

# SOA wordt mainstream

**'Middleware is een enorme business voor Oracle'**

**Steven G. Harris is volgens zijn visitekaartje Vice President van de Java Platform Group bij Oracle. Maar visitekaartjes zeggen vaak niet zo veel, en naar zijn eigen zeggen is hij development manager voor het Java application server team. Hij is verantwoordelijk voor alles wat het J(2)EE te maken heeft, voor de webservices runtime, O/R mapping en zelfs voor dingen als JDBC-drivers en Java-support in de database. Optimize sprak met hem.**

Harris: 'Mijn team doet veel van de standaardactiviteiten die door het hele bedrijf hun werking hebben. Samenwerking met andere bedrijven door standaarden begint vaak bij mijn team, op het webservices-front en het Java-front. SOA is een van die dingen die goed past bij System Integrators voor wat betreft de mogelijkheden om samen te werken en dingen in beweging te brengen. SOA is definitief op weg mainstream te worden in de manier waarop IT-afdelingen bezig zijn dingen te adopteren. Het wordt steeds meer gezien als een noodzaak in plaats van een optie, iets wat ze bijna als vanzelfsprekend aannemen.'

## Suite-concept

*Op het gebied van appservers is er nogal wat concurrentie, die voor een deel zijn spullen gratis levert. Denkt u dat Oracle er op termijn in zal blijven slagen er geld mee te blijven verdienen?*

Harris: 'Afgezien van de algemene issue open source, is commoditisation van pure J2EE applicatieservers definitief een trend. Klanten neigen er echter naar oplossingen van een hoger niveau te kopen in tegenstelling tot het kopen van de meest capabele J2EE-server op de markt. We waren één van de eerste bedrijven om het suite-concept te pushen, in middelzware omgevingen. Veel mensen vertelden ons dat we gek waren toen we het deden maar nu denk ik dat dit het dominante kooppatroon van bedrijven is (suites).'

*Er zijn tijden geweest dat Oracle zei dat het 300% Java was, (Harris lacht: 'Ja, dat heb ik ook wel eens opgeschreven'), de tijd ook dat een database zo 'plat' mogelijk moet zijn en alle logica in de middle tier*

*moest zitten. Hoe denkt Oracle daar nu over?*

Harris: 'Wij propageren niet het uit elkaar trekken van bestaande systemen teneinde ze te vervangen. Mensen hebben jaren in hun database geïnvesteerd en vaak applicatielogica ingebed in de database. Een van de dingen die we met betrekking tot SOA en webservices hebben gedaan, is het aanbieden van database-functionaliteit als service. Je kunt nu dus op een meer natuurlijke manier systemen die logica in de database bevatten samenweven. Of het nu bestaand spul is, of dat je ervoor kiest om

**'We waren één van de eerste bedrijven om het suite-concept te pushen'**

het nu zo te deployen.

Voor wat betreft de applicatieontwikkeling denk ik dat we een verandering zien in de richting van het ontwikkelen volgens SOA – wat iets anders is dan het ontwikkelen volgens het OO-paradigma. Het is anders omdat de architectuur zelf je iets van de discipline die je nodig hebt, oplegt om schaalbare applicaties te bouwen die gedistribueerd kunnen worden. Bij een technologie als Corba of zelfs tot op zekere hoogte EJB's moet je in feite heel gedisciplineerd zijn om het te doen. Bij SOA duwt de architectuur je in de richting van het bouwen van zeer *loosely coupled* systemen die als verzameling van afzonderlijke deeltjes met elkaar communiceren. Je zult applicaties zien die daaruit gebouwd zijn en die samengesteld zijn uit services die met elkaar verbonden kunnen zijn via een ESB, of via routing regels die op een flexibele manier gedefinieerd zijn met behulp van rule engines enzovoorts. In feite zullen deze evolueren in de richting van het aanbieden of gebruiken van services. Kijk ook naar de nieuwe dingen zoals service component architectuur, een gebied waarop Oracle, IBM, BEA, SAP samenwerken. Ik denk dat je daaruit een componentmodel ziet groeien om dien-



sten te combineren tot modules en systemen die dan verbruikt kunnen worden. In plaats van het inpakken van bestaande functionaliteit zal het meer gaan bewegen in de richting van kleine deeltjes van component services die op verschillende manieren deployed worden en die kunnen samenwerken om applicaties te produceren die business problemen oplossen.'

### Enorme business

*Een deel van de vraag was eigenlijk: wat doen we met al die functionaliteit die in de database zit? Wat het is devies van Oracle: moeten we meer op de Java-manier in de appserver stoppen, of toch meer van de functionaliteit van de database gebruiken?*

Harris: 'In de eerste plaats: middleware is een enorme business voor Oracle, het is erg succesvol geweest het laatste jaar, meer dan 28.000 klanten, 70% van de Fortune 500 gebruikt het, 24 % (uit het hoofd) groei over de laatste twaalf maanden. Oracle gelooft in middleware. We hebben een tijd als devies gehad: je

plaatst applicatielogica daar waar het zinvol is. Soms betekent dat het plaatsen van de applicatielogica bij de data, soms in de middleware. We hebben niet gezegd dat je het met PL/SQL of

***'Wij propageren niet het uit elkaar trekken van bestaande systemen om ze te vervangen'***

Java moet doen, ze zijn allebei beschikbaar in de database, ze gebruiken dezelfde runtime infrastructuur in de database enzovoorts. Het feit is: de meeste mensen stoppen hun applicatielogica in de middle tier. Ze stoppen soms ook applicatielogica in de database, bijvoorbeeld bij datatransformaties. Ik denk niet

## ***Advertentie Masterclass***

dat het óf óf is, we bevelen ook één van beide manieren aan. Wat we propageren is bouwen gericht op een SOA.'

