

BizTalk opent de poort voor WCF

BIZTALK SERVER, WCF OF BEIDE?

Met de 'connected enterprise strategy' brengt Microsoft technologie op de markt, waarmee je applicaties eenvoudig en efficiënt kunt ontsluiten en integreren. Aan de recentste versie van BizTalk, BizTalk Server 2006 R2, is een aantal krachtige features toegevoegd die deze strategie ondersteunt. Eén van deze features is ondersteuning voor Windows Communication Foundation (WCF) in de vorm van nieuwe adapters. Op deze WCF-adapters wordt hieronder verder ingegaan. Basiskennis van BizTalk Server en WCF is vereist bij het lezen van dit artikel.

Vaak is het verschil tussen BizTalk en WCF niet duidelijk en worden de technologieën onterecht gelijkgesteld. Daarom klinkt nogal eens de vraag: 'Waarom is BizTalk nog nodig nu je WCF kunt gebruiken?' WCF is een bouwsteen of API. Het voorziet in een aantal .NET-classes waarmee je op uniforme manier kunt communiceren over verschillende protocollen en transporten. BizTalk is geen bouwsteen, maar een product. Ook BizTalk heeft, door middel van de messaging engine en de verschillende adapters, mogelijkheden om te communiceren over verschillende protocollen en transporten. Er bestaat dus wel degelijk overlap tussen beide technologieën. Met de orchestration engine, de transformaties, de Business Rules-engine, Business Activity Monitoring en Business Activity Services heeft BizTalk echter nog veel meer functionaliteit aan boord. Je kunt beide dan ook het beste bekijken als complementaire technologieën. De combinatie van de twee biedt krachtige mogelijkheden om servicegeoriënteerde oplossingen te bouwen. Een voorbeeld hiervan is een scenario waarin Line Of Business (LOB) applicaties met behulp van WCF, eventueel op basis van de WCF LOB Adapter SDK, als services zijn ontsloten (zie ook kader WCF LOB Adapter SDK). BizTalk wordt dan ingezet als broker, die de aanroep van de verschillende services coördineert.

BizTalk en WCF

In BizTalk 2006 R2 zijn zeven nieuwe adapters voor communicatie met WCF toegevoegd. Binnen de messaging engine, die het hart van BizTalk Server vormt, zijn in deze versie geen wijzigingen voor WCF gemaakt. Het scheidingspunt tussen WCF en BizTalk ligt dus op het niveau van de BizTalk WCF-adapters. Tabel 1 geeft een overzicht van de verschillende WCF-adapters.

Er zijn vijf adapters die overeenkomen met de bindings die bekend zijn van WCF. Deze adapters zijn eenvoudig te configureren en voldoen in de meeste gevallen. Wil je geavanceerdere bindings of behaviours gebruiken, dan zijn de properties van de standaard WCF-adapters vaak niet toereikend. Kies in dat geval voor de Custom (in-process) of Custom Isolated-adapters. Ook kun je met

WCF-adapter	WCF-binding
WCF-BasicHttp-adapter	BasicHttp-binding
WCF-WSHttp-adapter	WSHttp-binding
WCF-NetTcp-adapter	NetTcp-binding
WCF-NetNamedPipe-adapter	NetNamedPipe-binding
WCF-NetMsmq-adapter	NetMsmq-binding
WCF-Custom-adapter	-
WCF-CustomIsolated-adapter	-

Tabel 1. WCF-adapters en bijbehorende WCF-bindings

deze adapters WCF Extensions in BizTalk gebruiken.

De WCF-BasicHttp-adapter is voor communicatie over HTTP(S) met legacy webservices. Deze adapter stelt je bijvoorbeeld in staat te communiceren met standaard .NET ASMX-webservices. Met de WCF-WSHttp-adapter is ook communicatie over HTTP(S) mogelijk, maar dan met webservices die aan de WS-*standaarden voldoen. De adapter ondersteunt in deze versie nog geen WS-Reliable Messaging en WS-Business Activities. Beide adapters draaien in een isolated Internet Information Server (IIS) proces. Ook de WCF-NetTcp-adapter ondersteunt net als de WCF-WSHttp-adapter de WS-*standaard. Communicatie loopt hierbij, zoals al uit de naam van de adapter blijkt, over het TCP-protocol. Dit garandeert een snelle en efficiënte communicatie. Voor het sturen naar en ontvangen van berichten uit een queue is er de WCF-NetMsmq-adapter. Deze gebruikt uiteraard MSMQ als transport.

