

Nieuwe generaties technieken bepalen grote prijsverschillen

Wie een automatiseringsprobleem heeft, wordt voor de oplossing ervan geconfronteerd met totaal verschillende prijzen. Het ene softwarebedrijf brengt zonder blikken of blozen vijf keer zoveel in rekening als een ander. Hoe is dit te verklaren? Het prijsverschil, concluderen Theo en Hans Mulder, wordt vooral bepaald door het selectief toepassen van nieuwe generaties methoden en technieken.

Hoe is het mogelijk dat sommige leveranciers nog steeds een factor drie tot vijf goedkoper zijn dan hun concurrenten met eenzelfde automatiseringsoplossing? Deze prijsverschillen gelden zowel voor het verrichten van vooronderzoek als de realisatie en de infrastructuur van informatiesystemen. Is hier sprake van een verkeerde marktwerking? Of ligt de oorzaak in een inferieure kwaliteit van de voorgestelde oplossing? In werkelijkheid zijn de prijsverschillen te verklaren uit het selectief toepassen van nieuwe generaties methoden en technieken.

Theo Mulder en Hans Mulder

Twee jaar geleden werden in Automatisering Gids (26 januari 1996) zeven praktijkgevallen beschreven waaruit onder meer kon worden geconcludeerd, dat kleine innovatieve PC-softwarebedrijven zich ontwikkelen als grote concurrenten van gevestigde softwarehuizen. Prijsverschillen van een factor vijf tussen PC- en minicomputeroplossingen bleken in de praktijk geen uitzondering. Deze verschillen bestaan nog steeds. Steeds vaker zijn het nu echter de PC-oplossingen en projecten op basis van evolutionaire ontwikkelingsmethoden die het nakijken hebben. Een paar voorbeelden:

1. In 1997 worden voor een offertetraject vijf hardwareleveranciers uitgenodigd. Vier stellen als minimumeisen voor de werkplek: Windows 3.11, een 133Mhz pentium-processor en een huurlijn voor een landelijk netwerk. De vijfde leverancier stelt eigenlijk geen specifieke eisen aan de besturingssystemen, processor of middleware. Elke bestaande werkplek met een Internet-browser en een ISDN-verbinding volstaat. De prijs van deze offerte is een factor drie lager.

2. De databaselicenties voor een landelijke applicatie (van het ministerie van Justitie) met 170 werkplekken variëren van 600.000 tot 45.000 gulden. Dit prijsverschil is des te opmerkelijker omdat het de databases betrof van dezelfde leverancier. Dit prijsverschil werd veroorzaakt door een andere prijsstrategie en wijze van afrekenen van licenties.

In de 'dure client/server-licenties' zijn de gebruikers en de werkplek met naam bekend en wordt voor 170 gebruikers afgerekend. In de 'goedkope Internet-licentie' kan niet per gelijktijdige gebruiker worden afgerekend, immers op het Internet zijn er geen gelijktijdige gebruikers, maar slechts 'hits'. Een goede licentieafdracht op basis van het aantal 'hits', ook wel micro-transacties genoemd, hoort nog niet tot de afrekenmogelijkheden van databaseleveranciers. Derhalve gold in dit praktijkvoorbeeld voor Internet-licenties een andere prijs, namelijk een die een factor dertien lager lag.

3. Binnen automatiseringsprojecten zijn extreme verschillen tussen offertes in doorlooptijd voor de definitiestudie geen uitzondering. Sommige leveranciers bieden een maanden durend vooronderzoek aan terwijl andere leveranciers ervan uitgaan dat de specificaties door de (deskundige) eindgebruikers vanzelf worden aangedragen tijdens prototypesessies. Het blijft voor de opdrachtgever een uitdaging het prijsverschil op waarde te schatten.

