

Gert Florijn over het hergebruik van software

STRUIKKELEND HARDLOPEN

Ontwikkelaars zijn vanaf hun eerste 'hello world'-applicatie steeds opnieuw bezig het wiel uit te vinden. Aan de andere kant wordt eveneens vanaf het begin hergebruik van software en applicaties nagestreefd. Als één techniek de belofte van hergebruik inhoudt, dan is het wel SOA.

Door Dré de Man

Volgens Gert Florijn, Director of Consulting bij DNV, zijn daar echter heel duidelijke voorwaarden aan verbonden. Achter de technische mogelijkheden gaan moeilijkheden schuil, die er altijd al geweest zijn.

Florijn: "De facto doen we al heel lang aan hergebruik. Programmeertalen zijn veel meer geconvergeerd en de raamwerken binnen de talen zijn het volgende stapje convergentie. Binnen applicaties is er dus al heel veel hergebruik, maar je ziet dat er ook hergebruik is van commerciële pakketten. In de wereld van productleveranciers zie je vooral een strategische benadering van hergebruik waarbij je niet een product gaat bouwen, maar een familie van producten. Je begint met een analyse om te onderzoeken welke dingen gelijk zijn en welke variabel."

Binnen de wereld van administratieve applicaties blijkt hergebruik wat minder eenvoudig te zijn.

"Waar we vroeger dachten dat technische voorzieningen noodzakelijk waren om tot hergebruik te komen, is dat nu het punt niet meer. Bij complexe toepassingen is met name het besturingsaspect, het eigenaarsaspect, veel dominantier. Je

kunt uitrekenen wat hergebruik oplevert, dat zijn heel overtuigende voordelen. Maar een projectmanager die gestimuleerd wordt om zijn software net wat generieker en dus beter algemeen toepasbaar te maken zegt: 'Ja, hoezo? Het is mijn budget, ik ga het niet voor iemand anders doen.' Wanneer de besturing en het eigendom niet centraal geregeld worden met de juiste machtsfactoren erbij, is het een heel lastig verhaal."

Softwarehergebruik is waarschijnlijk iets dat voor de business nog te abstract is, tenzij er een CIO is die het allemaal heel goed kan uitleggen.

"Ja, exact, dan kan het. Er moet een evident voordeel zijn om dingen op een bepaalde manier te gaan hergebruiken. Het moet ook op de juiste schaal zijn. Het is makkelijker te begrijpen als je het hebt over 'we gaan binnen onze organisatie maar een keer opslaan' met alle evidente normale voordelen die je daarvan hebt. Dat begrijpen mensen wel. Op het moment dat je bepaalde berekeningscomponenten wilt standaardiseren wordt het al een stuk lastiger."

Maar je zou toch mogen verwachten dat wanneer je een applicatie netjes opbouwt en dingen hergebruikt die hun nut al eer-



Foto: Dré de Man

Gert Florijn: "De afhankelijkheden bestaan wel degelijk nog, alleen niet meer in de techniek maar veel meer in het besturingsmodel."

der bewezen hebben, de functionaliteit er eerder is.

"Dat zou je denken, maar het probleem is dat je heel veel afhankelijkheden moet managen. Je moet diensten gaan gebruiken van anderen. In de meeste organisaties is het onmogelijk om dat allemaal centraal te beleggen. Dan moet je allemaal meeschakelen om dat nieuwe ding dat van al die andere diensten gebruik maakt werkelijk operationeel te krijgen. Op het moment dat je vele raakvlakken hebt, moet je gaan praten over tijdstippen wanneer je het gaat uitrollen. Vaak moet het dan net weer iets anders voor degene die de dienst afneemt. Dat moet qua timing ook weer geregeld worden. Dus waar we denken dat er geen afhankelijkheden zijn, bestaan die wel degelijk nog. Alleen niet meer in de techniek, maar veel meer in het besturingsmodel."

Als ik dat zo hoor dan denk ik dat dit het nog veel moeilijker maakt om op een agile manier met requirements om te gaan. Tegen de tijd dat je het uitonderhandeld hebt met je collega's zijn de requirements alweer veranderd en kun je opnieuw beginnen.

