

**De RAD Race van Software Release Magazine op 29 en 30 maart is voor de tweede keer in successie gewonnen door CrossmarX, met hun eigen Application Engine. Tweede werd een team van Finalist IT dat gebruik maakte van Ruby on Rails. Ook dit team haalde dezelfde plaats als het jaar daarvoor. Teams die gebruik maakten van nieuwe tools en talen leverden verreweg de beste prestaties.**

## Best of both worlds

### CrossmarX en Ruby-teams eerste drie bij RAD Race 2007

**D**e RAD Race zoals Software Release Magazine die nu organiseert is tien jaar geleden in een iets andere vorm gestart in een tijd dat RAD tools erg populair waren. Hij was vooral bedoeld om de prestaties van RAD Tools (gereedschappen voor het snel ontwikkelen van software toepassingen) te beoordelen. Vijf jaar geleden nam Software Release Magazine de fakkel over van het inmiddels ter ziele gegane

tijdschrift CM Corporate en organiseerde de RAD Race voor de eerste keer zelf. De uitslag van die wedstrijd was nogal onthutsend: teams die met oudere RAD tools zoals Clarion en Magic slaagden erin veel beter te scoren dan teams met nieuwe technologie zoals Java. In 2005 kwam daar echter een zekere kentering in toen een Java-team van Xebia goed wist te concurreren met de RAD tools.



Het team van CrossmarX won de race voor het tweede opeenvolgende jaar.

**Tekst en fotografie:**  
Dré de Man



Finalist IT behaalde de tweede plaats

Vorig jaar trad er weer een omslag op, in die zin dat een nieuwe generatie tools meedeed én een team dat gebruik maakte van een relatief onbekende programmeertaal en bijbehorend framework, Ruby on Rails. Beide scoorden beter dan de mededingers. Ook nieuw was dat een .NET-team (Class-A) goed scoorde, zelfs iets beter dan het beste Java-team, en op de derde plaats eindigde.

### Nieuwe generatie

Dit jaar blijken de verschillen tussen de nieuwe en de oudere generaties nog groter geworden te zijn. De nieuwe generatie neemt de eerste zes plaatsen in van het totale aantal van elf, met de Ruby-teams van respectievelijk Finalist IT Group en Holder nek aan nek op de plaatsen twee en drie en CrossmarX met de zelf ontwikkelde en gepatenteerde Application Engine op de eerste plaats.

De verschillen met de resultaten van voorgaande jaren heeft ook van doen met de in de loop van de jaren geëvolueerde opgave. Langzaam maar zeker is er steeds meer nadruk komen te liggen op iets gecompliceerdere business logica en minder op de dataset. De achtergrond daarvan is dat op het gebied van CRUD functionaliteit (CRUD: create, update, delete, standaard database-bewerkingen) de tools niet meer sterk verschillen. De eisen die in de praktijk aan de business logica worden gesteld nemen echter toe, zeker waar het de mogelijkheid tot aanpassen ervan betreft.

### Webwinkel

Om de deelnemers zo veel mogelijk gelijke kansen te geven, werd twee dagen voor de wedstrijd

bekend gemaakt dat de opgave zou bestaan uit het schrijven van een applicatie voor een webwinkel. Op donderdagochtend om 8.00 begon de race en werd duidelijk dat de webwinkel digitale fotocamera's en bijbehorende objectieven verkocht. Tegen de lunch bleken de teams – naar eigen inschatting – tussen de 8 en 14 % van het totale puntenaantal behaald te hebben. De eerste relatief gemakkelijke modules leverden echter niet zo veel punten op. Gemeten aan standaardfunctionaliteit was er dus al een indrukwekkende hoeveelheid werk verricht. In vier uur hadden veel teams al een vijftal invoerschermen geproduceerd voor het personeel, waarmee de drie verschillende producten aangepast konden worden, het personeelsbestand en het klantenbestand veranderd kon worden. De snelste teams hadden ook al werkende cataloguspagina's waarmee producten getoond konden worden aan consumenten. Ongetwijfeld waren nog niet alle integriteitsregels (voorwaarden die aan het invoeren van data werden gesteld) geïmplementeerd en de scores van de teams op dat moment waren voor een deel ook te verklaren uit verschillen in bescheidenheid. Niettemin: het binnen vier uur produceren van een webwinkelapplicatie met een afwijkende vorm en assortiment blijft toch een prestatie.

### Gelijk op

Om 15.00 uur zaten de beste teams al op meer dan 25% van het maximum aantal punten. Concreet betekende dat, dat ook het winkelmandje nu werkte en dat sommigen al begonnen waren aan het moeilijkste gedeelte van de opgave: de *virtual salesman*. Opvallend was dat de beide Ruby-teams nu een vrijwel gelijk aantal punten dachten gescoord te hebben. De teams zouden qua puntenaantal vanaf dat moment tot zelfs na de jurering gelijk op gaan. Ze bleken ook vergeleken met drie uur daarvoor behoorlijk gestegen in het algemeen klassement en stonden vrijwel vooraan.

