

برای ساخت یک WCF Client یا دسترسی به یک سرویس WCF دو راه وجود دارد.

استفاده از WCF Proxy

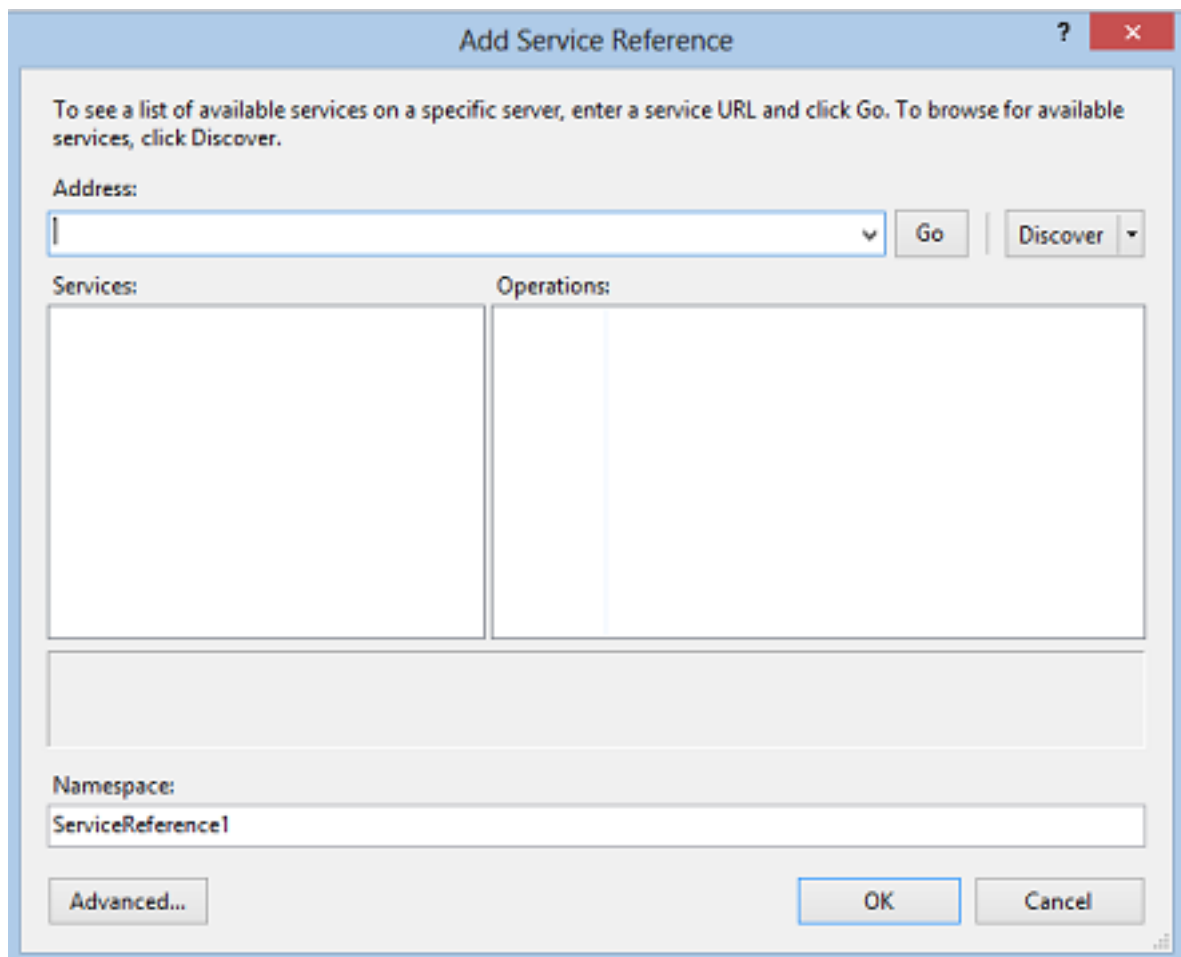
استفاده از ChannelFactory

قصد داریم طی یک مقایسه کوتاه این دو روش را بررسی کنیم:

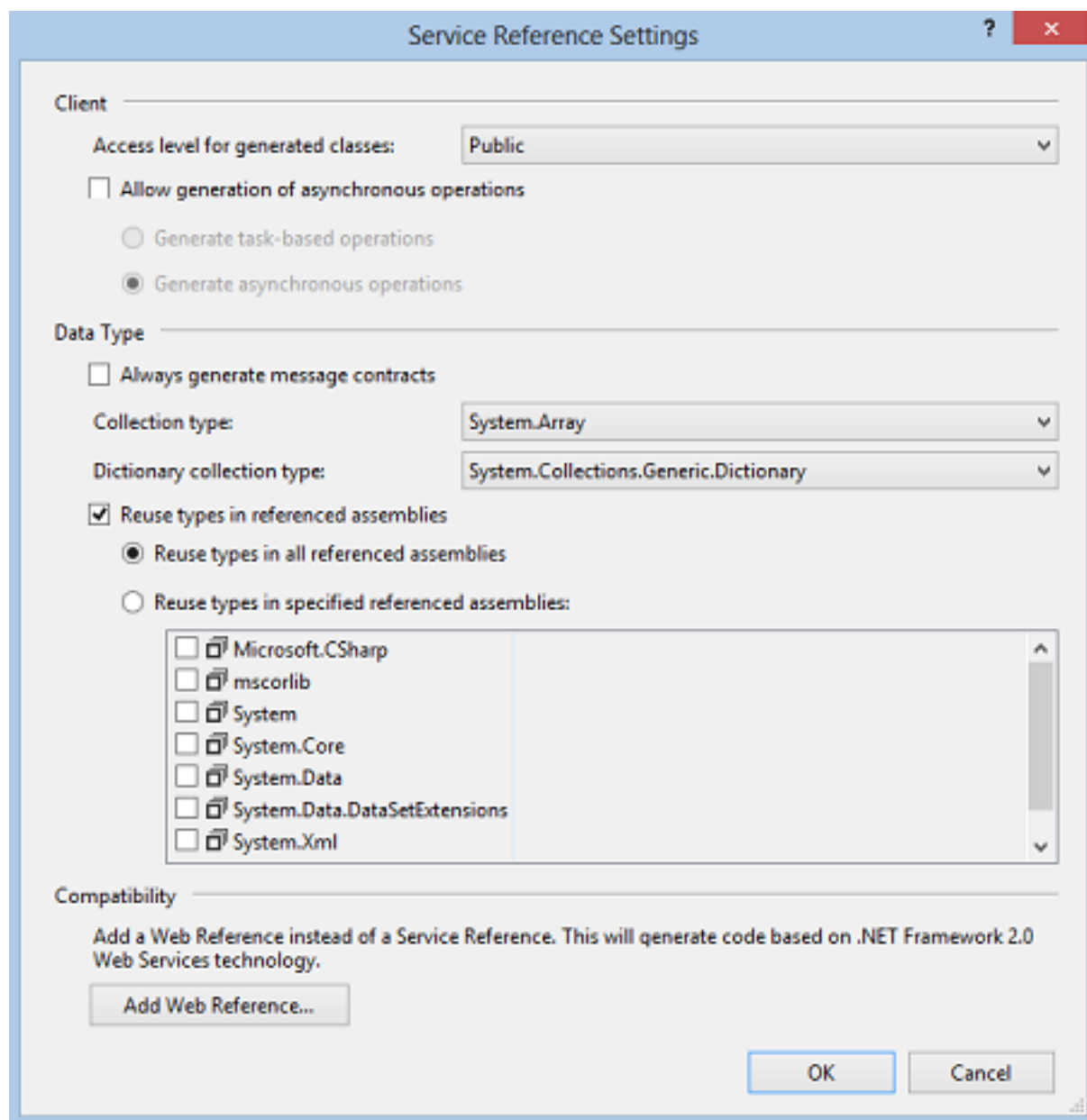
WCF Proxy:

Proxy در واقع یک کلاس CLR است که به عنوان نماینده یک اینترفیس که از نوع Service Contract است مورد استفاده قرار می‌گیرد. یا به زبان ساده تر، یک Proxy در واقع نماینده Service Contract ای که سمت سرور پیاده سازی شده است در سمت کلاینت خواهد بود. Proxy تمام متد یا Operation Contract های سمت سرور را داراست به همراه یک سری متدها و خواص دیگر برای مدیریت چرخه طول عمر سرویس، هم چنین اطلاعات مربوط به وضعیت سرویس و نحوه اتصال آن به سرور. ساخت Proxy به دو روش امکان پذیر است:

با استفاده از امکانات AddServiceReference موجود در Visual Studio. کافیسست از پنجره معروف زیر با استفاده از یک URL سرویس مورد نظر را به پروژه سمت کلاینت خود اضافه نمایید



همچنین می‌توانید از قسمت Advanced نیز برای تنظیمات خاص مورد نظر خود (مثل تولید کد برای متدهای Async یا تعیین نوع Collection ها در هنگام انتقال داده و ...) استفاده نمایید.



با استفاده از [SvcUtil.exe](#) . کاربرد در svcutil.exe موارد Metadata Export, Service Validation, XmlSerialization Type و Generator و Metadata Download و ... خلاصه می‌شود. با استفاده از Vs.Net Command Prompt و svcutil می‌توان به سرویس مورد نظر دسترسی داشت. مثال

```
svcutil.exe /language:vb /out:generatedProxy.vb /config:app.config
http://localhost:8000/ServiceModelSamples/service
```

:ChannelFactory

ChannelFactory یک کلاس تعبیه شده در دات نت می‌باشد که به وسیله یک اینترفیس که به عنوان تعاریف سرویس سمت سرور است یک نمونه از سرویس مورد نظر را برای ما خواهد ساخت. اما به خاطر داشته باشید از این روش زمانی می‌توان استفاده کرد

که دسترسی کامل به سمت سرور و کلاینت داشته باشید.