"Dat is zeker een heel groot risico. Het governance-vraagstuk is massaal onderschat. Het is veel te veel gegaan over het maken van dingen en niet over het beheer, het eigendom en hoe we dat aansturen in een organisatie. Als je dat professioneel geregeld hebt, kun je ook snel dingen deployen. Heb je dat niet, dan loop je de kans – en dat zie je ook bij vrij veel SOA-programma's - dat men dat gaandeweg gaat ontdekken. Dat wordt een beetje struikelend hardlopen. Dan steekt er weer zo'n takje uit de grond: 'o, daar hadden we niet aan gedacht'. Die hele beheers-, exploitatie- en eigendomsvraagstukken zijn cruciaal om het effectief te maken. Echt goed SOA doen vereist een zeer volwassen IT-organisatie. Terwijl de suggestie initieel was: 'we klikken het zo bij elkaar'. Dat is dus niet zo. Ja, je kunt het wel bij elkaar klikken, maar daarmee heb je het nog niet operationeel."

Dus technisch is het een stuk makkelijker geworden, maar organisatorisch is het nog net zo moeilijk.

"We komen nu bij vragen als: 'hoe doen ze dat dan, wie vertelt wie wat, wat is eigenlijk de informatie die we uitwisselen, wat is de betekenis daarvan?' We gaan het hebben over dingen als canonical datamodels, we moeten met elkaar gaan afspreken wat de betekenis van dingen is. Dat is ook nog niet makkelijk, want je moet dat doen met een bestaand systeemlandschap. Niemand kan opnieuw beginnen, iedere serieuze IT-organisatie sleurt de ballast van vijf, tot zelfs twintig jaar, met zich mee. Een voorbeeld: bij een telecomprovider hadden ze meerdere systemen waar klantgegevens in staan. Dan ga je kijken wat de betekenis van 'klant' in dat systeem is. Dat blijkt totaal uiteenlopend. Het kan betekenen: 'we hebben een offerte gestuurd, iemand heeft ooit een contract gehad', 'hij heeft tenminste één product in actief gebruik', 'hij heeft een informatieaanvraag gedaan'. Dat moet je dus over zo'n landschap heen zien te managen."

"Dat is des te belangrijker, omdat de fouten van een andere aard zijn dan technische fouten. Stel dat ik een systeem aanpas dat een service aanbiedt, dan heeft dat effect op de betekenis van de gegevens die dat ding aan de buitenwereld levert. Als degene die dat afneemt niet dezelfde interpretatie heeft, gaat het alsnog fout. Alleen op een subtielere manier, op een minder zichtbare manier dan bij technisch falen."

Als de betekenis subtiel verandert, kan dat pas op het moment dat je gaat auditen naar boven komen.

"Daarom ben ik ook blij dat de afgelopen jaren de discussie is gegaan van het bouwen van zo'n SOA-landschap naar vragen als: 'wat betekent traceerbaarheid?' Bij cursussen over architectuur geef ik wel eens de vraag: 'ik zie een scherm en op dat scherm staat: aantal kinderen 2. Dat gaat over mij. Ik heb drie kinderen. Waar is dat fout gegaan?' Dan zeggen mensen altijd: 'dat kun je vinden door goede logging', maar het is



Gartner SOA & Application Development and Integration Summit 2009

SOA Health Check: In Tough Times Reconsider Everything, Evolve, Enable Change

24 – 25 June 2009 | Royal Lancaster Hotel, London, UK

The “perfect storm” hitting the worldwide economy forces organizations to reconsider everything in their IT strategy. Including their approach to service-oriented architecture.