Met de WCF-NetNamedPipe-adapter kunnen twee processen op een veilige manier, binnen de grenzen van één machine, over named pipes met elkaar communiceren. De WCF-NetNamedPipe-, WCF-NetTcp- en WCF-NetMsmq-adapters kun je alleen in een volledig WCF (WCF-service en WCF-client) scenario toepassen. Vanuit BizTalk kun je WCF op twee manieren gebruiken:

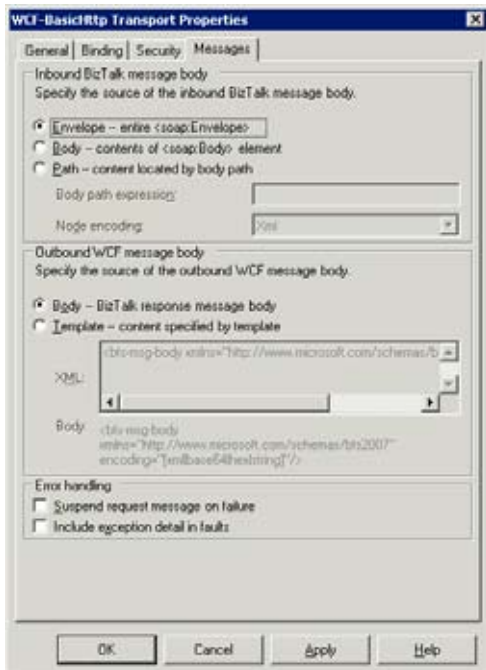
- WCF-services aanbieden
- WCF-services aanroepen

WCF-services aanbieden

BizTalk-oplossingen kun je met de WCF-adapters als WCF-service aanbieden. De client kan de WCF-service consumeren zonder te weten dat de achterliggende functionaliteit in BizTalk is geïmplementeerd. Wanneer de client de service aanroept, dan zal de WCF-

WCF LOB Adapter SDK

Het was al mogelijk om met behulp van de BizTalk adapter SDK custom adapters voor BizTalk te ontwikkelen en daarmee legacy systemen te ontsluiten. Een nadeel daarvan is dat je deze adapters alleen kunt gebruiken vanuit BizTalk. Met de WCF-LOB adapter SDK kun je WCF-adapters voor legacy applicaties, databases en dergelijke te ontwikkelen, die bruikbaar zijn vanuit meer omgevingen. Dit alles op basis van .NET 3.X WCF-technologie. Je kunt dan bijvoorbeeld .NET-maatwerkcode, BizTalk of SharePoint inzetten als consumer van de adapter. De WCF LOB adapter SDK heeft onder andere een simpel en uniform programmeermodel boven op het WCF-channelmodel, een runtime omgeving voor de adapter, een standaard manier om metadata over de LOB-applicatie aan te bieden en een aantal wizards om een adapter te bouwen of te gebruiken.



Afbeelding 1. De properties van de receive location

adapter per *receive location* een instantie van een van ServiceHost afgeleide class aanmaken. Eenmaal in BizTalk is een via de service binnengekomen bericht omgezet in een 'gewoon' BizTalk-bericht en zal de BizTalk-messaging engine het bericht op de normale manier verwerken. De properties van de receive location (zie afbeelding 1) stellen je nog wel in staat om aan te geven welk deel van het binnengekomen WCF-bericht BizTalk als *BizTalk-bericht* in de messagebox zal publishen. Er zijn drie mogelijkheden:

- **Envelope** - Inhoud van het hele soap:envelope-element
- **Body** - Inhoud van het soap:body-element wordt body
- **Path** - Een Xpath-expressie waarmee een deel van het bericht is te selecteren.

Voor het aanbieden van WCF-services is aan BizTalk de 'BizTalk WCF Service Publishing Wizard' toegevoegd. Deze nieuwe wizard heeft veel overeenkomsten met de 'BizTalk Web Services Publishing Wizard' die in eerdere versies van BizTalk al aanwezig was. Het belangrijkste verschil tussen de twee zit in het eerste scherm. De WCF-wizard vraagt namelijk eerst of je een service-endpoint wilt publishen of alleen een Metadata-endpoint. Het publishen van een service-endpoint is uiteraard alleen mogelijk voor de out-of-process WCF-adapters (WCF-basicHttp, WCF-WSHttp en WCF-CustomIsolated). Je kunt naast een service-endpoint ook een metadata-endpoint te publishen. Dit maakt het mogelijk voor de in-process adapters (WCF-NetTcp, WCF-NetNamedPipe WCF-NetMsmq en WCF-Custom) om metadata te publishen. Clients van dit type WCF-service kunnen daarmee meta-informatie van de service opvragen. In de volgende stappen is de wizard identiek aan de BizTalk Web Services Publishing Wizard. Je dient achtereenvolgens de volgende zaken aan te geven:

1. Of je een orchestration of een schema wilt publishen als WCF-service.
2. Wat de target namespace is van de service.
3. Wat de locatie (url) van de webservice is en of deze zonder authenticatie (anonymous) is te bereiken.

