make the Web what it is. REST emphasizes things like separation of concerns and layers, statelessness, and caching, which are common in many distributed architectures because of the benefits they provide. These benefits include interoperability, independent evolution, interception, improved scalability, efficiency, and overall performance."

Actually only the difference is how clients access our service. Normally, a WCF service will use SOAP, but if you build a REST service, clients will be accessing your service with a different architectural style (calls, serialization like JSON, etc.).

REST uses some common HTTP methods to insert/delete/update/retrieve information which is below:

1. **GET** - Requests a specific representation of a resource
2. **PUT** - Creates or updates a resource with the supplied representation
3. **DELETE** - Deletes the specified resource
4. **POST** - Submits data to be processed by the identified resource

Why and Where to Use REST?

Few days back, I was writing a service which was supposed to be accessed by heterogeneous language/platform/system. It can be used by iPhone, Android, Windows Mobile, .NET web application, JAVA or PHP. Using web service, it was bit complex for me to expose it to everyone using uniform system. Then we decided to use REST, which was easily espoused over cloud. This was a great example which shows the capability of **SIMPLE RESTful SERVICE** :). Below are some points which will help you to understand why to use the RESTful services.

1. Less overhead (no SOAP envelope to wrap every call in)
2. Less duplication (HTTP already represents operations like **DELETE**, **PUT**, **GET**, etc. that have to otherwise be represented in a SOAP envelope).
3. More standardized - HTTP operations are well understood and operate consistently. Some SOAP implementations can get finicky.
4. More human readable and testable (harder to test SOAP with just a browser).

Hot News: [Windows 8 review: Yes, it's that bad](#)
[The Code Project Insider](#). [Free](#) each morning.

Related Articles

- [CREATE RESTful WCF Service API Using POST: Step By Step Guide](#)
- [4 Simple Steps to Consume WCF Service using Silverlight](#)
- [Step by Step Guide to Delicious OAuth API](#)
- [First Step to REST via WCF](#)
- [6 Steps to Enable Transactions in WCF](#)
- [Create a REST service with WCF and consume it using jQuery](#)
- [Basic Authentication on a WCF REST Service](#)
- [A Step by Step WCF Small Program For Beginners](#)
- [Restful Crud Operation on a WCF Service](#)
- [Create REST Service using WCF](#)
- [Eight steps to enable Windows authentication on WCF BasicHttpBinding](#)
- [Step by Step SharePoint Server 2010 Installation Guide](#)
- [Nine simple steps to enable X.509 certificates on WCF](#)
- [WPF/Silverlight: Step By Step Guide to MVVM](#)
- [Digest Authentication on a WCF REST Service](#)
- [Step by Step Guide to Silverlight 4 Command Binding](#)
- [Consuming a WCF / ASMX / REST Service using jQuery](#)
- [Step By Step Guide to Factory Pattern Client Activated Object \(CAO\) Creation](#)
- [RESTful Web Services: A Quick-Start How-to Guide - Part 1](#)
- [4 Steps to Enable Instrumentation in WCF](#)

5. Don't need to use XML (well, you kind of don't have to for SOAP either but it hardly makes sense since you're already doing parsing of the envelope).
6. Libraries have made SOAP (kind of) easy. But you are abstracting away a lot of redundancy underneath as I have noted. Yes, in theory, SOAP can go over other transports so as to avoid riding atop a layer doing similar things, but in reality just about all SOAP work you'll ever do is over HTTP.

Step by Step Guide

Generally, a developer is scared to use WCF because of a lot of confusing configuration. I will try to use minimum configuration so that it will be easier to understand for us. We will develop **Restful WCS API** in 6 steps. So let's start now.

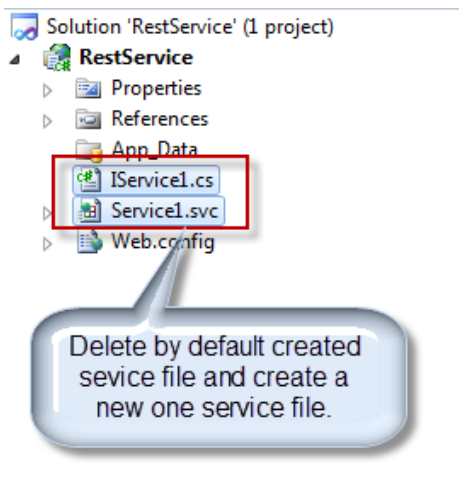
STEP 1

First of all launch Visual Studio 2010. Click **FILE->NEW->PROJECT**. Create new "**WCF Service Application**".



