

Op ontdekkingsreis naar de nieuwe functionaliteiten

VISUAL STUDIO 2008: EEN STERK VERBETERDE ONTWIKKELOMGEVING

Ten tijde van de komst van Visual Studio 2008 en het .NET Framework 3.5 wordt er ook veel geschreven over LINQ, de taalverbeteringen en de uitbreidingen in Team Foundation Server. Naast deze vernieuwingen is ook de ontwikkelomgeving van Visual Studio ingrijpend aangepast. In dit artikel wordt een overzicht gegeven van de belangrijkste verbeteringen in de Visual Studio-omgeving. Omdat Visual Studio het belangrijkste gereedschap is voor een ontwikkelaar, is het interessant om deze verbeteringen onder de loep te nemen.

Visual Studio 2008 voorziet de ontwikkelaar van een groot aantal verbeteringen, die zijn gericht op productiviteit. Er is door Microsoft de afgelopen jaren goed geluisterd naar de gebruikers van Visual Studio. Een flink aantal wijzigingen zijn wensen van de ontwikkelaars die door Microsoft wordt ingevuld. Hieronder volgt een opsomming van de belangrijkste aanpassingen. Dit is geen uitputtende lijst, maar een overzicht van de meest in het oog springende:

- Multi-targeting
- Javascript-IntelliSense
- Verbeterde Javascript-debugging
- Geneste ASP.NET master page-support at design time
- CSS-editing en lay-outondersteuning
- Splitview-ondersteuning om zowel de designer als de code te zien
- Een versnelde ASP.NET-designer
- Een WPF-designer.

Over de gehele linie is er veel aandacht besteed aan de performance van Visual Studio 2008. Zodra de tool wordt opgestart, valt de performance-verbetering op. Deze positieve performance-ervaring is te merken in het gehele product.

Multi-targeting ondersteuning

De vorige versies van Visual Studio ondersteunden alle een specifieke versie van het .NET Framework. Zo werkt Visual Studio 2005 uitsluitend met .NET Framework 2.0. Met de komst van Visual Studio 2008 is dit anders. De nieuwste versie van de ontwikkelomgeving richt zich op meer versies van het .NET Framework. De versies 2.0, 3.0 en 3.5 worden ondersteund. Concreet betekent dit dat een applicatie die is geschreven in Visual Studio 2005, en dus werkt op basis van .NET Framework 2.0, in de nieuwe versie van Visual Studio kan worden onderhouden. Deze applicatie kan zelfs gebruikmaken van de nieuwe voorzieningen in de ontwikkelomgeving die verder in dit artikel worden besproken. De multi-targeting support is een belangrijke feature van Visual Studio 2008, misschien wel de belangrijkste. Dankzij deze feature hoeft je bestaande applicaties niet meer te migreren naar de nieuwste versie van het framework wanneer je gebruik wilt maken van de nieuwe mogelijkheden van Visual Studio. De oudere applicaties die gebruikmaken van versie 1.0 of 1.1 van het .NET Framework moeten wel gemigreerd worden, omdat deze versies niet worden ondersteund. De achterliggende reden hiervoor is het feit dat de common language runtime in versie 2.0 sterk is verbeterd.

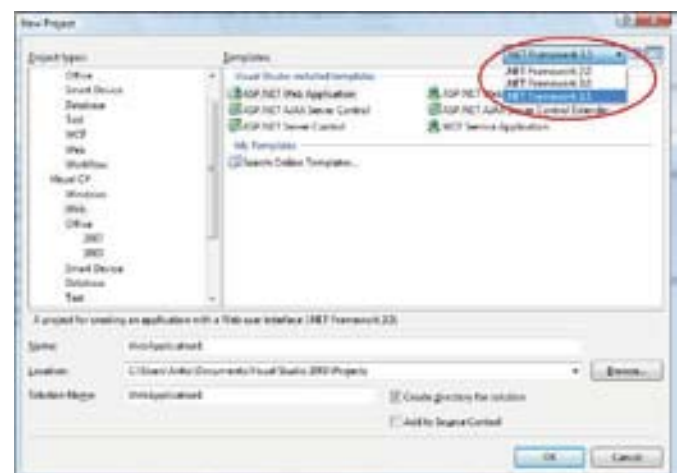
Bij het maken van een nieuw project in Visual Studio 2008 kan een target framework worden gekozen, zoals weergegeven in afbeelding 1.

