



visie

Een interview met
professor Chris Verhoef
van de UvA

Software moet je managen

"Veel beslissingen over ICT worden niet of niet optimaal genomen," zegt professor Chris Verhoef. ICT is voor bedrijven cruciaal geworden, het is volgens hem een bedrijfsmiddel dat inmiddels zou moeten worden behandeld zoals andere middelen, bijvoorbeeld het onroerend goed. Maar daarvoor ontbreekt het in sommige bedrijven aan voldoende kennis. Om dit te verhelpen werkt Verhoef aan Software Asset Management, een serieuze en strategische benadering van hoe je met ICT omgaat.

Marieke Vos

Sommige ICT is zo bedrijfskritisch dat het een 'asset' is, een verworvenheid. Het wordt met de jaren alleen maar meer waard, omdat er steeds meer kennis in gecodeerd wordt. De precieze waarde van ICT is echter moeilijk te meten, zo wordt vaak gezegd. Volgens Verhoef is dat niet helemaal waar: het is moeilijk te meten, maar je kunt er wel iets zinnigs over zeggen. "Trek de stekker maar uit het netwerk. Wat kost het dan als het een uur of een dag down is? Dat is een indicatie van hoeveel het waard is. Je kan hetzelfde doen met een apart systeem. Blijkt dat je hierdoor helemaal geen omzet misloopt, dan is het systeem of netwerk kennelijk geen bedrijfskritische asset. Blijkt echter dat het miljoenen kost, dan moet je ervoor zorgen dat uitval niet gebeurt, òf dat je kan uitwijken. En als je het hebt uitbesteed, dan moet er in het contract met die partij een boeteclausule zijn opgenomen die de kosten van de uitval dekt."

Niet alleen een kwestie van communicatie

Chris Verhoef is hoogleraar informatiesystemen aan de Vrije Universiteit in Amsterdam. Daarnaast werkt hij voor het Software Engineering Institute (SEI) in Pittsburgh en als wetenschappelijk adviseur voor de Deutsche Bank in Manhattan. Zijn diverse werkzaamheden voor wetenschap, semi-overheid (SEI) en bedrijfsleven geven hem een bijzonder inzicht in de problemen die mensen hebben met de vele software in bedrijven. "Veel beslissingen over ICT worden niet of niet optimaal opgenomen."

Hij constateert dat niet alle bedrijven een totaaloverzicht hebben van de software die zij gebruiken. "Op het niveau van de Raad van Bestuur weet niet iedereen welke systemen voor tachtig procent van de geldstromen zorgen. Ze weten niet wat hun software waard is, wat er aan ICT in het bedrijf aanwezig is en hoe bedrijfskritisch het is. Maar ze nemen soms wel beslissingen op grond van deze gebrekkige kennis. Je ziet soms dat politieke argumenten dan de overhand krijgen en dan bedoel ik dat in negatieve zin. De afwegingen zijn niet rationeel." Hoe dat komt, daarop is niet zo gemakkelijk antwoord te geven. "Als het een lijstje was met problemen, dan hadden we dat allang opgelost. Maar het is een complex van problemen. Een bron van deze problemen kan zijn dat ICT en de Raad van Bestuur niet altijd goed met elkaar communiceren. ICT-ers spreken de taal van de techniek en weten dat vaak niet te vertalen in juridische, financiële of risicoargumenten. Topmanagers kunnen strategische argumenten weer vaak moeilijk vertalen in ICT-taal. Maar ook al konden ze dat wel, dan nog was het probleem niet opgelost. Als je weet wat een baksteen is, kan je een metselaar niet automatisch uitleggen hoe het gebouw eruit moet zien."

Software is geen vrachtwagen

Het echte probleem ligt elders en dat heeft met name te maken met de kennis over ICT. Raden van Bestuur denken teveel in termen van software als waren het vrachtwagens, legt Verhoef uit. "Software wordt beter naarmate er meer aan wordt gewerkt. Elke dag weer wordt er kennis in gecodeerd en dat maakt dat software steeds waardevoller wordt. Dat is kennelijk heel moeilijk te begrijpen voor mensen die ICT zien als een middel dat je kunt vervangen als ware het een vrachtwagen. Je kan gecodeerde kennis niet gemakkelijk uit een oud systeem halen en omzetten in een nieuwe. Want de menselijke kennis, die in feite is neergeslagen in de soft-

ware, is niet meer allemaal paraat. En omdat niet altijd goed is gedocumenteerd welke aanpassingen er gedurende de jaren in systemen zijn gemaakt, is het lastig deze kennis helemaal over te nemen.