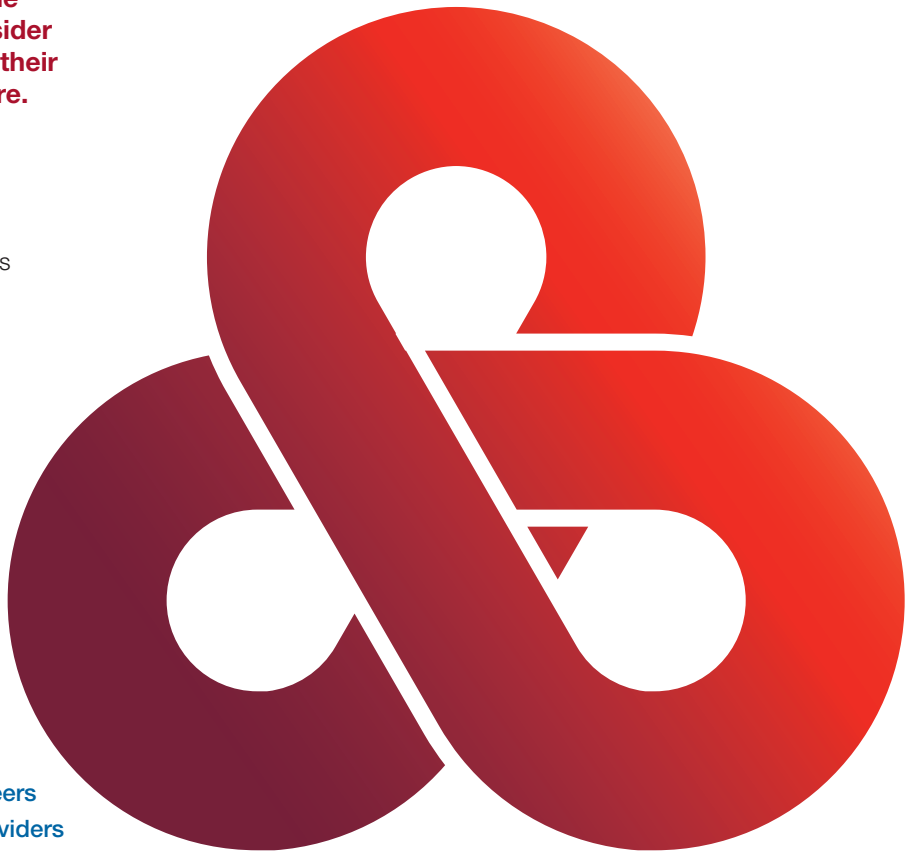
IT departments are asking themselves:

- What can SOA do for me right now?
- Is there real value in SOA or was it just hype?
- Can SOA help me get through the mayhem or is it a luxury I cannot afford anymore?

The SOA & Application Development and Integration Summit answers these critical questions by discussing how SOA and related technologies can help organizations efficiently support the challenges ahead, but also take advantage of the opportunities manifested in times of profound changes.

A completely new program:

- **Guest Keynote**
- **Gartner Keynotes**
- **Pioneering case studies**
- **Workshops**
- **In-depth pre-event Tutorials**
- **Top-tier networking with over 300 of your peers**
- **Meet innovative technology and service providers at the forefront of the Applications space**
- **An exclusive opportunity to speak one-on-one with a Gartner Analyst**



View the full Agenda online: europe.gartner.com/soa



Register using priority code MP-SOA3 and save €500 on the standard delegate rate!

