

**Van 19 tot en met 23 september vond in San Francisco de jaarlijkse JavaOne conferentie plaats. Voor het eerst in de geschiedenis van JavaOne werd de conferentie door een andere partij dan Sun georganiseerd, namelijk door Oracle en vond het gelijktijdig plaats met de jaarlijkse Oracle OpenWorld conferentie.**

# Een spannend JavaOne

## Doorgewinterde congresganger moet wennen

**O**p de aanplakbiljetten stond te lezen: “One city, one week, three conferences”. Drie zul je denken? Ja drie, want naast JavaOne en OpenWorld was er ook nog Oracle Develop. Oracle Develop is normaal gesproken de technische track van Oracle OpenWorld. Voor de gelegenheid was Oracle Develop dit jaar gekoppeld aan JavaOne. Registreerde je voor de JavaOne conferentie, dan mocht je ook naar Oracle Develop en vice versa.



### Hooggespannen verwachtingen

Voor Oracle was deze conferentie van enorm groot belang. Het had zich stellig voorgenomen te bewijzen een waardige ‘steward of Java’ te zijn, die Java revitaliseert, klaarstoomt voor de nabije en verder weggelegde toekomst en krachtig positioneert ten opzichte van concurrerende platforms en technologieën. Daarvoor wilde Oracle perse tijdens deze conferentie een serie heel concrete aankondigingen doen en plannen ontvouwen die gevoelens van een geslaagde doorstart, serieuze vooruitgang en versterkt vertrouwen zouden opwekken.

Voor de belangrijke onderdelen van Java – de taal, het platform, de community – moest deze JavaOne een helder en geloofwaardig statement afgeven: de korte en langere termijn plannen van Java SE, ME, FX en het Java EE enterprise platform. Daarnaast bestonden veel vragen over de toekomst van diverse voormalige Sun producten, zoals de JDK/JVM (Hotspot Virtual Machine), GlassFish applicatie server, NetBeans IDE en de MySQL database.

De afgelopen jaren – eerst deels door beperkte financiële ruimte bij Sun gekoppeld aan een zeker onvermogen in die organisatie om mooie voorname-ns in concrete actie om te zetten, daarna sterk beïnvloed door de langdurige overnameperiode en complexe integratie van Sun in Oracle – gebeurde er weinig concreets, richtinggeevends en enthousiasmerends rondom Java.

Het laatste wapenfeit – de release van Java EE 6 – was het resultaat van een proces dat al in 2006 in beweging was gezet, en daarmee niet maatgevend voor de toestand van Java. Tijdens JavaOne 2010 was het aan Oracle om de stagnatie en zelfs licht negatieve spiraal te doorbreken met echt nieuws, concrete plannen en reëel uitzicht op de volgende evolutieronde. Het weer aanhaken van de Java community – zeker geen vanzelfsprekendheid gezien de wijdverbreide scepsis met betrekking tot Oracle – moest met deze conferentie gerealiseerd worden.

Vaste bezoekers van JavaOne wilden eigenlijk hun feestje van een paar jaar geleden terug, waar tijdens de absolute hoogtijdagen in 2006 en 2007 het Moscone Conferentie Center door pakweg 15.000 Java enthousiastelingen werd beheerst en iedere hoek en ieder gaatje passie voor technologie ademde. De laatste jaren was al een deel van de omvang, de speciale sfeer en de focus verloren gegaan – ook al deed Sun haar best met doorzichtig geflatteerde bezoekersaantallen de schijn op te houden. Als dan niet de tijden van weleer zouden herleven, dan wilden deze alumni toch in elk geval kunnen constateren dat Java ‘alive & kicking’ is en dat onder nieuwe bewindvoerders ook een nieuwe toekomst gloort – weliswaar misschien anders maar toch zeker veelbelovend en geloofwaardig. Met een duidelijke plek voor de community, voor individuele initiatieven en open source projecten – en vooral zonder eenzijdige dominantie van Oracle.



**Lucas Jellema**

is Senior Consultant op het gebied van Java en SOA bij Amis.



**Paul Bakker**

is trainer/consultant bij InfoSupport.



**Bert Ertman**

is Technology Manager van het Java Competence Center bij InfoSupport.

En natuurlijk een JavaOne conferentie waar de Java-gemeenschap een honk heeft, waar een stevig technisch inhoudelijk programma staat met ruimte voor verschillende inzichten, leveranciers en creatieve ideeën. Hopelijk zouden ook de vertrouwde gezichten, de iconen en ‘thought-leaders’ aanwezig zijn en daarmee niet alleen hun inbreng hebben maar ook de geloofwaardigheid van de conferentie helpen vaststellen. Kortom, de verwachting waren redelijk hoog gespannen.

### Even wennen

Voor de doorgewinterde JavaOne-ganger was het allemaal even wennen. Het vertrouwde Moscone convention center was ingenomen door de stropdassen van Oracle OpenWorld en op een enkele keynote sessie na voelde je je eigenlijk niet echt welkom in Moscone. Soms werd je ook echt letterlijk weggestuurd. Voor de meeste OpenWorld badges binnengelaten: er waren twee soorten toegangspassen – een voor JavaOne en Oracle Develop (met name voor software ontwikkelaars) en een voor Oracle Open World (vooral voor IT en Business management, gebruikers van ERP & CRM applicaties en specialisten op het vlak van infrastructuur). De eerste speelde zich grotendeels af in drie hotels en de tent op de tussenliggende straat – voor de gelegenheid ‘The Zone’ gedoopt – terwijl de tweede plaatsvond in het Moscone-center op pakweg 15 minuten lopen.

De keynotes werden live gestreamed en voor de JavaOne gangers betekende dat kiezen tussen meekijken in de Grand Ballroom van het Hilton hotel, of je eigen laptop op je hotelkamer. Een groot onderdeel van het Hilton was dat er tijdens de keynotes



## Java SE 7 en 8

Eind vorig jaar is er een roadmap voor Java 7 gepresenteerd. Volgens die roadmap had Java 7, inclusief closures, rond JavaOne beschikbaar moeten komen. Als je het nieuws, of juist het ontbreken daarvan, rondom Java 7 de afgelopen maanden hebt gevolgd was het al enige tijd duidelijk dat dit niet realistisch was. Tot op de dag van vandaag is er nog geen JSR voor Java 7, en van een aantal prominente features is de exacte uitwerking nog niet duidelijk. De reden voor het volledig missen van de release-datum komt voor een deel ook door het concreet maken van de Oracle/Sun overname dit jaar, waardoor het werk behoorlijk vertraagd is. Op JavaOne dit jaar was Java SE dan ook een van de onderwerpen waar iedereen op nieuws hoopte en dat is absoluut waargemaakt.

Oracle heeft besloten de plannen voor Java SE in twee releases op te delen. Java 7 wordt een evolutionaire release met alleen nieuwe features die slechts beperkt impact hebben op de taal en het platform. Java 8 wordt vervolgens een revolutionaire release waarin grote vernieuwingen zoals Lamda expressies (closures) zijn opgenomen. Op Java 7 moeten we nog wel een jaar wachten. Het goede nieuws is dat Java 8 al een jaar daarna beschikbaar moet worden, wat ons dan inmiddels in 2012 brengt.

### Verstandig

Het is waarschijnlijk alleen maar heel verstandig dat Oracle heeft gekozen om de release op te splitsen en verder uit te stellen. Als features eenmaal in de taal of de virtual machine terecht komen is dat vrij definitief. Om backwards compatibiliteit te garanderen kunnen features achteraf niet meer zo maar aangepast of verbeterd worden. Met dit in het achterhoofd zijn we er als Java ontwikkelaar allemaal bij gebaat dat features op de juiste manier geïntroduceerd worden, ook al moeten we daar dan misschien wat langer op wachten.

