

# Service Oriented Architecture voor interne beheersing

Bedrijfsprocessen overschrijden steeds vaker de grenzen van de organisatie, bijvoorbeeld in het geval van processen met toeleveringsbedrijven. Dergelijke processen vereisen een correcte en tijdige uitwisseling van informatie, vaak ondersteund door informatiesystemen.

Jeffrey Kramer, Assurance  
Zouhair Taheri, Assurance  
Wim Hutten, Assurance

## 1. Inleiding

Omdat bedrijfsprocessen steeds vaker de grenzen van de organisatie overschrijden, is het vermogen van informatiesystemen om onderling te communiceren de laatste jaren een veelbesproken onderwerp. Een belangrijk aspect hierbij is de rol van softwarearchitecturen. Een van die architecturen is de Service Oriented Architecture (SOA), een softwarearchitectuur die organisaties helpt met de problematiek van het ontwerpen van gedistribueerde informatiesystemen in complexe en vaak ook verouderde (geautomatiseerde) omgevingen en de aandachtspunten vanuit interne beheersing.

## 2. Service Oriented Architecture: de ins en outs

Met de softwarearchitectuur SOA kunnen organisaties een informatiesysteem ontwerpen op basis van koppelbare, gedistribueerde 'services'. Binnen de context van SOA wordt met het concept 'service' verwezen naar een bouwsteen die een bepaalde functionaliteit aanbiedt ter ondersteuning van een (bedrijfs)proces. Het principe achter

services binnen SOA is dat ze eenvoudig en snel te combineren zijn tot een nieuwe service.

Een service bestaat altijd uit twee onderdelen: een welomschreven 'interface' en een achterliggende realisatie. De interface is het koppelpunt van de service. Hierin staat beschreven welke functionaliteit de service aanbiedt en hoe de service aangeroepen kan worden. De realisatie verzorgt de interne werking van een service, en is in principe verborgen voor de gebruiker van de service.

### Interface

Een service kan (een deel van) de functionaliteit van een bedrijfsproces aanbieden. Bijvoorbeeld het opvragen van actuele vluchtinformatie. De interface van een dergelijke service vereist als input 'vluchtnummer', en geeft als output de status van de betreffende vlucht. De gebruiker van de service komt tijdens het koppelen alleen in aanraking met de input en output, en niet met de achterliggende realisatie. In werkelijkheid communiceert de realisatie met een complexe IT-omgeving, bestaande uit verschillende mainframes. Dit blijft in een SOA-omgeving verborgen voor de gebruiker van een service.

Omdat services gekoppeld worden via hun interface, is er geen sprake van afhankelijkheid op het niveau van de realisatie. Dit wordt ook wel een 'zwakke afhankelijkheid' genoemd. Als systemen op basis van hun realisatie gekoppeld worden, is er sprake van een 'sterke afhankelijkheid'. Een veelgebruikte analogie om dit te illustreren is een elektrisch stopcontact. Wie met elektrische apparatuur naar het buitenland gaat, dient rekening te houden met mogelijke verschillende stekkers en voltages: de interne werking van het elektriciteitsnet. Er is in dat geval een sterke afhankelijkheid tussen het beschikbare stopcontact en het functioneren van de apparatuur. Het doel van SOA is het verbergen van deze complexiteit door de interne werking zodanig te abstraheren dat een servicegebruiker elk beschikbaar 'stopcontact' kan gebruiken, zonder enige kennis van de interne werking te hebben. De interface van een service speelt daarbij een belangrijke rol.

De interface van een service beschrijft welke functionaliteit de service aanbiedt, en hoe de service aangesproken dient te worden. De aangeboden functionaliteit kan door eindgebruikers of andere services gebruikt worden om informatie te verkrijgen, verwerken of op te slaan. De functionaliteit die een service kan aanbieden, is zeer omvangrijk. Op het hoogste niveau wordt er onderscheid gemaakt tussen simpele en complexe services. Simpele services bieden op verzoek ruwe informatie aan, bijvoorbeeld aandelenkoersen of de voorraad van een product in een ERP-systeem. Complexe services hebben de eigenschap dat ze simpele services (de bouwstenen) kunnen combineren tot een compleet bedrijfsproces dat van begin tot eind deels of volledig geautomatiseerd verloopt. SOA zou bijvoorbeeld ingezet kunnen worden om een inkoopproces te automatiseren. Een service die op basis van actuele marktprijzen de meest gunstige leverancier bepaalt, kan gebruikt worden om een dergelijk proces efficiënt te laten verlopen. Dankzij de welomschreven interface en de zwakke afhankelijkheid kan de service daarnaast opnieuw gebruikt worden in een ander proces.

