

# Heeft Oracle een tool-strategie?

## Interview met Ted Farell

*Voor veel mensen is de strategie rond Oracle ontwikkeltools en talen niet helemaal duidelijk. De grote nadruk op Java werd weer verkleind, en in versie 10 wordt de support voor PL/SQL zelfs uitgebreid. De Oracle-gemeenschap reageert nu – na aanvankelijk geschrokken te zijn – nogal laconiek en gaat in vele gevallen op de oude voet verder. Tijd voor een verklarend gesprek met Oracle's director of strategy voor applicatie-ontwikkeltools, Ted Farell.*

Voordat Ted Farell in dienst trad bij Oracle, was hij onder meer CTO bij WebGain en mede-oprichter van verschillende kleine software-bedrijven waaronder Night Owl Enterprises, Novasoft Systems en Tendril Software. Bij dat laatste bedrijf stond Farell aan het hoofd van een ontwikkelteam dat het product StructureBuilder ontwierp, een tool waarmee het UML-model en de Java source code continu kunnen worden gesynchroniseerd. Thans is Farell architect en director of strategy van de divisie voor de applicatie-ontwikkeltools bij Oracle Corporation. Hij is in die functie verantwoordelijk voor de technische en strategische richting van Oracle's ontwikkeltools, waaronder Oracle JDeveloper 10G en Oracle's Java- en XML-IDE's voor de ontwikkeling van J2EE applicaties en webservices. Tot zijn taken behoort ook het op één lijn brengen van de verschillende development technologieën onder een gemeenschappelijke visie, waarmee Oracle zich tot doel stelt een leidende positie te verwerven in J2EE development.

### Focus

*Een paar weken geleden werd bekend dat IBM een miljard dollar gaat investeren om ontwikkelaars over te halen te gaan ontwikkelen voor Java of beter gezegd Websphere. Dat geeft aan hoe belangrijk het voor hun is om ontwikkelaars te werven voor hun platform. Hoe is dat voor Oracle en hoe verhoudt zich dat tot de ontwikkeltools?* Farell: 'Die miljard dollars is marketing van IBM. Oracle heeft meer producten die onder Java draaien dan IBM. IBM heeft nog steeds Lotus Notes en de legacy software. Ze consolideren wel en tegen 2006 zal het er wel beter uitzien. Maar wij geven geen miljard dollar uit aan marketing, wij stoppen het in onze

producten. Oracle is helemaal toegewijd aan Java en J2EE applicatieservers, en alle producten die nog niet op dat platform draaien worden daarnaar gemigreerd.'

*Oracle probeerde jaren geleden al ontwikkelaars over te krijgen naar Java, dat werkte niet zo heel goed, in die zin dat de oude garde niet overgegaan is, dus de support voor de oude tools Designer en Forms is verlengd.*

Farell: 'Zeker. We hebben geen strategie om de support voor Designer en Forms te stoppen, maar onze huidige strategie berust op J2EE. Als je nog steeds Forms of Designer gebruikt, PL/SQL is nog steeds een taal die wij ondersteunen, en die twee producten ondersteunen we ook. In JDeveloper 10g zitten een aantal nieuwe dingen, zoals PL/SQL debugging en design. Het verschil tussen de oude tools en technologieën en de nieuwe is dat de voornaamste focus op nieuwe ontwikkeling en vooruitgang in de J2EE-hoek zal zitten. PL/SQL-technologie is echter nog steeds geldig. Wij ondersteunen die taal nog steeds; het zal ook in JDeveloper geïntegreerd worden.



Ted Farell, architect en director of strategy voor ontwikkeltools bij Oracle



*Farell: "Oracle heeft meer producten die onder Java draaien dan IBM"*

We hebben ook nog Forms en Designer. Dat zijn weer andere tools en runtimes zijn, maar ook die ondersteunen we nog steeds.'