Afhankelijk van deze keuze wordt de omgeving aangepast. Wanneer bijvoorbeeld wordt gekozen voor een webapplicatie op basis van het .NET Framework 2.0, dan is er een toolbox zichtbaar met ASP.NET 2.0-controls. Verder zorgt Visual Studio 2008 er voor dat naar de juiste assemblies wordt gerefereerd en dat de instellingen in de web.config juist staan. Wanneer je een assembly-referentie gaat toevoegen, dan zijn alleen die assemblies te selecteren die bij het geselecteerde .NET Framework passen; zie afbeelding 2.

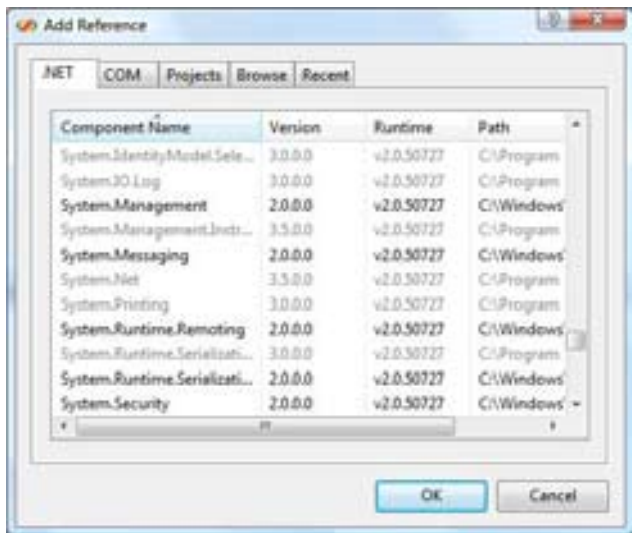
Verbeteringen voor de webontwikkelaar

Voor de webontwikkelaar is een aantal in het oogspringende verbeteringen aangebracht. Ook deze dragen bij aan een verhoogde productiviteit. De belangrijkste zijn weergegeven in tabel 1. Een voor de hand liggende en zeer handige verbetering is de splitview editing. Deze modus maakt het mogelijk om gelijktijdig de html-source en de designview te zien. Deze twee views werken goed samen. Wanneer een element wordt geselecteerd in de designview, dan wordt ook de bijbehorende source-code geselecteerd. Vanzelfsprekend werkt dit ook omgekeerd. Een splitview-scherm is te zien in afbeelding 3.

Ook is een aantal belangrijke uitbreidingen te vinden op het terrein van stylesheets. Het werken met CSS-files kan nogal complex worden. Om het werken met CSS-files overzichtelijk te houden, is een aantal contextgevoelige tool- en property-windows toegevoegd. Deze windows zijn bruikbaar in zowel de designer- als source-



Afbeelding 1. Multi-targeting in Visual Studio 2008



Afbeelding 2. Selecteren van een assembly in Visual Studio 2008

mode. Het 'manage styles window' is het basis-window. Dit window geeft aan welke styles worden gebruikt in het huidige document. De cirkels, zoals weergegeven in afbeelding 4, geven aan welke styles worden gebruikt. Wanneer je de muis over een regel beweegt, wordt de achterliggende CSS-regel met waarden getoond. Met een rechtermuisklik is het direct mogelijk door te springen naar een editor en de rule te bewerken.

Wanneer je moeite hebt met de vele en lange CSS-stylesheets, dan is de 'summary' view van de CSS-properties windows een uitkomst. Deze modus zorgt er voor dat er wordt gefilterd en dat alleen de gebruikte regels voor het huidige document worden getoond. In afbeelding 5 wordt duidelijk dat sommige regels worden gedupliceerd. De ene regel is rood doorstreept. Dit geeft aan dat de CSS-regelinstelling is overschreven met een andere waarde. Wanneer je op een waarde klikt, dan wordt de hiërarchie zichtbaar. Kortom, een zeer bruikbare functionaliteit om helderheid te verschaffen in de brij aan CSS-regels.

