

Linda DeMichiel won de JCP annual award. Ze leidde de JSR 220 EJB 3.0 maar daarvoor al tal van andere JSR's. Java Magazine sprak met haar over de praktijk van het leiden van deze veel besproken JSR, die onder meer 800 pagina's aan geschreven specificaties met zich mee bracht. Een deel van het werk kwam uiteindelijk terecht in de JEE specs, zoals annotaties voor call backs, wat ook veel extra werk betekende, net zoals het nieuwe persistentie-gedeelte.

thema

Hoe JSR 220 tot stand kwam

Interview met Linda DeMichiel

Wanneer je denkt aan EJB 3 en aan de toch wel heel verschillende bedrijven die zich er mee bezig gehouden hebben, dan zou je je kunnen voorstellen dat hier sprake is geweest van verschillende agenda's zoals men dat noemt: conflicterende belangen. Was het moeilijk daarmee om te gaan?

DeMichiel: 'Er zijn moeilijke momenten geweest. In het eerste jaar was er een sfeer waarin veel mensen dachten dat EJB een waardevolle technologie was en die op het gebied van *ease of use* aangepast moest worden. Er was dus een hoop actieve brainstorming en er waren veel creatieve ideeën die geëvalueerd werden. Toen we eenmaal gepubliceerd hadden was er meer druk van buiten en werd het wat conservatiever. Toen we verder gingen met persistence brachten we een aantal JDO-vendors in de expertgroep, wat de dynamiek van het persistence-werk een beetje veranderde. Omdat ze er later bij kwamen, was het de vraag of het wel voor hen zou gaan werken. Ze werden echter gerustgesteld in die zin dat ze op een gegeven moment wisten dat ze echt meetelden. In het begin waren mensen meer op een altruïstische manier aan het werk, en tegen het einde, toen het duidelijk werd dat deze technologie echt wat zou gaan worden, zag je meer defensief gedrag met betrekking tot productbelangen.'

COMPONENTEN *U heeft ook de 2.0 JSR voorgezeten, ik zou me kunnen voorstellen dat het moeite gekost heeft zo veel veranderingen aan te brengen.*

DeMichiel: 'In feite was bij 2.0 van het begin de visie van entity beans als componenten, we maakten ze

gewoon beter, niets radicaals. We voegden meer functionaliteit toe, maar het bleven zware componenten. We maakten ze in ieder geval lokaal bereikbaar, in plaats van die remote-componenten. Dat was in mijn optiek



Linda DeMichiel: 'Deze JSR heeft meer feedback gekregen dan ik ooit gezien heb.'

een grote vergissing in de vroege architecturen. Maar ze hadden nog steeds de smaak van componenten, ze waren geen persistent Pojo's, er waren geen radiale veranderingen. EJB QL was wel iets goeds. Dat heeft zich in de loop van de tijd bewezen als query-taal in het object-domeinmodel, maar wel een query-taal die zich richt op SQL. Maar verder was 2.0 – hoewel het wel werkte – zeker niet de beste manier en entity beans kregen een hoop kritiek.'

Welk andere uitgangspunten waren er behalve de genoemde?

DeMichiel: 'De pattern-boeken waren erg verhelderd in dat opzicht. Ik las ze en dacht: hoe moeten mensen dit gebruiken? Dit is geen patroon, dit is een anti-patroon. Ik las ook een hoop boeken die kritiek op EJB hadden. Je moet weten wat er stuk is om het te kunnen repareren. Ik had daar mijn beeld bij maar ik wilde dat van anderen ook kennen. Dat vormde een deel van de agenda: het ervoor zorgen dat veel belemmeringen zouden verdwijnen. Ik had dus mijn waslijst en anderen in de expertgroep eveneens en zo begon het.'

Bij deze JSR lijkt het wel alsof er veel meer invloed uit de Java-community is geweest, bijna alsof het al open source is.