### Rolverdeling

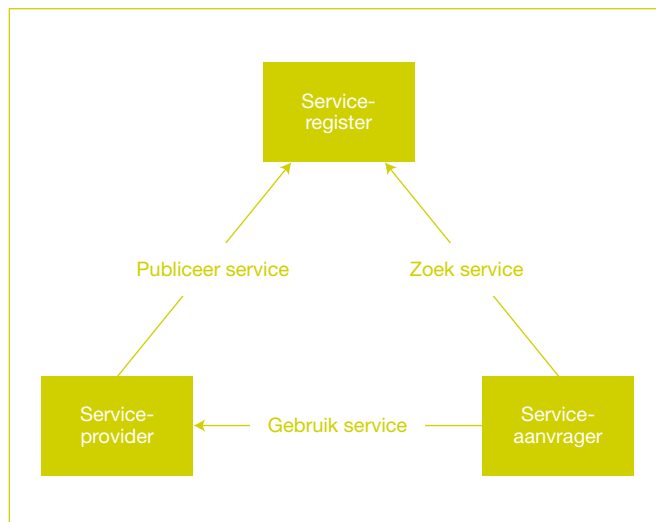
Binnen SOA is er sprake van een roloverdeling (figuur 1). De serviceprovider (aanbodzijde) stelt services beschikbaar aan serviceaanvragers (vraagzijde). Dit gebeurt echter niet direct, maar via een zogenoemde serviceregister, een soort Gouden Gids waar services in geregistreerd staan. Serviceaanvragers gebruiken het register om services te zoeken op basis van functionele en niet-functionele eisen. Bij functionele eisen kunt u denken aan een bepaalde

### Samenvatting

Service Oriented Architecture (SOA) is een softwarearchitectuur voor het ontwerpen van een informatiesysteem op basis van koppelbare, gedistribueerde 'services'. In dit artikel wordt nader ingegaan op wat SOA is, hoe het werkt, wat de voor- en nadelen zijn, hoe SOA in de praktijk werkt en wat de impact is op de interne beheersing. Ook wordt ingegaan op webservices, een platform- en programmeertaal die SOA realiseert op een concreter niveau, en op het COMPAS-project, waarmee nieuwe modellen, talen, en een raamwerk opgezet worden om compliance-eisen te modelleren, specificeren, valideren, en om compliance tijdens de uitvoering van een (geautomatiseerd) bedrijfsproces af te dwingen en monitoren.

functionaliteit die de service aanbiedt. Niet-functionele eisen hebben betrekking op aspecten als beschikbaarheid, performance en kosten van de service.

De gedachte achter SOA is niet gebonden aan een specifieke invulling, technologie of applicatie. Het is daardoor een zeer krachtig, doch abstract concept. De SOA-roloverdeling is hier deels een voorbeeld van: de rol van serviceaanvragers kan door zowel informatiesystemen als mensen worden vervuld.



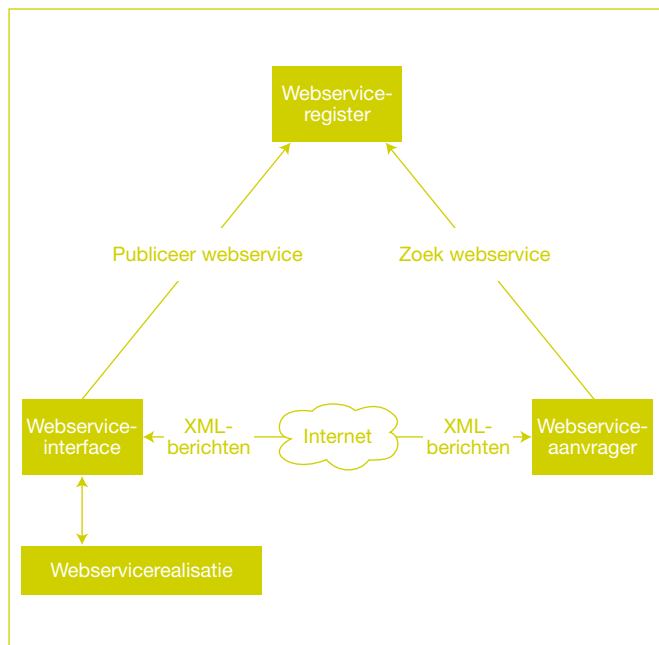
Figuur 1 – SOA-roloverdeling

### 3. Webservices

Webservices is een technologie die SOA realiseert op een concreter niveau. De technologie achter webservices maakt het mogelijk om een service-interface te beschrijven, te publiceren, te zoeken en aan te roepen via internet (zie figuur 2). De rolverdeling van SOA (figuur 1) komt hierin sterk naar voren.

