

DIT MISLUKT HOOGSTWAARSCHIJNLIJK: VERVANGEN VAN EEN SYSTEEM MEER DAN 7000 FUNCTIEPUNTEN GROOT



Chris Verhoef,
hoogleraar
Informatica aan de
Vrije Universiteit
in Amsterdam: 'Er
zullen nog heel wat
CIO's ontslagen
worden omdat
ze niet in staat
zijn gebleken de
errorcatastrofe te
vermijden.'

Bij vervanging van bestaande software wordt vaak vergeten om oude functionaliteit mee te nemen. Om die reden voldoen vernieuwde ict-systemen vaak niet. 'Ga daarom niet overhaast te werk bij een pakketselectie of bij nieuwbouw', zegt Chris Verhoef, hoogleraar Informatica aan de Vrije Universiteit in Amsterdam. 'Zeker als het gaat om wat grotere systemen. Dan werkt het principe grote-passen-snel-thuis niet. Zorg ervoor dat je alle oude bedrijfsregels boven tafel krijgt. Dat is mogelijk, hoewel goede methodieken nog wereldwijd onderwerp van onderzoek zijn, onder meer aan de Vrije Universiteit.' Volgens Verhoef is het zinvol om volgens een groei- en snoeimodel te werk te gaan. Om te voorkomen dat je met je nieuwe systeem snel weer in dezelfde problemen raakt, die je nu juist met het vervangen van de oude software wilde vermijden.

door: Cok de Zwart

Het valt Verhoef op dat de meeste organisaties pas aan het vervangen van een systeem gaan denken als een reactie op het disfunctioneren van het systeem of wanneer er zich een andere externe omstandigheid voordoet. 'Eigenlijk wordt er veel te weinig pro-actief of preventief nagedacht over het opzetten van een vervangingscyclus. Het kan verstandig zijn om over vervanging te gaan nadenken op een moment waarop er nog geen problemen zijn', aldus Verhoef die vertelt over zijn betrokkenheid bij de vervanging van ict-systemen bij de Deutsche Bank (noot 1). Deze bank had een mooi en goed functionerend *global transaction and settlement system* ter ondersteuning van de handel op de beurs. Toch besloot het management uit strategische overwegingen dit systeem te vervangen. Waarom? Verhoef: 'De bank had een groei-strategie geformuleerd op basis waarvan overnames zouden worden gedaan in het buitenland. De vraag was op welke manier die nieuwe partijen op het handelssysteem zouden moeten worden aangesloten. Dat zou kunnen door de buitenlandse banken hun eigen systemen te laten behouden en met interfaces aan de systemen van de Deutsche Bank te koppelen. Of door de bestaande

systemen te vervangen door het eigen systeem van de Deutsche Bank. Maar in het laatste geval zouden er in elk land aanpassingen aan de software nodig zijn. Al was het alleen maar om te kunnen voldoen aan de nationale wettelijke eisen. In beide gevallen zouden op den duur de onderhoudskosten buitenproportioneel gaan stijgen.' Dus werd er gekozen voor een oplossing waardoor uiteindelijk minder versie-onderhoud zou moeten worden gepleegd. Besloten werd om een nieuw handelssysteem te bouwen dat – door het instellen van landen-specifieke parameters – overal ter wereld bruikbaar zou zijn. Op die manier kon de ict de groei-strategie adequaat ondersteunen. Verhoef: 'Ik vind dit een goed voorbeeld van een beslissing waaraan geen negatieve aanleiding ten grondslag ligt. Je hoeft immers niet altijd ziek te zijn om beter te worden.'

Spagaat

Het bouwen van een nieuw systeem of het selecteren en implementeren van een nieuw pakket vergt enige tijd. En tijd is een kostbaar goed. Veel ict-managers menen dat het hun aan die tijd ontbreekt. Ze nemen dan veel te snel de stappen van probleemformulering naar

analyse en uiteindelijk de oplossing. Verhoef: 'Velen zitten in een vervelende spagaat. Ik kan het me wel voorstellen. Je komt als nieuwe CIO binnen en je wilt voortvarend te werk gaan om je positie als vertrouwen van de Raad van Bestuur waar te maken. Er wordt wel eens gezegd dat een CIO binnen honderd dagen – net als de president van de Verenigde Staten – zijn positie moet hebben duidelijk gemaakt. Maar je ziet ook dat veel CIO's gemiddeld na twee jaar zijn afgebrand en weer het veld moeten ruimen. Dan blijkt het project waarmee ze wilden scoren in het honderd te zijn gelopen. Of er wordt veel te veel geld aan uitgegeven. Of het project loopt vertraging op. Of het nieuwe systeem voldoet niet aan de verwachtingen en mist belangrijke functionaliteit. De filosofie van grote-stappen-snel-thuis die in eerste instantie veel goodwill kweekt, blijkt dan verkeerd uit te pakken.'