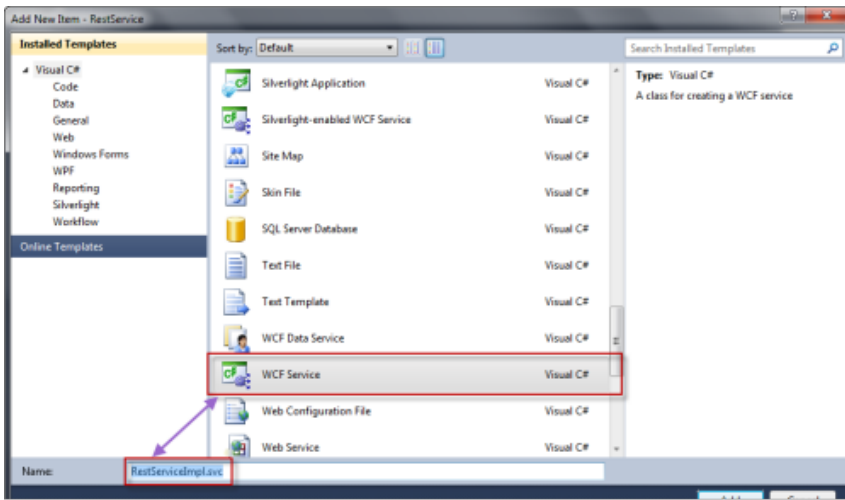
STEP 2

Once you create the project, you can see in solution that By Default WCF service and interface file are already created. Delete By default created file as we will create our own interface and WCF service file.



STEP 3

Now right click on solution and create one new WCF service file. I have given name to the service file as "**RestServiceImpl.svc**".



STEP 4

As I explained at the start of the article that we will be writing an API which can return data in XML and JSON format, here is the interface for that. In `IRestServiceImpl`, add the following code:

```
[ServiceContract]
public interface IRestServiceImpl
{
    [OperationContract]
    [WebInvoke(Method = "GET",
        ResponseFormat = WebMessageFormat.Xml,
        BodyStyle = WebMessageBodyStyle.Wrapped,
        UriTemplate = "xml/{id}")]
    string XMLData(string id);

    [OperationContract]
    [WebInvoke(Method = "GET",
```

Return data in XML format.

A UriTemplate is composed of two parts: a URL path and a query.

In the above code, you can see two different methods of `IRestService` which are `XMLData` and `JSONData`. `XMLData` returns result in XML whereas `JSONData` in JSON.

STEP 5

Open the file `RestServiceImpl.svc.cs` and write the following code over there:

```
public class RestServiceImpl : IRestServiceImpl
{
    #region IRestService Members

    public string XMLData(string id)
    {
        return "You requested product " + id;
    }

    public string JSONData(string id)
    {
        return "You requested product " + id;
    }

    #endregion
}
```

STEP 6

Now let's move to configuration part which is the last one. There will be two basic parts of the configurations file which we must have to understand.

[Collapse](#) | [Copy Code](#)

```
<services>
```

This part contains information about the End Point. Below are the code details.

```
<system.serviceModel>
  <services>
    <service name="RestService.RestServiceImpl" behaviorConfiguration="ServiceBehaviour">
      <!-- Service Endpoints -->
      <!-- Unless fully qualified, address is relative to base address supplied above -->
      <endpoint address="" binding="webHttpBinding" contract="RestService.IRestServiceImpl" behaviorConfiguration="web">
      <!--
      Upon deployment, the following identity element should be removed or replaced to reflect the
      identity under which the deployed service runs. If removed, WCF will infer an appropriate identity
      automatically.
    </--
  </services>
```

[Collapse](#) | [Copy Code](#)

```
<behaviors>
```

This part contains details about service and endpoint behavior.

```
<behaviors>
  <serviceBehaviors>
    <behavior name="ServiceBehaviour">
      <!-- To avoid disclosing metadata information, set the value below to false and remove the metadata:
      <serviceMetadata httpGetEnabled="true"/>
      <!-- To receive exception details in faults for debugging purposes, set the value below to true. S
      <serviceDebug includeExceptionDetailInFaults="false"/>
    </behavior>
  </serviceBehaviors>
  <endpointBehaviors>
    <behavior name="web">
      <webHttp/>
    </behavior>
  </endpointBehaviors>
</behaviors>
```

And that's it. Our Restful WCF service is ready for test purposes.