De kracht van webservices is de platform- en programmeertaal onafhankelijkheid. Web services zijn namelijk niet verbonden aan een specifieke applicatie zoals SAP of een programmeertaal als Java. Dit maakt het mogelijk om op eenvoudige wijze een webservice-interface beschikbaar te stellen en aan te spreken vanuit elk informatiesysteem dat verbonden is met internet. Dit is vergelijkbaar met de eerdere analogie van het stopcontact.

De communicatie tussen een webservice en service-aanvrager verloopt door middel van berichten in XML-formaat. XML, eXtensible Markup Language, is een platformonafhankelijke taal waarmee data zodanig gestructureerd kunnen worden, dat zowel mensen als informatiesystemen ze kunnen interpreteren. Een informatiesysteem kan XML-berichten geautomatiseerd creëren en verwerken.



Figuur 2 – Webservicearchitectuur

### Voorbeeld

Een voorbeeld van een webservice is een online-dienst die aangeboden wordt door een reisbureau. Reisbureaus kunnen een webservice als centraal toegangspunt aanbieden om informatie van verschillende (derde) partijen (vliegtuigmaatschappijen, hotels, verzekeringsmaatschappijen, autoverhuurders, enzovoort) te aggregeren. In plaats van te zoeken op de individuele websites kan de gebruiker vanuit de website van het reisbureau de informatie met één XML-bericht simultaan doorzoeken, raadplegen en combineren. De webservice kan daarnaast extra waarde toevoegen door bijvoorbeeld alleen vluchten en hotels te presenteren die aan bepaalde criteria voldoen. De technologie achter webservices maakt het mogelijk om dergelijke verzoeken automatisch af te handelen.

### 4. Voor- en nadelen van SOA

Wat heeft een organisatie nu precies aan SOA?

#### Interoperabiliteit

Het belangrijkste voordeel van SOA is de interoperabiliteit: organisaties kunnen met SOA hun informatiesystemen zeer snel koppelen aan een systeem waarvan de organisatie geen diepgaande kennis heeft. In een IT-omgeving die de grenzen van de organisatie overschrijdt, is SOA daardoor een sterke architectuur om met de complexiteit van heterogene informatiesystemen om te gaan.

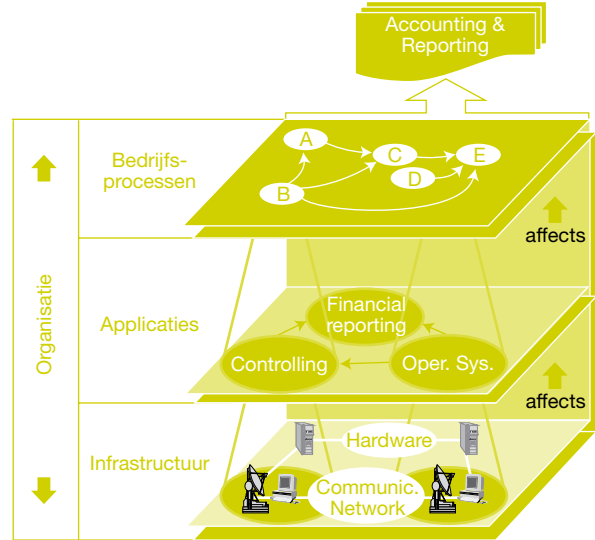
#### Verouderde informatiesystemen ontsluiten

SOA biedt ook de mogelijkheid om verouderde informatiesystemen te ontsluiten uit hun afgesloten omgeving. Bij veel organisaties vervullen verouderde informatiesystemen nog altijd bedrijfskritische taken. Het nadeel van deze systemen is dat ze niet eenvoudig te koppelen zijn met moderne systemen, terwijl dat wel noodzakelijk is om processen goed te ondersteunen. Door services toe te voegen aan deze verouderde informatiesystemen, kunnen ook zij platform- en taalafhankelijk aangesproken worden door andere informatiesystemen.