### **Structuur**

*Dan heb je in zekere zin twee problemen, het eerste is je mensen die met PL/SQL werken naar Java moet overhalen en aan de andere kant heb je de Java-ontwikkelaars die je moet overhalen om nu juist voor Oracle te gaan ontwikkelen.*

Farell: 'Ja, dat is juist. In het algemeen heb je meer details en meer keuzevrijheid in de J2EE-architectuur. Het is meer een

vrije vorm, developers hebben heel veel keuzemogelijkheden. Veel ontwikkelaars willen al die keuzes niet, die willen gewoon iets kunnen bouwen. Kijk maar naar de totale ontwikkelmarkt: die wordt dit jaar geschat op acht miljoen professionele ontwikkelaars waarvan ruim twintig procent Java-ontwikkelaars zijn. Dat zijn er bijna twee miljoen. Dan heb je nog Cobol, C++, Visual Basic, Forms, PowerBuilder, dus het aantal Java-ontwikkelaars is groot, maar als percentage van het geheel is het nog klein. Wij streven ernaar een oplossing en een architectuur te bieden die voor alle ontwikkelaars werkt, inclusief de Cobol en VB-mensen. Dat is min of meer onze strategie. De huidige Java-IDE's, ook JDeveloper, zijn echt gericht op Java 'coders', mensen die Java kennen, en die zijn niet erg vriendelijk voor andere mensen die beginnen met Java. JDeveloper 10 - dat nog niet zo lang uit is - was sinds vorig jaar in preview. Daarbij hebben we ons gericht op het verhogen van de productiviteit. Dat gebeurt door toevoegen van een laag bovenop J2EE. We hebben dus een oplossing die we aankondigen samen met JDeveloper en dat heet Application Developer Framework, of

***Tenzij je XML een taal noemt, concentreren wij ons op PL/SQL en Java***

## *Het aardige van het Application Development Framework is dat het objectgeoriënteerd is en modulair opgezet*

ADF. Dat is een set van runtime classes boven op de J2EE applicatieserver, allebei standaard. Die maken het ons als het ware mogelijk om een structuur bovenop J2EE te plaatsen. Je kunt altijd naar J2EE gaan, alle opties zijn open. ADF is de structuur. We hebben ook wat design patterns gecreëerd. Zo kan de developer zich richten op de business logica en niet zozeer op de J2EE-technologieën. We kunnen daarmee een meer diverse kring van gebruikers aantrekken: niet uitsluitend Java-gebruikers, maar ook anderen. Terwijl we verder gaan, kunnen we doorgaan met het opbouwen van meer declaratieve en visuele manieren naar ADF en JDeveloper om meer van die tachtig procent aan te trekken. De architectuur is MVC (model view controller): je business services - de business logica en de modelinterface daarvoor - en je model en je view, dus je presentatie en je model. ADF volgt een traditioneel MVC paradigma, en het propageert het gebruik van metadata en declaratief programmeren. Dit is waar Oracle in het verleden veel succes mee heeft gehad, als je denkt aan Forms en Designer. Anderen zoals Powerbuilder VB 6 hadden ook succes met metadata.'

### **Metadata**

*Java is een van de belangrijkste talen op dit moment. In de toekomst zullen andere talen belangrijker worden, misschien meer procesgeoriënteerd. Hoe staan jullie daar tegenover?*

Farell: 'Veel van het werk dat we in ADF en in andere projectintegratie producten doen, bestaat in het gebruik van metadata. Wanneer je die gebruikt, beschrijf je de business logica in een universele vorm. Je kunt XML ervoor gebruiken, je kunt het opslaan in een database of in een document. Een voorbeeld is ADF-M. Dat is een modellaag in deze architectuur. We hebben dat net als Java-standaard ingediend, JSR 227. Het maakt het mogelijk data van verschillende services - webservices of EJB of Java-classes of wat dan ook - te verbinden in user interfaces. Dat gebeurt allemaal declaratief dus je hoeft alleen maar die XML-paden te specificeren om de data eruit te krijgen. Voor ons integratieproduct geldt hetzelfde. Het zal in de volgende versie BPEL ondersteunen waardoor je het business proces beschrijft en tijdens runtime daarop laat uitvoeren. Wat de codeertalen betreft: wij ondersteunen Java voor de middle-tier en de client, en PL/SQL voor de database taken. We zien

