

# Concurrency vraagt nog veel aandacht voor talen

ANDERS HEJLSBERG KEYNOTE SPREKER OP DEVDAYS 2010

**Robert de Ruiter**

Onze hersens zijn geprogrammeerd om met taal om te gaan. Programmeertalen zijn gerelateerd aan gewone en gesproken taal. Ze omkaderen de manier waarop je over een probleem denkt. Daarom wordt programmeertaal nog steeds geschreven in gewone tekst, zegt Anders Hejlsberg, Technical Fellow in de developer divisie van Microsoft. En hij kan het weten, want hij is de architect achter verschillende programmeertalen.

“Over 25 jaar programmeren we nog steeds in tekstbestanden. Dat had ik 25 jaar geleden niet durven zeggen, maar het is toch zo. Het gaat dieper dan tekstbestanden. Wij hebben nu eenmaal iets met taal. Onze hersens zijn geprogrammeerd om daarmee om te gaan. Programmeertalen zijn gerelateerd aan gewone en gesproken taal. Ze omkaderen de manier waarop je over een probleem denkt. Als je in C# programmeert en iemand laat je een programma van Visual Basic zien raak je in de war. Je bent er visueel niet op ingesteld en je hersens leggen niet zo snel het juiste verband. Kijk eens naar wiskunde. De manier van opschrijven is de laatste 200 jaar niet veranderd en het is nog steeds tekst”. Anders Hejlsberg, die hier zijn ideeën naar voren brengt, is niet alleen een van de meest invloedrijke ontwikkelaars van programmeertalen en ontwikkeltools sinds het ontstaan van de IT, hij is ook nog welbespraakt en sympathiek. Zo'n man wil je graag hebben voor de keynote op DevDays. En daar was hij dan ook. Hejlsberg is de ontwerper van onder meer Turbo Pascal, hoofdarchitect van Delphi, hoofdarchitect van C# en werkt als Technical Fellow in de developer divisie van Microsoft.

## Allemaal Chinees

“Het is natuurlijk vreselijk inefficiënt dat we zo veel verschillende talen spreken. Waarom gaan we niet met z'n allen over op Engels of Chinees? Maar we neigen er nu eenmaal naar om ons te identificeren met datgene wat we hebben gekozen. Zo gaat dat ook met programmeertalen. In de loop der tijd zijn we meer verticaal gaan werken. Door de technologische ontwikkeling hebben we ons moeten specialiseren. Het is daardoor nauwelijks haalbaar om, wanneer je werkzaam bent in het medische vakgebied, je met een DSL voor een ander vakgebied te verdiepen. We hebben in de IT dus eerder meer dan minder talen nodig”.

“Nieuwe talen ontstaan vaak op basis van de bestaande. Er is ook geen reden om het oude overboord te zetten. Het is vooral een kwestie van hoe wij vooruitgang boeken. Er is nog steeds veel Fortran en Cobol in omloop en C# is ook al tien jaar oud en heeft nog minstens tien jaar te gaan. Ik denk dat als je meer zaken toe-

voegt de complexiteit ook toeneemt. En er is geen magische truc om die complexiteit eruit te krijgen zonder de compatibiliteit te schaden. Als die compatibiliteit dreigt te verminderen kun je beter opnieuw beginnen en jezelf weer vooraan de rij te plaatsen door iets geheel nieuws te maken”.

“Je kunt eigenlijk maar op twee manieren met de ontwikkeling van programmeertalen omgaan. Je gaat voor iets heel nieuws of je ontwikkelt voorzichtig aan een bestaande taal en zorgt dat deze compatibel blijft. Als je kiest voor de weg van de nieuwe taal zul je dat regelmatig moeten doen, want met de oude versie zet je ook je hele community overboord. Je begint als het ware een nieuwe kerk en wilt niet meer met de afvalligen in discussie”.

## Gestolen ideeën

“Bij het ontwikkelen van nieuwe talen stelen we ideeën van elkaar uit de verschillende silo's. Imperatieve talen als C# en Java voegen



ANDERS HEJLSBERG: “BIJ HET ONTWIKKELEN VAN NIEUWE TALEN STELEN WE IDEEËN VAN ELKAAR”.

bijvoorbeeld features van functionele en dynamische programmeertalen toe zoals lambda expressions, strong typing en dynamic typing. Zo groeien talen ook naar elkaar toe. Er is geen reden om één taal te hebben, het gaat er om de beste te kiezen voor wat je ermee wilt bereiken”.