Daarbij komt nog eens het financiële aspect. Afbreukrisico's van projecten voor grote, nieuwe systemen zijn namelijk heel hoog. Bijvoorbeeld: negentig procent van alle ERP-implementaties faalt! Eenderde van alle software projecten wordt afgebroken, de helft is twee keer te laat, twee keer te duur en levert de helft van de functionaliteit, en slechts twintig procent van de softwareontwikkeling is volgens planning. In harde dollars: er wordt jaarlijks alleen al in de USA 81 miljard dollar weggegooid aan die afgebroken projecten. In Nederland is die situatie niet anders."

Een fault injection rate van honderd procent

Een nieuwe vrachtwagen rijdt, maar nieuwe software beslist niet. Je moet er, net zoals bij de oude systemen, jarenlang aan sleutelen voordat het precies doet wat jij wilt. Als de Raad van Bestuur klaagt dat de oude systemen niet meer voldoen, dan ligt dat niet altijd aan die systemen maar soms ook aan het feit dat de zorg voor die systemen vaak niet goed is gemanaged. Er is bijvoorbeeld niet bijgehouden wat er aan is gesleuteld, want projecten moeten meestal snel klaar zijn en dan is er geen tijd voor goede documentatie en voor kwaliteitscontroles." Bedrijven hebben dus software die te waardevol en te ingewikkeld is om zomaar te vervangen. Het topmanagement begrijpt dat niet altijd en daardoor worden wel eens niet optimale ICT-beslissingen genomen. Het leidt ertoe dat sommige bedrijven in de problemen raken. "Ze komen in een complexity catastrofe of een error catastrofe", zegt Verhoef. De complexity catastrofe betekent dat software zo complex is geworden dat je er weinig in kan veranderen zonder dat er iets fout gaat. "Dan heeft het systeem

een hoge 'fault injection rate': er zijn sommige systemen met een FIR van honderd procent. Wat je ook doet, er gaat altijd elders iets fout. Bedrijven die hier genoeg van hebben en beslissen oude software door nieuwe te vervangen, maar die niet alle belangrijke kennis uit de oude overnemen, raken verzeild in de error catastrofe: alles wat fout kan gaan, gaat ook fout. Want doordat je het verleden uitbant en de goede dingen uit het verleden ook vergeet, maak je dezelfde fouten weer opnieuw en nu allemaal tegelijk. "Die bedrijven blijven uiteindelijk zitten met een mislukt project."

Software Asset Management

Het is dus noodzakelijk dat de top op een professionele manier leert omgaan met ICT-beslissingen. Hoe dat moet is nogal ingewikkeld, Chris Verhoef werkt aan Software Asset Management om dat te kunnen ondersteunen. Het is een manier om de ICT-middelen, met name software, te managen zoals ook andere bedrijfsmiddelen worden beheerd. "Je laat als bedrijf het managen van je onroerend goed toch ook niet over aan de portier? Dit is een belangrijke verantwoordelijkheid van de Raad van Bestuur." Software Asset Management (SAM) onderkent vijf belangrijke aspecten. Ten eerste is dat de politiek-strategische: hoe gebruik ik mijn ICT-budget zo effectief mogelijk voor de business, nu en in de toekomst? Het is een taak van de Raad van Bestuur om daar een visie over te hebben. Het tweede is het juridische aspect. "Bij outsourcen moet je heel duidelijke contracten afsluiten. En bij de bouw van software moeten de specificaties helder zijn. Daar zitten juridische aspecten aan, waar specifieke kennis voor nodig is." Het derde aspect heeft met het management, de organisatie van ICT, te maken. "Stel je hebt honderd miljoen regels sourcecode in zestig landen draaien. Hoe organiseer je dat? Hoe zorg je er bijvoorbeeld voor dat er een nieuwe versie wordt geïnstalleerd?" Het vierde aspect is een financiële/economische: dat een bedrijf zijn ICT-beslissingen kan rechtvaardigen op grond van economische redenen. "ICT moet bijdragen aan het bedrijfsresultaat en dat moet je kunnen onderbouwen."

