

Whats the difference between WCF and Web Services?

In this article I will explain the Difference between ASP.NET web service and WCF services like ASP.NET web services. I will also discuss how we use the both technologies for developing the web services.

Web Service in ASP.net:

A Web Service is programmable application logic accessible via standard Web protocols. One of these Web protocols is the Simple Object Access Protocol (SOAP). SOAP is a W3C submitted note (as of May 2000) that uses standards based technologies (XML for data description and HTTP for transport) to encode and transmit application data.

Consumers of a Web Service do not need to know anything about the platform, object model, or programming language used to implement the service; they only need to understand how to send and receive SOAP messages (HTTP and XML).

WCF Service:

Windows Communication Foundation (WCF) is a framework for building service-oriented applications. Using WCF, you can send data as asynchronous messages from one service endpoint to another. A service endpoint can be part of a continuously available service hosted by IIS, or it can be a service hosted in an application. An endpoint can be a client of a service that requests data from a service endpoint. The messages can be as simple as a single character or word sent as XML, or as complex as a stream of binary data.

In what scenarios WCF must be used:

- A secure service to process business transactions.
- A service that supplies current data to others, such as a traffic report or other monitoring service.
- A chat service that allows two people to communicate or exchange data in real time.
- A dashboard application that polls one or more services for data and presents it in a logical presentation.
- Exposing a workflow implemented using Windows Workflow Foundation as a WCF service.
- A Silverlight application to poll a service for the latest data feeds.

Features of WCF :

- Service Orientation
- Interoperability

- Multiple Message Patterns
- Service Metadata
- Data Contracts
- Security
- Multiple Transports and Encodings
- Reliable and Queued Messages
- Durable Messages
- Transactions
- AJAX and REST Support
- Extensibility

Difference between Web Service in ASP.net & WCF Service:

WCF is a replacement for all earlier web service technologies from Microsoft. It also does a lot more than what is traditionally considered as "web services".

WCF "web services" are part of a much broader spectrum of remote communication enabled through WCF. You will get a much higher degree of flexibility and portability doing things in WCF than through traditional ASMX because WCF is designed, from the ground up, to summarize all of the different distributed programming infrastructures offered by MS. An endpoint in WCF can be communicated with just as easily over SOAP/XML as it can over TCP/binary and to change this medium is simply a configuration file mod. In theory this reduces the amount of new code needed when porting or changing business needs, targets, etc.

ASMX is older than WCF, and anything ASMX can do so can WCF (and more). Basically you can see WCF as trying to logically group together all the different ways of getting two apps to communicate in the world of MS; ASMX

was just one of these many ways and so is now grouped under the WCF umbrella of capabilities.

Web Services can be accessed only over HTTP & it works in stateless environment, where WCF is flexible because its services can be hosted in different types of applications. Common scenarios for hosting WCF services are IIS ,WAS ,Self-hosting ,Managed Windows Service.

Major Difference is That Web Services Use XmlSerializer But WCF UsesDataContractSerializer which is better in Performance as Compared to XmlSerializer.

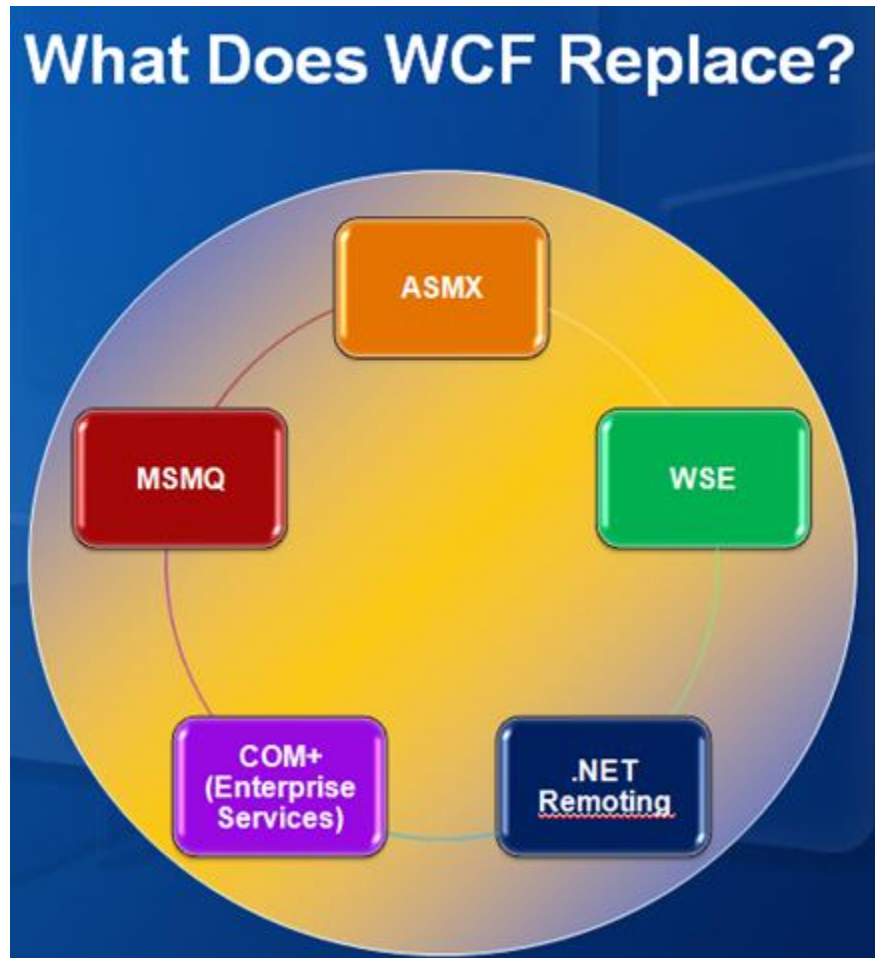
Key issues with XmlSerializer to serialize .NET types to XML