De laatste belangrijke vernieuwing die de moeite waard is om te beschrijven, is de 'nested master page' ondersteuning. Dit is nieuwe functionaliteit die hoog op het wensenlijstje stond van menig ASP.NET-ontwikkelaar. De master page op zichzelf is één van de meest gebruikte functionaliteiten van ASP.NET 2.0. Het is nauwelijks voor te stellen dat een webontwikkelaar geen gebruik zou maken van een master page om een consistente lay-out op te leveren. Binnen ASP.NET 2.0 was het al mogelijk geneste master pages te hebben, de tooling ondersteunde dit echter niet. Het was

Nieuwe functionaliteit	Omschrijving
Splitview editing	De mogelijkheid om zowel de 'html-source' als de 'design'-view gelijktijdig te zien.
CSS-style manager	Een tool window dat een overzicht geeft van de CSS-stylesheets met de bijbehorende regels voor de pagina die wordt bewerkt.
CSS-properties window	Een property window dat een overzicht geeft van alle toegepaste CSS-settings voor het geselecteerde html-element of de ASP.NET-server control
CSS-source view IntelliSense	De html-designer ondersteunt de mogelijkheid een element of control te selecteren en vervolgens grafisch een regel van de CSS-lijst toe te passen.
Nested master page support	De mogelijkheid om een 'root' master page te definiëren voor de gehele website. Vervolgens kunnen master pages worden gedefinieerd die afleiden van de 'root', maar een verdere invulling geven voor een speciaal type pagina. Visual Studio 2008 ondersteunt deze bestaande ASP.NET 2.0-functionaliteit ook volledig in de WYSIWYG-editor.

Tabel 1. De belangrijkste verbeteringen voor webontwikkelaars

niet mogelijk deze geneste pagina's te bewerken met de Visual Studio-editor. Met de komst van Visual Studio 2008 wordt dit anders. Nested master pages zijn een volwaardige functionaliteit geworden, ook in de designer.

Alle beschreven nieuwe webfunctionaliteiten zijn door de multi-targeting ook beschikbaar voor bestaande ASP.NET 2.0-applicaties. Bestaande applicaties kunnen gebruik blijven maken van het .NET Framework 2.0 en toch volledig profiteren van de nieuwe functionaliteiten om de productiviteit van softwareontwikkeling te verhogen.

Javascript: een eersteklas burger

Javascript wordt volledig ondersteund in Visual Studio 2008. De drie belangrijkste functionaliteiten zijn:

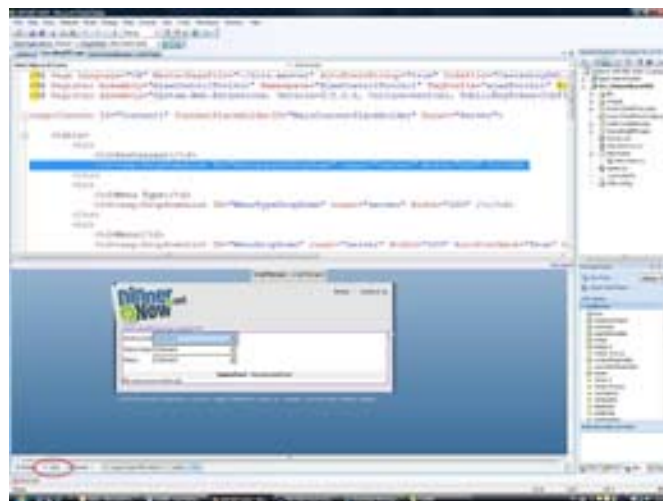
- IntelliSense
- Herbruikbare ASP.NET AJAX-controls, -behaviours en -libraries
- Debugging.

In de wereld van ASP.NET AJAX zijn dit onmisbare functionaliteiten. Deze nieuwe functionaliteiten zorgen er voor dat Visual Studio 2008 een uitstekende ontwikkelomgeving is om ASP.NET AJAX-applicaties te ontwikkelen. Omdat Javascript een dynamische taal is, was het in het verleden niet eenvoudig om IntelliSense toe te voegen. Dankzij een technologie genaamd 'type inference' beschikt Visual Studio 2008 wel over IntelliSense. Deze 'type inference' analyseert de context van Javascript en realiseert dynamische IntelliSense. In afbeelding 6 is een code-file weergegeven met context-gevoelige IntelliSense.

