

Aan de slag met Report Builder

RAPPORTAGE VOOR EINDGEBRUIKERS

Tot voor kort was het maken van rapporten in SQL Server Reporting Services voorbehouden aan ontwikkelaars. Met de nieuwe versie van Reporting Services en de nieuwe Report Builder wordt dit ook voor eindgebruikers mogelijk. Dit artikel laat aan de hand van een uitgewerkt voorbeeld zien wat Report Builder is en wat je moet doen vóórdat een eindgebruiker aan de slag kan. De informatie is gebaseerd op de Microsoft SQL Server June 2005 Community Technology Preview en de Northwind sample database.

De marketing- en sales-afdeling van het bedrijf Northwind wil inzicht krijgen in orders met een totaalbedrag boven de \$ 5000. Medewerkers van deze afdeling moeten zo'n rapport zelf met de Report Builder kunnen samenstellen. Maar voordat ze aan de slag kunnen, moet er eerst een Report Model worden gemaakt door de ontwikkelaar.

Het maken van een Report Model

Het Report Model (ook wel het Semantic Model ofwel het semantische model genoemd) vormt straks de basis voor rapporten die met de Report Builder kunnen worden gemaakt. Dit nieuwe model moet de kloof dichten die vaak bestaat tussen de ontwikkelaar en de eindgebruiker. Een eindgebruiker denkt in termen van: een klant heeft orders en een order heeft orderregels. Een ontwikkelaar denkt echter in termen van: de tabel Orders heeft een foreign key-relatie met de tabel Customers en de tabel Order Details heeft een foreign key-relatie met de tabel Orders. Tot voor kort werd alleen het perspectief van de ontwikkelaar ondersteund in Reporting Services. Het nieuwe Report Model moet het perspectief van de eindgebruiker vertegenwoordigen.

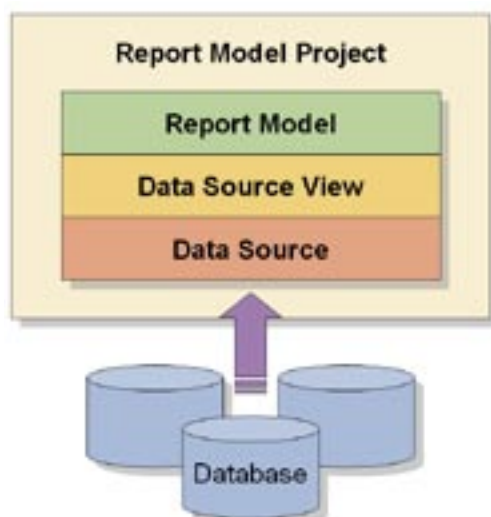
In de SQL Server Business Intelligence (BI) Development Studio vind je hiervoor een nieuwe projecttemplate genaamd Report Model Project. De nieuwe projecttemplate bevat alle ingrediënten om te komen tot zo'n Report Model. In chronologische volgorde:

een Data Source, een Data Source View en het Report Model zelf (zie afbeelding 1). Aan de hand van een uitgewerkt voorbeeld zullen deze onderdelen één voor één de revue passeren. Het voorbeeldproject uit dit artikel is te downloaden via de website van het .NET Magazine en Qurius ETX op www.quriusetx.nl