Service Ready to Test Now

Now I launch the application in the browser to see the result. I launch this service in Internet Explorer and my URL is now <http://localhost:35798/RestServiceImpl.svc>. Now if I use <http://localhost:35798/RestServiceImpl.svc/xml/123> URL, I get the following response on the browser which is an XML format and that was my task to achieve.



Now if I use <http://localhost:35798/RestServiceImpl.svc/json/123> URL, I get the following response on the browser which is an XML format and that was my task to achieve.

Hope the article is useful for the community. Comments, suggestions and criticisms are all welcome.

History

- 30th August, 2010: Initial post

License

This article, along with any associated source code and files, is licensed under [The Code Project Open License \(CPOL\)](#)

About the Author

ashutosh k. shukla

A versatile person having 6.0 Years of IT Industry experience in software development on .NET Technology presently leading the team successfully.

Software Developer
(Senior)
<http://mr-ashu.blogspot.com/>
Singapore

Member

 [Follow on Twitter](#)

[Article Top](#)

[Sign Up](#) to vote *Poor* *Excellent*

Comments and Discussions

You must [Sign In](#) to use this message board.

Search this forum

Profile popups Noise Layout Per page

[Refresh](#)

[First](#) [Prev](#) [Next](#)

wonderful! I give you a 5!

 [Janilane](#)

21hrs 21mins ago

Thanks for sharing your knowledge, this really helped when the net is so full of explanation but not much How-Tos.

[Sign In](#) · [View Thread](#) · [Permalink](#)

Re: wonderful! I give you a 5!

 [ashutosh k. shukla](#)

20hrs 58mins ago

Thanks. I am glad that my article is useful for you.

[Sign In](#) · [View Thread](#) · [Permalink](#)

Great write up ,How to add security?

 [Member 9300316](#)

9:18 13 Aug '12

Thanks for your great tutorial , How can I add security in to this and host this on IIS?

[Sign In](#) · [View Thread](#) · [Permalink](#)

Re: Great write up ,How to add security?



ashutosh k. shukla

16:11 15 Aug '12

Actually i have not added security there in the sample example. But you can add this easily as WCF supports multiple security modes. You can get those details over here : <http://msdn.microsoft.com/en-us/library/ff405740.aspx> [^]

Thanks
Ashu

[Sign In](#) · [View Thread](#) · [Permalink](#)

Simple & great



PREMSONBABY

22:34 9 Aug '12

Really appreciate the effort. A sample code should be like this fully backed ready to eat cake.

Good work.

Regards,
Premson.

[Sign In](#) · [View Thread](#) · [Permalink](#)

Re: Simple & great



ashutosh k. shukla

23:23 9 Aug '12

Thanks

[Sign In](#) · [View Thread](#) · [Permalink](#)

Web.config



Strascina

14:26 8 Aug '12

I've been following these steps (outlined in <http://www.codeproject.com/Articles/105273/Create-RESTful-WCF-Service-API-Step-By-Step-Guide>) and got my service to work well. I've been trying to implement it into an existing ASP.net solution and have been unable to get everything to work. This "new" webservice works but my other applications don't. I'm thinking maybe something is wrong in my solution web.config file (that's the error I'm getting, anyway). I've put the contents of the web.config stuff you've outlined (the whole piece) in the file. Could there be another attribute I'm missing that gets a webservice to work with non-webservice stuff?

Any help you could provide would be great.

[Sign In](#) · [View Thread](#) · [Permalink](#)

Re: Web.config



ashutosh k. shukla

16:09 8 Aug '12

There may be many reason behind this issue. Can u please explain a bit more with example? I will try to solve this.

[Sign In](#) · [View Thread](#) · [Permalink](#)

Re: Web.config



Strascina

4:26 9 Aug '12

Thanks for the speedy reply - I'm in Colorado (the time is 8:25 am on 8/9/2012).

I have a asp.net (c#) application I built using the 3.5 framework that does some DB reads/writes and does a lot of HTML stuff. I built this REST application as a stand alone item (for development and testing) and now want to bring it into my other app. I first changed the target framework on the main app to 4.0. Then I brought in the REST project and added the service model references. I modified the solution web.config - added the service model stuff from the REST app. It compiles successfully but my main app will not fire, but the REST app does.

[Sign In](#) · [View Thread](#) · [Permalink](#)

Re: Web.config



ashutosh k. shukla

5:05 9 Aug '12

I think this code will only work for framework 4.0. It will not work on 3.5 .

[Sign In](#) · [View Thread](#) · [Permalink](#)

