

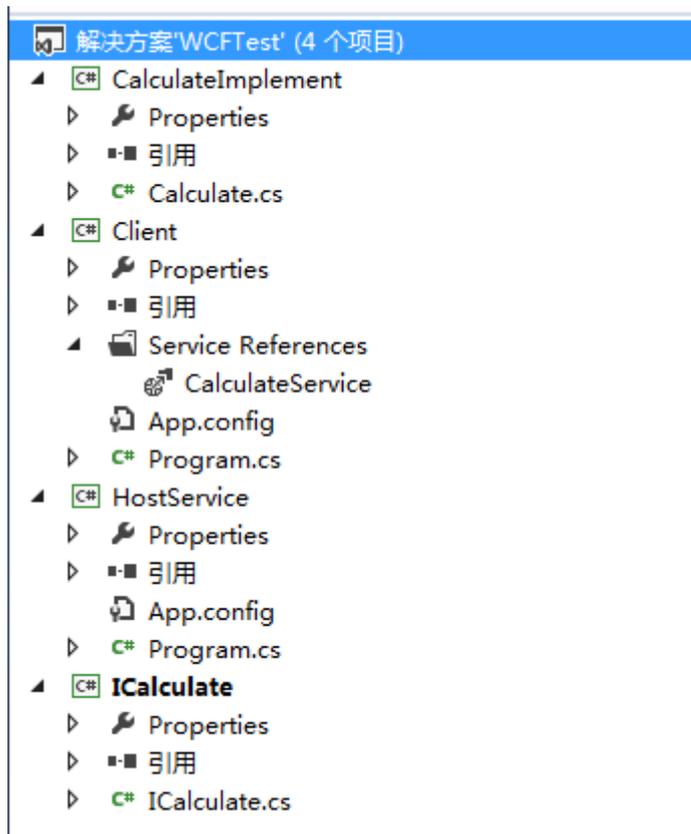
The Demo of SOA

Made by HuaSheng

2014-10-13

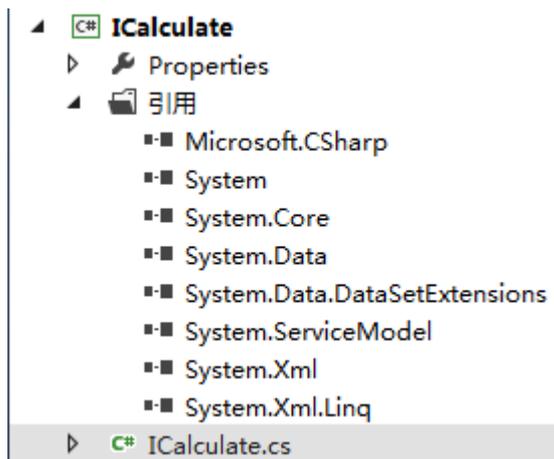
Blog: <http://www.cnblogs.com/chenyongblog>

使用 **SOA** 来实现两个数字的相加，不包含验证。



1、首先定义一个接口 **ICalculate**

(1)引入 **System.ServiceModel** 程序集



(2)公开接口，使用 **ServiceContract** 特性定义服务契约（标注 **interface**），**OperationContract** 特性标注 **Method**

```

using System.ServiceModel;

namespace CalculateImplement
{
    [ServiceContract]
    public interface ICalculate
    {
        [OperationContract]
        double Add(double x, double y);
    }
}

```

2、接口的实现

```

namespace CalculateImplement
{
    public class Calculate : ICalculate
    {
        public double Add(double x, double y)
        {
            return x + y;
        }
    }
}

```

3、Host 管理服务

```

namespace HostService
{
    class Program
    {
        static void Main(string[] args)
        {
            ServiceHost host = new ServiceHost(typeof(Calculate));

            try
            {
                host.Open();
                Console.WriteLine("Service is open.....");
                Console.ReadLine();
                host.Close();
            }
            catch (Exception et)
            {
                throw et;
            }
        }
    }
}

