

Java, daar hoort deze column over te gaan. Dat is al moeilijk genoeg. Meestal begint het schrijven van m'n maandelijkse Java column met het nalopen van alle nieuws op Java-gebied. Tijdens het lezen van al die Java-nieuwtjes – nieuwe frameworks, library's, tools – drijven mijn gedachten gewoonlijk af naar de problemen waar ik (jullie ook?) werkelijk mee te maken krijg. De uiteindelijke column gaat meestal meer over mensen en hun werkwijzen dan over Java; meer over programmeer-paradigma's dan over echte Java.

Was dit het?

Helaas, ik kan niet anders. Dit zijn nou eenmaal de gedachtespinselen in mijn hoofd, dit is wat ik belangrijk vind. Dit wil niet zeggen dat ik Java niet belangrijk of leuk vindt. Java is mijn taal. Vraag mij iets te programmeren en ik gebruik Java. Vraag mij een systeem te designen en Java-library's schieten me te binnen. Vraag me een architectuur uit te werken en de standaard Java-architecturen zijn mijn referentie (maar zelden ook mijn eindpunt). Nogmaals; Java is mijn taal.

De laatste tijd is het zoeken naar Java-nieuwtjes een beetje een formaliteit geworden. Je googelt wat, bezoekt wat bekende sites en komt er weer achter dat er weer een nieuwe versie van een dertien-in-een-dozijn library is en dat we er weer een Eclipse plugin bij hebben. Je vraagt je af waarom de bouwers van al dit moois nog de moeite doen. Natuurlijk zijn veel van de betreffende projectjes nog behoorlijk groen, er is nog veel te verbeteren. Maar; er zijn al zo veel library's, frameworks, plugins en tools! Is deze ene tool écht beter? Anders? Nah ...

We hadden fase I in de Java-geschiedenis, de fase van de oorspronkelijke JDK, applets, kleine

applicaties zonder al te veel extra library's. Dat was een tijd met veel energie. Het web kwam net om de hoek dus was er een hoop leuks te bouwen, Java bruiste. De grote omslag naar fase II kwam met J2EE. Java werd volwassen en was toepasbaar voor echte 'enterprise'-systemen. Heuse banken en verzekeraars gingen Java-systemen bouwen. Java werd groot. In eerste instantie werd J2EE toegepast 'as is', zonder al te veel extra's. J2EE werd gebouwd zoals de Petstore aangaf dat je J2EE moest bouwen. Dat bleek niet zo heel eenvoudig. Om allerlei omslachtige onderdelen van J2EE te omzeilen kwamen er steeds meer frameworks. STRUTS kwam in de mode. J2EE werd steeds groter, ingewikkelder en steeds minder beheersbaar. Applicatieservers werden topzwaar. Tijd voor fase III; de vereenvoudiging. In fase III is herontdekt hoe je met relatief eenvoudige middelen de systemen kunt bouwen die je nodig hebt. JBoss, Spring en Hibernate speelden een grote rol hierbij. POJO's en AOP kregen een plek binnen J2EE. Tegelijkertijd kregen ook steeds meer oude bekende ontwikkelmodellen weer een plek binnen Java – zie het

Oracle-ADF. Ziehier, dit was de geschiedenis van Java in één alinea.

Nu die laatste fase van de Java-geschiedenis alweer een tijdje bezig is, begint het nogal stilletjes te worden. Natuurlijk zijn er meer mensen met Java bezig dan ooit tevoren. En toch gebeurt er niks echt belangrijks meer. Zijn we klaar? Is Java 'af'? Is dit het? Was dit het? Mijn angst is dat Java op het punt is gekomen waar we met C++ waren geëindigd, dat het spannende er af is. Dat betekent nog niet dat Java onbelangrijk geworden is. Het betekent alleen dat de lol er een beetje af dreigt te gaan. Dat betekent óók dat er ruimte ontstaat voor nieuwe talen en oplossingen. Wat dat betreft zijn dit misschien wel heel spannende tijden. Misschien gloort er aan de horizon een nieuwe Java. Een nieuwe taal waarmee we de komende vijftien jaar aan de slag gaan? Tot die tijd is er nog genoeg ruimte voor vele nieuwe library's, frameworks, plugins en columns. Al gaan die columns misschien nóg minder over Java.

*Daan Kalmeijer is docent consultant bij
CIBIT-adviseurs | opleiders
(e-mail: daan@cibit.nl).*