

In de SRM Webtoolgids van 2000 is ColdFusion nog opgenomen als een interessante, efficiënte maar proprietary webtool. Sinds de MX/versie zijn ColdFusion-applicaties echter in feite Java-applicaties. Daarmee is ColdFusion een voorbeeld van een ease of development tool. Daarnaast is het ook zeer interessant gezien de toenadering tussen JSP (2.0) en de eigen scripttaal CFML. Java Magazine sprak met ColdFusion-goeroe Ben Forta.

interview

Ben Forta:

‘Niet iedere applicatie hoeft n-tier MVC te zijn’

Ben Forta was ooit een klant van Allaire: hij kocht in 1995 versie 1.1 van ColdFusion, via een website. CFML (ColdFusion Markup Language) is er dus ook al heel lang, het is ouder dan ASP, PHP en JSP. Het oorspronkelijke doel ervan was om het verplaatsen van data van back-ends naar webpagina's en vice versa te vergemakkelijken. Aanvankelijk was het uitsluitend bestemd voor het Windows-platform. Het was geschreven als een extensie op een webserver, aanvankelijk CGI (later IIS).

Forta: ‘Het was een extensie om ervoor te zorgen dat je van alles met de webserver kon doen wat je anders niet zou kunnen doen. Tot ColdFusion 5 was het geschreven als extensie voor web servers. Het was native code geschreven in C++ voor Windows, later ook voor Linux en andere platforms, maar ook weer native code voor ieder platform. Het is in feite een page preprocessor, dus je maakt pagina's zoals je webpagina's maakt. Daarbinnen maak je extra scripts, en er is een proces dat daar doorheen loopt en kijkt naar die instructies: haal gegevens uit een database, verstuur een e-mail, maak een pagina design. Het lijkt in feite erg op ASP. Er zijn talen die moeite doen om design patterns en best practices af te dwingen, dat is voor veel gebruikers voordelig. Het heeft echter ook een groot nadeel voor veel gebruikers, omdat het automatisch een belangrijk aantal gebruikers uitsluit van het gebruik van de applicaties. Architectuur-puristen houden er niet van wanneer ik dit zeg, maar de waarheid is: niet alles hoeft een n-tier mvc-applicatie te zijn. Als je doet aan datamining, als je een database

hebt met stored procedures en je wilt n-tiers daarbovenop, *who cares*. Maar er is een evenwicht, sommige dingen moeten meer gestructureerd zijn, andere niet. Wat ColdFusion altijd heeft proberen te doen, is een soort middenweg aan te bieden. Als je een structuur wilt maken dan kan dat, maar het wordt nooit echt aan je opgelegd. Dat heeft tevens een keerzijde. ColdFusion heeft ook een beetje geleden onder het stigma dat het entry level is. En gemakkelijk wordt nogal eens gelijk gesteld aan ‘niet schaalbaar’ of aan ‘programmeurs gebruiken het niet’, nietwaar?’

STIGMA Forta: ‘Dat stigma is met de komst van ColdFusion MX verdwenen. ColdFusion is toen helemaal opnieuw geschreven. Technisch gezien is ColdFusion nu een Java-applicatie die op een J2EE

‘ColdFusion is er alleen tijdens het programmeren’

server naar keuze draait, het wordt gedeployed als iedere ander Java applicatie. Het is daarbij tevens een compiler. Dus als je ColdFusion code schrijft, compileert het die in feite in Java byte code. ColdFusion is er alleen tijdens het programmeren, tijdens runtime is er alleen Java. Dat weerlegt het argument dat ColdFusion niet schaalbaar is volledig: het is net zo



Ben Forta: 'CFML is net zo open als Java'

schaalbaar als Java. Het weerlegt ook het argument dat het proprietary is, want het wordt ondersteund door andere vendors (*bijvoorbeeld Bluedragon, het is een open standaard, red.*) en verder als applicatie. Het is net zo open als Java. Maar het allerbelangrijkste is, dat developers die zeer gestructureerde applicaties willen bouwen, dat met ColdFusion gewoon kunnen doen. Als je objecten wilt bouwen, JSP's, als je JDBC wilt gebruiken, dan kan dat. Je kunt allerlei dingen met elkaar mixen zoals je dat zelf wilt.'

