

Force.com werkt met BizTalk Server 2010

SALESFORCE GEÏNTEGREERD MET BIZTALK

Steef-Jan Wiggers

Integratie tussen Salesforce en on-premise applicaties kan worden verzorgd met behulp van BizTalk, dat je in staat stelt data van en naar Salesforce te laten gaan. In dit artikel komt een van de mogelijkheden naar voren om data uit Salesforce te verkrijgen met behulp van BizTalk Server.

Software as a Service (SaaS) is het opkomende fenomeen waar nu vrijwel iedereen in de IT industrie wel mee bekend is. Een SaaS oplossing bevindt zich in de cloud en is schaalbaar, betrouwbaar en overal beschikbaar. Salesforce is een aanbieder van een Customer Relation Management (CRM) SaaS oplossing, welke in de cloud beschikbaar is. Een organisatie kan er voor kiezen om van een dergelijke oplossing gebruik te maken in combinatie met andere oplossingen in de Cloud en/of binnen de organisatie (on-premise). Het laatste is niet ondenkbaar, omdat organisaties om verschillende redenen bepaalde gegevens en processen graag binnen de bedrijfsmuren willen houden. Er ontstaat daardoor dus behoefte aan integratie tussen beide.

SalesForce.com

SalesForce heeft nu al meer dan twee miljoen gebruikers. De oplossing stelt je in staat als menig ander CRM pakket, bijvoorbeeld Microsoft Dynamics CRM, de klantgegevens en -interactie te beheren. Alles wat de gebruiker nodig heeft is centraal opgeslagen en toegankelijk vanaf elke willekeurige locatie. Salesforce biedt haar oplossing als een dienst op het internet, ofwel software die als een online dienst wordt aangeboden. Daarmee zorgt Salesforce voor de installatie, onderhoud en beheer en de gebruiker benadert de software over het internet. De gebruiker betaalt uiteindelijk alleen voor het gebruik en hoeft software niet aan te schaffen. Salesforce als SaaS-aanbieder wil natuurlijk een zo hoog mogelijke beschikbaarheid garanderen voor de CRM dienst. Daarom zal de data op meerdere servers en locaties zijn gepositioneerd. De beschikbaarheid en stabiliteit worden hierdoor fors vergroot, zoals ook geldt voor Windows Azure als Platform as a Service (PaaS) dienst.

Force.com

Voor ontwikkelaars is het onderliggende Force.com platform interessant. Dit platform, waar Salesforce op is gebouwd, wordt beschouwd als een Platform as a Service, omdat het aan ontwikkelaars wordt aangeboden om bestaande en nieuwe applicaties te bouwen en te verbeteren. Binnen een Force.com applicatie of web service wordt een Java/C# lijkende taal gebruikt, genaamd APEX. De taal is specifiek voor dit platform en is eenvoudig op te pakken als je met eerder genoemde talen bekend bent. Elk object in een

Force.com applicatie kan communicatie over en weer aangaan met de buitenwereld via SOAP en REST service interfaces. Hierdoor worden dus diverse integratie scenario's mogelijk, waarbij data ergens anders opgeslagen is.

Met behulp van BizTalk Server kan een ontwikkelaar verzoeken naar Salesforce.com en/of Force.com applicaties sturen. Hij heeft daarbij de mogelijkheid te kiezen voor een adapter gebouwd door TwoConnect of Nuclei, zodat hij de APEX API niet hoeft te leren begrijpen en de tekortkomingen van BizTalk adapters kan omzeilen. De TwoConnect adapter ondersteunt zelfs push/pull interactie met Salesforce, hetgeen met standaard adapters van BizTalk niet kan. Toch kan een ontwikkelaar met de WCF adapters van BizTalk communiceren met web service interfaces gebouwd in Force.com. Menig organisatie geeft namelijk voorkeur aan communicatie met web services en het consumeren van web services is bij iedere ontwikkelaar wel bekend. De focus in dit artikel ligt op het gebruik van de WCF adapters in BizTalk. In het onderstaande figuur wordt een dergelijk scenario geschetst.