Oracle maakt hier ook duidelijk een statement mee; Java is belangrijk en wordt met een lange termijn strategie doorontwikkeld. Oracle is ook een partij die pas uitspraken over roadmaps doet als het zeker is dat die ook haalbaar zijn. Overigens wordt binnenkort toch nog een JSR voor Java 7 opgesteld.

### Java SE 7

Als we naar de features van Java 7 kijken is dit zoals eerder gezegd een evolutionaire release. Net zoals in Java 1.4 en Java 6 zijn de echte vernieuwingen daarmee beperkt, maar niet onbelangrijk. Het lijstje van vernieuwingen is als volgt:

- \* JSR 292: Support for dynamically-typed languages (InvokeDynamic)
- \* Languages update of the project Coin
- \* Concurrency and collections updates (jsr166y)
- \* Ionet JSR 203: More new I/O APIs for the Java platform (NIO.2)
- \* SCTP (Stream Control Transmission Protocol)
- \* SDP (Sockets Direct Protocol)
- \* Elliptic-curve cryptography (ECC)
- \* Client XRender pipeline for Java 2D
- \* Create new platform APIs for 6u10 graphics features
- \* Nimbus look-and-feel for Swing
- \* Swing JLayer component

### InvokeDynamic

De eerste drie vernieuwingen zijn voor de meeste ontwikkelaars waarschijnlijk het interessantst. InvokeDynamic zorgt voor betere ondersteuning van dynamische talen op de Java Virtual Machine en is van groot belang voor bijvoorbeeld Groovy en JRuby. Dynamische methoden (methoden die alleen runtime op een klasse bestaan) worden nu niet in de Virtual Machine ondersteund en worden nu met allerlei omwegen door dynamische talen geïmplementeerd. De performance hiervan is dan ook verre van optimaal.

InvokeDynamic in Java 7 geeft de Virtual Machine wel directe ondersteuning voor dynamische methoden. Dit is voor de taal Java zelf niet zo heel interessant maar voor alternatieve van uitermate belang.

### Project Coin

Project Coin is een verzamelnaam voor een aantal kleine taal vernieuwingen. Dit zijn weliswaar geen enorm vernieuwde wijzigingen maar wel handigheidjes. Denk hierbij bijvoorbeeld aan de Diamond Syntax om het gebruik van Generics eenvoudiger te maken zoals in het voorbeeld hieronder is te zien.



## Java 6

```
Map<String, Collection<Integer>> map=new
LinkedHashMap<String,
Collection<Integer>>();
```

## Java 7

```
Map<String, Collection<Integer>> map=new
LinkedHashMap<>();
```

Het type aan de rechterkant van de declaratie wordt door de compiler afgeleid en hoeft niet meer expliciet opgegeven te worden. Andere voorbeelden van kleine vernieuwingen zijn het gebruik van Strings in switch statements en automatisch resource management (het automatisch sluiten van resources zoals files).

## Fork-Join framework

De derde interessante Java 7 vernieuwing is een geheel nieuwe API om efficiënt taken over meerdere threads uit te smeren. Deze API is bekend als het Fork-Join framework. Gezien het groeiende aantal cores in systemen is dit een essentiële API om ook in de toekomst optimaal gebruik te maken van beschikbare hardware. Deze API wordt echter pas echt interessant met de komst van Java 8, waarbij de API is te gebruiken in combinatie met Lambda expressies. Tot die tijd blijft de API weliswaar zinvol, maar met een wat onhandige syntax.

## Java SE 8

Het lijstje met vernieuwing voor Java 8 is op dit moment weliswaar nog wat korter, maar wel een stuk indrukwekkender. De meest in het oog springende vernieuwing is de ondersteuning voor Lambda expressies (closures). Dit gaat het mogelijk maken om bepaalde APIs en code constructies veel eleganter te maken en heeft vooral voordelen bij het werken met collecties en concurrent code. Belangrijk hierbij is dat er heel goed wordt gekeken hoe dit in Java kan worden ingepast zonder backwards compatibility te breken, maar ook hoe dit op bestaande APIs kan worden aangesloten. Het zou de JDK een stuk complexer maken (vooral voor nieuwe ontwikkelaars) als we straks bijvoorbeeld twee Collection APIs zouden hebben; de bestaande en een nieuwe met ondersteuning voor Lambda expressies.

Hier is echter een elegante oplossing voor bedacht. Iedere bestaande methode die als argument een geïmplementeerde interface met een enkele methode heeft, kan straks worden aangeroepen met een Lambda expressie als argument. Hiervoor zijn geen wijzigingen in de API noodzakelijk terwijl Lambda expressies wel op allerlei plaatsen direct bruikbaar zijn. Hieronder staan twee voorbeelden van het gebruik van Lambda expressies binnen bestaande APIs.

```
executor.submit({ -> println("Boo"); });
btn.addActionListener({ ActionEvent e ->
println("Boo") });
```

## Project Jigsaw

De andere grote vernieuwing in Java 8 is terug te vinden in Project Jigsaw en heeft alles te maken met modulariteit. Hiermee moeten twee problemen worden opgelost; de welbekende "JAR Hell" en opsplitsing van de JRE zelf.

De enige mogelijkheid die we nu binnen Java hebben om gebruik te maken van libraries is het classpath. Het classpath houdt echter geen rekening met bijvoorbeeld versionering of transitieve afhankelijkheden. Dit model is veel te zwak bij het ontwikkelen van grotere applicaties en zorgt al snel voor lastig op te lossen problemen. OSGi biedt hier weliswaar oplossingen voor, maar dit staat nog te ver af van standaard Java. Afhankelijkheden naar andere libraries moeten we in Java 8 op de volgende manier kunnen definiëren.

```
// module-info.java
module com.foo @ 1.0.0 {
requires org.bar.lib @ 2.1-alpha;
requires edu.baz.util @ 5.2.11;
}
```

Dit mechanisme wordt vervolgens ook gebruikt om de JRE zelf op te splitsen. Daarmee wordt het mogelijk om applicaties veel sneller op te starten als er nog geen JRE geïnstalleerd is door alleen de benodigde onderdelen van de JRE te downloaden die een applicatie ook echt nodig heeft. Hiermee kan onder andere de performance van Applets enorm worden verbeterd.

een bar was geopend waar je een biertje (of iets anders) kon halen. Wat dat betreft dus weer niets te klagen. Deze keynotes zijn overigens voor iedereen nog steeds te bekijken, via de URL: <http://onemandpreview.vportal.net/>.

Waar je in het verleden in een van de vele grote zalen van het Moscone zat, moest je het nu af en toe doen met kleine achterafzaaltjes in een van drie JavaOne conferentiehôtels. Na een doolhof van gangen stond je dan opeens in een zaaltje waar soms amper 50 man in paste. Een vervelende bijkomstigheid was dat vanwege de variatie in zaalgrootte tussen de verschillende hotels er regelmatig sessies opnieuw werden ingedeeld en als je dan eindelijk het juiste zaaltje had gevonden bleek de sessie waar je voor kwam te zijn verdwenen. Een moderne oplossing voor dit probleem was de iPhone/iPad applicatie waarin – als je een internetverbinding had – de meest recente informatie wel terug te vinden was. Wat minder goed werkte met deze applicatie was last-minute van sessie wisselen. Net als de laatste jaren bij JavaOne het geval was, moest je je ook nu van te voren aanmelden voor een sessie. Stond je niet op de lijst dan werd je resoluut naar een wachtrij verwezen. De wachtrij mocht de zaal pas in als de sessie was begonnen en er nog plaatsen vrij waren in de zaal. Op zich een goed systeem zou je denken, echter niet alle updates kwamen ook daadwerkelijk terecht bij de zaalwachter, waardoor je dus soms onterecht naar de wachtrij werd verwezen.

## Logistiek

De logistiek van JavaOne was op zich prima geregeld. Bij de roltrappen stond personeel dat bewaakte dat er niet teveel mensen tegelijk van verdieping wisselden zodat grote verstoppingen uitbleven. Tussen de sessies door werd er netjes voor een natje en een droogje gezorgd en de lunches waren weliswaar enigszins van vliegtuigmaaltijd kwaliteit, maar verder goed binnen te houden.