```

这里需要配置 App.config

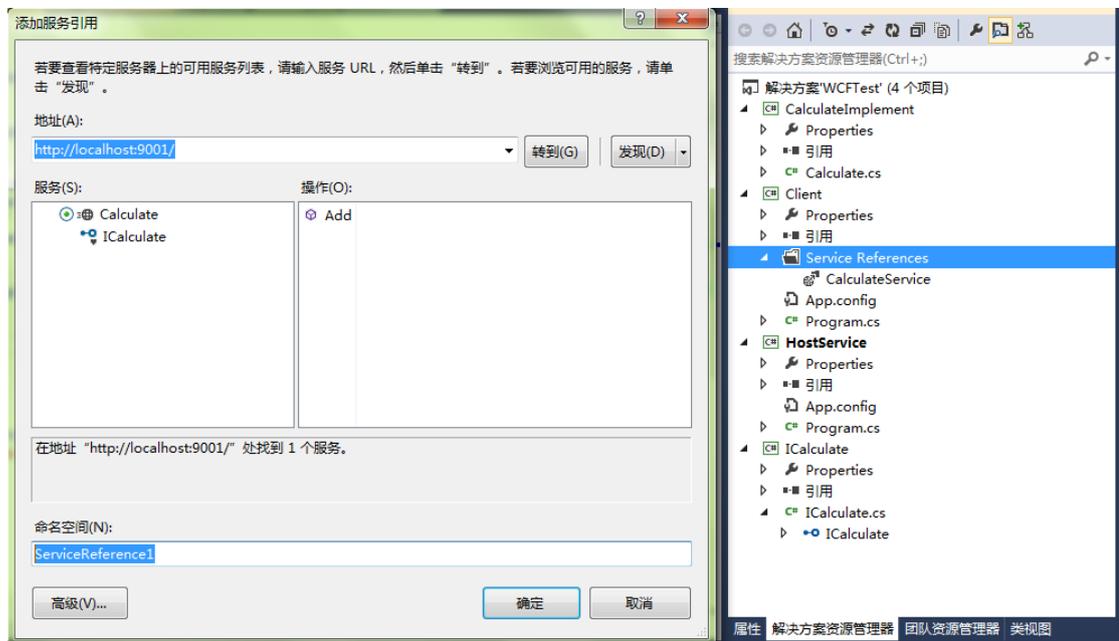
```

<!--WCF Setting-->
<system.serviceModel>
  <services>
    <service name="CalculateImplement.Calculate" behaviorConfiguration="serviceBehavior">
      <host>
        <baseAddresses>
          <add baseAddress="http://localhost:9001"/>
        </baseAddresses>
      </host>
      <endpoint name="CalculateImplementEndPoint"
        address="CalculateImplement"
        binding="basicHttpBinding"
        contract="CalculateImplement.ICalculate"/>
      <endpoint name="mex"
        binding="mexHttpBinding"
        contract="IMetadataExchange"
        address="mex"/>
    </service>
  </services>

  <behaviors>
    <serviceBehaviors>
      <behavior name="serviceBehavior">
        <serviceMetadata httpGetEnabled="true" httpsGetEnabled="false"/>
        <serviceDebug includeExceptionDetailInFaults="false"/>
      </behavior>
    </serviceBehaviors>
  </behaviors>
</system.serviceModel>
</configuration>

```

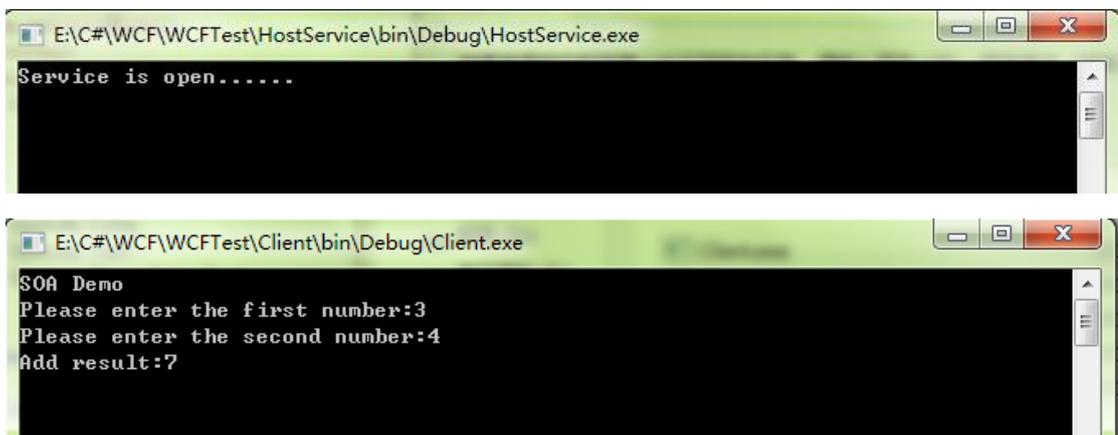
4、开启 Host，在 Client 端添加 Service，修改命名空间



Client 代码:

```
class Program
{
    static void Main(string[] args)
    {
        CalculateService.CalculateClient calculate = new CalculateService.CalculateClient();
        Console.WriteLine("SOA Demo");
        Console.Write("Please enter the first number:");
        double num1 = Convert.ToDouble(Console.ReadLine());
        Console.Write("Please enter the second number:");
        double num2 = Convert.ToDouble(Console.ReadLine());
        double result = calculate.Add(num1, num2);
        Console.WriteLine("Add result:" + result);
        Console.ReadLine();
    }
}
```

程序运行:



The image shows two separate console windows. The top window, titled 'E:\C#\WCF\WCFTest\HostService\bin\Debug\HostService.exe', displays the text 'Service is open.....'. The bottom window, titled 'E:\C#\WCF\WCFTest\Client\bin\Debug\Client.exe', displays the following text: 'SOA Demo', 'Please enter the first number:3', 'Please enter the second number:4', and 'Add result:7'. This demonstrates the client successfully connecting to the host service and performing a calculation.

Thanks,
Made by Luke