Ervaringscijfers en vuistregels voor het schatten van automatiseringsuitgaven raken snel achterhaald door de introductie van nieuwe technologieën. Er bestaat daarom behoefte aan een overzicht dat inzicht geeft in het ontstaan van de prijsverschillen tussen automatiseringsoplossingen. Een automatiseringsoplossing is een combinatie van de infrastructuur (computers, randapparaten, netwerken, generieke software), ontwikkelingsgereedschap (programmeertalen, case- en testgereedschappen), de toepassing (de presentatie-, applicatiesoftware met gegevensverzamelingen, maar ook de bijbehorende handmatige procedures) en de methodische werkwijze van automatiseren (zie tabel).

Productiever

In de automatiseringsbranche worden overwegend drie generaties infrastructures aangeboden. De eerste is de centrale, grote en dure computer: het mainframe. De tweede is ontstaan in de jaren zeventig door de introductie van de minicomputer zoals de PDP-serie van Digital. De mini was kleiner en goedkoper dan het mainframe. De minicomputer betekende een doorbraak voor middelgrote ondernemingen en departementale automatisering.

In het midden van de jaren tachtig vond de introductie plaats van de PC. Hoewel de infrastructuur in eerste instantie gericht was op de werkplek, kon de PC in het begin van de jaren negentig met behulp van lokale netwerken en krachtige PC-servers de rol van de minicomputer overnemen. Het prijsverschil van tienduizenden gulden tussen een mini en PC's werd hierdoor duidelijk zichtbaar.

Toch is het goed op te merken dat het succes van een bepaalde infrastructuur snel kan worden overdreven. Zo is bijvoorbeeld 'downsizing' van zware mainframetoepassingen naar een client/server-omgeving met PC's (nog) niet haalbaar gebleken. Na vervanging van een mainframe of minicomputer blijken de beheerkosten per werkplek vaak dramatisch toe te nemen.

Vaak melden softwarebureaus dat zij een vierde-generatie-ontwikkelingsgereedschap gebruiken, om daarmee de mate van productiviteit en flexibiliteit te benadrukken. De eerste generatie - machinecode - staat het dichtst bij de computer, terwijl elke volgende generatie programmeertalen meer begrijpelijk wordt voor de mens. Er is echter nooit meer een tijd gekomen waarin de infrastructuur zo vergaand werd benut als in die beginjaren.

Tot de tweede generatie ontwikkelingsgereedschappen worden bijvoorbeeld Autocode en (later) RPG gerekend. De voordelen van de tweede generatie waren de symbolische naamgevingen en het efficiënter programmeren van algoritmen per regel code. Deze efficiency leverde veel rendement op, gezien de dure uren van programmeurs. De aanhoudende behoefte aan meer en vooral ook goedkopere informatiesystemen deed de roep ontstaan naar eenvoudigere en productievere manieren van programmeren.

De derde generatie, waaronder Cobol, introduceerde allerlei gestructureerde faciliteiten voor bestandsorganisaties. Vanaf de jaren zeventig zijn in Cobol-achtige programmeertalen allerlei toepassingen ontwikkeld voor specifieke functies, zoals de verkoop-, inkoop-, debiteuren-, crediteuren- en financiële administratie. Een onvoorzien gevolg van deze gescheiden bestandsorganisaties was eilandautomatisering, die resulteerde in onnodige gegevensopslag en handelingen. In die tijd ontstond de vraag naar een groot geïntegreerd managementinformatiesysteem (Mis), ook 'total system' genoemd.

De vierde generatie ontstond in de jaren tachtig en combineerde het schrijven van regels code met het grafisch programmeren van schermlayouts en een relationele database. Deze database beheert onder meer de relaties tussen de gegevens en biedt SQL goede mogelijkheden voor de bevraging van de gegevens.

Een succesvolle toepassing van deze vierde generatie zijn de bedrijfsbrede informatiesystemen, bekend onder de naam Enterprise Resource Planning (ERP). Deze bedrijfsbrede systemen betekenden feitelijk een realisatie van de 'total system'-gedachte, een integratie van de voorheen intern gescheiden systemen voor verkoop, inkoop, logistiek en financiën.