Dat is wel eens anders geweest. Nog niet zo heel lang geleden gunden de Java- en .NET-programmeurs elkaar het licht in de ogen niet. Tegenwoordig lijken beide kampen veel beter door een deur te kunnen.

Hejlsberg heeft daar wel een verklaring voor. “De twee kampen zijn volwassener geworden. Programmeertalen doorlopen allemaal dezelfde lifecycle. Het begint bij de extremistische adepten, die bij wijze van spreken in staat zouden zijn om je huis in brand te steken als je niet overneemt wat zij prediken. Neem de Ruby on Rails-lieden van een paar jaar geleden. Dat was pure religie. Daar sluiten zich dan gematigder programmeurs bij aan, die meer methodologie inbouwen en een zijweg inslaan. De extremisten verlaten daarop het pand, roepen naar wie het horen wil ‘Ruby on Rails is oude koek’ en beginnen iets nieuws te ontwikkelen”.

“Java en C# hebben dat proces ook doorlopen. Java was ooit als Ruby on Rails. Volledig religieus en irrationeel. Ondanks alle tekortkomingen geloofden velen erin. Nu is het een volwassen taal

---

## De truc is om het abstractieniveau omhoog te krijgen, zodat we op een hoger niveau kunnen denken.

---

en er is een hele industrie omheen gebouwd. Maar geef het nog tien jaar en het is de nieuwe Cobol”.

### Hardware

“Ontwikkelingen in de hardware hebben uiteraard een flinke invloed op de ontwikkeling van programmeertalen. De Wet van Moore gaat nog altijd op en zal zich nog tenminste een of twee decennia doen gelden. Miniaturisatie van geïntegreerde circuits kan doorgaan tot we letterlijk op atomair niveau uitkomen en straling een probleem gaat vormen. Daarnaast kunnen we nog iets verwachten van kwantum computers en wie weet wat nog meer”. “Zeker is dat de hardware zich blijft ontwikkelen. Software gaat veel langzamer. Dat komt doordat software veel meer is gebonden aan hoe wij denken. Onze hersens zijn de afgelopen vijftwintig jaar niet 10.000 maal groter geworden. We denken nog steeds als vroeger. De truc is om het abstractieniveau omhoog te krijgen, zodat we op een hoger niveau kunnen denken. We moeten ons niet meer druk maken om het low-level-loodgieterswerk van het programmeren. Maar dat gaat niet snel, daar is veel inspanning voor nodig”.

Er zal volgens Hejlsberg overigens altijd ruimte blijven voor het ‘ambachtelijke’ programmeerwerk. “De omgeving wordt echter veel vergeeflijker als je geen ‘ambachtsman’ bent. Als zodanig zullen er dus wel wat grenzen vervagen. Dit wil niet zeggen dat de ontwikkelingen vertragen, integendeel. Overal ter wereld worden de research-inspanningen verhoogd”.



ANDERS HEJLSBERG: “DE PROGRAMMEEROMGEVING WORDT VERGEEFLIJKER”.

### Concurrency

Concurrency vraagt de nodige aanpassingen in de programmeeromgeving. “In zijn algemeenheid is het ontwikkelen van een nieuwe taal geen goede zaak, omdat je veel bestaande technologie overboord zet. Aan de andere kant: als een omslagpunt wordt bereikt kan het wel zinvol zijn om een nieuwe taal te ontwikkelen. Niemand weet op dit moment waar het antwoord ligt. We kunnen – en moeten - zeker nog de bestaande talen verbeteren om met concurrency om te gaan. .NET noch Java zijn ooit ontwikkeld om met concurrency om te gaan. Deze zullen dus zeker nog de nodige aandacht vragen”.