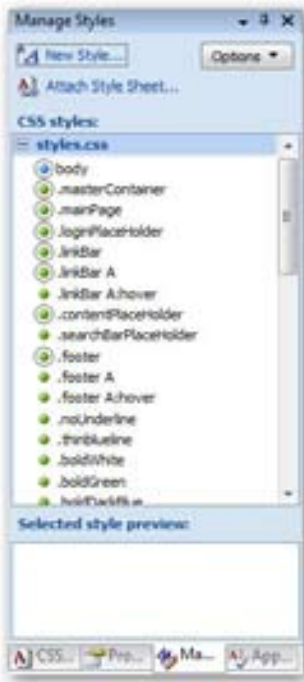
Deze IntelliSense werkt ook voor externe Javascript-libraries en files. Wanneer de Javascript-functies zijn voorzien van commentaar, dan kan de IntelliSense laten zien wanneer een functie wordt gebruikt. In voorbeeldcode 1 is een voorbeeld opgenomen van een Javascript-functie met commentaar van welke de IntelliSense gebruik kan maken. Wanneer gebruik wordt gemaakt van een scenario waarin ASP.NET AJAX wordt toegepast en een webservice wordt aangesproken, is IntelliSense ook beschikbaar. Als in Visual Studio 2008 een referentie wordt gemaakt naar een webservice, dan zorgt de omgeving er voor dat Javascript ook IntelliSense heeft voor de webservice. Naast IntelliSense wordt het ASP.NET AJAX-programmeren verbeterd door een drietal templates dat wordt aangeboden voor het maken van herbruikbare AJAX-componenten. De drie templates die worden aangeboden zijn: AJAX Client Behaviour, AJAX Client Library en AJAX Client Control. Ook bij het maken van deze componenten is IntelliSense beschikbaar.

Debugging

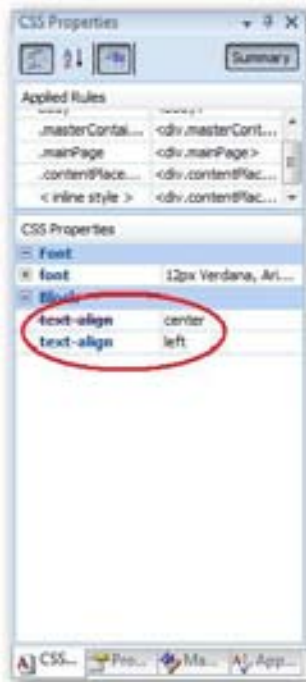
Volledige ondersteuning tijdens het debuggen is een voorwaarde om Javascript- en ASP.NET AJAX-applicaties professioneel te ontwikkelen. Met Visual Studio 2005 is dit beperkt mogelijk. Met de nieuwe versie van Visual Studio is debugging vernieuwd en verbeterd en volledig geïntegreerd in de ontwikkelomge-