Nu alle informatie is ingevoerd, kan de wizard in Internet Information Server een WCF-service-applicatie aanmaken. Indien aangegeven, zal de wizard ook de benodigde *receive locations* in BizTalk creëren. De WCF-service is nu klaar voor aanroep door een client. De service zal bij een aanroep een bericht ontvangen en deze via één van de WCF-adapters en aan BizTalk doorgeven. BizTalk kan dit bericht vervolgens verder verwerken door bijvoorbeeld een orchestration op te starten of het naar een backend-systeem te sturen.

WCF-services consumeren

Voor het consumeren van WCF-services vanuit een orchestration kun je Visual Studio (VS) gebruiken. VS beschikt over een nieuwe wizard speciaal voor het consumeren van WCF-services vanuit een orchestration. In afbeelding 2 is te zien dat je deze wizard via de opties 'Add Generated Items' en vervolgens 'Consume WCF Service' kunt starten. In de eerste stap geef je aan waar de wizard de metadata (WSDL) van de service kan ophalen. Dit ophalen kan via een metadata exchange (MEX) endpoint of via WSDL- en XSD-bestanden als die beschikbaar zijn. Kies je voor een MEX-endpoint, dan moet je in de volgende stap van de wizard de url naar het MEX-endpoint van de WCF-service aangeven. Als laatste stap vraagt de wizard om de invoer van een namespace. Na bevestiging genereert de wizard op basis van de WSDL van de service een aantal BizTalk-onderdelen of artefacts. Hiermee kun je op een eenvoudige manier vanuit een orchestration met de service te communiceren. De wizard maakt, onder de aangegeven namespace, drie verschillende soorten artefacts aan. Dit zijn:

- Service Schema
- Orchestration met WCF-types
- Binding files

Serviceschema

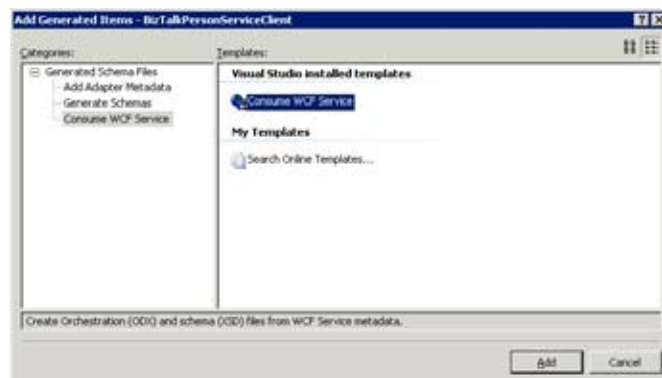
Het serviceschema bevat de definitie van de berichten die BizTalk naar de WCF-service zal sturen en de berichten die de WCF-service retourneert. In afbeelding 3 zie je welke schema's de wizard maakt voor de service uit codevoorbeeld 1.

Orchestration met WCF-types

Deze orchestration bevat een aantal BizTalk-types. Alle types hebben een 'internal' scope, zodat ze ook vanuit andere orchestrations in dezelfde Visual Studio Solution zijn te gebruiken. Het scheiden van de gegenereerde types en het gebruik ervan in een andere orchestration is overigens ook de aanbevolen werkwijze. Op die manier is het mogelijk de artefacts opnieuw te genereren, als er een nieuwe versie van de WCF-service beschikbaar is. Op basis van de genoemde serviceschema's zijn multipart message-types gegenereerd. Instanties van deze message-types gaan uiteindelijk heen en weer naar de WCF-service. Ook is een orchestration-poorttype gegenereerd dat voor iedere methode van de WCF-service een poortoperatie bevat. De koppeling tussen de BizTalk-poortoperatie en de SOAP-action van de WCF-service is configureerbaar. In afbeelding 4 is de SOAP Action header-sectie te zien van de adapter-properties. Codevoorbeeld 2 laat zien dat in dit voorbeeld de BizTalk-operation 'GetByName' is gekoppeld aan een bepaalde SOAP Action van de WCF-service.