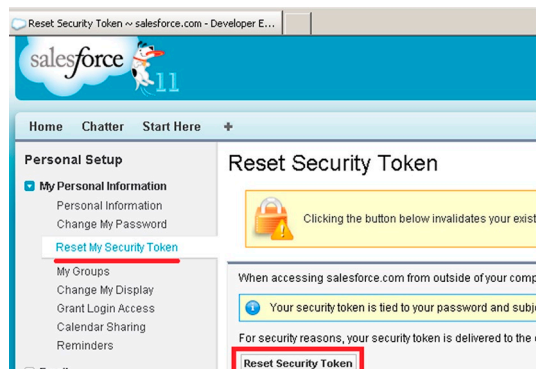


FIGUUR 1. AANROEP VAN EEN SERVICE GEBOUWD IN FORCE.COM.

Toegang tot Force.com

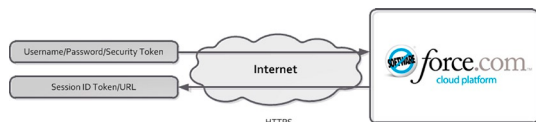
Om in Force.com zaken voor elkaar te krijgen zal eerst een account moeten worden aangemaakt via <https://na7.salesforce.com/>. Wanneer een ontwikkelaar eenmaal een account heeft dan is een 'sandbox' omgeving beschikbaar. Vervolgens is hij in staat om applicaties en services te ontwikkelen, de omgeving te beheren, objecten aan te maken, enzovoort. Een van de belangrijke zaken, die een ontwikkelaar zal moeten regelen is toegang tot zijn services, welke hij zal gaan ontwikkelen. De toegang tot in Force.com gemaakte services en applicaties wordt geboden via een standaard protocol als HTTP of de veilige variant HTTPS. De veiligheidseisen voor authenticatie zijn hoog en vindt plaats via een combinatie van gebruikersnaam, wachtwoord en veiligheidstoken. Het laatste is te verkrijgen in Force.com en een ontwikkelaar kan

door in te loggen met het verkregen account naar zijn persoonlijke informatie gaan. Daar kan een verzoek gedaan worden voor een security token via het "Reset My Security Token" (zie figuur 2). Vervolgens zal naar het email adres, dat is gekoppeld aan het account, een email met het token worden gestuurd.



FIGUUR 2. VERZOEK VOOR EEN SECURITY TOKEN.

Een van de belangrijkste service operaties binnen Salesforce.com is de login. Elke call naar een operatie vereist een sessie token in de HTTP header. Het wordt verkregen via het account, dat de operatie aanroept. Gebruikersnaam, wachtwoord en veiligheids-token worden naar de login functie gestuurd. Als deze valide zijn dan zal de functie een sessie token en service URL retourneren (zie figuur 3). De levensduur van de sessie token is wel beperkt en na de service operatie aanroep wordt time-out counter gereset.

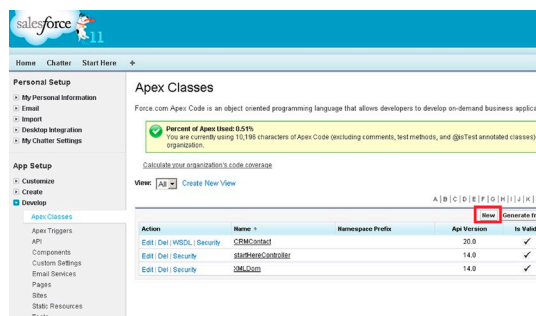


FIGUUR 3. VERKRIJGEN VAN EEN SESSIE TOKEN EN SERVICE URL.

Bouwen van APEX web service

Out-of-box API van Salesforce levert een uitgebreide set aan operaties, wat een ontwikkelaar in staat stelt gegevens in Salesforce aan te passen, te verwijderen en/of nieuwe toe te voegen in objecten, bijvoorbeeld het Account object. De API voorziet echter niet in alles, bijvoorbeeld het aanpassen van diverse objecten tegelijkertijd. In force.com kun je echter zelf services bouwen waarbij je wel in staat bent acties te bundelen of dat je bijvoorbeeld van een object maar een bepaalde subset aan data retourneert. Zowel de standaard als zelfgemaakte services leveren een makkelijk te consumeren WSDL definitie.

