Volume 1,Issue-1,December-2011 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

www.ijarcsse.com

Research paper

Available online at www.ijarcsse.com

Importance of WCF in Web Applications

Dr. SamratVivekanand O Khanna
Institute of Sci. & Tech. for Adv Std & Res.
sonukhanna@yahoo.com

Mijal Mistry
Institute of Sci. & Tech. for Adv Std & Res.
mijalmistry@rediffmail.com

Abstract – **This paper represents the importance of WCF services into the real world. In some complex scenario it is not possible to update the data at several location and also difficult to update the information at all the places. To address this type of issues WCF is useful and the WCF services available into.net framework which plays an important role for achieving this. WCF services are different compare to standard web services. Using it several complex problems can be addressed. The implementation between the different applications can be achieved by interoperability and transactions by WCF services. It is useful in establishing communication between diverse systems and to gain performance benefits on similar systems. It can produce best result in heterogeneous environment.**

Keywords: **WCF, Web Application, Web, Flight, Searching, Data Binding**

I. INTRODUCTION

WCF (Windows Communication Foundation) is a runtime and a set of APIs for creating systems that send messages between services and clients. The same infrastructure and APIs are used to create applications that communicate with other applications on the same computer system or on a system that resides in another company and is accessed over the Internet. [1]

The Fig.1 shows the terms used in the WCF architecture. [1] Following are the important components of WCF. Contracts and Descriptions: Contracts define various aspects of the message system. The data contract describes every parameter that makes up every message that a service can create or consume.

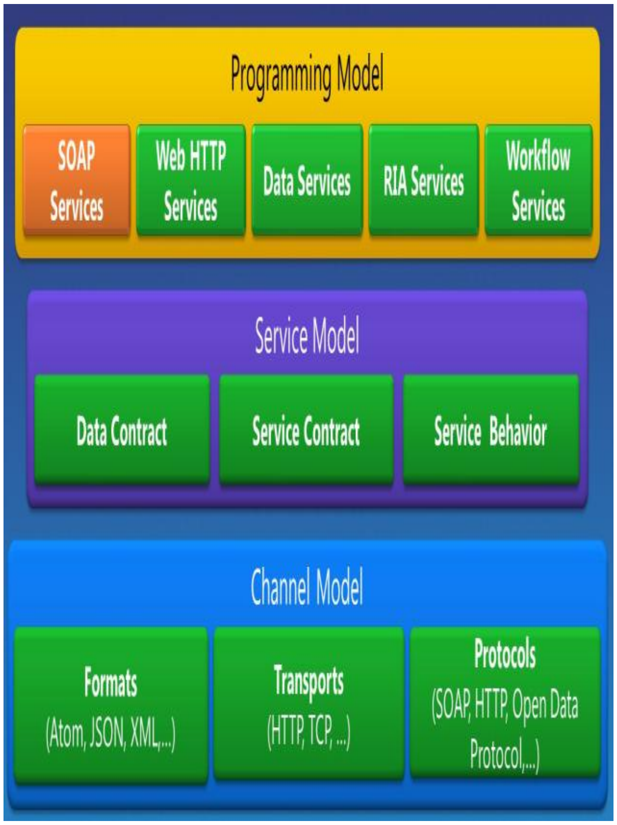

Fig. 1 Silverlight Architecture

The message parameters are defined by XML Schema definition language (XSD) documents, enabling any system that understands XML to process the documents.

Service Runtime: The service runtime layer contains the behaviors that occur only during the actual operation of the service, that is, the runtime behaviors of the service. Throttling controls how many messages are processed, which can be varied if the demand for the service grows to a preset limit. An error behavior specifies what occurs when an internal error occurs on the service, for example, by controlling what information is communicated to the client.

Messaging: The messaging layer is composed of channels. A channel is a component that processes a message in some way, for example, by authenticating a message. A set of channels is also known as a channel stack. Channels operate on messages and message headers. This is different from the service runtime layer, which is primarily concerned about processing the contents of message bodies.

There are two types of channels: transport channels and protocol channels.

Hosting and Activation: Services can also be hosted, or run in an executable managed by an external agent, such as IIS or Windows Activation Service (WAS). WAS enables WCF applications to be activated automatically when deployed on a computer running WAS. Services can also be manually run as executables (.exe files). A service can also be run automatically as a Windows service. COM+ components can also be hosted as WCF services.

II. WCF TERMS

There are several terms used in WCF. Here we are going to see the important one which we have used while developing the system.

Message: A self-contained unit of data that can consist of several parts, including a body and headers.

Service: A construct that exposes one or more endpoints, with each endpoint exposing one or more service operations.