De voordelen van deze integratie zijn onder meer het voorkomen van dubbele administraties en een betere afstemming tussen de diverse bedrijfsfuncties. Een nadeel is dat organisaties niet meer alleen te maken hebben met het eigen informatiesysteem.

Een ander nadeel is de inflexibiliteit om een groot bedrijfsbreed systeem aan te passen aan de steeds veranderende bedrijfsprocessen van een organisatie.

Generaties

In de praktijk spreekt men ook over generaties toepassingen. De eerste generatie zijn berekeningen van salarissen van ambtenaren, het registreren van basisbestanden, het verwerken van relatief simpele maar grote hoeveelheden transacties. De tweede generatie - eveneens gericht op de back office - ontstaat bij de introductie van de minicomputer. Dat zijn toepassingen waarvoor min of meer standaardpakketten worden geschreven en daardoor ook economisch verantwoord kunnen worden aangeschaft door het MKB.

De derde generatie toepassingen zijn gericht op de front office. Voorbeelden zijn reserveringssystemen, point of sale-toepassingen, telefonische inlichtingen et cetera. Deze toepassingen worden gekoppeld of geïntegreerd met back office-systemen. In de laatste fase van deze generatie zien we de ontwikkeling van bedrijfsbrede ERP-systemen.

Het onderscheiden van generaties bij infrastructuur en gereedschappen, kan ook worden toegepast bij de werkwijzen van automatisering. De eerste generatie is de zogenoemde levenscyclus. Een groot voordeel van deze werkwijze was dat de beheersbaarheid verbeterde door het project op te delen in fasen en mijlpalen. Tevens fixeerde de levenscyclus het aantal en de volgorde van de activiteiten en vormde daardoor een goede checklist voor de

(tussen)resultaten.

Een vooral in Nederland en Engeland bekende levenscyclusmethode is SDM. De nadelen van dergelijke waterval-methoden (te langzaam en inflexibel) worden de laatste jaren breed uitgemeten. Maar we moeten niet onderschatten hoeveel projecten zonder een dergelijke methode nog meer uit de rails zouden zijn gelopen.

De tweede generatie methoden (meer gestructureerd en abstract) ontstond in de jaren zeventig toen de automatisering zich verder verspreidde in het bedrijfsleven en de toepassingen zich, in plaats van op specifieke registraties en calculaties, meer gingen richten op de directe ondersteuning van de verschillende bedrijfsfuncties. Deze periode valt grotendeels samen met de introductie van de minicomputer en de derde-generatietalen.

Een gevolg van een meer bedrijfsgerichte automatisering was dat de rol van de automatiseerder verschuift van expert naar analist. De automatisering van de verkoop- of inkoopafdeling was immers niet zo voorgestructureerd als een annuïteitenberekening en bovendien was de automatiseerder vaak onbekend met die specifieke bedrijfsvoering. In deze periode werden net zoals eerder het geval was allerlei JBF (Jan Boerenfluitjes)-methoden gebruikt voor het analyseren en ontwerpen van systemen.

De gevolgen van deze particuliere methoden waren soms niet te overzien. Een gebrekkig begrip van de bedrijfsvoering leidde regelmatig tot technisch correcte, maar functioneel onbruikbare software. Het was daarom noodzakelijk dat de automatiseerder eerst een gestructureerd vooronderzoek deed om de juiste specificaties boven water te krijgen voor het te bouwen systeem. Het zijn de computerdeskundigen Edward Yourdon, De Marco en Chen geweest die structuur hebben aangebracht in het analyseren en ontwerpen van informatiesystemen.

Onder meer door de fysieke vormgeving van de logische (informatie-inhoud) te onderscheiden en de gegevensstromen en -verzamelingen grafisch weer te geven, werd het mogelijk de modellen te controleren op juistheid, volledigheid en consistentie. Helaas bleken deze onderzoeken in de praktijk zeer arbeidsintensief en kostbaar en resulteerden zij vaak in uitgebreide en onoverzichtelijke modellen en rapportages voor de opdrachtgever en gebruikers.