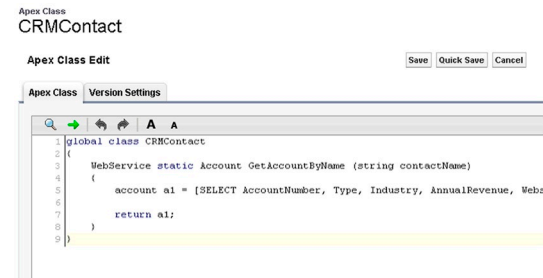
Een web service kun je bouwen met APEX code, welke is georganiseerd in klassen. Een APEX code klasse kan als web service



FIGUUR 4. AANMAKEN NIEUWE APEX KLASSE.

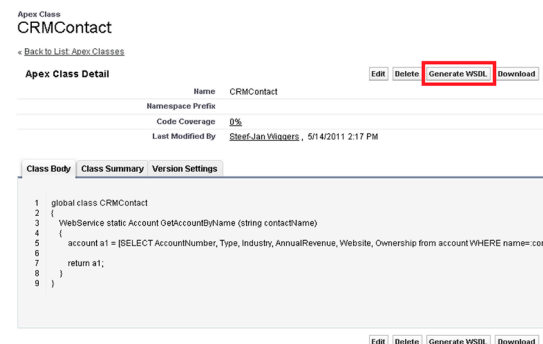
blootgesteld worden door voor de operatiename het sleutelwoord web service te plaatsen. Binnen het force.com portaal kan de ontwikkelaar via zijn account/setup navigeren naar "Develop" en klikken op "Apex Classes". Hij ziet vervolgens een overzicht, waarbij hij een nieuwe APEX klasse kan aanmaken.

Als de ontwikkelaar een nieuwe APEX klasse heeft aangemaakt krijgt hij beschikking over een code editor, waar hij de code voor de klasse in kan schrijven zoals in het onderstaande voorbeeld.



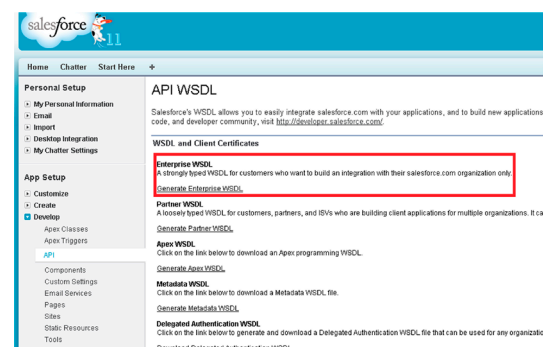
FIGUUR 5. APEX KLASSE CODE EDITOR.

Met behulp van save wordt de code gecompileerd en opgeslagen. De volgende actie is het generen van een WSDL, die de ontwikkelaar kan gebruiken om de service te kunnen consumeren vanuit BizTalk of .NET.



FIGUUR 6. GENEREN VAN DE WSDL.

De gegenereerde WSDL wordt in de browser getoond en kan lokaal op een schijf van het werkstation van de ontwikkelaar worden opgeslagen. Om de service te kunnen consumeren in BizTalk of met behulp van een .NET client zal eveneens de Enterprise WSDL, waar zich onder andere de login operatie bevindt, moeten worden gegenereerd en lokaal worden opgeslagen. Zoals eerder is beschreven dient namelijk voor elke web service operatie aanroep een sessie token te worden meegegeven. De Enterprise WSDL kan gevonden worden in menu item API binnen 'Develop'.



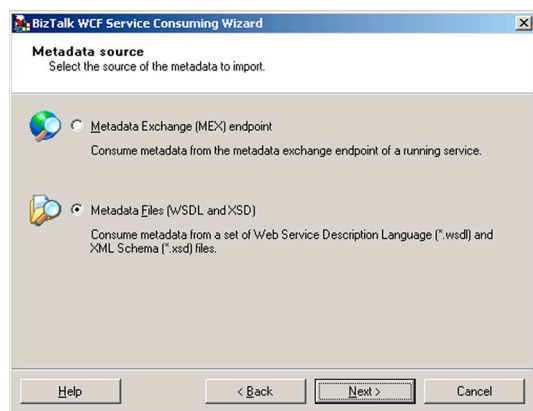
FIGUUR 7. GENEREN VAN ENTERPRISE WSDL.