Binding files

Naast de 'logische' orchestration-poorten zijn voor communicatie met de WCF-service ook 'fysieke' BizTalk-poorten nodig. Om deze poorten aan te maken, levert de wizard een tweetal binding-files. De files bevatten alle informatie die nodig is om de send-poorten



Afbeelding 2. Het starten van de wizard via de opties 'Add Generated Items' en 'Consume WCF Service'



Abbeelding 3. De schema's die de wizard maakt voor de service uit codevoorbeeld 1

binnen BizTalk aan te maken. De eerste file is specifiek voor de gekozen WCF-adapter, bijvoorbeeld WCF-BasicHttp of WCF-WSHttp. De tweede binding file bevat dezelfde informatie maar dan voor het geval je de WCF-Custom-adapter gebruikt. Dankzij de wizard is gebruik van een WCF-service vanuit een orchestration eenvoudig en in lijn met de werkwijze die een BizTalk-ontwikkelaar gewend is.

De toekomst van BizTalk Server en WCF

BizTalk Server heeft dus vanaf de 2006 R2-versie goede ondersteuning voor de WCF-technologie aan boord. Wizards specifiek voor WCF zijn hierbij een goede hulp. BizTalk-onderdelen zijn eenvoudig als WCF-service aan te bieden. Ook het aanroepen van WCF-services vanuit een orchestration is een fluitje van een cent. Op het moment van schrijven van dit artikel is er nog weinig bekend over de toekomstige versie van BizTalk Server. Er is dus ook niets te zeggen over de ondersteuning daarin van WCF-technologie. Al gissend ligt wel voor de hand dat de integratie en ondersteuning van WCF in de toekomst nog groter, beter en efficiënter zal zijn. Zoals de orchestration engine in de toekomst gebaseerd zal zijn op Windows Workflow Foundation-technologie, zo zal WCF-technologie, of in ieder geval bepaalde onderdelen daarvan, terug te vinden zijn in de BizTalk-messaging engine. De eerder genoemde overlap in de communicatiemogelijkheden tussen de technologieën zal dan wellicht verdwijnen. Daarover nadenkend is een aantal zeer inte-

```
[ServiceBehavior(Namespace =
    "http://Aviva.Samples.WCF.CSharpPersonService/2007/12")]
public class CSharpPersonService : ICSharpPersonService
{
    public Person GetByName(string lastName)
    {
        Console.WriteLine("Request received with parameter '{0}', last ame);

        // dummy implementation will always return same person and
        // does not care about input parameter 'lastName'.
        Person person = new Person();
        person.FirstName = "John";
        person.LastName = "Doe";

        return person;
    }
}
```

Codevoorbeeld 1. Voorbeeld van een WCF-service

```
<BtsActionMapping
    xmlns:xsi = "http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd = "http://www.w3.org/2001/XMLSchema">
    <Operation Name = "GetByName" Action =
        "http://Aviva.Samples.WCF.CSharpPersonService/2007/12/GetByName">
    </Operation>
</BtsActionMapping>
```

Codevoorbeeld 2. Mapping tussen BizTalk-operation en SOAP Action



Abbeelding 4. De SOAP Action header-sectie van de adapter-properties

ressante features te verzinnen. Een nog grotere samenwerking van de twee roept echter minimaal met zo veel vragen als antwoorden op. Totdat Microsoft de details van BizTalk vNext als onderdeel van het toekomstige 'oslo' SOA-platform openbaar maakt, zullen deze vragen onbeantwoord blijven. Met BizTalk Server 2006 R2 is de eerste grote stap gezet richting een nauwe samenwerking van de twee technologieën. Het brengt vooral BizTalk-ontwikkelaars goed gereedschap om met WCF te integreren.

Randal van Splunteren is consultant bij Aviva Solutions (www.avivasolutions.nl). Hij richt zich op integratieoplossingen op basis van Microsoft-technologie in het algemeen en Microsoft BizTalk Server in het bijzonder. Randal is bereikbaar via: randal.van.splunteren@avivasolutions.nl

Referenties

- BizTalk Server 2006 R2: <http://www.microsoft.com/biztalk/default.mspx>
- Microsoft WCF Line of Business Adapter SDK: <http://www.microsoft.com/biztalk/technologies/wcflobadaptersdk.mspx>
- The Future of BizTalk/WCF/WF: <http://www.winterdom.com/weblog/2007/09/03/TheFutureOfBizTalkWCFWF.aspx>
- Microsoft 'oslo': geekswithblog.net/cyoung/articels/116456.aspx