De derde generatie werkwijzen ontstond in reactie op deze problemen. In de jaren negentig werden nieuwe, snellere en iteratieve werkwijzen voorgesteld, zoals prototyping en Rapid Application Development van James Martin. In het begin waren deze evolutionaire methoden gericht op geavanceerde vierde-generatieontwikkelingsgereedschappen, zodat de voorlopige versies van het systeem snel konden worden teruggekoppeld naar de gebruikers. In een nieuwe versie werd het systeem flexibel aangepast aan het commentaar uit deze projectsessies.

De voordelen ten opzichte van de tweede generatie werkwijzen zijn de toegenomen snelheid van ontwikkelen en een grotere betrokkenheid van de organisatie. Door de inbreng van de (deskundige) eindgebruiker verschuift de rol van de automatiseerder van analist naar facilitator. De automatiseerder volgt daarbij vaak een 'U vraagt wij draaien'-aanpak, wat kan leiden tot twee uitkomsten.

De eerste is het 'eeuwigdurende project', waarin de projectgroep in staat is steeds nieuwe functionaliteiten toe te voegen aan het systeem. De andere projectuitkomst is die waarin de klant, na verloop van tijd, ontevreden raakt over het resultaat en de automatiseerder alle verantwoordelijkheid van de hand wijst, omdat de eindgebruiker per slot van rekening het systeem heeft gekregen waarom hij had gevraagd. Het gevaar van de evolutionaire generatie is dat het ontwikkelen van steeds meer (een overdosis aan) functionaliteit als oplossing wordt gezien, terwijl het feitelijk symptoombestrijding is voor het ontbreken van precieze specificaties voor de informatisering van de bedrijfsprocessen.

Bovendien zijn de gevolgen van het nieuwe informatiesysteem voor de bedrijfsprocessen met prototypingtechnieken vaak niet te overzien voor de eindgebruikers en facilitator. Een en ander resulteert dan ook regelmatig in het opnieuw automatiseren van de bestaande (handmatige) systemen, waardoor grote rendementsvoordelen achterwege blijven. Het gevaar van het ontbreken van precieze bedrijfsproces-specificaties heeft overigens ook betrekking op veel ERP-projecten, die de nadruk leggen op een overdosis aan gestandaardiseerde functionaliteit als blauwdruk voor de bedrijfsprocessen, terwijl bij de implementatie vaak pas blijkt welke gevolgen dit heeft voor de bedrijfsvoering.

Concluderend kan het volgende worden gezegd (zie tabel).

1. Ook een methodische werkwijze kan worden ingehaald door een productievare, dus snellere en goedkopere methode.
2. Generatieverschillen leiden tot problemen in het schatten van de productiviteit van de systeemontwikkeling en dus ook van de automatiseringsuitgaven. Dit plaatst met name vraagtekens bij het gebruik van ervaringscijfers in onder meer benchmarking en functiepunanalyse.
3. De keuze voor een methodische werkwijze wordt in grote mate bepaald door de te bouwen

toepassing. Raketbesturingssoftware kan men beter niet bouwen met een evolutionaire methode, dergelijke software moet gestructureerd worden ontworpen.

4. Goede eigenschappen worden van generatie op generatie doorgegeven. Een voorbeeld is de toepassing van de levenscyclusgedachte voor projectfasering in latere generaties.

5. Zolang de informatie- en communicatietechnologie zich blijven ontwikkelen, zullen nieuwe automatiseringswerkwijzen blijven ontstaan.

Componeren

Tegenwoordig zien we in selectietrajecten nieuwe generaties infrastructuren, ontwikkelingsgereedschappen, toepassingen en werkwijzen die sterk in prijs verschillen met de gangbare oplossingen.