برای آشنایی با نحوه پیاده سازی به این روش نیز می‌توانید از این [مقاله](#) کمک بگیرید.

مثال:

```
public static TChannel CreateChannel()
{
    IBookService service;

    var endPointAddress = new EndpointAddress( "http://localhost:7000/service.svc" );
    var httpBinding = new BasicHttpBinding();

    ChannelFactory<TChannel> factory = new ChannelFactory<TChannel>( httpBinding,
endPointAddress );

    instance= factory.CreateChannel();

    return instance;
}
```

همان طور که مشاهده می‌کنید در این روش نیاز به یک EndpointAddress به همراه یک نمونه از نوع Binding مورد نظر دارید. نوع این Binding حتما باید با نوع نمونه ساخته شده در سمت سرور که برای هاست کردن سرویس‌ها مورد استفاده قرار گرفته است یکی باشد. این نوع‌ها می‌تواند شامل MSMQ, WSDualHttpBinding, WSHttpBinding, BasicHttpBinding, NetTcpBinding و البته چند نوع دیگر نیز باشد.

در نتیجه برای ساخت یک سرویس به روش ChannelFactory باید مراحل زیر را طی نمایید:
یک نمونه از WCF Binding بسازید

یک نمونه از کلاس EndPointAddress به همراه آدرس سرویس مورد نظر در سمت سرور بسازید(البته می‌توان این مرحله را نادیده گرفت و آدرس سرویس را مستقیماً به نمونه ChannelFactory به عنوان پارامتر پاس داد)

یک نمونه از کلاس ChannelFactory یا استفاده از EndPointAddress بسازید
با استفاده از ChannelFactory یک نمونه از Channel مورد نظر را فراخوانی نمایید(فراخوانی متد CreateChannel)

تفاوت‌های دو روش

ChannelFactory	Proxy
شما نیاز به دسترسی مستقیم به اسمبلی حاوی Service Contract پروژه خود دارید.	فقط نیاز به یک URL برای ساخت سرویس مورد نظر دارد. بقیه مراحل توسط ابزارهای مرتبط انجام خواهد شد
پیاده سازی آن پیچیدگی بیشتر دارد	استفاده از این روش بسیار آسان و ساده است
نیاز به دانش اولیه از مفاهیم WCF برای پیاده سازی دارد	فهم مفاهیم این روش بسیار راحت است
زمانی که اطمینان دارید که مدل و entityها پروژه زیاد تغییر نخواهند کرد و از طرفی نیاز به کد نویسی کمتر در سمت کلاینت دارید، این روش موثرتر خواهد بود	زمانی که میزان تغییرات در کلاس‌های مدل و Entityها زیاد باشد این روش بسیار موثر است. (مدیریت تغییرات در WCF)
به تمام اینترفیس‌های تعریف شده در بخش Contracts دسترسی داریم.	فقط به اینترفیس‌هایی که دارای ServiceContractAttribute هستند دسترسی خواهیم داشت.
به تمام متدهای عمومی سرویس دسترسی داریم.	فقط به متدهای که دارای OperationContractAttribute هستند دسترسی خواهیم داشت.

آیا می‌توان از روش AddServiceReference تعبیه شده در Vs.Net، برای ساخت ChannelFactory استفاده کرد؟

بله! کافیت هنگام ساخت سرویس، در پنجره AddServiceReference از قسمت Advanced وارد برگه تنظیمات شوید. سپس تیک مربوط به قسمت های Reused Type in referenced assemblies و Reused Types in specified referenced assemblies را بزنید. بعد از لیست پایین، اسمبلی های مربوط به Domain Model و هم چنین Contract های سمت سرور را انتخاب نمایید. در این حالت شما از روش Channel Factory برای ساخت سرویس WCF استفاده کرده اید.

- ☒ Reuse types in referenced assemblies
- ☐ Reuse types in all referenced assemblies
- ☒ Reuse types in specified referenced assemblies:

<input type="checkbox"/>	<input type="checkbox"/>	Microsoft.CSharp
<input type="checkbox"/>	<input type="checkbox"/>	mscorlib
<input checked="" type="checkbox"/>	<input type="checkbox"/>	Service
<input type="checkbox"/>	<input type="checkbox"/>	System
<input type="checkbox"/>	<input type="checkbox"/>	System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089
<input type="checkbox"/>	<input type="checkbox"/>	System.Data.DataSetExtensions

نظرات خوانندگان

نویسنده: افشار محبی
تاریخ: ۹:۴۳ ۱۳۹۲/۰۸/۰۶

استفاده از Channel Factory باعث می‌شود سورس کد پروژه بسیار تمیز و خلوت باشد. این موضوع اگر با سورس کنترل و محیط کار دسته جمعی همراه باشد خیلی تاثیر گذار خواهد بود چون Service Reference مقدار زیادی فایل تولید می‌کند. به علاوه با کمک Channel Factory امکان بعضی خودکار سازی‌ها هم بهتر فراهم خواهد بود.