Mede door enkele sessies die je toch in Moscone kon volgen en wat gezamenlijke evenementen en feestjes om de conferenties heen, had je wel het gevoel deel uit te maken van het grote geheel. Een evenement waarvan de gezamenlijke omvang ruim 41.000 bezoekers betreft heeft natuurlijk bizarre feitjes tot gevolg.

Zo waren er bezoekers uit 116 verschillende landen, die gezamenlijk ruim 67.000 hotelovernachtingen hadden geboekt, was er in totaal voor 2300 uur aan sessies te volgen, van 1735 sprekers, werden er 141.000 koppen koffie geschonken en lag er ongeveer 450 kilometer aan netwerkkabel.

Nog indrukwekkender dan voorgaande cijfers is de berekening dat de Bay Area, waarvan San Francisco deel uitmaakt, gedurende de OpenWorld week een economische impuls van ongeveer 100 miljoen dollar krijgt!

De auteurs van dit artikel kunnen bevestigen dat de koffers een stuk voller terug kwamen dan dat ze heen gingen.

Een ding moet je Oracle zeker nageven. Ze weten absoluut hoe je een feestje moet geven.

Het grote hoogtepunt op dat gebied was natuurlijk het ‘Appreciation Event’ op woensdagavond, waar headliners als Don Henley en de Black Eyed Peas spetterende optredens gaven, maar ook de verschillende bijeenkomsten, netwerkborels van lokale Oracle afdelingen, bedrijfsfeestjes of andere uitjes waren allemaal op tot de verbeelding sprekende locaties en rijkelijk voorzien van eten en drinken.

## Belangrijkste aankondigingen

“De ogen van de wereld zijn op ons gericht...” begon Thomas Kurian zijn keynote. “De Java wereld verwacht van ons een duidelijke product roadmap voor de komende drie jaar voor alle belangrijke technologieën op het Java platform.” En dat was ook precies wat hij in de anderhalf durende keynote ging doen. De eerste horde was genomen, de Java wereld hing aan zijn lippen.



## Java SE

De belangrijkste aankondigingen op het gebied van Java SE waren dat wat tot nu toe werd gezien als Java SE Release 7 is gesplitst in Java SE 7 – zomer 2011 – en Java SE 8 – 2012. Java SE 7 bevat de lichtere verbeteringen – inclusief eenvoudiger integratie tussen Java en andere (dynamische) talen – JRuby,

## De toekomst van Enterprise Java

Eind 2009 is Java EE 6 officieel uitgekomen. Hoewel dit al bijna een jaar geleden is begint dit nu pas langzaam in de praktijk beschikbaar te komen. Het wachten is uiteraard op ondersteuning voor Java EE 6 in applicatieservers, en dat heeft altijd even tijd nodig. Omdat daarmee voor de meeste ontwikkelaars Java EE 6 nog ontzettend nieuw is werd hier ook nog uitgebreid aandacht aan besteed tijdens de keynotes op JavaOne. Inmiddels begint de ondersteuning in applicatieservers langzaam op gang te komen en zijn in een aantal applicatieserver in ieder geval delen van Java EE 6 te gebruiken. Denk hierbij aan onder andere Glassfish, WebSphere (o.a. JPA 2), JBoss (vrijwel volledig), WebLogic (JSF 2.0 en JPA 2.0; gepland volledige JEE 6 compliance in december 2011) en Resin (volledig Web Profile).

Ondertussen wordt er uiteraard ook nagedacht over de volgende versie van Java EE. De focus ligt hierbij op dit moment bij de volgende onderwerpen:

- Cloud computing
- Modularity
- Web Tier
- API enhancements

Cloud computing is een trend die inmiddels eigenlijk niet meer te negeren is. Dat betekent dat ook Java EE hier beter op moet gaan aansluiten. Binnen de grote verzamelaar “Cloud” is een van de trends NOSQL, meestal uitgelegd als “Not Only SQL”. Alternatieve datastores – memory-grids spelen frequent een rol – worden steeds vaker ingezet

om met de schaalbaarheid eisen waar internet applicaties steeds vaker mee te maken krijgen om te kunnen gaan. Een van de ideeën voor Java EE 7 is het uitwerken van een standaard API voor dit soort datastores zoals we voor relationele databases JDBC en JPA hebben. Een ander probleem waar Java EE mee te maken krijgt is de andere manier van het hosten van applicaties. In een PaaS omgeving draaien vele applicaties, mogelijk van verschillende klanten, binnen dezelfde omgeving. Hierdoor wordt het belangrijk applicaties beter van elkaar te kunnen isoleren en applicaties beter te kunnen versioneren.

Modulariteit van applicaties is ook een belangrijk thema voor de volgende versie van Java EE. Dit heeft natuurlijk ook alles te maken met de mogelijkheden die Java 8 hiervoor gaat bieden. Ook de nieuwe taalfeatures, met in het bijzonder Lambda expressies, gaan een grote rol spelen in het verbeteren van de Java EE APIs in de toekomst. Dat betekent echter wel dat we Java EE 7 afhankelijk wordt van Java 8, en daarmee zeker niet voor 2012 beschikbaar komt. Dat hoeft geen probleem te zijn, versie 6 van Enterprise Java begint nu pas langzaam in de markt te landen. Aan de andere kant blijft de ontwikkeling van alternatieve frameworks in die tijd natuurlijk niet stil staan.

lets concreter wordt het als we over de WebTier van Java EE praten. Hier is vooral betere integratie met HTML 5 nodig. Denk daarbij aan een standaard API voor de serverkant van Web Sockets en een gestandaardiseerde JSON library. Al met al zijn de plannen voor Java EE 7 nog vrij vaag, maar de ideeën zijn in ieder geval zeer interessant.

**De product managers hebben het nu bij Oracle voor het zeggen.**



**Na een doolhof van gangen stond je in een zaaltje waar amper 50 man in paste.**

Scala, Groovy, JavaScript – die op de JVM draaien. Het ingrijpender en hier en daar controversiële werk – waaronder closures en modularization – is naar Java SE 8 doorgeschoven. De Sun JDK gaat verder als de Oracle JDK, met hetzelfde licentiemodel. Deze JDK wordt voor Java 7, 8 en verder gebaseerd op OpenJDK, een Open Source project

### MySQL Sunday

Op de zondag voorafgaand aan de conferentie vond een tot voor kort ondenkbaar evenement plaats: onder de vlag van Oracle, op de conferentie van de grootste commerciële database leverancier in de wereld, vond MySQL Sunday plaats, een dag volledig gewijd aan de gratis, open source database MySQL. Met maar liefst 6000 vooraf geregistreerde bezoekers (die overigens niet allemaal kwamen opdagen)

Op MySQL Sunday kondigde Oracle de sterk verbeterde open source 5.5 release aan van MySQL. Met ingang van deze release bevat MySQL (weer) InnoDB als standaard database engine. Daarmee komt nu standaard ondersteuning voor ondermeer Foreign Keys, ACID transacties en Crash Recovery in MySQL. Verdere verbeteringen in MySQL zijn er op het gebied van availability (replicatie) en gebruiksgemak, naast verhoogde schaalbaarheid – bijvoorbeeld via partitionering van tabellen en indexen en vereenvoudigde clustering. Ook de MySQL Workbench 5.2 – een IDE voor ontwerp en (eenvoudige) administratie van de database en ook SQL ontwikkeling – werd warm aanbevolen. Maar de meest in het oog lopende verbeteringen liggen op het vlak van performance. In vergelijking met MySQL 5.1 zijn in benchmarks onwaarschijnlijke verbeteringen gevonden, op Windows tot een factor 15, onder Linux tot een factor 3.