europe.gartner.com/soa

Tel: +44 (0)208 879 2430

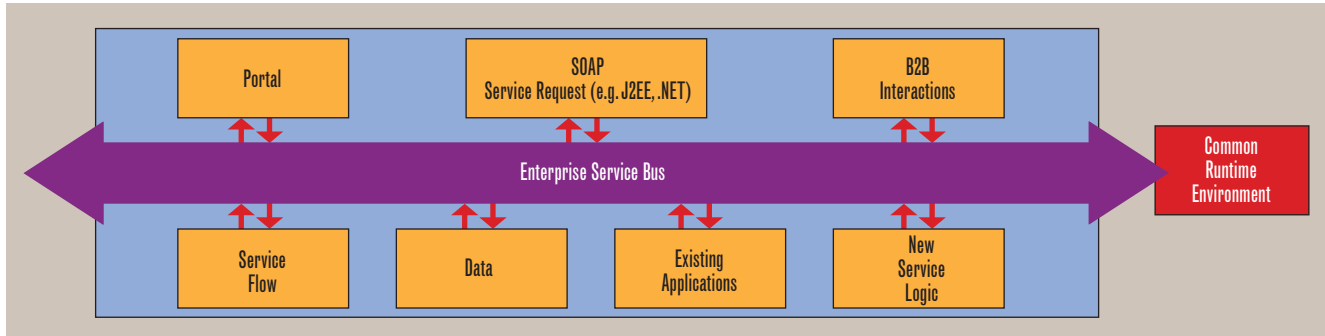
Email: emea.registration@gartner.com

Gartner

SOA & Application
Development and
Integration
Summit 2009

24 – 25 June | London

europe.gartner.com/soa



Het SOA-ideaalplaatje waar Florijn op doelt.

meer. Het gaat om dat totaalplaatje waar dit getal dan van daan komt, welke dingen het allemaal meegemaakt heeft op zijn reis van de bron naar hier. Dat inzicht moet je hebben, vanuit beheersperspectief. Dat soort vraagstukken over zo'n landschap als geheel dat zijn interessante dingen. Daar werd tot nu toe in de SOA-discussie heel weinig aan gedaan."

Gartner zegt ook over hergebruik van applicaties dat het een 'mess waiting to happen' is. Dat klinkt niet geruststellend.
 "Nee dat is zeker zo. (Toont SOA-ideaalplaatje.) Het punt is dat op het moment dat er hier iets uitkomt, het in werkelijkheid allemaal uit verschillende systemen komt. Om dat te traceren en om daar de kwaliteit van te beoordelen, maar ook om de kwaliteit van dienstverlening te kunnen bepalen, heb je heel veel inzicht nodig. Je moet heel veel expliciet maken. Over gegevens in die datamodels, maar ook volume, beschikbaarheid, capaciteit en dat moet je ook allemaal managen."

Maar daar heb je toch ook software zoals service repositories voor?

"Ja, je moet beginnen met het beschrijven van welke diensten je überhaupt hebt en je moet vervolgens de kwaliteit van die dienst gaan beschrijven. Maar de gedachte dat we alles beschreven hebben en dat dit soort dingen niet optreden, is een ontkenning van het feit dat er zich subtielere dingen afspelen tussen die werelden. Beheerders hebben ook een eenheid van werken nodig. Die moeten backups maken, nieuwe versies installeren, recovery doen. Dat is overigens een interessante vraag, hoe je backup en recovery doet in een service oriented landschap. Stel het klantsysteem is gecrasht en het orderstelsel niet. Wat moet ik nu restoren, zodat er weer een goed beeld uitkomt? Wat als nu de helft van mijn systemen crasht en de ander helft niet? Wat is nu een consistent beeld? Dat zijn heel relevante beheer- en governance-vraagstukken. Dat was in het begin van SOA geen onderwerp van gesprek. SOA was zeker in het begin een hype in de ontwikkelwereld, nieuw is beter, dat soort beloften."

Maar, gaat het nog goedkomen? We willen ook nog communiceren via webservices met andere bedrijven en om het nog moeilijker te maken hebben we ook cloud computing.

