

VELDWIJK

e-Migratie (3 en slot)

In de voorgaande twee columns besprak ik twee verschijnselen die ik allebei aanduidde met de term e-migratie: het verbouwen van bedrijfsbrede gegevenshuishoudingen en het grootschalig uitbesteden van systeemontwikkeling en -beheer naar lage-lonenlanden. Ik sluit af met een derde, meer technisch e-migratiebegrip: het migreren (converteren) van gegevens van een oud naar een nieuw systeem. e-Migratie in deze beperkte zin is iets dat me al jaren fascineert, zowel technisch als tactisch. Wat mij vooral steeds weer verbaast is de gewoonte om gegevensmigratie niet te zien als behorend tot systeemontwikkeling, maar exclusief als iets dat te maken heeft met implementatie. Voor systemen waarin gegevens niet lang worden vastgelegd is dat nog enigszins begrijpelijk, maar het gebeurt ook bij systemen die staan of vallen bij het hebben van een goed geheugen. Migratie (of conversie) komt bijna overal ergens achteraan.

Theoretisch zou je zeggen dat conversie op de agenda zou moeten staan bij de start van een ontwikkelproject. Je wilt immers zeker weten dat de gegevensinhoud van een bestaande database waar gewenst kan worden overgetankt naar een nieuwe. Gegevens*mapping* moet dus meestal zo snel als mogelijk gebeuren. Bij maatwerk systeemontwikkeling moet dit op zijn laatst wanneer een database-ontwerp gereed is en bij pakketaanschaf voor de pakketkeuze. Gebeurt dat later dan is de basis gelegd voor een probleem of een ramp, afhankelijk van het belang van de gegevens.

Conversiegerelateerde problemen en rampen (in wording) zie ik op meerdere plekken optreden, zelfs bij organisaties die uit ervaring beter zouden moeten weten. Twee columns terug noemde ik enkele grote verandertrajecten die naar mijn gevoel tot dito administratieve ellende zullen leiden. De belangrijkste grond voor mijn onheilvoorspellingen ligt in conversies die zijn mislukt of gaan mislukken.

Deels is de schade van een mislukte conversie niet goed zichtbaar. Zo hoorde ik van een verzekeraar die bezig is met de invoering van een nieuw systeem. Deadline 1 januari 2006, want dan verandert de wetgeving ingrijpend. Helaas is de gegevensconversie grotendeels mislukt en zal men op zijn best nog moeten doordraaien met delen van het oude systeem. Op zijn slechtst mislukt ook de invoering van het nieuwe systeem. Die schade kun je berekenen. Minder zichtbaar is dat er op grote schaal 'nee' wordt verkocht aan klanten die juist vanwege die nieuwe wetgeving willen veranderen van verzekeraar.

Naar de vraag waarom gegevensconversie zozeer een project-sluitpost is kan ik slechts raden en dat wil ik graag doen.

Ik zie twee oorzaken. Allereerst denk ik dat er bij systeemvernieuwing bijna alleen maar wordt gekeken naar functionele aspecten, waarbij de nadruk ligt op functionaliteit die het bestaande systeem niet heeft. Een dergelijke focus op bestaande functionele knelpunten is vragen om nieuwe gegevensknelpunten, ook al omdat juist op gegevensgebied de duivel in het detail zit. Wat niet helpt is dat het bestaande systeem op gegevensgebied veelal een zwarte doos is en dat de ICT-beheerders vaak geen belangrijke speler zijn in het selectieproces – zeker wanneer ze in India zitten. Naar mijn gevoel speelt dit focusprobleem bij nagenoeg elk uit de hand gelopen of mislukt conversietraject.

De tweede, grotendeels onbekende oorzaak van problemen ligt in de aard van legacy databases. Dergelijke databases zijn vaak kaartenbakken vol met gegevens die niet voldoen aan de documentatie – voorzover aanwezig. Heel vaak gaan ontbrekende of niet actuele documentatie en corrupte gegevensverzamelingen hand in hand. De conversie naar een nieuwe gegevensbank is dan nagenoeg per definitie een traject vol met negatieve verrassingen. Vooral bij conversie naar een pakket levert dit problemen op. Nog acuter worden deze problemen bij gegevensconversie van een kaartenbak naar een echte database door de toepassing van controles op database niveau. Wie bij de conversie tegen een *check rule* of een *referential constraint* aanloopt, heeft een probleem dat niet kan worden genegeerd of weggemoffeld, temeer daar de systeemontwikkelaar of de pakketleverancier meestal niet happig is om constraints te *disablen*.

Verstandig management betekent haast per definitie dat dergelijke risico's zoveel mogelijk worden weggenomen en waar dat onmogelijk is beheerst – wat dus niet de gewoonte is. Die praktijk is des te zorgwekkender omdat corruptie van de bestaande gegevensverzamelingen eerst en vooral een probleem is dat moet worden opgelost door de gebruikers zelf. Als ik als ICT'er constateer dat het veld 'Geslacht' niet alleen de waarden 'M' en 'V' bevat maar ook allerlei andere waarden waarvan de betekenis duister is, dan betekent dit doorgaans vooral werk voor de gebruikers. De ICT'er constateert en bewaakt en de gebruiker lost op. Het slechtste moment om dit te doen is op het moment dat het nieuwe systeem in gebruik wordt genomen. En omdat 'later' vaak niet kan moet het dus eerder. Het kennelijke onvermogen om dit in te zien en er naar te handelen kost naar mijn mening jaarlijks miljarden.

René Veldwijk

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