Oracle werkt aan integratie van MySQL met haar enterprise producten voor database beheer, data integratie en biedt daarnaast support voor MySQL via de Oracle World Wide support organisatie. Daarnaast ziet Oracle MySQL als ‘wapen’ op Windows tegen SQLServer. Naast MySQL biedt Oracle overigens nog twee gratis database producten: Berkeley DB (open source, zowel in C als C++ als Java beschikbaar) en Oracle XE (de gratis, gelimiteerde versie van Oracle RDBMS).

Van Google sijpelde het bericht uit dat de Google AppEngine naast de huidige niet-relatieve BigTable storage oplossing op korte termijn ook MySQL zal ondersteunen. Zoals het door een Google medewerker werd omschreven: “We hebben MySQL genomen, de open source bron-code, en deze geschikt gemaakt (lees: een fork gecreëerd) om in de AppEngine toe te passen.” (vervang de woorden AppEngine door Android en MySQL door OpenJDK en je hebt een zeer gevoelige discussie te pakken tussen Oracle en Google; het is interessant om te horen wat Oracle van deze toepassing van MySQL vindt).

waar Oracle tientallen ontwikkelaars voor inbrengt. De Hotspot JVM wordt door Oracle samengevoegd met de JRockit JVM die Oracle via BEA in handen kreeg. Het resultaat van deze combinatie wordt door Oracle ingebracht in het OpenJDK project.

### Java EE 7

Het belangrijkste nieuws op het vlak van Java EE 7 is dat het JSR project voor Java EE 7 in de komende maanden wordt opgestart. De verwachting is dat er meerdere profiles – naast het Web Profile – worden gedefinieerd om lichtere Java EE servers mogelijk te maken, binnen de standaard specificaties. Een overkoepelend thema voor Java EE 7 is een streven naar meer modulariteit in applicatie servers en meer programmeergemak via dependency injection en vereenvoudigde configuratie (overigens een trends die al vanaf Java EE5 is ingezet).

### Java FX

JavaFX wordt geïntegreerd in Java SE – en komt daarmee ook beschikbaar voor andere (dynamische) talen die op de JVM draaien (bijvoorbeeld Groovy, JRuby en Scala). JavaFX script gaat verdwijnen. JavaFX krijgt twee-weg interoperabiliteit met JavaScript en HTML 5/DOM. JavaScript kan de FX SceneGraph benaderen en manipuleren; FX kan evenzo de DOM tree manipuleren en JavaFX kan rich HTML bevatten en renderen. FX krijgt ook ‘native’ data binding faciliteiten met ondermeer REST, JDBC, JSON en XML. De JavaFX UI Controls zullen worden gereleased onder een Open Source licentie.

Oracle werkt aan een nieuwe 2D en 3D graphics engine – onderdeel van de Prism stack – die van moderne hardwareversnellers gebruik maakt om real-time graphics te renderen. Deze engine zal ondermeer JavaFX gaan bedienen.

### Java ME

Verbeteringen voor Java ME – de Mobile Edition – werden ook aangekondigd. ME zal de WebKit Engine en een JavaScript Engine gaan bevatten waarmee HTML, CSS en JavaScript kunnen worden gerenderd en uitgevoerd. Ook onder ME komen verbeterde, op moderne hardware versnellers gebaseerde grafische render engines – 60 frames/s in 2D en 3D.

ME gaat APIs bevatten waarmee telefoon-functies als betalingen, GPS, telefoon en SMS vanuit de Java ME applicatie aangesproken kunnen worden.

In een JavaOne keynote mogen ook de gebruikelijke feel-good statistieken niet ontbreken. We willen ze jullie uiteraard niet onthouden:

- 1.1 miljard desktops draaien Java
- 930 miljoen Java Runtime Environment downloads per jaar
- 3 miljard mobiele telefoons draaien Java

## JDeveloper vs Eclipse vs NetBeans

Een paar jaar terug was de wereld een stuk eenvoudiger: Oracle had haar eigen Java IDE – JDeveloper – met een betrekkelijk klein marktaandeel – en ondersteunde een aantal Eclipse gerelateerde projecten (zoals Dali, Facelets tool, JSF tooling). BEA leverde een IDE onder de naam WebLogic Workshop, in feite een uitgebreide set plugins voor Eclipse. Sun was de trotse eigenaar van NetBeans, een derde Java IDE.

Veel verandert er niet: Oracle gaat door met ondersteuning van Eclipse, zowel via diverse projecten van de Eclipse Foundation als via het zogenoemde OEPE tool (Oracle Enterprise Pack for Eclipse), plugging die productiviteit en functionaliteit toevoegen aan Eclipse, met ondermeer de interessante App X-Ray feature die applicaties en de afhankelijkheden van de applicatie componenten doorlichten en rapporteren, bijvoorbeeld voor impact analyse of release management. Ook NetBeans wordt voortgezet – in 2011 zullen er twee releases verschijnen. Oracle wil van NetBeans de Java IDE maken – met ook ondersteuning voor bijvoorbeeld Java ME en Java FX. NetBeans functionaliteit voor JVM talen als Groovy en Scala verwacht Oracle van de ‘community’. Tenslotte JDeveloper is vooreerst de IDE voor Fusion Middleware – het wordt niet gepusht om een andere IDE te gaan vervangen, maar zal ook zelf niet plaats maken voor een van de andere twee.

- Per jaar worden er (nu nog) 31 keer meer Java telefoons verkocht dan Apple en Android toestellen samen
- 100% van alle Blu-ray afspeelapparaten draaien Java
- Er worden 1.4 miljard Java Cards geproduceerd per jaar

### Oracle OpenWorld

Oracle maakte duidelijk serieus werk te maken van de via de Sun acquisitie in huis gehaalde hardware divisie. De voornaamste productlijnen ondergaan gestage evolutie. Bijvoorbeeld de nieuwste generatie Sun ZFS Storage Appliance werd aangekondigd, evenals de Sparc T3 – de eerste 16-core processor ter wereld. Daarnaast zet Oracle zwaar in op de combinatie van hardware en software en de onderlinge afstemming daarvan, onder de slogan “Software. Hardware. Complete. Engineered to work together.” Het hoogtepunt in dat verband was de aankondiging door Larry Ellison van de Exalogic Elastic Cloud middleware appliance. Deze appliance bestaat uit een onderling geoptimaliseerde combinatie van hardware – maximaal 30 compute nodes van elk 12 cores met 96 Gb snel RAM geheugen, onderling en met storage verbonden via ultrasnel Infiniband netwerk – en software – Oracle Virtual Machine, Linux en WebLogic Server met Coherence (memory grid)

als ‘secret sauce’. Oracle claimt 12-voudige performance verbetering voor web applicaties en 4,5 keer snellere messaging – tot respectievelijk 1 miljoen HTTP-requests/seconde en 1,8 miljoen JMS berichten per seconde. Hoewel deze appliance, de tweede na de Exadata database machine, een trend lijkt te bevestigen, is de relevantie voor ons gewone stervelingen waarschijnlijk vooralsnog beperkt, gezien het pittige prijskaartje van dit type apparaat.

Naast de aankondiging van Oracle Solaris 11 (in de loop van 2011) werd een tweede Operating System release gemeld: de Oracle Unbreakable Enterprise [Linux] Kernel, onderdeel van Oracle Linux – voorheen bekend als Oracle Enterprise Linux. Deze Kernel is geoptimaliseerd voor het draaien van Oracle software – bijvoorbeeld Fusion Middleware – en het benutten van de Exadata en Exalogic hardware configuraties.