Op het ogenblik werkt Hejlsberg meestentijds aan ‘upgrades’ de ontwikkeling van C#. Verder richt hij zijn aandacht op een nieuwe ontwikkeling: Compiler-as-a-Service. Er worden nieuwe implementaties van C# en VB compiler geschreven, die al in managed code staan. De bedoeling is om het een stuk eenvoudiger te maken om meer begrip voor de code in de programma’s te krijgen. Het zou verschillen tussen dynamische en statische ontwikkeltaalen op te heffen en een extra level van abstractie toe te voegen. Dit zal nog niet zichtbaar zijn in C# 5.0. Dat is een doorontwikkeling van de huidige versie 4.0.



# Meer werk naar client, minder naar de server

SCOTT HANSELMAN OVER WEBDEVELOPMENT

**Robert de Ruiter**

Scott Hanselman werkt als zelfstandige voor Microsoft als Principal Program Manager. Hij verspreidt informatie over softwareontwikkeling en dan met name de Microsoft stack. Zijn aanwezigheid op DevDays 2010 werd zeer op prijs gesteld, gezien de volle zalen die hij trok. Hanselman geeft een beeld van huidige en toekomstige ontwikkelingen in webdevelopment. Zijn idee: doe meer met de rekenkracht van de cliënt.

“In webdevelopment is een beweging weg van de server, die veel werk verricht. We realiseren ons dat computers steeds zwaarder worden, maar weinig werk doen. In feite zijn het ondanks alle capaciteiten maar domme terminals. Met hun vier gigabyte RAM en een dual processor staan ze te wachten tot een server het werk heeft gedaan. Tot de view ergens anders is gered. De HTML die we over het internet duwden bestond alleen maar uit mooie beelden”.

“Nu zijn er vier grote browsers: Internet Explorer, Chrome, Firefox en Opera. Zij verhogen het niveau voor iedereen. Als je nu goede HTML of Javascript schrijft, maken de browsers de applicaties beter en sneller. IE 9 biedt multiprocessor ondersteuning voor Javascript, accelerator, graphics. Dat is bepaald geen domme terminal meer. Hierachter zit de gedachte dat de computers zo snel zijn geworden, dat we daarvan gebruik moeten maken om het web ook sneller te maken”, aldus Hanselman.

## Templates versturen

“De ontwikkelaar moet zich nu afvragen wat dit alles voor het development proces betekent. Liever dan een view te renderen en als HTML te versturen kunnen we templates overbrengen die zeggen hoe een pagina eruit moet zien en afzonderlijk de data overbrengen. De template en de data worden op de terminal bij elkaar gebracht. AJAX is zo'n concept waarbij je alleen de data ontvangt en die bij de client verwerkt: validatie, Javascript grids, applicaties al gmail en hotmail. Om deze reden omarmt Microsoft ook jQuery. MS en de developer division realiseren zich dat als ze iets niet zelf hoeven bouwen, ze dat heel goed aan anderen kunnen overlaten. jQuery is daar een mooi voorbeeld van”.

“ASP.NET 4, dat wordt geleverd met 2010, komt met Webforms en MVC. Dit is kiezen tussen een motor of een minibusje, afhankelijk van je eigen wensen. Als je voor een motor kiest wil je snelheid en flexibiliteit, maar hoeft je geen boodschappen te doen of je kinderen naar voetbal te brengen. Als je voor de minibus kiest heb je duidelijk andere wensen. Daarin heb je de ruimte, maar hoeft je van de snelheid niet veel te verwachten. Zo zijn Webforms en

MVC ook verschillende 'vehicles', die draaien op het ASP.NET platform. Je kunt ze ook beide gebruiken, afhankelijk van het doel dat je ermee wilt bereiken”.

In het verleden was ASP.NET sterk gefocust op controls en de server. MVC daarentegen richt zich meer op de clients en heeft meer Javascript en meer user control. Omdat het allemaal op ASP.NET draait kun je ook hybride applicaties maken met pagina's van MVC en Webforms. De CodePlex-website is ook gedeeltelijk MVC en gedeeltelijk Webforms.

Hanselman: “Als je snel veel productiviteit wilt hebben is Webforms geweldig. Als je gaat voor low-level Javascript en HTML ben je beter af met MVC”.



SCOTT HANSELMAN: “TEMPLATES VERZENDEN EN OP DE CLIENT DE DATA VERWERKEN”.