Binnen API zal de Enterprise WSDL moeten worden gegenereerd en opgeslagen. Deze standaard WSDL is specifiek bedoeld voor integratie doeleinden.

De WSDL verschijnt in de browser, zodra de ontwikkelaar op 'Generate Enterprise WSDL' heeft geklikt. Hij kan vervolgens de WSDL definitie lokaal op zijn werkstation opslaan.

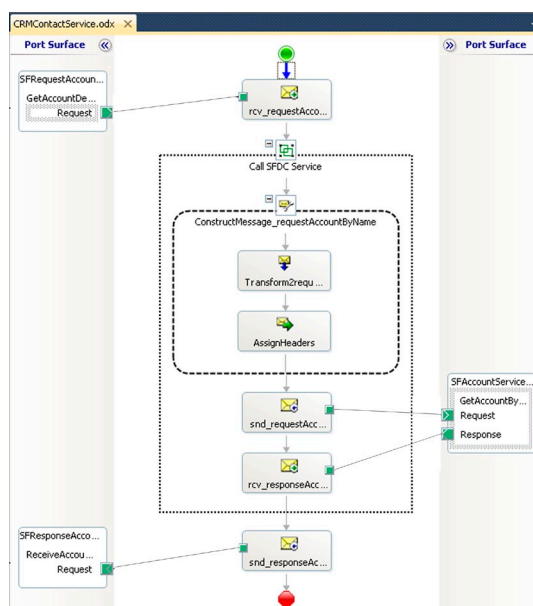
Consumeren van service in BizTalk

Een ontwikkelaar kan de zelfgemaakte service in BizTalk consumeren. Als eerste zal hij een BizTalk project moeten aanmaken alvorens hij door middel van Add, Generate Items via Consume WCF Service de WSDL van de zelfgemaakte service kan importeren. De ontwikkelaar navigeert naar de desbetreffende WSDL, die lokaal op een schijf is opgeslagen.



FIGUUR 8. IMPORTEREN VAN DE WSDL FILE.

Alvorens de WSDL wordt geïmporteerd kan hij de namespace aanpassen of de default hanteren. Na de import actie wordt een binding- en orchestration file (.odx) gegenereerd. De orchestration neemt de naam aan van de geïmporteerde WSDL en heeft een enkele Port Type definitie, die alle operaties bevat die in de WSDL voorkomen. Daarnaast bevat het ook de multi-part bericht definities voor elk in- en output bericht. De binding file is een WCF-Custom adapter binding, die de ontwikkelaar in staat

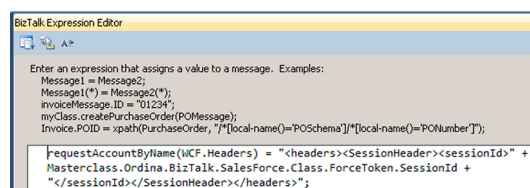


FIGUUR 9. CONSUMEREN VAN ZELFGEMAakte SALESFORCE.COM SERVICE IN EEN ORCHESTRATION

stelt de aanroep naar de web service te configureren. De ontwikkelaar kan de orchestration verder gebruiken om het geheel af te maken.

Zoals eerder beschreven zal bij elke aanroep van een service een sessietoken moeten worden meegeleverd. Een ontwikkelaar kan er voor kiezen om een extra aanroep binnen een orchestration te doen. Daarvoor zal hij de Enterprise WSDL binnen de orchestration moeten consumeren, door deze te importeren. Een andere mogelijkheid is een in c# gemaakte helper klasse te maken en verantwoordelijkheid van het ophalen van de sessietoken daar te laten. De klasse kan hierdoor worden hergebruikt in andere orchestrations.

In figuur 9 is een orchestration weergegeven waar in de 'Assign-Headers' Expression shape de aanroep voor een sessietoken naar de helper klasse wordt verzorgd. De sessietoken wordt vervolgens in de header van het request bericht geplaatst.