De wellicht voor Oracle belangrijkste aankondiging van deze conferentie: de uitrol in Q4 en algemene beschikbaarheid in Q1 (2011) van Fusion Applications – het nieuwe Enterprise Applicatie

## GlassFish

Na de aankoop van BEA beschikte Oracle al over twee Java EE applicatie servers – OAS of iAS op basis van OC4J en WebLogic Server. De laatste werd al snel als strategisch bestempeld en is ondermeer het fundament onder Fusion Middleware. Sun bracht ook een Java EE applicatie server in, in de vorm van GlassFish, ondermeer de referentie-implementatie voor de Java EE specificaties. Voor zover er al onduidelijkheid was over de toekomst van GlassFish, dan heeft Oracle die nu wel weggenomen: GlassFish wordt voortgezet door Oracle. In 2011 komt Oracle met twee nieuwe releases van de open source applicatie server GlassFish. In beide releases worden ondermeer verbeteringen doorgevoerd op basis van de commerciële WebLogic Server (en andersom overigens ook levert GlassFish verbeteringen voor WebLogic Server).

Oracle geeft aan meer dan 2000 betalende GlassFish klanten te hebben (organisaties die betalen voor enterprise support). De positionering is als volgt: organisaties die Oracle’s Fusion Middleware willen gebruiken moeten WebLogic Server inzetten (of eventueel WebSphere of JBoss) – maar niet GlassFish. Standaard Java EE applicaties kunnen op beide applicatie servers worden gedeployed – en dus zeker ook op GlassFish. NB: GlassFish – als referentie implementatie – heeft altijd als eerste de nieuwste Java EE functionaliteit aan boord (WebLogic Server zal pas tegen het eind van 2011 volledig Java EE 6 compliant zijn), en is dus aan te bevelen aan organisaties die altijd vooraan willen lopen bij de adoptie van nieuwe technologie.

**Oracle  
maakt echt  
werk van de  
in huis  
gehaalde  
hardware  
divisie.**

## Glimpje van JCP & JSR proces: Willen we JMS 2.0?

Het is dinsdagavond, tegen 9 uur. In de gangen en zalen van het Hilton hotel hangt iets van de originele JavaOne sfeer. De bezoekers van Oracle Open World zijn naar hun recepties en cocktailparties en dineetjes. Maar de echte Java enthousiastelingen, de kern van de JavaOne-gangers, zitten bij een van de vele BoF (Birds of a Feather) sessies. Dit zijn intieme, in-depth sessies, in kleinere zalen met overzichtelijke groepen en vaak een niche-onderwerp, zoals een heel specifiek open source project, een zeer geavanceerde toepassing of, zoals in dit geval, een discussie over de toekomst van een Java specificatie: JMS.

Een groep van zo'n 30 geïnteresseerden heeft zich verzameld voor een sessie met de titel: "Java Message Service: Time for Version 2.0?", op initiatief van Tom Barnes en Ed Bratt, beide van Oracle (voormalig Sun). Hun stelling: sinds JMS was geïntroduceerd is er slechts een update van de specificatie geweest, JMS 1.1, en dat was 8 jaar geleden. Is er zo langzamerhand geen behoefte aan een update van de specificatie? En zou dat dan JMS 1.2 moeten worden of is er echt een substantiele herziening van de specificatie aan de orde en zou het 2.0 moeten worden?

Deze groep gaat dat uiteraard niet ter plekke beslissen. Maar de sessie demonstreert wel een stap in het community-proces dat uiteindelijk leidt tot een specificatie voorstel (JSR) dat in verschillende gremia moet worden goedgekeurd voor het daadwerkelijk onderdeel van het Java platform (JEE 7) wordt. Aanwezig zijn enkele vertegenwoordigers van vendors met message servers en JMS implementaties, de vroegere spec-lead voor JMS, gebruikers van JMS en andere geïnteresseerden.

### JMS relevant?

Er zijn verschillende vrij kleine aanleidingen voor deze bijeenkomst: de JMS specificatie staat al zo lang stil – dat op zich zou al als ongewenst kunnen worden beschouwd (de indruk zou kunnen ontstaan dat JMS niet meer relevant is. Op het gebied van messaging is sinds 2002 veel ontwikkeling geweest; allerlei leveranciers hebben op JMS gebaseerde messaging-producten, waar ze elk hun eigen, proprietary uitbreidingen op hebben geïmplementeerd. Het lijkt waardevol om een deel van die extensies onder de JMS

product – volledig gebaseerd op Java & Java EE technologie en SOA & WebService standaarden. Zie verder ook het kader over Fusion Applications.

### Antwoorden op vragen

In de vooruitblik op JavaOne in de vorige uitgave van Java Magazine (#4, 2010) hebben we een groot aantal vragen en twijfels geformuleerd waarop we graag een antwoord, danwel een uitweg zouden

specificatie te brengen zodat ontwikkelaars en beheerders met een standaard model aan de slag kunnen. Tenslotte bevat JMS enkele inconsistenties en onnodige omwegen die vanuit Oracle (Sun) graag eindelijk een keer opgelost worden. Graag zouden Tom en Ed meer integratie willen met andere Java specificaties en ook met niet-Java messaging protocollen. En JMS zou wat hen betreft eigenlijk de nieuwste ontwikkelingen op messaging gebied moeten gaan bevatten. In deze sessie proberen ze steun te verwerven voor een nieuwe JMS specificatie en een gevoel voor de richting die de aanwezigen zouden willen voor deze specificatie.

### Veel ideeën

Ze lopen langs een aantal ideeën voor uitbreiding van de standaard (NB: veel van deze suggesties zijn aanwezig in veel JMS implementaties, maar zijn dan leverancier- en implementatiespecifiek): acknowledgements, meerdere receivers op een JSM Queue, expiry, flow control, content based selectors and poison message handling, (meer) XML payload ondersteuning, wild card topic subscription en een topic hiërarchie, cluster-capable JMS providers, integratie met WS\* protocollen zoals SOAP over JMS, WS-RM en REST.

### Serius ontwikkelen

Ook meer voorzieningen voor administratie en monitoring worden voorgesteld. Integratie met andere talen en platformen: een C API en een API voor .NET. Ook: aansluiting bij AMQP. En dan enkele wat vagere maar populaire thema's: real time messaging, messaging in the cloud, notification-only use cases (focus op schaalbaarheid en grote berichten-volumes) en integratie met SOA.

Met de (overigens weinig concrete) feedback van deze sessie, verdere reacties van messaging leveranciers, het formele overleg in JCP kringen maar ook artikelen zoals dit en verdere berichten binnen en signalen uit de wereldwijde Java gemeenschap wordt stukje bij beetje het voorstel voor een nieuwe JMS specificatie gevormd. Totdat het mogelijk in de loop van 2011 JMS als onderdeel van het Java EE 7 pakket serieus te gaan ontwikkelen.

Zien. Zijn deze antwoorden er ook gekomen? Voor het grootste deel wel. Hoewel de antwoorden niet altijd waren wat we graag hadden willen horen is toch duidelijk te merken dat – hoewel Oracle flink heeft moeten wennen aan die rare Java developer community – er nu een realistische inslag zit in de antwoorden die worden gegeven waaraan je kunt afleiden dat ze het schip dat Java heet onder controle aan het krijgen zijn.

## Fusion Applications

Een van de strategisch belangrijkste software ontwikkelactiviteiten van Oracle in de afgelopen jaren, was de totstandbrenging van Fusion Applications. Deze nieuwe portfolio van Enterprise applicaties vormt de uiteindelijke opvolging voor bestaande Oracle ERP en CRM producten als Siebel, PeopleSoft, JDEdwards en E-Business Suite en gaat de concurrentie aan met vooral SAP. Fusion Applications (FAPPS) is misschien wel de grootste business applicatie die ooit is ontwikkeld. En zij is volledig gebaseerd op Java en JEE.

## Servicegeoriënteerd

De meeste Enterprise applicaties stellen klanten in staat om met proprietary tools – zoals PeopleTools, Oracle Forms en ABAP – extensies te ontwikkelen of volledige maatwerk applicaties te creëren. Oracle heeft FAPPS gebouwd met dezelfde JEE Fusion Middleware 11g stack die voor klanten beschikbaar is voor eigen software ontwikkeling. Deze JEE stack maakt ondermeer veel gebruik van JSF 1.2 – via Oracle's ADF Faces Rich Client Components – en gebruikt ook de Portlet 2.0 (JSR-286) specificatie. FAPPS is volledig servicegeoriënteerd – voor iedere op menselijke gebruikers gerichte schermfunctionaliteit is een

WebService equivalent beschikbaar. Onderliggende standaarden zijn ondermeer JMS, SCA, SOAP, JAX-WS, BPEL, REST – JAX-RS, SDO en JPA (EclipseLink).