DeMichiel: 'Ik hield een presentatie op het TSS symposium, maart 2004, en dat was een soort *coming out*-feestje voor EJB 3. Er was nog geen spec, en mensen vielen zowat van hun stoel, vooral door het persistence-deel. Kort daarop op JavaOne gaven we de eerste echte draft vrij. We kregen veel exposure, die we constructief konden gebruiken. Veel mensen gaven presentaties op JUG's. Wij blogden, vroegen om feedback. Ik moet zeggen dat van alle JSR's die ik ken deze veel meer feedback heeft gekregen dan ik ooit gezien heb. Iedere spec draft had een comment mailadres waar mensen commentaar naartoe konden sturen en dat stuurde ik door naar de groep. Er ontstonden op deze wijze ook dialogen.'

INPUT

Dat lijkt er veel op alsof het al open source zou zijn.

DeMichiel: 'Nee, ik zie het zo niet helemaal. We gaven er toch leiding aan als een expertgroep, we hadden alleen het voordeel dat we veel input op deze manier kregen.'

Misschien was het ook wel nodig omdat er vrij veel kritiek op EJB 2 was.

DeMichiel: 'Ja, de API was toen hij geïntroduceerd werd tien jaar geleden echt bedoeld om gemakkelijker enterprise-applicaties te maken. Maar het deed dat op een heel zwaarwichtige manier, de interfaces, de mechanismen, en de wereld is dat ontgroeid. Het programmeermodel is niet acceptabel meer. Wij wilden dat wat

goed was behouden – en er was veel goeds – maar het wel veel gemakkelijker maken om het te gebruiken. Ik hoop dat we daarin geslaagd zijn. Ik denk dat het mogelijk simpeler kan. De expert groep heeft heel goed werk geleverd en heeft de mechanismen geleverd die nodig waren. Natuurlijk is er ruimte voor verbeteringen. We hebben niet zo verschrikkelijk veel nieuwe functionaliteit toegevoegd, omdat het zeker aan de EJB-kant vooral op *ease of use* gericht was.'

Welke punten zullen in de toekomst nog veranderen, vooral in persistence?

DeMichiel: 'Voor *persistence* was dit echt een 1.0 release, dus we wisten dat er nog dingen moesten veranderen in de kernfunctionaliteit. Er zullen mapping types ingebracht moeten worden, mapping strategieën. Een ander compromis was, dat er ruimte in de *persistence* API is gebleven om te ontsnappen naar vendor-specifiek gedrag. Er zullen wat bugs zijn en gaatjes in de specs, maar de core functionaliteit is goed.'

Terwijl u aan het werk was gingen de ontwikkelingen met betrekking tot allerlei frameworks als Spring gewoon door, was dat niet moeilijk?

DeMichiel: 'De IT-wereld is nu eenmaal voortdurend in beweging. Dat zal ook tussen nu en de volgende release gebeuren. Anders blijf je aan het werk en komt het werk nooit af. Er zijn JSR's - ik wil geen namen noemen - waar ik jaren geleden bij betrokken was en die nog steeds in hun early draft-stadium zijn. Je moet je groep in beweging houden. Ook al lijkt drie jaar een lange periode, het was toch een hectische tijd. Voor mij waren het eigenlijk ook twee JSR's: de kern-EJB en *persistence*, met dezelfde expert-group. Dat was een erg ambitieuze doelstelling voor de tweeënehalf jaar dat we daarop focusten.'

GAVIN *Wanneer werd het idee geboren om persistence apart te behandelen?*

DeMichiel: 'Dat was rond de tijd dat JBoss in de expert-groep zitting nam. Marc Fleury was toevallig in de stad, wilde graag langskomen en dus legde ik hem uit wat we deden, en toen zei hij op een bijna kinderlijk enthousiaste manier: 'Ah, dit is geweldig, je moet met Gavin (*King, red.*) praten'. Om eerlijk te zijn was het ook heel handig, want voor het werk in *persistence* was het heel handig dat Gavin en de Hibernate-gemeenschap achter ons stonden. Het bracht een hoop expertise binnen. Oracle zag de richting die we insloegen en sloot zich er ook bij aan. Dat vormde het begin om *persistence* goed van de grond te krijgen.'

Tekst en fotografie Dré de Man