

VELDWIJK

Paradigma Stigma

U leest een blad over data(bases). De lezers zijn database professionals. Deze columnist is een volbloed database man. U en ik verschillen vast en zeker over allerlei data(base) zaken van mening, maar in de basis kijken we waarschijnlijk gelijk tegen 'datadingen' aan. U en ik delen zogezegd op ons vakgebied een basaal denkkader – een *paradigma*. Paradigma's veranderen vaak heel moeizaam, vooral in omgevingen waar de werkelijkheid fuzzy en plooibaar is, zoals in de informatica. Het blijkt gemakkelijker om mensen ervan te overtuigen dat de aarde rond is dan van het feit dat de relationele visie op gegevensstructuren beter is dan een hiërarchische visie. Paradigma's veranderen vaak pas wanneer een generatie manager is geworden of gepensioneerd is.

Het bijzondere van de informaticawereld is dat paradigma's lang naast elkaar kunnen bestaan en elkaar nauwelijks beconcurreren. In database-land domineert bijvoorbeeld nog steeds het relationele paradigma en in programmeerland het Object-Oriented paradigma. Van wederzijdse beïnvloeding is nauwelijks sprak. Het lijkt erop dat we in de ICT te maken hebben met een aantal succesvolle en vrij stabiele paradigma's voor even zovele deelgebieden. Verandering vindt plaats maar, net als in andere takken van sport, heel erg langzaam. Dat het in de ICT zo vaak anders lijkt is hype en schijn, maar daarover wil ik het even niet hebben. Wat ik hier met u wil delen is het probleem dat ontstaat wanneer een applicatie zwaar moet leunen op meer dan één paradigma. Met dat probleem hebben mijn collega's en ik een half jaar geworsteld.

Aanleiding is een applicatie die op een uitermate flexibele manier de facturatie moet verzorgen voor een bedrijf dat daarmee zijn brood verdient. Facturatie-functionaliteit zit in allerlei bedrijfsapplicaties en mede daarom hebben veel bedrijven moeite met het binnenkrijgen van geld van klanten die niet meteen betalen. Het flexibele facturatie-systeem dat wij mochten bedenken en bouwen is een zware database-applicatie, maar tegelijk een zware workflow-toepassing. Honderdduizenden brieven (facturen, herinneringen, aanmaningen, afrekeningen, overzichten) en financiële transacties gaan hand in hand met workflow-based verwerking van telefoontjes, betalingen, brieven en deadline-overschrijdingen. Zo'n systeem is alleen goed te bouwen wanneer het workflow paradigma en het database-paradigma op de een of andere manier compatibel worden gemaakt. Het probleem daarbij is dat het workflow-paradigma intrinsiek *record-at-a-time* is terwijl het database-paradigma de wereld bekijkt als verzamelingen records die *set-at-a-time* moeten worden verwerkt. Op allerlei punten in zo'n applicatie ontstaan er dan paradigmaconflicten en daar gaat het standaard verkeerd met

ofwel de performance ofwel de gevraagde workflow-flexibiliteit.

Wij besloten om het probleem te lijf te gaan door een workflow engine te bouwen en die volledig te baseren op het workflow-paradigma en om de database-processen compromisloos relationeel te laten afhandelen. Ik ben 40+ en dus wat minder flexibel, terwijl mijn collega (en DB/M auteur) Van Orden jonger en bovendien briljant is; dus werd hij de workflow-kampioen en mocht ik het database-fort verdedigen. En toen gebeurde er iets fascinerends: Van Orden leek zijn achtergrond als database-expert volledig achter zich te laten en werd binnen twee maanden helemaal 'workflow'. Het gevolg: discussies en hoog oplopende woordenwisselingen tussen meneer Workflow en meneer Database. Beiden deden we geen stap opzij om onze missie hoog te houden. Van Ordens workflow software met allerlei Java/Oracle-interactie die complexe workflow-processen moest ondersteunen, mocht geen invloed hebben op de opzet van zware transacties door Veldwijk c.s. We hebben deze problemen uiteindelijk gekraakt en een systeem neergezet dat intrinsiek trage, flexibele workflow combineert met het vermogen om data in bulk te verwerken. De manier waarop is misschien stof voor een artikel in DB/M (en Business Process Magazine), maar dat is niet eens zo interessant.

Wat mij fascineert is dat mensen die gepokt en gemazeld zijn in een paradigma, kennelijk probleemloos de stap kunnen maken naar een heel ander paradigma. Het lijkt er daarbij werkelijk op dat het 'thuisparadigma' in dat geval helemaal moet worden losgelaten. Als dit zo werkt dan hebben we weer eens een voorbeeld bij de hand van de Wet van Behoud van Ellende ofwel het Paradigma Stigma: oude en starre mensen kunnen niet switchen tussen paradigma's en de jonge en meer flexibele mensen die dat wel kunnen, laten alle mooie dingen van hun eerdere paradigma thuis. Voor de manier waarop paradigma-overschrijdende projecten worden uitgevoerd zijn er ook flinke consequenties. Je moet verschillende geloven op één kussen brengen en ze echt laten communiceren. Ik weet nu dat het moeilijk is maar niet hopeloos. Het wordt pas hopeloos wanneer we niet willen erkennen dat denken in één paradigma in veel gevallen niet voldoet.

En collega Van Orden zal door zijn DB/M schrijverschap vast wel weer terugkomen in het warme database-nest.

René Veldwijk

Dr. R.J. Veldwijk (rene.veldwijk@faapartners.com) is partner bij FAA Partners, een onderdeel van de Ockham Groep.