## 3000 Ontwikkelaars

Aan FAPPS hebben naar verluid pakweg 3000 ontwikkelaars gewerkt die in totaal zo'n 11000 scherm-eenheden hebben opgeleverd tegen rond de 3500 database tabellen. Deze ultramoderne productlijn vormt een levend bewijs van de mogelijkheden van browser based Java/JEE technologie en kan ook een inspiratie vormen voor andere applicaties.

Functioneel gezien zijn ondermeer opvallend naast de rijke user interface, de integratie van real-time business intelligence, focus op 'management by exception' en taak-beheer met duidelijkheid in wat de gebruiker om welke reden zou moeten doen, de geïntegreerde communicatie-faciliteiten en de 'design time at runtime' opties die applicatiebeheerders in staat stellen de inrichting van de applicatie op runtime aan de organisatie, de afdeling of de gebruiker aan te passen.

Een hele wezenlijke bijdrage daartoe is een verschuiving van de macht binnen de nieuwe Java organisatie. Waar bij Sun voornamelijk de technische lead architects het voor het zeggen hadden op het gebied van doorontwikkeling en innovatie is dat bij Oracle niet meer het geval. Net als bij het voortbrengen van hun eigen commerciële producten wordt er nu slechts op de trom geslagen door de product managers. Hoewel product managers en developers het in de echte wereld niet altijd met elkaar kunnen vinden, zorgt het wel voor realistischere plannings, nadenken voordat er over nieuwe features wordt gecommuniceerd en zou het op de langere termijn wel eens kunnen leiden tot een betere kadans in de technologie-releases.

Het vertrek van enkele technische Sun kopstukken heeft wellicht ook met deze nieuwe manier van aansturing te maken. Het afstaan van de macht is natuurlijk niet voor iedereen even makkelijk. Voorafgaand aan de echte conferentie waren er voor de Oracle ACE en Java Champions communities een aantal speciale bijeenkomsten georganiseerd waar – weliswaar onder NDA – een vooruitblik werd gegeven van de belangrijkste aankondigingen van de conferentie en meteen om feedback werd gevraagd. Bij deze bijeenkomsten waren alle belangrijke mensen uit de Oracle Java organisatie aanwezig. Er werd dan ook druk genetwerkt tussen de Oracle en Java ambassadeurs in het veld en

de aanwezige product managers, lead architects en technologie evangelisten. Er waren slechts een paar onderwerpen die hardnekkig werden vermeden of waar een heel duidelijk "no comment" werd uitgesproken. Een goed voorbeeld daarvan was de rechtszaak tegen Google (zie ook het kader) en ook het vertrek van James Gosling – the father of Java – bij Oracle werd redelijk omzeild.

## Content en trends

Zoals reeds geconstateerd in de vooruitblik op de conferentie was het argument dat Oracle slechts product pitches voor haar eigen producten zou houden absoluut naar het rijk der fabelen te verwijzen als je ook maar een halve blik op het conferentieprogramma zou werpen. De overgrote meerderheid van de honderden JavaOne sessies ging echt over Java. Daar waar het Oracle producten betrof, kon je deze terugvinden in de Oracle Develop track, die zich voor het overgrote deel in een van de drie hotels afspeelde. Naast de in-depth verhandelingen over de aankondigingen tijdens de keynote en de beoogde features voor de komende releases van het Java platform waren er ook nog een aantal andere duidelijke trends te bemerken. Cloud computing in al zijn facetten (PaaS, SaaS, IaaS, oneindige schaalbaarheid, parallel computing, etc), NOSQL, polyglot programming en HTML 5 staken er als trends redelijk bovenuit gezien het grote aantal sessies over dergelijke onderwerpen of zaken die daarmee verband houden.

**Java wordt  
de primaire  
taal voor  
Java FX.**



## ‘JavaFX platform for the richest, most advanced graphical user interfaces’

Ondanks dat JavaFX al sinds 2007 ieder jaar een van de hoogtepunten van JavaOne keynotes is heeft de technologie tot dusver niet kunnen overtuigen. Bugs, slechte tooling en gebrek aan kant en klare componenten zorgen ervoor dat de meeste Java developers JavaFX tot nu volledig links laten liggen. En dat terwijl het keer op keer wordt gepresenteerd als de belangrijkste Java technologie sinds lange tijd. En niet alleen in de wereld van Rich Internet Applications, maar ook bijvoorbeeld voor mobiele apparaten en televisies.

“For all the screens of your life” was de slogan vorig jaar op JavaOne, terwijl JavaFX op geen enkel platform echt is doorgebroken. Niet direct een succesverhaal, en eigenlijk leken er ook nog maar twee mogelijkheden voor JavaFX; de technologie de nek omdraaien, of drastische stappen nemen om het platform weer op de rails te krijgen.

### Strategisch JavaFX

Oracle heeft gekozen voor de laatste optie en geeft duidelijk aan JavaFX als strategisch te zien. “We want to make Java [Fx] the platform for the richest, most advanced graphical user interfaces”. Met ondersteuning voor rich media, animaties, gradienten, blurs, transformaties, schaduwen en reflecties en functionaliteit om rich HTML te renderen. En gericht op de ontwikkelaar en niet zozeer de grafische ontwerper.

De meest opvallende beslissing is om JavaFX Script volledig te laten verdwijnen. De reden hiervoor is dat JavaFX Script een hele declaratieve taal is. Dat is fijn als je user interfaces ontwikkelt, maar een enorm nadeel bij het ontwikkelen van APIs. Tegen dit laatste probleem liep het ontwikkelteam van JavaFX aan bij het doorontwikkelen van het platform. Bovendien zorgt de nieuwe taal ervoor dat bestaande Java tooling niet meer te gebruiken is.

Daarom is besloten om JavaFX een pure Java API te geven en JavaFX Script volledig te laten verdwijnen. Het platform kan dan veel sneller worden doorontwikkeld en alle bestaande Java tooling is direct weer te gebruiken. Dit betekent zeker niet dat het werk van de afgelopen drie jaar volledig weggegooid wordt. Het belangrijkste onderdeel van JavaFX is de runtime die zorgt voor de rendering en zaken zoals media ondersteuning. De runtime hoeft niet aangepast te worden, alleen gaat de runtime anders gebruikt worden.

JavaFX met al zijn rijke grafische mogelijkheden is

straks beschikbaar in Java Client applicaties en in Applets, en zal ook kunnen renderen naar HTML 5 + JavaScript voor omgevingen die geen Java plugin ondersteunen (zoals iPad). Bovendien is FX ook bereikbaar voor alle overige talen op de JVM. Voor de rijke effecten zal Oracle de ontwikkeling van een moderne grafische engine ter hand nemen – met 2D en 3D ondersteuning – , op basis ondermeer van hardware-versnelling en met gebruik van DirectX (onder Windows) en OpenGL (op andere platformen).

### Opnieuw beginnen?

Het laten verdwijnen van JavaFX Script zal niet door iedereen positief ontvangen worden, hoe goed de argumenten hiervoor ook zijn. Bedrijven die de afgelopen jaren geïnvesteerd hebben in JavaFX kennis en applicaties kunnen eigenlijk weer opnieuw beginnen.



Op de conferentie sprak een aantal bezoekers zich hier ook duidelijk over uit.

Een ander nadeel van het verdwijnen van JavaFX Script is dat we weer een stukje terug het Swing tijdperk instappen. Java is geen declaratieve taal. Aan een stuk Swing code is vaak lastig te zien wat de samenhang tussen componenten in de UI is, wat bij JavaFX Script door het declaratieve karakter een stuk eenvoudiger is.