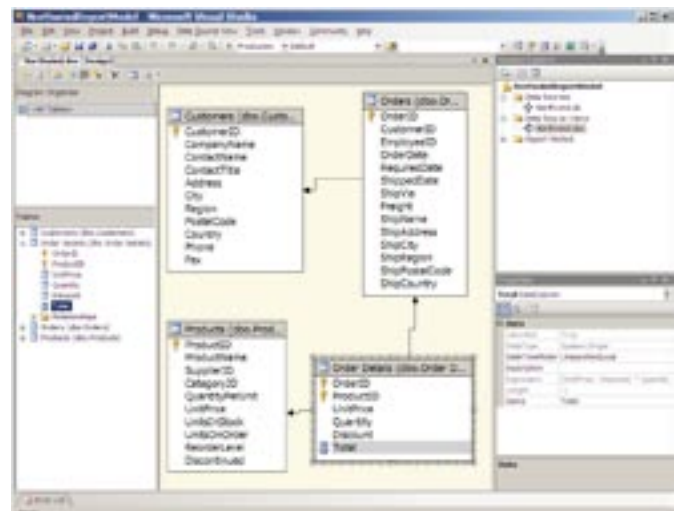
Data Source

We beginnen met het aanmaken van een Report Model-project en een Data Source. De Data Source representeert de connectie met de database waarin de tabellen staan waarover we gaan rapporteren. In dit geval de Northwind-database.

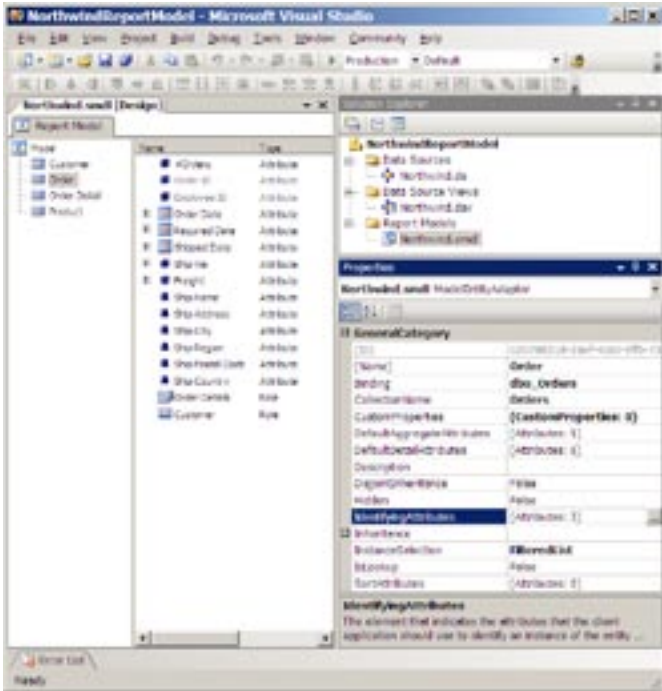
1. Start de BI Development Studio.
2. Kies voor New Project en selecteer vervolgens bij Business Intelligence Projects de Report Model Project-template. Noem het nieuwe project NorthwindReportModel en kies OK.
3. Klik met de rechtermuisknop op de Data Sources-folder in de Solution Explorer en kies Add New Data Source.
4. Doorloop de wizard om een nieuwe connectie aan te maken naar de Northwind-database (download en installeer zondig de Northwind-database van de Microsoft-site of gebruik het instnwnd.sql script dat onderdeel uitmaakt van de download bij dit artikel).
5. Noem de nieuwe data source Northwind en kies Finish om de wizard te beëindigen. In de Solution Explorer staat nu een Northwind.ds-file.



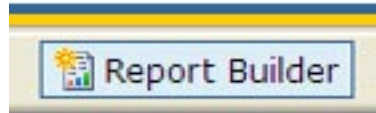
Afbeelding 1. De onderdelen van een Report Model.



Afbeelding 2. De Data Source View Designer.



Afbeelding 3. Het standaard gegenereerde model.



Afbeelding 4. De nieuwe Report Builder-knop op de Report Manager-site.

- **Role:** de relatie tussen twee entiteiten. Hiermee kan aan de eindgebruiker duidelijk worden gemaakt dat een klant orders heeft en dat een order bestaat uit orderregels.