dus geen noodzaak daar een nieuwe taal aan toe te voegen, maar we zien wel het steeds toenemende gebruik van metadata om de business logica te beschrijven. Sommige mensen noemen XML een taal. Voor mij is het meer een dataformaat. Dus tenzij je XML een taal noemt concentreren wij ons op PL/SQL en Java.'

*Toen ik vanmorgen iemand vertelde dat ik een gesprek ging voeren over de Oracle-strategie met betrekking tot tools, vroeg hij: 'Hebben ze dan een strategie?' Zou u in een of twee zinnen die strategie kunnen samenvatten?*

Farell: 'De Oracle toolstrategie bestaat uit twee elementen. De eerste prioriteit is dat we de beste developer-omgeving willen zijn voor onze runtime, voor onze database, applicatieserver et cetera. De tweede prioriteit is om - vanwege alle industriestandaarden - de toonaangevende development tool te zijn voor Java en webservices, onafhankelijk van het platform. We hebben laatst een poll gedaan bij de gebruikers van OTN, waaruit bleek dat 52% van de respondenten niet naar een Oracle omgeving deployden. Ze deden dat dan naar bijvoorbeeld BEA of IBM, of ze maakten gewoon rich clients. We hebben dus een goede mix. Oracle is geen grote marketing machine, we praten liever over de dingen die we doen. Oracle gaat dus geen marketingaankondiging doen dat we een miljard aan iets gaan uitgeven. We doen het gewoon. Daardoor zijn we niet zo zichtbaar en weten mensen niet dat we een toolstrategie hebben, of dat we zeer gecommiteerd zijn aan standaards en daarin deelnemen. Dat is een van de barrières waar we tegenop lopen: de perceptie is dat we proprietary zijn, en dat we mensen aan een vendor lock-in willen onderwerpen. Onze applicatieserver is helemaal standaard, certified. Hij heeft de hoogste performance in de markt, onze tools zijn allemaal gestandaardiseerd, en je kunt naar andere applicatieservers deployen. Tegelijkertijd zien we ook de waarde van de stack. Onze applicatieserver zal altijd de beste applicatieserver zijn voor onze database, ook al werkt die ook met andere databases, en de tools zullen de beste tools zijn voor de applicatieserver en de database. Dat is de strategie in een notendop.'

*Bij Microsoft zijn er productgroepen die echt verliesgevend zijn, terwijl andere extreem winstgevend zijn. Hoe is dat bij Oracle met de toolsdivisie?*

Farell: 'De toolsdivisie is geen winstmaker. We hebben drie productlijnen: de database, E-Business Suite en de applicatieserver (met integration, portal en wireless). Het is de taak van de tools om die producten te ondersteunen, en om meer developers met die producten te laten werken. In sommige gevallen geven we bijvoorbeeld geld uit aan tools om extensies te bouwen naar andere IDE's. Als mensen voor een ander tool kiezen - want die vrijheid hebben ze bij ons - willen we er wel zeker van zijn dat ze nog steeds effectief en efficiënt kunnen schrijven

voor onze runtime. We willen er niet zozeer geld mee verdienen, we proberen ontwikkelaars productiever te laten zijn en het aantrekkelijker te maken om voor de Oracle runtime te ontwikkelen.'