“Visual Studio 2010 heeft ook een aantal elementen die webdevelopment eenvoudiger maken. Met name Javascript Intellisense. Javascript kan aanvankelijk een intimiderende taal zijn. Mensen die met Webforms hebben gewerkt zijn bekend met Intellisense, waarmee je een object kunt typen als ‘Person.’ en je krijgt een keuzeschermbalk met veel mogelijkheden ‘Person.Firstname’ of ‘Person.Surname’. De IDE kijkt naar het object en geeft je verschillende mogelijkheden. Maar Javascript is een hele dynamische taal. Objecten krijgen en verliezen vaak properties. Het is ook allemaal tekstgebaseerd”.

## Sophisticated parsing

“Aan de serverkant kan ik deze introspectie plegen omdat ik er de DLLs heb. Aan de clientkant heb ik alleen de Javascript, alleen de tekst. We hebben een sophisticated parsing engine die al dat Javascript op de achtergrond bekijkt en evalueert. Als je nu Javascript inbrengt van de jQuery library hebben we een manier om documentatie op te vragen. Als je in jQuery ‘OpenParenthesis’ type, krijg je een popdown met alle jQuery methodes en een tooltip die verklaart hoe de methode werkt”.

“Het verschil tussen editen van Javascript in het kladblok of in Visual Studio is gigantisch. Mensen die hun eigen Javascript libraries schrijven kunnen deze documentatie als afzonderlijk bestand insluiten. Als ik iemand mijn Scott.js bestand geef, kan ik er het Scott.vsdoc.js bestand bij doen. Je kunt dan precies de methodes en helpbestanden inzien. De parsing gebeurt in de achtergrond en een kwestie van seconden. Dat gaat razendsnel”.

---

## Multimonitor is een kleine, maar erg belangrijke verbetering. Je werkt nu veel makkelijker en sneller

---

“Er zitten nog meer verbeteringen in de Visual Studio editor om je code eenvoudiger te manipuleren. Je kunt in Visual Studio ook diagrammen genereren, sequence diagrammen bijvoorbeeld. Deze zijn vergelijkbaar met wat je nu met de hand doet in Visio. Met de sequence diagrammen kun je bestaande code sneller doorgronden. Er zijn nog meer items die het werk van de developer eenvoudiger maken. Tot dusver kon je met Ctrl-F wel de ‘shopping cart’ als tekst in de code vinden, maar verder ging dit niet. Met VS2010 kun je nu ook de methodes en classes ‘shopping cart’ vinden”.

“Nog een item: sommige klassenamen worden erg lang. Volgens de Camel-casing noem je een klasse bijvoorbeeld HttpCPolicy. Je hoeft nu alleen HCP te typen en VS 2010 vindt waar je naar zoekt. Het lijkt niet belangrijk, maar als je weet hoe vaak je als developer deze functie gebruikt levert het een behoorlijke tijdbesparing op”.

## Multimonitor

Multimonitor is nog zo’n kleine, maar belangrijke vernieuwing. Hanselman: “Ik heb zelf vijf monitoren staan, maar kon tot dusver slechts op één tegelijk werken. Met VS 2010 kun je nu ieder scherm oppakken en naar een andere monitor slepen. Je kunt je HTML in een scherm, de browser in een ander en het werkoverzicht in een derde scherm plaatsen. Dat werkt wel heel erg makkelijk”.



SCOTT HANSELMAN: “ASP.NET 4 EN .NET FRAMEWORK 4 SIDE-BY-SIDE”.

“De editor ondersteunt nu ook Ctrl-scroll. Als je de Ctrl-knop ingedrukt houdt tijdens het scrollen kun je de tekst vergroten en verkleinen. Deze functie werkt al in Internet Explorer, Word en Excel, maar nu dus ook in VS 2010.

Alles bij elkaar genomen – VS 2010 biedt in totaal circa 1900 nieuwe features – wordt dankzij de editor softwaredevelopment in het algemeen en webdevelopment in het bijzonder een veel plezieriger ervaring, vindt Scott Hanselman.

## Toekomst

Wat Hanselman opvalt aan de nieuwe pakketten ASP.NET 4 en .NET 4 is dat deze als side-by-side release zijn uitgebracht. “Het installeren van .NET 4 geeft dus geen problemen met bestaande applicaties. Ze leven naast elkaar. Dit betekent dat op één Windows server een .NET 1.1, 2, 3.5 en 4 kan draaien. .NET 4 installeren verandert dus niets aan bestaande applicaties. Dat geeft mensen de flexibiliteit om applicaties geleidelijk te installeren, of zelfs delen van applicaties. Heb je een applicatie draaien op 3.5, dan zal die het ook doen op 4”.