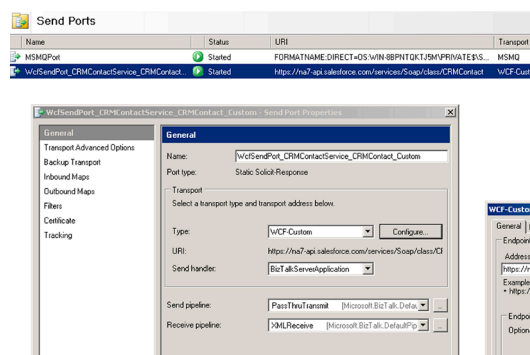


FIGUUR 10. PLAATSEN VAN DE SESSIE-TOKEN IN DE HEADER VAN REQUEST BERICHT.

De code van de methode ForceToken, die wordt aangeroepen in de expression shape staat hieronder weergegeven.

```
private static string GetSession()
{
    DateTime now = DateTime.Now;
    TimeSpan diff = now.Subtract(_sessionDate);
    if (_sessionId == string.Empty || (diff.Minutes > 60))
    {
        string uname = "sj.wiggers@xxxx.com";
        string password = "xxxxxx";
        string securityToken = "xxxxxxxxxxxxxxxxxxxxxx";
        SFSvcRef.SforceService proxy = new SFSvcRef.SforceService();
        proxy.Url = "https://login.salesforce.com/services/Soap/c/20.0";
        SFSvcRef.LoginResult result = proxy.login(uname, password + securityToken);
        _sessionId = result.sessionId;
    }
    return _sessionId;
}
```

Wanneer de orchestration klaar is en de ontwikkelaar het voorzien heeft van een applicatie naam en strongkey kan het naar de BizTalk runtime worden uitgerold. Daar kunnen verder de fysieke poorten worden geconfigureerd. Voor de zelfgemaakte service is



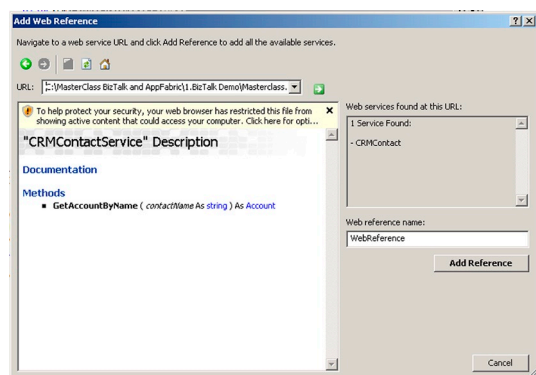
FIGUUR 11. SEND PORT SALESFORCE SERVICE

een binding file aanwezig, die is ontstaan na importeren van de WSDL. Door importeren van deze binding file wordt automatisch een send port, verantwoordelijk voor communicatie met de zelfgemaakte service in force.com gegenereerd.

Naast deze poort kan de ontwikkelaar ook de andere poorten configureren, de bijhorende host instance(s) van de applicatie herstarten en vervolgens de applicatie zelf starten. Met behulp van BizTalk kan nu de service worden aangeroepen.

.NET

Het is ook mogelijk om vanuit .NET zelf een in force.com gemaakte service te consumeren. De ontwikkelaar kan dit doen door eerst een web reference toe te voegen aan zijn project van zowel de service als Enterprise WSDL.



FIGUUR 12. TOEVOEGEN VAN EEN WEB REFERENCE.