## Geen ramp

*De Oracle PL/SQL gemeenschap is hier in Nederland nog sterk. Toch zal men vroeg of laat naar Java over moeten. Wanneer de huidige situatie over tien jaar onveranderd is, dan zou dat een ramp voor Oracle zijn.*

Farell: 'In ieder geval zouden ze nog steeds op onze database zitten. PL/SQL is onze voorkeurstaal voor de database en Java is dat voor de middle tier. Het gaat om de vraag, of je de business logica in de middle tier stopt of in de database. Zelfs in een ideale situatie zijn er argumenten voor beide. Veel mensen stoppen dingen in de database die beter in mid-tier zouden kunnen zitten. Er zijn activiteiten die zeer data-gecentreerd zijn, die we kunnen optimaliseren en waarmee we een veel betere performance kunnen krijgen in de database, en er is andere business logica die je zeker in de mid-tier wilt hebben omdat we het daar beter kunnen beheren. Ik ben het met je eens dat het slecht zou zijn wanneer we over jaren nog steeds die groep mensen hebben die programmeren op de database, omdat we geloven dat de middle tier van waarde is, maar het is niet het ergste wat er bestaat. De database is immers schaalbaar in clusters en kan het aan. We adviseren developers niet alles te laten vallen. Wanneer je een applicatie in PL/SQL hebt geschreven dan is er in feite geen reden om dat vandaag te veranderen. De middle tier is gewoon een nieuwe ontwikkeling die je zou moeten overwegen.'

*Maar de onderhoudbaarheid zal slecht zijn, en ook de architectuur van zo'n applicatie is niet optimaal.*

Farell: 'Er zijn zeker beperkingen, maar de database performance zal veel compenseren. Je kunt bovendien stukjes PL/SQL als webservice verpakken, dus je kunt met alles en iedereen praten. Het is geen ramp, maar we geloven niet dat je de volledige architectuur gebruikt op de manier waarop je dat zou kunnen doen. Ons plan is echt om mensen aan het denken te zetten hoe je de middle tier kunt gebruiken. We slagen daarin, het begint vorm aan te nemen. Het is een proces dat we moeten continueren.'

*Is er nog speciale hulp die u biedt bij de overgang, afgezien van tools en de cursussen?*

***PL/SQL is onze voorkeurstaal voor de database en Java is dat voor de middle tier***



*Farell: 'We proberen ontwikkelaars productiever te laten zijn.'*

Farell: 'Ja, het aardige van ADF is dat het objectgeoriënteerd is en modulair opgezet. Het is niet alles of niets. Je kunt dus beginnen met het toevoegen van deelgebieden en andere functionaliteit. Dat is de grootste sleutel naar adaptatie. Het is anders dan veel frameworks in het verleden. Voor de rest houden wij van PL/SQL. We denken dat het de juiste taal is voor zaken die in de database gebeuren, maar we willen tegelijkertijd de juiste scheiding van logica bevorderen.'

## Aardigheidjes

*Hoewel ik me kan voorstellen, dat over een paar jaar en misschien eerder dan we denken, Oracle zegt: "Niemand hoeft meer PL/SQL te schrijven. Wij doen dat voor jullie want we kunnen het ook beter dan jullie".*

Farell: 'Wij houden van die aanpak waarbij iemand die uiteindelijk weet wat hij doet altijd in de low level api's kan, J2EE of de database PL/SQL. Daarnaast voegen ADF en JDeveloper nog een aantal abstracties en aardigheidjes toe. Mensen zullen waarschijnlijk nog steeds in staat zijn in PL/SQL te programmeren maar ook in ADF dan heel visueel en declaratief iets doen. PL/SQL is waarschijnlijk de juiste oplossing maar het is ook de vraag: kunnen we een omgeving aanbieden waar ze niet een hoop code hoeven te schrijven? We kunnen het op een andere, meer mda-achtige manier beschrijven. Dan genereren we de geoptimaliseerde PL/SQL. Maar Java vertalen in PL/SQL is moeilijk. Het zijn twee functioneel verschillende dingen. We gaven mensen de mogelijkheid om Java in de database laten draaien maar we realiseerden ons dat het niet de juiste tier er voor was.'

**Tekst en fotografie: Dré de Man**