“We hebben mensen niet willen dwingen om voor een bepaalde versie te kiezen. In VS2010 krijg je zelfs een uitrolmenu waarin je kunt aangeven op welke versie je de applicaties wilt laten draaien.”





# DevDays 2010 in Beeld



# De 'drawer of broken dreams' kan nu dicht

CHARLIE KINDEL OVER WINDOWS PHONE 7

Robert de Ruiter

De mobiele ruimte is al gigantisch en uiteraard erg belangrijk voor de software-industrie. Je zou kunnen stellen dat de mobiele telefoon de meest personal computer is. Het is dus logisch dat Microsoft zich op deze markt wil manifesteren. Het pad dat tot dusver was ingeslagen met Windows Mobile wierp vragen op voor de toekomstige mogelijkheden. Charlie Kindel, bij Microsoft verantwoordelijk voor Windows Phone 7, legt uit.

**Onze concurrenten hebben** een andere manier van denken ontwikkeld en veel succes gehad met smartphones. We hebben afstand genomen van het probleem en ons afgevraagd hoe we op de lange termijn succesvol kunnen zijn. Hier toe is de 'three screen' cloud visie ontwikkeld. Niet alleen om op mobiel gebied meerwaarde te kunnen bieden aan zakelijke klanten en business partners, maar ook om PC en televisie als middel in te schakelen om mensen toegang te geven tot hun persoonlijke gegevens en informatie en diensten van derden.

Op de vraag of de inspanningen van Microsoft niet een beetje 'too little too late' komen zegt Kindel: "De smartphone markt staat nog in de kinderschoenen. Op het ogenblik zijn er een half tot een miljard mobieltjes in omloop. Als je dit afzet tegen de wereldbevolking staat er nog heel veel te gebeuren. We praten voortdurend met potentiële klanten en velen van hen hebben producten van de concurrentie en zijn daar niet altijd tevreden mee. We zien dus nog voldoende kansen om ons marktaandeel op mobiel gebied te vergroten".



CHARLIE KINDEL.

Klanten en ontwikkelaars vonden dat we met Windows Mobile veel keus boden. Bijna te veel keus, te veel flexibiliteit, te veel fragmentatie.

Ontwikkelaars wisten vaak niet of hun applicaties op bepaalde telefoons werkte. Gebruikers waren erg onzeker over de vraag welke mobiel ze moesten aanschaffen en met welk operating system. Met Windows Phone 7 hebben we ons gefocust op een veel homogener hardware

ecosysteem. Niet volledig homogeen, er is nog flexibiliteit voor de hardwareproducenten om hun apparaten aan de wensen van de klant aan te passen. Tegelijkertijd willen we ook duidelijkheid en ruimte scheppen voor de developers van de software. We specificeren heel duidelijk de karakteristiek van de hardware. Alle hardwarefabrikanten kunnen hun apparaten daarop bouwen. Maar ook daar zit wel wat flexibiliteit in. Sommige mobieltjes hebben een toetsenbord; andere niet".

De mobiele telefoons met Windows Mobile 7 die op de markt komen zullen afkomstig zijn van verschillende hardware-leveranciers. Met toestellen van Samsung en LG zijn al demonstraties gegeven. Daar zullen ongetwijfeld andere merken bij komen, maar deze zullen qua hardware wel grote overeenkomsten vertonen. Iedereen kan met Windows Phone 7 applicaties bouwen. Over de manier waarop die op de markt verschijnen houdt Microsoft niet bezig. Het is niet van plan om een 'marktplaats' voor mobiele applicaties op te zetten zoals die bijvoorbeeld wel bestaat voor de tools die via CodePlex worden aangeboden. "Ik heb zelf een simpele applicatie gebouwd. Je kunt hiermee je mobiel als zaklamp gebruiken. Dit is een goed voorbeeld van een Windows Phone applicatie. Ik heb het daarom op CodePlex gezet als open source. Vooral om als voorbeeld van een eenvoudige applicatie te dienen. Maar ik ken ook mensen die applicaties hebben gebouwd, die er niet over denken om de code beschikbaar te stellen".