Afbeelding 3. Splitview window

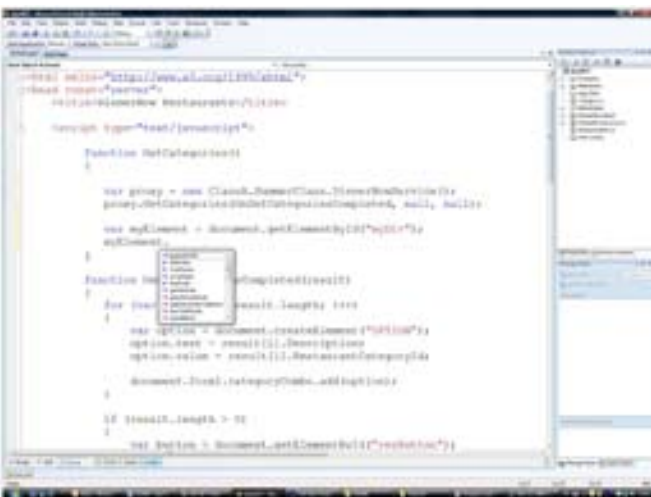


Afbeelding 4. Manage styles-window

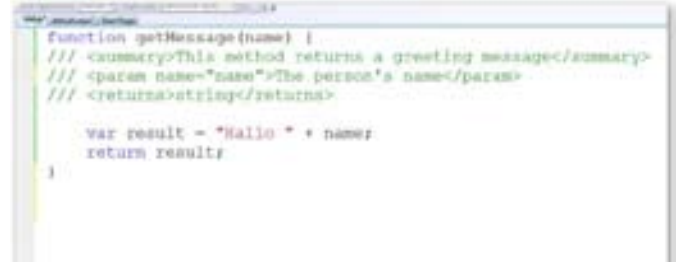


Afbeelding 5. CSS-properties summary

ving. Met de komst van de nieuwe ontwikkelomgeving kunnen breakpoints op elk moment worden geplaatst. Hiervoor hoeft de webapplicatie niet te draaien, dit in tegenstelling tot Visual Studio 2005. Breakpoints kunnen direct worden geplaatst in de server-side .aspx- en master sourcefiles. Visual Studio 2008 zorgt er voor dat zo'n breakpoint automatisch wordt vertaald naar een breakpoint in de dynamisch gegenereerde client-html. Wanneer breakpoints worden toegevoegd of verwijderd bij een draaiende applicatie, dan zorgt de omgeving er voor dat deze opnieuw automatisch worden afgebeeld in de originele sourcefile op de server. Hierdoor ontstaat een prettige situatie voor de ontwikkelaar om een ASP.NET AJAX-applicatie te debuggen. Een situatie die voor een Visual Studio-ontwikkelaar heel vertrouwd aanvoelt. Het verschil tussen een C# / VB server-side breakpoint en een Javascript-client is volledig weggenomen. Vanzelfsprekend worden de breakpoints bewaard wanneer de solution of het project wordt afgesloten. Tijdens het debuggen kan het handig zijn een overzicht te hebben van alle gebruikte url's waarin Javascript naar de client is verzonden. In de zogenaamde ScriptDocument-functionaliteit is dit mogelijk. Een aparte tool was al aanwezig voor Visual Studio 2005, nu is dit volledig geïntegreerd in de ontwikkelomgeving, zoals zichtbaar is in afbeelding 8. Bij een dubbelklik op één van de url's wordt



Afbeelding 6. Javascript IntelliSense



Voorbeeldcode 1. Javascript-functie met commentaar voor IntelliSense

de juiste Javascript-file geladen.

Tijdens het debuggen is het nu ook mogelijk gedetailleerde objectinformatie te bekijken in de Locals/Watch-windows. Deze informatie is beschikbaar dankzij een sterk verbeterd object-executie- en inspectiemodel dat door Visual Studio 2008 wordt gebruikt. Zo is het nu bijvoorbeeld mogelijk runtime-objecten en events te bekijken. De properties-structuur en waarden kunnen ook tot in detail worden bekeken. Het Immediate-window kan ook worden gebruikt om in runtime Javascript op te vragen en aan te passen.

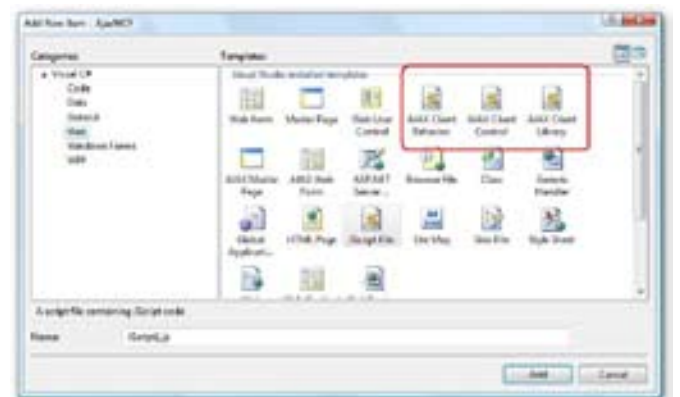
In Visual Studio 2008 is een flink aantal verbeteringen en vernieuwingen aangebracht om het werken met Javascript eenvoudiger, maar vooral productiever te maken. Deze Javascript-verbeteringen zijn van groot belang wanneer ASP.NET AJAX-applicaties worden gebouwd. Net als de eerder genoemde functionaliteiten, zijn ook de Javascript-functionaliteiten beschikbaar voor bestaande applicaties geschreven op basis van een eerdere versie van het .NET Framework.

Er is meer...