Hier is echter wel een hele elegante oplossing voor in de vorm van een DSL in een alternatieve JVM taal zoals Groovy of Scala. Zo'n DSL kan gebouwd worden op basis van de beschikbare Java APIs en voor de applicatieontwikkelaar het schrijven van een UI net zo eenvoudig maken als in JavaFX Script. Op de vraag of Oracle een DSL gaat leveren werd door Richard Bair, de architect van JavaFX, wifflend antwoord gegeven. Hier wordt naar gekeken, maar zoals het er nu naar uitziet wordt dit aan de community overgelaten. Het gebruik van bestaande alternatieve JVM talen is wel duidelijk onderdeel van de strategie. Dit kan overigens prima werken, zo is er bijvoorbeeld ook een Groovy DSL voor het schrijven van Swing applicaties.

Het werk voor het omschrijven van de JavaFX APIs naar Java is al in volle gang. Begin 2011 komt er een early access release beschikbaar binnen een partner programma. Later in het jaar wordt er een publieke beta gestart. De uiteindelijke release staat voor herfst 2011 op de planning. Dat Oracle deze planning serieus neemt blijkt ook uit het feit dat ze op JavaOne steeds aangaven hard op zoek te zijn naar ontwikkelaars voor het JavaFX team. Naast het omschrijven van de API staat er ook een groot aantal interessante nieuwe features op de planning.

### HTML DOM

Een van de meest belangrijke is de volledige integratie tussen de HTML DOM en JavaFX. Het idee is dat JavaFX nodes als het ware onderdeel worden van de DOM van de pagina en er zodoende naadloze samenwerking tussen JavaScript en JavaFX mogelijk is. JavaScript kan de nodes in de JavaFX SceneGraph benaderen en manipuleren. Ook komt er een aantal belangrijke nieuwe componenten zoals een TableView en wordt er hard aan de performance gewerkt.

Het was ook zeer opvallend dat er helemaal niets meer werd gezegd over JavaFX op andere platforms dan de desktop en browser. Wellicht dat dit in de toekomst weer aandacht krijgt, maar op dit moment is het waarschijnlijk alleen maar heel goed om een duidelijk focus te hebben.

Met nu een duidelijke en realistische focus gaat het Oracle ongetwijfeld lukken over een jaar iets bruikbaar neer te zetten. Of dit nog voldoende is om de concurrentie bij te benen en om het vertrouwen van developers weer terug te winnen zal nog moeten blijken.



Over het vertrek van James Gosling bij Oracle werd niet gesproken.

### JCP blijft een zorgenkindje

Ondanks het goede nieuws dat er vanuit Oracle een duidelijke en heldere product roadmap ligt voor de komende twee a drie jaar voor alle belangrijke onderdelen van het Java platform is er ook aanleiding voor grote zorgen, namelijk de relatie tussen Oracle en het Java Community Process (JCP). Niet alleen de relatie met de JCP vormt een probleem, ook het feit dat zo goed als al het werk binnen de JCP op dit moment tot stilstand is gekomen vormt reden tot ongerustheid. Het is zeer belangrijk dat deze impasse op zeer korte termijn wordt doorbroken om de JCP bestaansrecht te laten houden en te voorkomen dat er door Oracle powerplay gespeeld moet gaan worden zoals Sun dat in het verleden al te vaak heeft moeten doen. De repu-

tatie en credibility van Oracle bij de andere JCP partners zal er echter voor zorgen dat als dit gebeurt het ook wellicht meteen afgelopen zal zijn met de JCP.

Wat niet meehelpt is de draai die Oracle heeft gemaakt toen ze in 2007 opriepen om Java volledig open te maken en de opstelling die ze nu hebben, nu ze de trotse eigenaar van Sun en alle Java patenten zijn. Een aantal van de belangrijke partijen binnen de JCP helpen Oracle maar wat graag herinneren aan het pleidooi uit 2007. Ook James Gosling voert op zijn eigen wijze actie tegen zijn ex-werkgever en heeft zelfs speciale “free Java” merchandize laten ontwerpen. Enkele van deze t-shirts en buttons werden ook daadwerkelijk gedragen door bezoekers van JavaOne.

Heel belangrijk zal de aankomende face-to-face meeting van de JCP zijn die medio oktober wordt gehouden in Bonn. Zal het Oracle lukken om partijen a la Google, IBM, Apache Software Foundation en SAP mee te krijgen, of valt deze meeting ten prooi aan verder politiek steekspel. De hangende rechtszaak tegen Google zal het geheel er niet gemakkelijker op maken. Tijdens JavaOne is door Oracle de toezegging gedaan dat ze naar Bonn gaan met in ieder geval de voorstel-JSR voor Java SE 7 onder de arm. De vraag is of de situatie in Bonn ook zodanig zal zijn dat Oracle deze ook gaat indienen. Als deze JSR er niet komt, dan komt daarmee automatisch het timeframe van Java 7 in gevaar.

Minder duidelijk waren de posities van de JCP-partijen ten aanzien van JavaOne (en Oracle). Waar staan leveranciers als IBM, SAP, VMWare/SpringSource en Red Hat/JBoss maar ook pure open source organisaties als Apache? Accepteren zij Oracle als hoeder van Java? En gaan ze mee in de initiatieven die Oracle ontplooit voor het platform en de taal? Tijdens de conferentie is dit niet geheel duidelijk geworden. We kunnen ons natuurlijk wel afvragen welke opties er voor dit soort partijen eigenlijk zijn. Uiteindelijk is Java voor elk van deze organisaties essentieel. Het succesvol door-evolueren is dus noodzakelijk. De JCP spelregels – waar ook Oracle zich aan te houden heeft – laten niet heel veel ruimte voor allerlei eigenzinnige initiatieven. Voortgang frustreren en blokkades opwerpen lijkt uiteindelijk in niemands belang te zijn. Hoewel de spelers eigen agenda's hebben en tot op zekere hoogte concurreren met elkaar zouden ze elkaar hopelijk moeten kunnen vinden in het JCP overleg en de bekrachtiging van acties die Java weer in beweging gaan zetten. Wellicht dat de decennia-lange constructieve samenwerking van IBM en Oracle in het overleg rond de ANSI SQL standaarden een goed voorbeeld kan vormen.

### JavaOne 2011

De conferentie van dit jaar was nog niet afgelopen of de discussie over volgend jaar werd al gestart. Wordt

**De vraag is of de andere leveranciers Oracle als hoeder van Java zien.**

het wederom samengevoegd met OpenWorld? Wordt JavaOne dan in Moscone West gehouden? In tegenstelling tot voorgaande jaren waren er in de JavaOne hotels geen banners te vinden met “bedankt, tot ziens en tot volgend jaar ...” teksten erop. Volgens bronnen binnen Oracle is de definitieve beslissing over het wel of niet samenvoegen van JavaOne en OpenWorld nog niet genomen. Eerst dient een uitgebreide evaluatie plaats te vinden en wordt er goed gekeken naar de geluiden uit de community. Die lijkt redelijk eensgezind: dit jaar viel zeker niet tegen, alleen de vibe is er op deze manier wel een beetje vanaf. JavaOne hoort in Moscone en blijft toch vooralsnog het toonaangevende Java event van het jaar. Afwachten dus, we zullen zien waar het toe zal leiden.

### Satellietconferenties

Al wel aangekondigd zijn de JavaOne satellietconferenties in Brazilië en China (met Rusland en India ook nog op de planning). De eerste twee vinden plaats op 7 tot 9 december 2010 in Brazilië en 13 tot 16 december 2010 in China. Zie bijvoorbeeld <http://www.openworldlad.com.br/java-one-en/abertura.php> voor de Latijns Amerikaanse Call for Papers. Dit is voor het eerst sinds 2001 en 2002 – toen er een JavaOne in Japan plaatsvond – dat de JavaOne

conferentie buiten San Francisco zal plaatsvinden. NB: ook deze JavaOne conferenties worden door Oracle gecombineerd met gelijktijdige Oracle Open World en Oracle Develop conferenties – het zijn geen zelfstandige evenementen.