- Only Public fields or Properties of .NET types can be translated into XML.
- Only the classes which implement IEnumerable interface.
- Classes that implement the IDictionary interface, such as Hash table can not be serialized.

Important difference between DataContractSerializer and XMLSerializer.

- A practical benefit of the design of the DataContractSerializer is better performance over Xmlserializer.

- XML Serialization does not indicate the which fields or properties of the type are serialized into XML where as DataContractSerializer Explicitly shows the which fields or properties are serialized into XML.
- The DataContractSerializer can translate the HashTable into XML.



The development of web service with ASP.NET relies on defining data and relies on the [XmlSerializer](#) to transform data to or from a service.

Key issues with **XmlSerializer** to serialize .NET types to XML

- Only Public fields or Properties of .NET types can be translated into XML.
- Only the classes which implement [IEnumerable](#) interface.
- Classes that implement the [IDictionary](#) interface, such as Hash table can not be serialized.

The WCF uses the [DataContractAttribute](#) and [DataMemberAttribute](#) to translate .NET FW types in to XML.

```
<pre lang="C#">
```

```
[DataContract]
public class Item
{
    [DataMember]
    public string ItemID;
    [DataMember]
    public decimal ItemQuantity;
    [DataMember]
    public decimal ItemPrice;
}
```

```
</pre>
```

The [DataContractAttribute](#) can be applied to the class or a structure. [DataMemberAttribute](#) can be applied to field or a property and these fields or properties can be either public or private.

Important difference between [DataContractSerializer](#) and [XMLSerializer](#).

- A practical benefit of the design of the [DataContractSerializer](#) is better performance over XML serialization.
- XML serialization does not indicate the which fields or properties of the type are serialized into XML where as [DataContractSerializer](#) Explicitly shows the which fields or properties are serialized into XML.
- The [DataContractSerializer](#) can translate the HashTable into XML.

Developing Service

To develop a service using ASP.NET we must add the [WebService](#) attribute to the class and [WebMethodAttribute](#) to any of the class methods.

```
<pre lang="c#">
```

Example

```
[WebService]
public class Service : System.Web.Services.WebService
{
    [WebMethod]
    public string Test(string strMsg)
    {
        return strMsg;
    }
}
```

```
}  
}
```

```
</pre>
```

To develop a service in WCF we will write the following code

```
<pre lang="C#">
```

```
[ServiceContract]  
public interface ITest  
{  
    [OperationContract]  
    string ShowMessage(string strMsg);  
}  
public class Service : ITest  
{  
    public string ShowMessage(string strMsg)  
    {  
        return strMsg;  
    }  
}
```

```
</pre>
```

The ServiceContractAttribute specifies that an interface defines a WCF service contract, OperationContractAttribute indicates which of the methods of the interface defines the operations of the service contract.

A class that implements the service contract is referred to as a service type in WCF.

Hosting the Service

ASP.NET web services are compiled into a class library assembly and a service file with an extension .asmx will have the code for the service. The service file is copied into the root of the ASP.NET application and Assembly will be copied to the bin directory. The application is accessible using url of the service file.

WCF Service can be hosted within IIS or WindowsActivationService.

- Compile the service type into a class library
- Copy the service file with an extension .SVC into a virtual directory and assembly into bin sub directory of the virtual directory.
- Copy the web.config file into the virtual directory.

Client Development

Clients for the ASP.NET Web services are generated using the command-line tool WSDL.EXE.

WCF uses the ServiceMetadata tool(svcutil.exe) to generate the client for the service.

Message Representation

The Header of the SOAP Message can be customized in ASP.NET Web service.

WCF provides attributes MessageContractAttribute , MessageHeaderAttribute and MessageBodyMemberAttribute to describe the structure of the SOAP Message.

Service Description

Issuing a HTTP GET Request with query WSDL causes ASP.NET to generate WSDL to describe the service. It returns the WSDL as response to the request.

The generated WSDL can be customized by deriving the class of **ServiceDescriptionFormatExtension**.

Issuing a Request with the query WSDL for the .svc file generates the WSDL. The WSDL that generated by WCF can customized by using ServiceMetadataBehavior class.

Exception Handling

In ASP.NET Web services, Unhandled exceptions are returned to the client as SOAP faults.

In WCF Services, unhandled exceptions are not returned to clients as SOAP faults. A configuration setting is provided to have the unhandled exceptions returned to clients for the purpose of debugging.