In eerste oogopslag lijkt zo'n Report Model misschien niets anders dan een vermomde Data Source View, maar niets is minder waar. Het Report Model voegt namelijk een heleboel informatie toe aan het Data Source View-model. Zo kun je in het Report Model entiteiten en attributen voor de eindgebruiker verbergen of een andere naam geven of je kunt aangeven dat het attribuut Phone niet aggregaerbaar is. Dit helpt de eindgebruiker straks bij het maken van zinnige rapporten. Het Report Model biedt dus de mogelijkheid de interactie van de eindgebruiker met het model in Report Builder te beïnvloeden. Gelukkig beschikt de BI Development Studio over een Report Model Wizard om snel een basis Report Model te genereren op basis van de aanwezige Data Source View. Tabellen, kolommen en relaties zullen zich immers vertalen naar entities, attributen en roles.

Data Source View

In de volgende stappen zullen we een zogenaamde Data Source View gaan aanmaken. De Data Source View representeert als het ware het fysieke datamodel waaraan in een latere fase het Report Model wordt gekoppeld.

6. Klik met de rechtermuisknop op de Data Source Views-folder in de Solution Explorer en kies Add New Data Source View.
7. Doorloop de wizard en kies Northwind als Data Source.
8. Vervolgens worden alle beschikbare tables en views van de betreffende Data Source getoond. Kies in dit geval de volgende tabellen: Customers, Orders, Order Details en Products. Kies Next om verder te gaan.
9. Accepteer Northwind als naam voor de Data Source View en kies Finish om de wizard te beëindigen. In de Solution Explorer staat nu een Northwind.dsv-file. Dubbelklik op deze file.
10. Klik met de rechtermuisknop op de tabel Order Details en kies New Named Calculation.
11. Voer Total in als Column name en voer de volgende expressie in: $(UnitPrice - Discount) * Quantity$. Kies vervolgens voor OK. Het resultaat is een nieuwe kolom Total onder aan de tabel Order Details. Het icoontje ervoor geeft aan dat het een berekende kolom betreft. Zie afbeelding 2 voor het eindresultaat. Kies Save All om de verrichte werkzaamheden op te slaan.

De Data Source View zal meestal bestaan uit een subset van tabellen en kolommen uit een database, eventueel aangevuld met kolommen op basis van expressies. Merk op dat het ook mogelijk is tabellen in de Data Source View te baseren op eigen queries (of stored procedures).

Report Model

Nu we beschikken over een Data Source View kunnen we starten met het Report Model. Het Report Model wordt beschreven in een op XML-gebaseerde taal genaamd Semantic Model Definition Language (SMDL). Het Report Model beschrijft de volgende typen objecten:

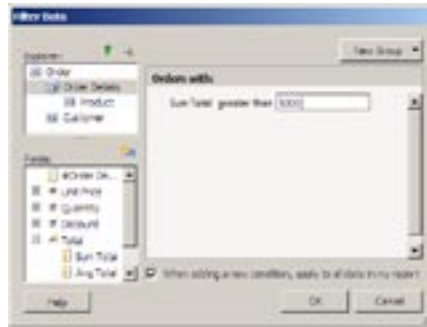
- **Entity:** een voor de eindgebruiker vatbaar begrip zoals Order of Customer.
- **Attribute:** een eigenschap van een entity. Bijvoorbeeld Order Date of Number of Customer Orders.

12. Klik met de rechtermuisknop op de Report Models-folder in de Solution Explorer, kies Add New Report Model en doorloop de volledige wizard waarbij alle defaults worden geaccepteerd.
13. Dit resulteert uiteindelijk in een Northwind.smdl-file in de Solution Explorer. Dubbelklik op deze file. Het Report Model opent nu in de designer. Helemaal links zie je de entiteiten staan en in het midden verschijnen de bijbehorende attributen. Zie ook afbeelding 3.
14. Kies links de entiteit Order en open via het Properties-venster de collection editor (knop met drie puntjes) achter de property IdentifyingAttributes. Verwijder alle attributen behalve Order Date en Freight. Voeg Company Name (via Customer) toe. Dit worden de drie attributen die de eindgebruiker initieel op zijn rapport ziet verschijnen wanneer hij de entiteit Order op zijn rapport sleept. Sluit de editor.
15. Kies vervolgens het attribuut Employee ID van de entiteit Order en zet de Hidden-property op True. Dit voorkomt dat de eindgebruiker dit attribuut ziet.
16. Kies ten slotte het attribuut Total en zet de Format-property op c2. Doe hetzelfde voor Sum Total. Hiermee geven we aan dat het een attribuut betreft dat een bedrag representeert en dat we 2 decimalen willen tonen. De eindgebruiker hoeft dit nu zelf niet meer te doen in zijn rapport.
17. Nu gaan we alles wat we tot nu toe hebben gebouwd publiceren. Klik met de rechtermuisknop op NorthwindReportModel in de Solution Explorer en kies Deploy. (Indien je SQL Server 2005 als named instance draait kan het zijn dat er nu een connectiefout optreedt. Open in dat geval de properties van het NorthwindReportModel en controleer of de TargetServerURL gelijk is aan `http://localhost/ReportServer$InstanceName`.)