De problemen waarin de CIO verzeild raakt kennen dikwijls een complex aan oorzaken. Verhoef: 'Nieuwbouw lijdt vaak aan wat men de errorcatastrofe noemt. Dat gebeurt als je radicaal breekt met de oude systemen. In die systemen zit allerlei bedrijfslogica verborgen die over decennia is opgebouwd. Een

NIEUWBOW LIJDT VAAK AAN WAT MEN DE ERRORCATASTROFE NOEMT. DAT GEBEURT ALS JE RADICAAL BREEKT MET DE OUDE SYSTEMEN

nieuw ontwerp herhaalt gewoon nogmaals alle fouten die indertijd werden gemaakt. Vroeg of laat loop je dan tegen allerlei uitzonderingen aan: speciale gevallen, u-bocht constructies, najeffecten van obsoleete regelgeving, en meer. Ik herinner me een geval van een verzekeraar. Die besloot een aantal pensioensystemen te vervangen door nieuwbouw. Bij oplevering bleek dat een categorie gepensioneerden nog rechten genoot die waren opgebouwd tijdens de Tweede Wereldoorlog. Die waren bij het ontwerp van het nieuwe systeem niet meegenomen, omdat er domweg niet goed gekeken was naar de bedrijfsregels die in de oude systemen zaten. De vraag die iedere CIO zich zou moeten stellen bij het bouwen van een nieuw systeem is: hoe ver reikt de horizon van de functionaliteit? Maak je daarin een vergissing, dan is de kans levensgroot aanwezig dat je te maken krijgt met de *errorcatastrofe*.

Versleutelde bedrijfslogica

Dat geldt volgens Verhoef niet alleen wanneer je besluit een oud systeem door nieuwbouw te vervangen, maar ook in tal van andere gevallen. 'Neem bijvoorbeeld een organisatie die groeit door overnames en die besluit om op één van de aanwezige systemen te gaan consolideren. Dikwijls is de keuze dan gebaseerd op een aantal overwegingen, zoals: het modernste systeem, het systeem dat draait op het voorkeursplatform, dat met de voorkeurstaal is geprogrammeerd, of dat bepaalde features bezit. Mijn ervaring is dat men meestal kiest voor het meest recente systeem. Dat is beter dan niets, maar leidt uiteindelijk tot diezelfde *errorcatastrofe*. Een systeem wordt snel redundant genoemd, maar is het bijna nooit als je er echt naar kijkt. Dus door het afschrijven van de andere systemen kom je dan

iets later toch weer terecht in de *errorcatastrofe*. Er wordt namelijk nog steeds allerlei versleutelde bedrijfslogica weggegooid die in de oude systemen begraven ligt. Dit effect wordt versterkt als men voor het nieuwste systeem kiest, waar op kleinere schaal ook al functionaliteit vergeten is mee te nemen. Een voorbeeld? Door overnames was een verzekeraar groter geworden en wilde een nieuw systeem voor de levensverzekeringen. Het meest recente systeem werd gekozen, maar verzekerden uit de voormalige DDR hadden heel andere rechten opgebouwd die behouden moesten blijven. Het ging om zoveel uitzonderingen dat het nieuwe systeem waardeloos bleek.'

Revolutionair of evolutionair

De grote vraag is natuurlijk hoe een CIO die *errorcatastrofe* kan ontlopen. 'Hoe kun je nu zeker weten dat je alle functionaliteit en bedrijfsregels uit je oude systeem meeneemt naar je nieuwe systeem? Wel, dat kun je nooit zeker weten. Het enige wat je redelijk zeker weet is, dat vervanging van een bestaand systeem dat de 5000 functiepunten grens te boven gaat, door nieuwbouw of een nieuw standaardpakket, een grote kans heeft te falen en zal leiden tot een stortvloed van fouten. Software die maximaal 100.000 regels code bevat – wat neerkomt op zo'n 1000 functiepunten – die kun je volgens mij wel goed vervangen door iets nieuws. Maar je moet er wel goed bij en over blijven denken. Op basis van de oude documentatie en de functionele specificaties die je van de gebruikers krijgt, kun je een heel eind komen met wat ik noem de revolutionaire aanpak. Passeer je die grens dan wordt de kans dat je in de problemen komt, steeds groter. Kom je in de buurt van de 5000 puntengrens dan is de strategie van volledige vervanging bijzonder