Het einde van dag één gaf een verwarrend beeld te zien: veel teams hadden het 'klassieke' deel van de applicatie vrijwel af en ook al een groot deel van de nieuwere onderdelen. Het had er alle schijn van dat de teams sneller waren dan die van voorgaande jaren. Anders dan vele voorgaande jaren was het noodzakelijk een aantal aanpassingen en uitbreidingen op de opgave ook werkelijk uit te delen. Verwarrender was dat een team beweerde dat de moeilijkste opgave niet op te lossen zou zijn. Twee andere teams (de latere nummers één en twee) dachten er echter anders over.

### Miljoenen mogelijke combinaties

's Avonds werd moeilijkste gedeelte van de opgave aangepast, voor het geval dat de opgave toch te moeilijk zou zijn. De moeilijkheid zat in een module waarin de *virtual salesman* sets van

**De teams hebben getoond dat ze onder druk een zeer hoge productiviteit kunnen leveren**

objectieven zou uitkiezen die geschikt zouden zijn voor reisfotografie. Deze sets moesten uitgekozen worden op basis van een vijftal criteria en daarna op combinaties van die criteria geordend, volgens dynamisch te veranderen wegingsfactoren. Voor alle criteria waren berekeningen nodig en de moeilijkheid van de opgave bestond er vooral in dat het aantal mogelijke combinaties van objectieven in de miljoenen kon lopen. Tools en teams moesten dus in staat zijn om op een slimme en dus snelle manier selecties en berekeningen te maken. Wanneer dat niet zou lukken zouden ze eerst iedere keer opnieuw voor alle criteria afzonderlijk en voor alle mogelijke combinaties tijdelijke tabellen moeten genereren, waardoor veel te veel tijd verloren zou gaan.

Door het enigszins aanpassen van de opgave werd dit aantal iets verkleind en bovendien werden er twee alternatieve, iets gemakkelijkere opgaven toegevoegd. Uiteindelijk koos geen van de teams voor de (minder punten opleverende) eenvoudigere alternatieven.

Uit de resultaten van de verschillende teams, blijkt vooral dat – anders dan een paar jaar geleden – hoe modernere de tools en talen waarmee de teams ten strijde trekken zijn, hoe beter hun score is. Ruby is weliswaar geen nieuwe taal, maar de combinatie met het framework Rails is wel vrij nieuw. Bovendien is ROR zoals het in de wandeling genoemd wordt, vooral nieuw in die zin dat het de laatste tijd sterk aan populariteit groeit. De ondersteuning door het Java-platform zal daar zeker een belangrijke rol in spelen. Dankzij JRuby is het sinds kort mogelijk om vanuit Ruby-applicaties met Ruby-syntax Java-functionaliteit te gebruiken.

CrossmarX is de meest uitgesproken vertegenwoordiger van de flexibele nieuwe open standaard RAD-tools. De functionaliteit die het tool standaard meebrengt maakt het mogelijk zeer productief te zijn, terwijl de flexibiliteit het oplossen van moeilijke business logica vraagstukken mogelijk maakt. Aangezien het tool standaard Java-code produceert voldoet het aan alle eisen met betrekking tot openheid. Michel Vogler van CrossmarX verklaarde tegenover Software Release Magazine voordat de uitslag bekend was, dat het doel van CrossmarX niet was de snelste te zijn, maar om het meeste gemak te bieden. Waarschijnlijk heeft juist dat gemak CrossmarX geholpen om zeer snel ook de moeilijkste opgave op te lossen.

De op het eerste gezicht zo verschillende nieuwe generatie winnaars blijkt een belangrijke overeenkomst te hebben. Anders dan de oorspronkelijke RAD-tools werken ze met open standaarden en produceren code die binnen het Java en/of .NET-platform volledig ondersteund wordt.

Gecombineerd met de hoge productiviteit levert dat een aantrekkelijk combinatie op: *best of both worlds*. Dat geldt voor CrossmarX en de beide Ruby teams, maar ook voor Outsystems en Servoy. Qua productiviteit geldt het in iets mindere mate voor Genwise en Unisys, maar je zou mogen verwachten dat die bij beide tools nog zal toenemen.

De nieuwste editie van de ook nu weer door Capgemini in hun Accelerated Delivery Center *gehoste* RAD Race laat over de gehele linie een vooruitgang zien. In die zin zou je ook kunnen stellen dat alle deelnemers gewonnen hebben. Ze hebben getoond dat ze onder druk een zeer hoge productiviteit kunnen leveren. Daarmee onderscheiden ze zich in gunstige zin van gangbare IT-projecten.

#### Beste vijf

- 1 CrossmarX met CrossmarX Application Engine (108 %)
- 2 Finalist IT Group met Ruby on Rails (81,3 %)
- 3 Holder met Ruby on Rails (81,1 %)
- 4 OutSystems met OutSystems Platform (79,6 %)
- 5 Servoy met Servoy (75,6 %)



De topdrie: v.l.n.r. de teams van Holder, CrossmarX en Finalist IT