PRODUCTIVITEIT Forta: 'Het internet was de grootste marketing hype in de geschiedenis. Eigenlijk bestaat Internet niet eens. Het bestaat uit websites met een slot op de deur, dus wie kan er naar binnen? Intranets bestaan uit websites. Het echt aantrekkelijke aan internet was niet zozeer de technologie. Wat het belangrijk maakte is dat het mensen de mogelijkheid gaf hun eigen werelden te beheren.

Iedereen kon een computer nemen, er wat software opgooien, een paar links maken en vervolgens kon je in principe erg productief zijn. Je hebt dan waarschijnlijk

Nu we deze push in de richting van architectuur hebben, of het nu J2EE of .NET of wat dan ook is, introduceren we weer complexiteit. Het op één na grootste dilemma van ondernemingen die veel geld in J2EE hebben geïnvesteerd, is hoe ze die investeringen kunnen omzetten in productiviteit. Er zijn veel interessante onderzoeken, surveys van analisten, die proberen na te gaan hoeveel geld organisaties aan J2EE infrastructuur hebben uitgegeven. Een jaar geleden was er eentje - ik geloof dat het Gartner was -, die concludeerde dat de meerderheid van J2EE implementaties in feite worden gebruikt als webserver en niets anders. Dus die enorme infrastructuur, want drie jaar geleden móest iedereen in infrastructuur investeren, wordt niet gebruikt.'

COMPLEXITEIT Forta: 'In iedere organisatie zijn er plaatsen waar structuur nodig is, dus schrijf je bijvoorbeeld een EJB of wat maar nodig is. Waar het mis gaat, is waar de interfaces gebouwd wordt, de UI en de user experience. JSP is er nog niet, maar probeert daar wel te komen. Wij zijn nauw bij JSP betrokken, we dragen in sterke mate bij aan de JSP specificatie. ColdFusion in Java-organisaties zou een goed alternatief voor JSP kunnen zijn, productiever ook. De architect die het mooie plaatje in zijn hoofd heeft waarin alles past, moet zich ColdFusion voorstellen als een verzameling tag-libraries. Dus wanneer je onze ColdFusion mailtag aanroept, die je een mailtje laat genereren in drie regels code, dan roept het dezelfde Java mail-api aan, die je direct zou aanroepen, maar het is een honderdste van de code. Dat is een voorbeeld dat ik Java-ontwikkelaars heb laten zien, en hun mond viel gewoon open. Wil je iets heel simpels doen, zoals een dropdown listbox gebruiken op een form, en data uit de database halen om die box te bevolken, dat is vier regels code in ColdFusion.'

Zou een Java-ontwikkelaar delen van ColdFusion ook zelf kunnen schrijven?

Forta: 'Zeker, alleen is het de vraag, of dat zinvol is. Bedenk maar wat een goed ontwikkelaar per uur kost. De basis ColdFusion versie begint bij 1500 dollar, dus als je een ontwikkelaar 100 dollar per uur betaalt dan heb je voor hetzelfde geld maar 15 uur. Bovendien is de developer versie gratis, wij berekenen alleen de productielicenties. Java developers kunnen JSP's en EJB's schrijven. Je kunt ze ook gewoon aanroepen in ColdFusion. Het is niet ontwikkeld om Java te vervangen, het is ontwikkeld om je een productiviteitsboost te geven voor de dingen waarin Java opmerkelijk upper markt in is. Verplaatst jezelf maar eens in de positie van J2EE vendors: hun grootste angst met betrekking tot .NET is ook niet zozeer dat het krachtiger en schaalbaarder zou zijn, maar dat het gemakkelijker te gebruiken zou zijn.'

'Eigenlijk bestaat Internet niet eens'

niet de meest schaalbare applicatie, als het een soort directory is, hoe schaalbaar moet het dan zijn? Dat is wat internet was: een manier die het iedereen mogelijk maakte, zeer productief te zijn.