Met deze acties worden proxy's gegenereerd voor de ontwikkelaar, die hij nodig heeft om de service te kunnen aanroepen. De onderstaande code geeft weer hoe eerst de login wordt aangeroepen om een sessietoken te verkrijgen alvorens met deze token de service zelf wordt aangeroepen. De aanroep van de service resulteert uiteindelijk in een subset van gegevens van een account object.

```
private void btnAccountDetails_Click(object sender, EventArgs e)
{
    //Get Sessionid first
    string uname = "sj.wiggers@xxxx.com";
    string password = "xxxxxx";
    string securityToken = "xxxxxxxxxxxxxxxxxxxx";
    SFSvcRef.SforceService proxy = new SFSvcRef.
    SforceService();

    proxy.Url = "https://login.salesforce.com/services/Soap/c/20.0";

    SFSvcRef.LoginResult result = proxy.login(uname, password +
    securityToken);

    SFGGetAccountSrv.CRMContactService crm =
    new SFGGetAccountSrv.CRMContactService();
    SFGGetAccountSrv.Account account =
    new SFGGetAccountSrv.Account();

    SFGGetAccountSrv.SessionHeader session
    = new SFGGetAccountSrv.SessionHeader();
    session.sessionId = result.sessionId;
    crm.SessionHeaderValue = session;
    account = crm.GetAccountByName
    ("Unicorn Advertisement");

    txtAccountDetails.Text = "AccountNumber : " +
    account.AccountNumber;
    txtAccountDetails.Text += "Ownership : " +
    account.Ownership;
}
```

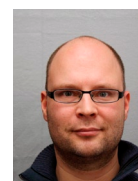
Conclusie

In dit artikel is een aspect van integratie met Salesforce aan bod gekomen. Er zijn echter veel meer patronen denkbaar als het gaat om integratie met Salesforce. Salesforce kent ook het concept van "outbound messaging", waarbij verandering aan een object in Salesforce resulteert in een uitgaand bericht. Dit bericht kan dan door bijvoorbeeld BizTalk worden opgevangen. Verder is een aantal zaken rondom Salesforce in dit artikel niet naar voren gekomen zoals het zoeken van data met behulp van Salesforce Object Query Language (SOQL), wat vergelijkbaar is met T-SQL, of het Salesforce Object Search Language, wat een tekstgerelateerde zoekactie uitvoert op entiteiten. Integratie met Salesforce kent vele facetten en enkele zijn in dit artikel belicht. Tevens zijn enkele mogelijkheden, welke de ontwikkelaar heeft binnen force.com portaal aan bod gekomen.



Links:

- Software as a Service: http://nl.wikipedia.org/wiki/Software_as_a_Service
- Salesforce: <https://www.salesforce.com/>
- Experience of BizTalk and Salesforce.com Integration: <http://geekswithblogs.net/BizTalkUnleashed/archive/2009/06/19/experience-of-biztalk-and-salesforce-com-integration.aspx>
- Orchestrating the Cloud: Part I – Creating and Consuming A Google App Engine Service From BizTalk Server: <http://seroter.wordpress.com/2009/10/01/orchestrating-the-cloud-part-i-creating-and-consuming-a-google-app-engine-service-from-biztalk-server/>
- Orchestrating the Cloud: Part II – Creating and Consuming a Salesforce.com Service From BizTalk Server: <http://seroter.wordpress.com/2009/10/11/orchestrating-the-cloud-part-ii-creating-and-consuming-a-salesforce-com-service-from-biztalk-server/>
- BizTalk, Salesforce and MSMQ – Part I: <http://soa-thoughts.blogspot.com/2010/08/biztalk-salesforce-and-msmq-part-i.html>
- BizTalk, Salesforce and MSMQ – Part II: <http://soa-thoughts.blogspot.com/2010/08/biztalk-salesforce-and-msmq-part-ii.html>
- TwoConnect Adapter: <http://www.twoconnect.com/TwoConnect-BizTalk-Adapter-for-Salesforce.com.aspx>
- Salesforce .NET: <http://code.google.com/p/salesforce-dotnet/>
- SaleForce For Dummies boek: http://sales.force.com/bookoffer/new_dummies_showNewCode_n1
- Microsoft BizTalk 2010: Line of Business Systems Integration: RAW: <https://www.packtpub.com/microsoft-biztalk-2010-line-of-business-systems-integration/book>
- Developer Force: <http://developer.force.com>



.....
Steef-Jan Wiggers is Architect bij de unit Microsoft Solutions van Ordina en MVP BizTalk Server. Hij is te bereiken via Steef-Jan. Wiggers@ordina.nl en heeft een blog over o.a. BizTalk (<http://soa-thoughts.blogspot.com>)