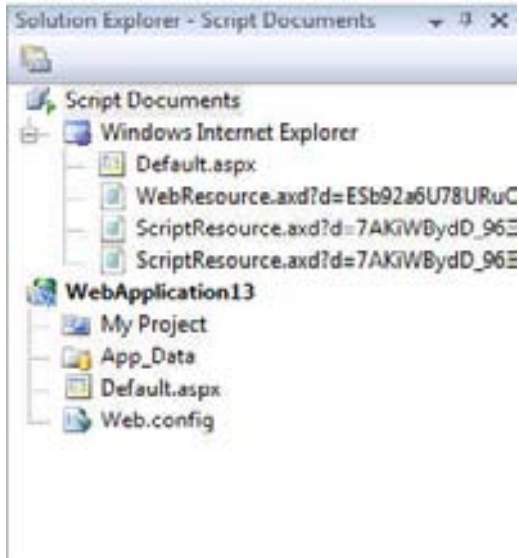
Naast de genoemde grote blokken aan vernieuwde functionaliteit zijn er ook nog diverse kleine verbeteringen. Deze zijn er op gericht om het leven van de Visual Studio-ontwikkelaar gemakkelijker te maken. Tijdens het werken met Visual Studio 2008 loop je iedere keer tegen een kleine, maar handige vernieuwing aan. In de editor is het bijvoorbeeld heel handig dat er een mogelijkheid is tot transparante IntelliSense. Wanneer je code aan het schrijven bent, komt het regelmatig voor dat een IntelliSense dropdown-box over een stuk code heen valt dat je graag wilt zien. Door het indrukken van de 'ctrl'-toets wordt de lijst transparant en kan de onderliggende code worden gelezen. Een andere verbetering is te vinden in de Visual Basic-code-editor. De IntelliSense is een stuk slimmer geworden en zorgt met behulp van filtering en de selectie van de waarschijnlijkste keuze voor tijdswinst. De komende tijd komt er steeds meer informatie naar buiten over dit soort handige vernieuwingen. Door met Visual Studio 2008 te werken, ontdek je al snel dit soort verbeteringen.

Grotere productiviteit

Het is dit artikel niet mogelijk alle verbeteringen te beschrijven. Daarom heb ik de belangrijkste vernieuwingen op een rijtje

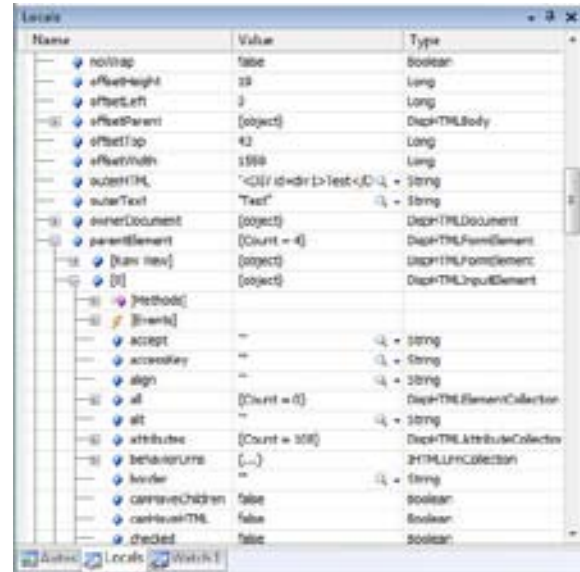


Afbeelding 7. ASP.NET AJAX-templates



Afbeelding 8. Script Documents

gezet. Naast de in het oog springende veranderingen in Visual Studio 2008 en het .NET Framework 3.5, zoals LINQ, is er ook het nodige verbeterd aan de ontwikkelomgeving. Veel van deze verbeteringen staan in het teken van een grotere productiviteit. Een aantal wensen vanuit het veld is direct terug te vinden in Visual Studio 2008. Belangrijke technologieën en talen zoals CSS en Javascript zijn volwaardige onderdelen geworden in de Visual Studio-omgeving. Het werken met de nieuwste versie van Visual Studio is een aangename ervaring waar iedere dag wat nieuws valt te ontdekken.



Afbeelding 9. Een uitgebreid Javascript-runtime-model

Anko Duizer is werkzaam als trainer/coach bij Class-A (www.class-a.nl). Daarvoor heeft hij vijf jaar als consultant bij Microsoft gewerkt. Onder zijn klantenkring bevinden zich voornamelijk top100-bedrijven in Nederland. Sinds begin 2001 is hij bezig met .NET. Speciale interesse heeft Anko voor de architectuur en het ontwerp van een gedistribueerde applicatie. Sinds 2006 is Anko Regional Director voor Microsoft in Nederland. Het blog van Anko is te lezen op <http://blogs.class-a.nl/blogs/anko/>. Voor vragen is hij bereikbaar via anko.duizer@class-a.nl

Referenties

<http://weblogs.asp.net/scottgu/>

<http://msdn2.microsoft.com/en-us/vstudio/aa700830.aspx>