VERGELIJKBAAR

Lijken J2EE en .NET in feite ook niet erg op elkaar?

Forta: 'NET en J2EE zijn ontwikkeld om nek aan nek tegen elkaar te concurreren. Ze zijn vaak met elkaar in debat. Wat Microsoft heeft gedaan en waarin ze voorlopen op de J2EE wereld, is hoe succesvol het is in gebruiksgemak en hoe snel je iets draaiend kunt krijgen. De waarheid is, dat er niets vergelijkbaars is in de Java-wereld (*Daar is niet iedereen het mee eens, zie ook het gesprek tussen Floyd Marinescu, Marc Portier, Andy Longshaw en Olav Maassen in het volgend nummer, red.*). Je moet je ook realiseren dat het grootste deel van de ColdFusion gebruikers ons product niet gebruiken als een manier om Java-apps te schrijven. Een kleine groep binnen de ColdFusion-gebruikers doet dat wel. Dat is nieuw, opwindend, en die groep groeit heel erg snel. De meerderheid van ColdFusion gebruikers bouwt reporting applicaties, Internet-applicaties, portal applicaties, ze hebben een grote Oracle database ergens en hebben een snelle interface daarnaar nodig, en dit is de snelste weg om die klus te klaren. Er zijn third party vendors die ColdFusion Markup Language ondersteunen. Dat is prima, talen moeten open zijn. Het JSP-voorstel dat in de richting van ColdFusion Markup Language opschuift, juichen wij heel erg toe. Sterker nog, veel onderdelen van dat voorstel komen van ons want wij zitten in die commissie en wij zitten daar ook erg achterheen.'

Wat gebeurt er in het geval dat JSP zoveel eigenschappen van ColdFusion Markup Language overneemt dat ColdFusion Markup Language geen eigen bestaansrecht meer heeft?

Forta: 'Dan moeten we zien wat het juiste business-model is, misschien moeten we dan extensies gaan schrijven, misschien webservices. Maar mocht het ooit zover komen, dan zien we wel weer verder.'

TOEKOMST Forta: 'JSP gaat de goede kant op, maar is er nog lang niet. JSP doet het niet goed, Java-ontwikkelaars horen dat niet graag maar het is wel zo. Om na te gaan hoe succesvol iets is, kun je eens kijken hoeveel boeken er over verschijnen. Ik houd alle trends in de gaten, over SQL, WAP, Java noem maar op. Op het JSP-front is het al maanden angstwekkend rustig, JSP loopt achter om twee redenen: in de eerste plaats levert het niet genoeg voor hardcore Java developers en ten tweede heb je er nog steeds teveel Java kennis voor nodig voor de niet-Java ontwikkelaars. Het maakt het mogelijk om een if-statement wat gemakkelijker te schrijven, maar je moet nog steeds Java-api's schrijven. Het gaat niet zo goed met JSP, het zal beter worden en het moet ook beter worden anders zal Java eronder lijden. En wanneer JSP meer zoals ColdFusion Markup Language wordt, dat zou ik dat geweldig vinden.'

'Van 1999 tot en met 2001 kende ColdFusion een enorme groei. Alle start-ups boden enorme kansen voor ColdFusion: voor iedereen die in twee weken een internetbedrijf wilde oprichten, was ColdFusion perfect, je kunt het in een week bouwen en je bent op het net. Veel van die bedrijven zijn er niet meer. Toch weet ColdFusion zijn positie binnen die groep te behouden. Momenteel zijn er twee nieuwe markten die we gaan veroveren: de eerste is de groep Dreamweaver-klanten. Dreamweaver heeft twee miljoen gebruikers, van wie tachtig procent statische webpagina's maakt. De meerderheid van Internet is nog steeds statisch, ongelooflijk. Het andere gebied dat belangrijk wordt is J2EE.'

Wat verwacht je voor de toekomst?