Endpoint: A construct at which messages are sent or received (or both).

Address: Specifies the location where messages are received. It is specified as a Uniform Resource Identifier (URI). The URI schema part names the transport mechanism to use to reach the address, such as HTTP and TCP. The hierarchical part of the URI contains a unique location whose format is dependent on the transport mechanism.

Following code snippet shows the address which we have used.

```
HTTPS://localhost:8005/Services/RetrievalService
```

Binding: Defines how an endpoint communicates to the world. It is constructed of a set of components called binding elements that "stack" one on top of the other to create the communication infrastructure.

Hosting: A service must be hosted in some process. A *host* is an application that controls the lifetime of the service. Services can be self-hosted or managed by an existing hosting process.

Client Application: A program that exchanges messages with one or more endpoints. The client application begins by creating an instance of a WCF client and calling methods of the WCF client.

III. DEVELOPED SYSTEM

To keep these things into mind we have developed a system using WCF services. We have developed a system for targeting the airline reservation process. Our system will communicate with clients system and fetch the results based on parameters supplied. Normally the process goes like this, we place a query say we want to search the available flights between Mumbai to London. When we supply the query it will fetch the results from different clients application. There are several flights from Mumbai to London and different airlines provide the frequency for traveling say Air India, Emirates, British Airways etc. All are having different timings and stoppage. Some flights are direct and some flights have 1 or 2 stoppages based on the airlines and passengers demands.

We have developed a WCF service which passed the parameters to the client’s application and return the results from different client’s applications. The service communicates with two applications, one is our own developed application and one is client’s application. We have defined one class called ShowFlightsResults which has following properties. The class has properties for storing the information for FlightID, Timings for Flights, Starting and Ending dates for Flights, Source and Destination location and trip way information. Trip can round trip or single trip.

```
[DataContract]
public class ShowFlightsResults
{
        [DataMember]
        public string FlightID;

        [DataMember]
        public string FlightTime;

        [DataMember]
        public string FlightStartDates;

        [DataMember]
        public string FlightEndDates;

        [DataMember]
        public string FlightSourceLocation;

        [DataMember]
        public string
        FlightDestinationLocation;

        [DataMember]
        public string FlightTripInformation;

}
```

A data contract is a formal agreement between a service and a client that abstractly describes the data to be exchanged. That is, to communicate, the client and the service do not have to share the same types, only the same data contracts. A data contract precisely defines, for each parameter or return type, what data is serialized (turned into XML) to be exchanged. [3] Windows Communication Foundation (WCF) uses a serialization engine called the Data Contract Serializer by default to serialize and deserialize data. [3]

```
public List GetAllFlights(string
source, string destination)
{
        // statements
}
```

For hosting and deployment following changes need to be performed. Following code snippet gives the idea about it.

```
<service name="
ShowFlightsResults.GetAllFlights ">
        <endpoint
binding="basicHttpBinding"
contract="ShowFlightsResults.GetAll
Flights "></endpoint>
      </service>
```

The system.serviceModel/services tag contains the classes, enumerations, and interfaces necessary to build service and client applications that can be used to build widely distributed applications. [3]

The GetAllFlights function returns the list of all the flights from different clients applications and then combine it into a single list and the result will be enumerated and return the combines values from all the clients and display.

IV. IMPORTANCE ON WCF

Imagine a situation above that if we have different web services for different clients then it might be difficult during updating. We have to update each web services individually and then host it again. Imagine how much time it will take to finish updating. To overcome this situation WCF is the answer. WCF, with managed code, using power and features of .NET 4.0, giving all the benefits of performance, security, transaction, reliability by supporting all open industry standards makes life easy for the developer and the organization by unifying the programming model with the best features from all available technologies. [4]

By combining WCF and Ajax and using set of techniques we can deliver a highly interactive, desktop-like user experience for Web applications in popular HTML browsers. [6]

V. CONCLUSION

While developing the above system we came to know that if there are more communication between other clients application, WCF communication framework provides the effective way for binding the data and we can securely retrieve the data. Our future work involve more secure way for retrieving the information and we also use some cryptography methods.

REFERENCES

[1] http://msdn.microsoft.com/en-us/library/ms733128.aspx

[2] http://msdn.microsoft.com/en-us/library/ms731079.aspx

[3]http://s.codeproject.com/KB/WCF/WCF-Service-Create-Config.aspx

[4]http://www.codeproject.com/KB/WCF/WCF_in_details.aspx

[5]http://msdn.microsoft.com/en-us/magazine/cc163354.aspx

[6]http://www.openajax.org/whitepapers/Introducing%20Ajax%20and%20OpenAjax.php