### Meer informatie

Hoe kun je de JavaOne conferentie nog nabeleven en waar kun je de content van de conferentie terugvinden?

Keynotes zijn vrij toegankelijk op <http://on-demandpreview.vportal.net/>. Alle overige (2000+) sessies kunnen hier ook bekeken (slides, soms video) en beluisterd worden – na betaling van \$245, of als je iemand kan vinden die bezoeker was van de JavaOne conferentie en daar gratis toegang aan overgehouden heeft. Veel van de presentaties zijn overigens ook te vinden op SlideShare en via talloze blogs. Een flink aantal demo's kan worden gevonden op YouTube – bijvoorbeeld JavaFX, LiveScribe en NetBeans. Zoek op “javaone 2010”.

Zie <http://openjdk.java.net/projects/jdk7/features> voor de voorgestelde functionaliteit van Java SE 7 en 8. De JavaFX demo in de JavaOne Keynote staat op YouTube: <http://www.youtube.com/watch?v=UXSmJYFrulY>.

## IBM laat Apache Harmony vallen.

### Oracle and IBM gaan samenwerken aan OpenJDK

Oracle en IBM gaan gezamenlijk de openJDK community ondersteunen, zo is kort na JavaOne bekendgemaakt. De OpenJDK community wordt de plek waar de ontwikkeling van Java SE gaat plaatsvinden. Het Java Community Process (JCP) blijft wat IBM en Oracle betreft het belangrijkste orgaan voor de standaardisering van de Java specificaties.

De samenwerking concentreert zich op het OpenJDK project, de open source implementatie van het Java platform, Standard Edition (Java SE) specificaties de Java taal, de Java Development Kit (JDK) en de Java Runtime Development (JRE). Beide bedrijven staan achter de roadmap voor de ontwikkeling van OpenJDK.

Hasan Rizvi, senior vice president van Oracle, noemt in de aankondiging van de samenwerking de Java community van ‘vitaal belang voor de ontwikkeling van het Java platform’. Zijn collega Rod Smith, vice president emerging technologies bij IBM, verwacht dat de samenwerking van IBM, Oracle en de andere leden van de Java community de innovatie van het Java platform flink kan versnellen. “De samenwerking is ook een signaal voor de enterprise klanten dat zij erop kunnen rekenen dat Java meer open, flexibele en nieuwe technologieën gaat bieden om hun business te ondersteunen”, zo voegt Smith er aan toe. Robert (Bob) Sutor, vice president Open Systems en Linux van de IBM Software Group, licht in zijn blog de samen-

werking nog eens toe. “In de open source wereld worden we nogal eens geconfronteerd met het fenomeen ‘forking’. Sommige deelnemers aan een project vinden dat het project niet de goede kant op gaat, stappen eruit met een kopie van de broncode en beginnen voor zichzelf. Er is geen garantie dat dit nieuwe ‘fork’-project zal slagen, maar het heeft wel de functie dat het mensen in staat stelt het project te voltooien zoals het hen voor ogen stond. ‘Reverse fork’ komt veel minder voor. Dit komt erop neer dat twee partijen met verschillende projecten besluiten om samen te werken, nadat zij de strijdbijl hebben begraven omwille van het belang van het project”.

Dat is nu precies wat er volgens Bob Sutor is gebeurd rond de discussie met Oracle over het OpenJDK project. IBM verplaatst de developmentactiviteiten van het Apache Harmony project naar OpenJDK. “Met anderen, die hetzelfde willen doen, zullen we samenwerken om de overstap zo eenvoudig mogelijk te maken”, aldus Sutor. IBM blijft wel betrokken bij andere Apache-projecten.

Sutor noemt de keuze van IBM ‘pragmatisch’. Het was duidelijk dat Sun noch Oracle van plan waren om de belangrijke test en certificerings test voor Java, de Java SE TCK, beschikbaar te stellen aan Apache. Dus liever dan van Harmony een onofficiële en on gecertificeerde Java-variant te maken hebben we gekozen voor OpenJDK”



## Wat wil Oracle echt? Thomas Kurian exclusief

‘Wat gebeurt er nou? Wat doet hij? Gaat dit wel goed?’ Geschrokken, verrast en ook geïntrigeerd zijn de reacties van de Oracle staf die aanwezig is bij de product briefing voor pakweg 40 Oracle ACE Directors voorafgaand aan JavaOne op het hoofdkantoor van Oracle, als Thomas Kurian, de hoogste man binnen Oracle als het gaat om software ontwikkeling, zijn emoties toont – over Java, het conflict met Google en de waardering in de Java gemeenschap voor Oracle.

Oracle kent niet de cultuur van iconen, rock stars en cult figuren die binnen Sun en in de Java Community vrij gebruikelijk waren. Afgezien van Larry Ellison zijn er vrijwel geen OracleVIPs, die het gezicht zijn van een specifiek product of technologie. Als dat wel zo zou zijn, was Thomas Kurian waarschijnlijk een eersteklas bekendheid. Vijftien jaar geleden als programmeur begonnen bij Oracle is hij in een aantal stappen binnen Oracle de ‘top 10’ binnengestroomd, als Executive Vice President verantwoordelijk voor product ontwikkeling, van Fusion Applications tot Database, en daarmee ook de ‘baas van Java’. Op maandagavond verzorgde hij de grote JavaOne Keynote, waarin hij de strategie voor het Java platform uiteenzette, in zijn kenmerkende glasheldere, uiterst correcte en volledige gecontroleerde maar vrijwel emotionele stijl. Hij staat er binnen Oracle om bekend tot in detail op de hoogte te zijn van alle producten en technologie waarvoor hij verantwoordelijkheid draagt en ondermeer alle software zelf te installeren en te testen en ook de documentatie tot in details te corrigeren.

Tijdens de genoemde product briefing liet hij op de vraag ‘wat is de verhouding tussen de JCP partners – hoe reage-

ren ze op Oracle’s initiatieven en met name: waar staat IBM?’ zijn controle varen. In een emotioneel, geïmproviseerd betoog van meer dan 30 minuten legde hij uit waarom Oracle de strijd met Google is aangegaan en vooral hoezeer het hem dwarszit dat talloze bloggers, journalisten en concullega’s allerlei verhalen – grotendeels nonsens volgens hem – verspreiden over de achtergronden, feiten en motieven van Oracle. En hij vanwege de juridische situatie geen publiek commentaar kan geven op al die verhalen.

Kurian gaf aan hoe hij persoonlijk verantwoordelijkheid voelt voor de 9 miljoen Java ontwikkelaars en de toekomst van het Java platform. En dat de achtergrond van het dispuut met Google niets te maken heeft met licentieopbrengsten van Android, maar puur met de angst voor een fork in Java. Zoals Sun jaren geleden Microsoft aanpakte vanwege Visual J++ , omdat een afsplitsing van Java dreigde, zo probeert Oracle te voorkomen dat Google via DalvikVM op Android een eigen variant van Java gaat afsplitsen. Er zijn 27 verschillende JVMs – die allemaal dezelfde specificaties naleven en zorgdragen voor het write once run everywhere principe. Als dat doorbroken wordt, zoals Google volgens Oracle doet, is dat een directe bedreiging voor de kern (waarden van) Java.

De vergaande persoonlijke betrokkenheid en de passie waarmee Kurian zich blootgaf zou de vele skeptici van Oracle een heel ander beeld geven dan ze normaalgesproken krijgen, een veel aansprekender, geloofwaardiger en gevoelvoller beeld. Helaas kon hij, om wat voor reden dan ook, deze spontane intensiteit niet oproepen in zijn publieke optredens tijdens JavaOne.

**‘Wat gebeurt er nou? Wat doet hij? Gaat dit wel goed?’**