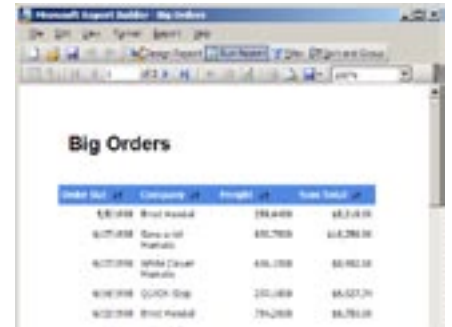
Tot zover het werk van de ontwikkelaar. Nu het Report Model is gepubliceerd naar de Report Server kan een beheerder rechten toewijzen. De toegang tot het Report Model kan zelfs tot op entiteit- en attribuutniveau worden bepaald! Heeft een gebruiker geen rechten op een entiteit of attribuut, dan kan hij deze later ook niet selecteren in Report Builder. Het toewijzen van rechten gebeurt via de SQL Server Management Studio. Daarna is het de beurt aan de eindgebruiker.



Afbeelding 5. Het Big Orders-rapport in design-modus.



Afbeelding 6. Eenvoudig gegevens filteren in Report Builder.



Afbeelding 7. De uitvoer van een rapport.

Het maken van een rapport met Report Builder

De Report Builder is een nieuwe WinForms smart client-applicatie in .NET 2.0. Je kunt de Report Builder starten door te klikken op de Report Builder-knop in het menu van de Report Manager site (zie afbeelding 4) of direct met de volgende URL: <http://localhost/ReportServer/ReportBuilder/ReportBuilder.application>.

1. Start de Report Manager-site op <http://localhost/Reports> (indien je SQL Server 2005 als named instance draait, moet je waarschijnlijk de instance-name met een \$-teken achter de genoemde URL typen, bijvoorbeeld: [http://localhost/Reports\\$InstanceName](http://localhost/Reports$InstanceName)).
2. Druk op de Report Builder-knop. Via click-once deployment in .NET 2.0 zal de Report Builder nu opstarten. Kies vervolgens het Northwind Report Model als databron.

De Report Builder opent nu in design-modus. Wat direct opvalt, is de intuïtieve en gebruiksvriendelijke Office look-a-like interface, een beetje vergelijkbaar met Powerpoint 2003. Helemaal rechts zie je een aantal standaard reporttemplates, helemaal links de entiteiten en attributen (fields) uit het Northwind-model. In het midden zie je een nog lege reporttemplate.

3. Klik op de titel van het rapport en geef het de titel Big Orders.
4. Sleep vanuit het Explorer-venster aan de linkerkant de entiteit Order op de tabel in het rapport. Meteen verschijnen de drie kolommen die we eerder hadden opgegeven in het Report Model. Merk op dat in het Explorer-venster de context is verplaatst naar Order. De aan Order gerelateerde entiteiten worden nu ingesprongen getoond.
5. Kies Order Details en sleep het Sum Total-field naar een positie direct achter de laatste kolom van de tabel op het rapport. Sum Total wordt nu als kolom getoond met een \$-teken en twee cijfers achter de komma. Herinner dat we dit eerder in het Report Model hebben aangegeven; zie afbeelding 5.
6. Echter, we zijn alleen geïnteresseerd in orders met een ordertotaal van méér dan \$5000. Klik op de Filter-knop onder de menubalk.