In de oude infrastructuren is de computer het uitgangspunt. In de nieuwe vierde generatie is het netwerk de computer. De infrastructuur verbindt interne en externe functionaliteit en gedistribueerde gegevensverzamelingen en is toegankelijk via een protocol. Gereedschappen richten zich meer op het componeren van bedrijfsproces-specificaties in plaats van het programmeren van gebruikersspecificaties.

Innovatieve softwareleveranciers zullen zich in de toekomst niet alleen meer onderscheiden door een veelvoud van functies aan te bieden in een groot systeem. Zij zullen vooral faciliteiten leveren die deze bedrijfsprocessen kunnen aansturen. We spreken dan inmiddels over de vijfde generatie gereedschappen die bedrijfsproces- en componentgericht is. Bij het nieuwe type (vierde generatie) toepassingen gaat het erom de acties en informatie-uitwisseling tussen verschillende en wisselende partijen te coördineren. Met andere woorden de toepassingen zijn niet meer alleen functie-, maar vooral actiegericht. De moderne naam hiervoor is workflow management. De nadruk komt in plaats van op registratieve en calculerende informatiesystemen te liggen op actiegerichte communicatiesystemen.

De nieuwe generatie methodische werkwijzen stelt een goede communicatie in de organisatie centraal en helpt de opdrachtgever en automatiseerder zich te richten op de organisatorische aspecten van het informatiesysteem. De opdrachtgever bouwt samen met een automatiseerder, nu in de rol van therapeut, eerst een model van de bedrijfsprocessen.

De prijsverschillen tussen automatiseringsoplossingen worden grotendeels verklaard door het selectief toepassen van nieuwe generaties methoden en technieken. Nieuwe technieken bieden altijd oplossingen voor enkele grote problemen van de voorgaande generatie. De PC's raken hun dominante positie kwijt aan de snelle ontwikkeling van in- en externe netwerken, hetgeen uiteindelijk zal leiden tot lagere kosten voor de werkplek.

Het ligt voor de hand dat de methodische werkwijzen zich meer en meer zullen richten op communicatie, met alle mogelijke voordelen voor nieuwe automatiseringsoplossingen. Hoe snel deze voordelen zullen worden benut, hangt mede af van de bereidheid van opdrachtgevers te willen leren van de generatieconflicten van de laatste vijftig jaar. Elke generatie technieken zal immers worden opgevolgd door een nieuwe generatie en leiden tot een conflict.

Prof. Th.J. Mulder en drs ing. J.B.F. Mulder zijn beiden werkzaam bij de Viagroep NV in Rijswijk, een groep samenwerkende IT-bedrijven op het gebied van organisatieadvies, systeemontwikkeling en infrastructuren.

Periode	1955-1965	1965-1975
	1975-1985	1985-1995
	1995-2005 ?	
Aanpak	-	Levenscyclus Evolutionair
	Gestructureerd	
	Communicatief	
	particuliere	(gefaseerd)
	(logisch/ (participatie/	(bedrijfsproces)
	methoden mijlpalen)	fysiek) prototyping)
Voorbeeld	JBF-methode	SDM Yourdons
	James Demo	
	Structured	Martins
	Method	RAD
Rol	Pionier	Analist Facilitator
	Expert	

Therapeut
Toepassingen
Back-office
Front-office
(calculaties)
(standaardmodulair)
(Orgware + WFM)

Gereedschap Machinecode
3 GL (Cobol)
5GL(Componenten)

Infrastructuur
decentraal/lokaal
(mainframe)
(mini-computer)

Prijs *ffff.....ffffff*
 ff f

Doorlooptijd jaren
 maanden maanden/weken

Experimenten
Back-office
Actiegestuurd
(maatwerk/massa)
(geïntegreerd ERP)

2 GL (RPG)
4 GL + RDMS

centraal mainframe
gedistribueerd
host/terminals
(PC, C/S) (Internet)
ffffff ffff

jaren/maanden
 weken