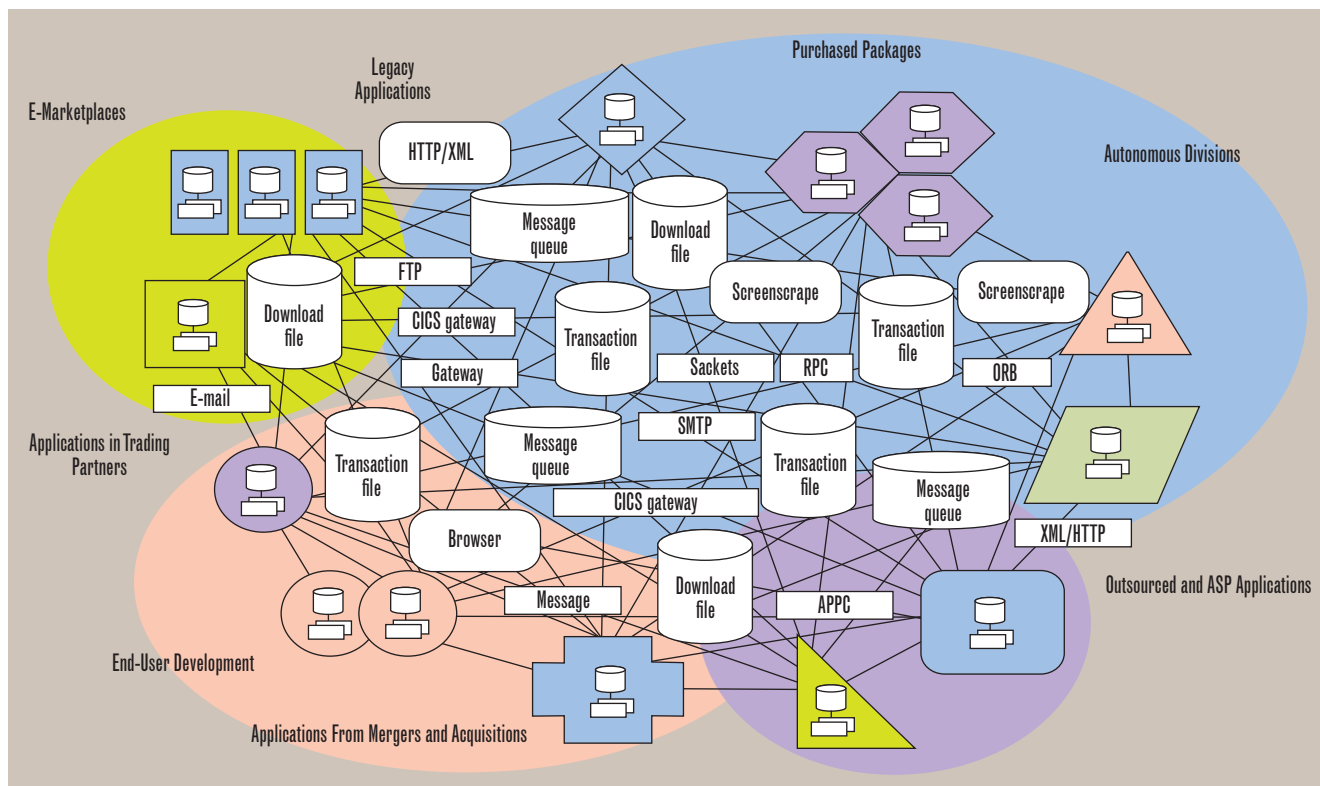
"Een van de sleutelbegrippen in dit hele verhaal is natuurlijk ontkoppeling. Je kunt ontkoppelen en vervolgens dingen maken door ontkoppelde dingen te integreren of te combineren. Die integrale conceptie van kwaliteit wordt te gemakkelijk onder tafel geschoven. Met frameworks maken we steeds grotere applicaties waarbij die code op steeds meer plekken zit. Overall zit software, maar als ik nu hier trek, waar valt het dan om? Bij SOA zie ik hetzelfde, alleen op een groter niveau."

Bovendien zitten er soms nog heel oude applicaties achter die services. Misschien is het wel twintig jaar oude software die met spuug en elastiek aan elkaar zit.

"Over een systeem dat twintig jaar meegaat, kun je een interessante discussie voeren. Als het om softwarehergebruik gaat, ben ik een groot fan van het koesteren van legacy. Die systemen hebben bewezen dat ze meekunnen. Grotere informatiesystemen, die kosten sowieso al meerder jaren om te bouwen. Iedereen die die dingen al eens van dichtbij gezien heeft, weet dat het daarna nog zeker twee jaar duurt, voordat alle eigenaardigheden eruit zijn. Als je die uit een bestaand ding kunt laten komen, waar we al door dat hele proces heen gegaan zijn, is dat de moeite waard. Dus koester je legacy."