## Multiple devices

Windows Phone 7 moet het eenvoudiger maken om voor verschillende devices te ontwikkelen. Microsoft heeft op dit gebied lering getrokken uit het verleden. Het heeft veel gefrustreerde developers gesproken omdat zij hun applicaties of spellen op ieder afzonderlijk toestel moesten testen. "Ik was onlangs bij een developer, die een lade met 30 verschillende toestellen had. Hij testte zijn applicaties op alle toestellen. Toch gebeurde het regelmatig dat hij klachten kreeg omdat de software op weer een ander model toch niet goed draaide. Hij noemde die lade met mobieltjes de 'Drawer of Broken Dreams'".





# Ben jij onze nieuwe C#.NET ontwikkelaar?

Ga dan met ons mee naar de  
**DevConnections 2010**  
in Las Vegas!



**Bezoek de DevConnections en  
haal je MCPD certificering voor  
Visual Studio 2010**

*Als Microsoft Gold Certified  
Partner werken wij met de  
nieuwste technieken. Steeds  
blijven op je vakgebied is dan  
een must. Daarom bezoeken alle  
collega's jaarlijks de Microsoft  
DevDays.*

Omdat 2010 een jaar wordt waarin allerlei nieuwe releases van Microsoft op de rol staan, gaan we dit jaar nog een stap verder. We gaan in 2010 naar de DevConnections in Las Vegas. En we zoeken nog een nieuwe collega die met ons mee wil gaan en zich in de onderstaande kenmerken herkent:

- Ben je een beginnende of ervaren C#.NET ontwikkelaar die wil werken met de nieuwste tools en technieken bij een Microsoft Gold Certified Partner?
- Heb jij een passie voor het ontwikkelen in .NET?
- Weet je perfect de weg in C#.NET te vinden en ben je initiatiefrijk, nauwkeurig en kwaliteitsgericht?
- Zijn je analytische denkvermogen en communicatieve vaardigheden uitstekend?

Dan zijn wij op zoek naar jou om ons groeiende, enthousiaste en hechte team van 6 ontwikkelaars te versterken. We gaan er natuurlijk van uit dat jij je ook prima kunt vinden in het ontwikkelen in een team. Hiervoor gebruiken we Team Foundation Server 2010 en de Agile development methode Scrum. Daarnaast heeft iedereen een MCPD certificering en we vernieuwen die in 2010 zodra de opleidingen en examens daarvoor beschikbaar zijn. Het spreekt voor zich dat jij deze certificering ook binnen sleept (of die al hebt).

#### **Wie is Driessen?**

Driessen is dé HRM-dienstverlener voor de overheid, onderwijs en zorg & welzijn. Onze 130 medewerkers zijn dagelijks ca. 600 non-profit organisaties door heel Nederland van dienst. Het brede dienstenpakket bestaat uit payroll, consultancy, outsourcing van personeels- en salarisadministraties, software en opleidingen. Wij zijn gevestigd in een modern kantoorpand in Helmond.

Nu al een kijkje achter de schermen nemen op misschien wel jouw toekomstige werkplek? Kijk op <http://www.driessen.nl/techtalk>. Ook kun je contact opnemen met onze manager automatisering Patrick Adriaansen. Hij informeert je graag verder over de functie, het team en het bedrijf. Je kunt Patrick bereiken via: 0492 - 50 66 66.

#### **Reageer snel!**

Is het bovenstaande profiel jou op het lijf geschreven? Aarzel dan niet en stuur je sollicitatiebrief en CV o.v.v. vacaturenummer 09.13 naar [vacatures@driessen.nl](mailto:vacatures@driessen.nl) of naar Driessen HRM\_Payroll, T.a.v. de heer Patrick Adriaansen, Postbus 748, 5700 AS Helmond.

Schootense Dreef 15  
5708 HZ Helmond  
Postbus 748  
5700 AS Helmond  
t. 0492 - 50 66 66  
f. 0492 - 50 66 50

[info@driessen.nl](mailto:info@driessen.nl)  
[www.driessen.nl](http://www.driessen.nl)

**driessen**

HRM \_ Payroll