Het laatste aspect is de techniek. Als je de bovenstaande vier aspecten hebt ondergebracht in je organisatie, dan moet de software nog wel worden ontwikkeld en onderhouden. "Als het gaat om bedrijfskritische systemen, dan kan je niet kiezen voor een complete nieuwbouw om de redenen die ik eerder heb genoemd. Het zal je niet zonder meer lukken die gecodeerde kennis uit het oude systeem volledig te reproduceren." Hoe het dan wel moet, daarover heeft Verhoef wel ideeën. Maar eerst vertelt hij nog dat Software Asset Management een benadering is die nog niet zoveel bedrijven gebruiken. "Het vereist vergaande kennis van je ICT en die is er in de meeste bedrijven niet." SAM is ook geen kant en klare methodiek, die je uit één boek kunt leren. "Er staat toch ook niet in één boekje uitgelegd hoe je onroerend goed beheert?" Om SAM goed in te vullen, heb je soms gespecialiseerde medewerkers nodig. Zoals bij het beheer van onroerend goed ook aannemers, makelaars, schilders en notarissen betrokken zijn. Die uitsplitsing van functies bestaat nog

niet in de ICT. Terwijl Verhoef zich heel goed kan voorstellen dat dat ooit wel gaat gebeuren. "Je zou als bedrijf bijvoorbeeld een ICT-accountant kunnen hebben. Stel dat je een externe partij software laat bouwen. Die accountant moet met de specificaties in de hand de source code van de software kunnen nalopen en constateren of alle gevraagde functionaliteit erin zit."

Centrale verwarming in een kathedraal

Als je als bedrijf software wilt vervangen, dan kan dat op twee manieren. Je kan een nieuw systeem bouwen. "Om dit goed te kunnen doen, moet je soms precies begrijpen wat er in de code van het oude systeem gebeurt. En omdat men negatieve ervaringen heeft met het oude systeem, wordt vaak alles op de schroothoop gegooid. Dat is heel risicovol." Ik pleit dan ook liever voor de tweede manier, de evolutionaire manier. Dit kost ook veel tijd, maar is door juist management haalbaarder dan de eerste manier. Vergelijk het hiermee: als je een kathedraal uit 1346 warmer wilt maken, dan kan je hem helemaal afbreken en steen voor steen opnieuw opbouwen, voorzien van nieuwe techniek. Wedden dat die kerk er niet zo uitziet als de oude? De evolutionaire manier is anders: dan bouw je er centrale verwarming in, zonder hem eerst af te breken."

Het is bepaald niet gemakkelijk om software assets aan te passen, weet ook Verhoef. Welk bedrijf heeft niet zulke systemen, die met houtje-touwtje methodes aan elkaar hangen en waarvan het beheer nogal problematisch is? Toch zal je deze systemen zo moeten aanpassen dat ze nog jaren kunnen draaien, denkt Verhoef. Gewoon omdat het afbreukrisico van een project dat een nieuw systeem gaat bouwen te groot is. "Dan praat je wel over grote, bedrijfskritische systemen. Voor eenvoudige systemen geldt dit niet."

Modificatie is dus voor de vernieuwing van grote systemen meestal de minst slechte methode. Met de technische invulling daarvan, het vijfde aspect van SAM, houdt Verhoef zich ook bezig. "Deze modificatie van software kan je vrijwel niet met de hand doen. Maar je kan er wel hulpmiddelen, technieken en tools, voor ontwikkelen. Hiermee kan je een set van problemen oplossen. Je analyseert bepaalde delen van de software en zet die vervolgens in de steigers. Je probeert de software minder complex te maken, door die functionaliteit te houden die je nodig hebt en de mogelijkheden in te bouwen om functionaliteit die je niet meer nodig hebt te veranderen. Met de hulpmiddelen die je hiervoor ontwikkelt, kan je proberen wat er gebeurt als je een deel van de functionaliteit verandert. Zo pak je het stukje bij beetje aan."

Hoe dit precies werkt en welke hulpmiddelen je daarvoor nodig hebt, daar houdt Verhoef zich mee bezig. Het is een essentieel onderdeel van SAM, een nieuw vakgebied dat Verhoef 'uit de grond probeert te trekken'. "Het is mijn bedoeling de wetenschappelijke kant hiervan de komende jaren sterk te ontwikkelen." Voor geïnteresseerden: handboeken hierover zijn er niet. "Als mensen hier meer over willen weten, dan rest ze nu nog niets anders dan mij gewoon eens te bellen."

marieke.vos@ngn.nl