Niemand kan opnieuw beginnen, iedere serieuze IT-organisatie sleurt de ballast van vijf tot twintig jaar met zich mee

Dat is meer dan alleen wrappen achter een interface en er vervolgens niet meer naar kijken. Wat moet ik dan doen om dat ding in die context goed te laten functioneren. Veel systemen zijn te complex om elke vijf jaar te vervangen en toch is er van buiten af de druk om te zeggen: 'okay, het moet nieuw en het moet anders'. Ik zou graag wat meer balans zien in de discussie over wat we nu hebben, wat daar de voordelen van



Een grote verzameling van applicaties met de verbindingen tussen de applicaties.

zijn voor het het servicegedachtegoed, hoe het past, en hoe we daar überhaupt komen. Want dat is niet in een week of zelfs een jaar te doen. Dan komen we terug bij het begin. Het vergt ook heel veel organisatorisch commitment, want een programma van drie jaar, dat is voor elke organisatie zwaar. Dan zijn we ook weer terug bij governance, een op ietwat abstracter niveau gevoerde discussie over wat nu eigenlijk het probleem is dat we proberen op te lossen. Ik zie te vaak dat het alleen maar gaat om welke nieuwe dingen we gaan doen. (Flo-rijn toont een plaatje met een grote verzameling aan applicaties met een wirwar aan verbindingen tussen de applicaties.) Dit wordt vaak als het probleem aangeprezen. Dit soort plaatjes zijn natuurlijk bij uitstek geschikt om mensen het gevoel aan te praten dat hier iets niet goed is. Als ik dit plaatje op een cursus laat zien, dan roepen alle cursisten: 'oh, wat is dit erg'. Maar wat is hier het probleem? Het feit dat hier veel lijntjes zitten? So what! Als ik die lijntjes onder controle heb, is er niets aan de hand. Maar het argument dat ik ze onder controle heb, wanneer ik al die lijntjes aan een Enterprise Service Bus hang, dat is kul. Ze zijn er nog steeds, maar in een wat andere vorm. En er moet nog steeds iemand beseffen dat als de data die hier uitkomt niet daar bij dit ding terechtkomt, dat er dan een majeur probleem is."

Maar betekent dat nu dat als je eenmaal het overzicht hebt, in veel gevallen blijkt dat je misschien helemaal geen SOA nodig hebt?

"Dat is inderdaad soms de vraag. Veel organisaties beginnen met BPM en dan blijkt dat er in dat proces eigenlijk geen variatie zit. Waarom doen we dan BPM? Maar een misschien wel fundamentele vraag is of een organisatie de volwassenheid kan opbrengen om die dingen ook daadwerkelijk te realiseren."

Stel dat die vraag nu van bovenaf komt en ik ben verantwoordelijk voor een bepaalde afdeling, dan wordt dat een heel moeilijke vraag.

"Ik houd van lastige vragen. Veel mensen gaan die uit de weg. Als je een politiek verdeelde organisatie hebt en je wilt dan die centrale governance voor SOA helemaal gaan opzetten, dan is dat erg risicovol. Er zijn heel goede voorbeelden van SOA-benaderingen waarin het zich uitbetaalt. Maar het is geen Haarlemmer olie. Het wordt een vanzelfsprekendheid en dat suggereert dat het de enige stijl is waarmee we een probleem kunnen oplossen en dat is aperte onzin. Er wordt teveel vanuit de oplossing gedacht en te weinig vanuit het probleem. Hergebruik vereist volwassenheid in de organisatie en dus een volwassen governance-model. En de voorwaarde voor goed governance is inzicht in waar we nu staan en waar we naartoe willen. Dat hoeft niet per se SOA zijn. Dat is een vraag die je dan pas kunt beantwoorden. Ik zie het nu te vaak omgedraaid worden. We weten de oplossing al. Maar als het uiteindelijk niet werkt, heb je alleen maar verliezers."

Dré de Man is freelance journalist.