Report Builder	Report Designer
Doelgroep is business-users	Doelgroep is IT-professionals en ontwikkelaars
Automatisch gegenereerde queries op basis van een semantisch model	Eigen queries
Eenvoudige rapporten op basis van een vaste set templates	Geavanceerde 'free-form'-rapporten
Click-once WinForms-applicatie	Designer geïntegreerd in Visual Studio / SQL Server BI Development Studio
Kan niet met rapporten werken die zijn ontwikkeld in Report Designer	Kan werken met rapporten die zijn ontwikkeld in Report Builder

Tabel 1. Belangrijkste verschillen tussen de Report Builder en de Report Designer

7. Kies Order Details en dubbelklik vervolgens op het Sum Total-field. Bewerk de expressie die rechts verschijnt naar 'Sum Total greater than 5000' (zie afbeelding 6).
8. Sla het rapport op onder de naam Big Orders. Het rapport is nu later via de Report Manager-site weer op te vragen.

Het resultaat van het opgeslagen rapport is overigens gewoon een rapportdefinitie in RDL (Report Definition Language). Deze zou dus met de Report Designer in de BI Development Studio kunnen worden aangepast. Andersom geldt dat helaas niet, want rapporten die zijn ontwikkeld in de Report Designer kunnen niet in de Report Builder worden bewerkt. Zie tabel 1 voor een overzicht van de belangrijkste verschillen tussen de Report Builder en de Report Designer.

9. Ten slotte kunnen we het rapport ook binnen de Report Builder 'uitvoeren'. Klik hiertoe op de Run Report-knop.

Zonder dat de gebruiker het merkt, wordt nu een webservice-call gedaan naar een nieuwe Report Server-webservice genaamd ReportExecution2005. De koppeling tussen het Report Model en de Data Source View maakt het mogelijk om op basis van het Report Model queries af te vuren op de database. Een nieuwe component van Reporting Services, de Query Translator, is hiervoor verantwoordelijk. Zo kunnen acties van eindgebruikers op het Report Model uiteindelijk toch worden vertaald in gewone database-queries. Het resultaat staat weergegeven in afbeelding 7.

Indien je nu bij een willekeurige order op de Company Name klikt verschijnt er een detailoverzicht van de betreffende Company. Indien je op een Sum Total klikt, verschijnen de onderliggende Order Details. Dit gedrag wordt 'Infinite Drillthrough' genoemd en biedt de eindgebruiker de mogelijkheid om bijna eindeloos te navigeren tussen gerelateerde entiteiten. En dat terwijl de gebruiker hier niets voor hoeft te doen. De navigatiepaden worden afgeleid uit het onderliggende Report Model.

Snel ad-hoc rapportages

De eindgebruiker kan met Report Builder snel eenvoudige 'ad-hoc'-rapportages fabriceren op basis van een Report Model. De ontwikkelaar speelt een belangrijke rol in het maken van zo'n Report Model. Voor geavanceerde rapportages blijft men toegewezen tot de Report Designer in de BI Development Studio.

René Sterenburg is als technisch architect werkzaam bij Qurius ETX (www.etx.nl) in Rijswijk. Voor vragen en opmerkingen is hij te bereiken via renes@quriusetx.nl.

Nuttige internetadressen

SQL Server 2005 Reporting Services (SSRS):
www.microsoft.com/technet/prodtechnol/sql/2005/2005ssrs.msp
 MSDN Webcast End-User Ad Hoc Reporting with SSRS:
www.microsoft.com/events/webcasts/ondemand.msp
 SQL Server 2005: www.microsoft.com/sql/2005/
 Voorbeeldcode: www.microsoft.nl/netmagazine10