risicovol. Is een systeem meer dan 7000 functiepunten groot, dan is het hoogst onwaarschijnlijk dat het gaat lukken. Er zit dan zoveel functionaliteit in de code versleuteld, dat het je echt niet meer lukt om een systeem volledig te vervangen. Op dat moment zul je een evolutionaire strategie moeten volgen. Ik vergelijk het wel met stedenbouw. Stel dat je niet tevreden bent over Amsterdam als stad. Dan kun je toch ook niet zeggen: we gooien alles maar plat en beginnen een compleet nieuwe stad te bouwen? Dan zul je stukje bij beetje aan de gang moeten. Wanneer je scholen wilt bouwen, dan zul je het aantal kinderen moeten tellen, de geografische verspreiding in kaart moeten brengen, toekomstverkenningen moeten uitvoeren om uiteindelijk te bepalen hoeveel scholen je op welke plekken in de stad moet bouwen. Naar deze analogie zie ik ook het vervangen van grote en complexe ict-systemen. Je zult moeten uitzoeken wat je hebt en wat je wilt vervangen. Dat kost tijd en de meeste CIO's wordt de tijd niet gegund om volgens een evolutionaire strategie te werk te gaan. En dat is de spagaat waarin ze dikwijls terecht komen.'

Federatieve architectuur

Bij de Deutsche Bank heeft men wel voor de evolutionaire strategie gekozen door de bestaande architectuur om te bouwen tot een federatieve architectuur. 'Men is begonnen om op algemeen niveau generieke functionaliteiten vast te stellen. Met andere woorden, eerst is onderzocht welke functionaliteit in het handeldssysteem generiek is en op alle locaties ter wereld is te gebruiken. Vervolgens is men een laag dieper gegaan en is men op regionaal niveau binnen de verschillende *lines of business* de benodigde functionaliteit gaan specificeren. Uiteindelijk is men op het laagste niveau locatie-specifieke functies gaan definiëren. Zo is de bestaande architectuur uiteengegafd en opnieuw in federatief verband opgezet. Op die manier is voorkomen dat specifieke veranderingen in het systeem dat in Oostenrijk wordt gebruikt, problemen veroorzaakt in de software die in Singapore het handelen op de beurs ondersteunt. Overigens is voor het ontrafelen van de bestaande systemen gebruik

gemaakt van zogeheten *logic mining*. Met deze tools is het mogelijk om een aantal verborgen bedrijfsregels te ontdekken', aldus Verhoef die met zijn staf op de Vrije Universiteit veel werk verzet om dergelijke logic mining tools te ontwikkelen en te evalueren door ze los te laten op in productie zijnde systemen.

Groei- en snoeimodel

'Een ict-systeem is natuurlijk niet rigide, maar groeit mee met zijn omgeving, met nieuwe wetgeving, nieuwe wensen en technologische vernieuwing. De kans is dus levensgroot aanwezig dat een nieuw geïmplementeerd systeem op den duur dezelfde nukken en grillen gaat vertonen als het oude dat het heeft vervangen. Om dat te voorkomen is het nuttig om te werk te gaan volgens het zogeheten groei- en snoeimodel. In het groeiseizoen laten we tussen de geconsolideerde code wat zaailingen toe, om te kijken of er wellicht een mooie oogst in het verschiet ligt. In het snoeiseizoen gaan we het onkruid wieden, en cultiveren we vruchtdragend zaaigoed. Dat is dan een consolidatieslag die je regelmatig als aparte activiteit uitvoert. Om een dergelijke situatie te bereiken, moeten systeem, organisatie, en mensen bestendig zijn tegen de getijden van zo generiek mogelijke consolidatie tot aan uiterst specifieke wildgroei. Nu is het zo dat veel organisaties grote delen van hun ict hebben uitbesteed en dat een groei- en snoeimodel niet of nauwelijks in zo'n uitbestedingsrelatie past. Formele afspraken, ingegeven door cost-leadership blokkeren zo'n aanpak, en zullen bij substantiële consolidaties de focus vooral op kostenbeheersing leggen, zodat er geen mogelijkheden ontstaan voor experimenten vanuit de business. Pas als het kostendekken bij zowel probleemeigenaren als van leverancierszijde omslaat in batendenken, zal de weg open zijn voor dit soort modellen. Tot het zover is zullen we nog heel wat CIO's ontslagen zien worden omdat ze niet in staat zijn gebleken de errorcatastrofe te vermijden.' ■

Noot 1: Voor een uitgebreide beschrijving van de Deutsche Bank case:
www.cs.vu.nl/~x/pl/pl.pdf

Over de auteur:
Cok de Zwart is hoofdredacteur van TIEM



Halen uw prestaties de top?

Ondernemen is topsport waarbij u voordurend moet inspelen op de veranderende marktomstandigheden. Een krachtige informatie strategie, heldere processen en effectieve instrumenten leiden tot succes.

TopForce draagt bij aan uw succes door de invoer van Enterprise Portal Services. Uw medewerkers hebben hiermee directe toegang tot relevante diensten en gepersonaliseerde informatie. Deze zijn altijd en overal beschikbaar met het gebruik van slechts één wachtwoord. Als topsporter heeft u immers al genoeg aan het hoofd.

Ga voor het hoogste niveau en laat de TopForce expert u informeren over de ROI van uw Enterprise Portal.

where Business meets IT

www.topforce.com