Forta: 'Integratie wordt heel erg belangrijk, en tegelijk met het uitbreiden van nieuwe standaarden en protocollen, webservices. Integratie zal zeker de sleutel zijn. Daarnaast moet je goed op de workflow te letten, om te zien welk gedeelte van het ontwikkelproces we wat kunnen veranderen. ColdFusion zal dicht bij de dingen blijven die ons succesvol hebben gemaakt: onze kracht is



Ben Forta: 'Integratie wordt heel erg belangrijk'

om zaken heel erg eenvoudig te maken. In de toekomst zul je van ColdFusion nog meer daarvan gaan zien, evenals meer integratie. Ook onze klanten vragen daarom.'

BUGS EN FEATURES

Is ColdFusion gebruikt om ColdFusion te schrijven?

Forta: 'Deels wel. Vanzelfsprekend moet de kern van ColdFusion Java zijn. Er zijn echt ook gedeeltes van ColdFusion die in ColdFusion source worden geleverd, en die tijdens runtime gecompileerd worden, inderdaad.'

'Het is ook mogelijk je eigen ColdFusion tags te schrijven, dus met de taal kun je de taal uitbreiden. Is dat open? Als je wilt veranderen hoe debugging werkt, hoe performance monitoring werkt, als je onze webserverlaag wilt vervangen door een andere, je kunt alles wat je maar wilt veranderen.'

Is er ook ondersteuning voor UML?

Forta: 'Er is geen UML-support, maar je kunt ieder UML modelleringsstool gebruiken. De huidige versie is 6.1, en dat is een update op MX, de Java versie van een jaar geleden. We hebben een miljoen regels C sourcecode naar Java geport. Dat was een enorme klus en ik was ook erg nerveus daarover. Je moet neerwaartse compatibiliteit scheppen, je moet zelfs bugs opnieuw maken. Er was een aantal bugs in de eerdere versies, die mensen als features gebruikten, en die moesten neerwaarts compatibel zijn met bugs! Er waren echt gebieden die wat problematischer waren: de installatie was complexer bij



garbage collection, dat soort dingen hebben een heel ander niveau nodig. Het veronderstelt een ander abstractieniveau wanneer je een JVM gebruikt. De initiële compile stap is langzamer, maar de runtime compile is sneller. Dat was in de vorige versie een issue, dus in de nieuwe versie hebben we nu een nieuwe compiler opgenomen die sneller is.'

'Hoe kan ColdFusion in Java net zo snel zijn als de C-versie?'

voorbeeld. Maar in de loop van het jaar hebben we veel goede feedback van klanten gekregen. Versie 6.1 was een kans om het strak te trekken en beter te maken.'

TRAAGHEID Forta: 'Zoals ColdFusion last heeft van vooroordelen, zo heeft Java dat ook. Veel mensen denken dat Java langzaam is. Er zijn wel degelijk bepaalde delen van Java die langzaam zijn, maar daar hebben we omheen gewerkt. Uiteindelijk hadden we een veel sneller product dan de C++-versie.'

Wat was er trager?

Forta: 'Het is niet zozeer een kwestie van traagheid, maar meer een radicaal andere manier waarop je zaken aanpakt. memory management, thread management,

COMPILER Forta: 'Mensen vragen me vaak: hoe kan ColdFusion in Java net zo snel zijn als de C-versie? Als je een Java applicatie draait, gebruik je een JVM die bovenop een OS draait, dus er moet wat verlies zijn. Daarin hebben ze gelijk. Maar wat ze vergeten, is dat ColdFusion geen C was, maar een interpreterproces in C. ColdFusion is nu een compiler, het is Java. Je moet dus niet C met Java vergelijken, maar een compiler met een interpreter. Iedere theoretische performances achteruitgang, wordt meer dan goedgemaakt doordat Java een echte compiler heeft. We geloven -nog -niet in Java op de client. Maar Java op de server is heel erg goed. We hebben ook ons nieuwe serverproduct, Breeze, in Java geschreven. Wij geloven echt in Java op de server.'

Tekst en fotografie: Dré de Man