*Binnen de open source beweging zie je allerlei ontwikkelingen waarvan ik me afvraag of Oracle daar blij mee is: veel frameworks, een enorme wildgroei, waarbij sommige mensen spreken van een XML-'tagsoup'. Wordt dat niet geen enorme legacy over een paar jaar?*  
Harris: 'Tja, in zekere zin plaats je je framework bovenop frameworks, Struts is een goed voorbeeld, het is geen standaard maar een de facto standaard. We certificeren deze frameworks over de hele stack. Er is een tendens om combinaties van dingen te gebruiken, de LAMP stack, (Linux, Apache, MySQL, PHP) en de SASH stack (Spring, Apache Axis, Struts, Hibernate) zijn daar voorbeelden van. Wij certificeerden de SASH stack tegen OC4J. Ja, je hebt gelijk, er is een proliferatie van frameworks, mensen neigen ernaar dingen te gebruiken die voor hen werken. Een deel van ons werk bestaat eruit het werken met deze frameworks te vergemakkelijken. Er zijn sommige dingen die niet helemaal zo slecht zijn als je

## ***'Met Project Fusion stemmen we de toekomstige versies van de applicaties op elkaar af'***

het karakteriseert, een daarvan het verdergaande gebruik van annotatie als een mechanisme. Er zijn annotaties met JSR 181, waar je een annotatie doet en zegt dit is een EJB en het aan de runtime overlaat om al die dingen voor je te doen. Die trend naar annotaties is een afkortingsmechanisme om dingen meer benaderbaar te maken. Als je op die standaardisatie van annotaties een laag zet - en dat gebeurt in de Java-gemeenschap - dan heb je een manier om mensen die macht te geven zonder het een chaos wordt. Een andere trend: Spring is heel handig en wordt door veel mensen gebruikt. Spring maakte dependency-injectie populair. Het werkt op een heel *loosely coupled* manier, als trend, het ziet uit naar meer frameworks maar het is ook een manier van programmeren zoals AOP. Je ziet toch een convergentie op een beperkt aantal zaken: annotaties, standaardisering daarvan, dependency-injectie, inversie van controletechnieken. De trend is ook gebruik te maken van tools. Daardoor gaan dingen natuurlijk aanvoelen, al betreft het dan bijvoorbeeld BPEL. Het wordt meer benaderbaar.'

## **Middleware**

*Project Fusion moet een enorme taak zijn, gaat dat op tijd lukken?*  
Harris: 'Ik sta er middenin. Een deel van de commitment is de bescherming van bestaande investeringen van klanten, en voor-

uitgang van die onderdelen. Er komen voor alledrie de producten in mei weer releases. Die dingen zijn allemaal gecertificeerd op de gewone Fusion-middleware. Eén van de verwarrende aspecten van Fusion is dat het middleware betreft die al door 28.000 bedrijven en organisaties gebruikt wordt. Project Fusion is de inspanning om de toekomstige versie van de applicaties op elkaar af te stemmen. Ook moet nu alle software getest worden voor WebSphere.'

*Dat probleem was er in het verleden niet.*

Harris: 'Ja, dat soort problemen hadden we niet, maar het is een minder groot probleem dan het lijkt. Als we het over *hot pluggable* hebben, dan zijn dat niet alleen woorden, het werkt ook. Voor het project Fusion is de requirements-fase voorbij. De specificaties en de aankomende release van Fusion Middleware zijn zo gecoördineerd dat we de nieuwe applicaties meteen voor de *fusion middleware* bouwen. We willen ook een standaard bouwen voor de manier waarop we jobs scheduleren. Dat wordt dan een deel van de middleware en de applicaties kunnen gebruik maken van die faciliteit. Als je een *Applications* klant was, krijg je wat je gewend was, maar als je middleware-klant bent krijg je ineens een extra mechanisme voor jobscheduling. Zowel middleware en applicaties hebben er voordeel van.'

*Wat is de positie van Oracle ESB naast BPEL en Interconnect? Zit InterConnect onder ESB bijvoorbeeld? (LJ)*

Harris: 'ESB is een centraal stuk van het SOA-platform; van de SOA-suite in feite. Interconnect is een subset van ESB, we gebruiken er ook delen van. Ik kan echter niet zeggen hoe de verhouding licentietechnisch precies zit, maar technisch gezien zal InterConnect opgaan in ESB. ESB wordt ook gebouwd door hetzelfde team, dus ook in die zin groeit het samen.'

*Met dank aan Lucas Jellema voor de vragen waar (LJ) bij staat. Lucas suggereerde overigens nog meer vragen, maar dat waren vragen waar Harris geen antwoord op gaf.*

Tekst en fotografie: **Dré de Man.**