#### Gevaar van wildgroei aan services

Een nadeel van SOA is dat een organisatie snel de neiging heeft om voor elk probleem een service te ontwikkelen. Het resultaat is vaak een wildgroei aan services zonder enige samenhang. Het ontwikkelen van een SOA-strategie en visie is daarom van cruciaal belang.

In figuur 3 wordt weergegeven welke lagen in een organisatie onderdeel uitmaken van de informatievoorziening die uiteindelijk leidt tot de jaarrekening. SOA heeft impact op al deze lagen: de bedrijfsprocessen, applicaties, infrastructuur en organisatie. De impact van SOA vraagt dus om een adequate interne beheersing. Het ontbreekt echter momenteel aan methodieken en tools om die interne beheersing adequaat te organiseren in een SOA. In dit kader wordt momenteel veel onderzoek gedaan naar de impact van SOA op bedrijfscontinuïteit, (informatie) beveiliging en (gereguleerde) compliance.



Figuur 3 – IT infrastructuur

### Risico's

Het koppelen van informatiesystemen door middel van SOA brengt ook risico's met zich mee. Denk bijvoorbeeld aan de continuïteit van een bedrijfsproces die in gevaar gebracht wordt als een service niet beschikbaar is. En ook de beveiliging in een servicegeoriënteerde omgeving is een behoorlijke uitdaging.

### 5. SOA in de praktijk

In de praktijk wordt SOA steeds vaker toegepast. Enerzijds ontwikkelen de belangrijkste softwareleveranciers steeds meer SOA-gerelateerde toepassingen voor hun softwareomgevingen. SAP bijvoorbeeld, biedt de SAP NetWeaver-omgeving aan, Oracle de SOA Suite. IBM heeft de WebSphere-omgeving volledig voorbereid voor SOA, en Microsoft biedt SOA-oplossingen aan door middel van BizTalk Server. Deze oplossingen maken het mogelijk om een bestaand ERP systeem te koppelen met een ander intern systeem, of een systeem van een externe partij.

Organisaties zetten SOA-oplossingen daarnaast steeds meer in om vraagstukken over verouderde informatiesystemen en interoperabiliteit op te lossen. Vooral ESB-oplossingen (ESB: Enterprise Service Business) worden in de praktijk steeds vaker ingezet om een SOA te

realiseren. Een ESB is een systeem dat functionaliteit ter ondersteuning van SOA-omgevingen aanbiedt. Dit kan gaan om het ondersteunen en beveiligen van XML-communicatie tussen services, het coördineren van complexe services en transacties, en het monitoren van services.

### 6. Impact op de interne beheersing

Het gebruik van SOA heeft ook gevolgen voor de interne beheersing binnen organisaties. Door informatiesystemen te koppelen met services wordt de rol van geautomatiseerde gegevensverwerking voor organisaties belangrijker maar ook complexer. Daarnaast ontstaan meer afhankelijkheden tussen organisaties. Via services krijgt een organisatie data (informatie) binnen vanuit andere organisaties. Hoe waarborg je dat deze data betrouwbaar zijn? Hoe garandeer je dat de data voldoende beveiligd zijn? De adequaatheid van de informatievoorziening van een organisatie wordt daardoor afhankelijk van (externe) services.

### 7. COMPAS

Via het COMPAS-project, een initiatief van de EU, wordt gedurende drie jaar een door de Europese commissie gesubsidieerd onderzoek uitgevoerd. Het project wordt

uitgevoerd door een groep van Europese universiteiten en bedrijven, waaronder de Universiteit van Tilburg en PricewaterhouseCoopers (PwC). Met COMPAS moeten er nieuwe modellen, talen, en een raamwerk ontwikkeld kunnen worden om compliance-eisen te kunnen modelleren, specificeren, valideren in een SOA, en om compliance tijdens de uitvoering van een (geautomatiseerd) bedrijfsproces in een SOA te kunnen afdwingen en monitoren.

Aanleiding voor het COMPAS-project is de volgende. In de huidige situatie worden compliancevraagstukken in een SOA nog niet op een integrale manier aangepakt, maar op basis van een specifieke casus. Dit heeft een aantal nadelen. Allereerst zijn de complianceoplossingen lastig te onderhouden door het gebrek aan architectuur en strategie. Ten tweede zijn de oplossingen, de services, niet opnieuw te gebruiken omdat ze specifiek zijn ingericht voor de betreffende casus. En ten derde is het tijdrovend om elke service up-to-date te houden met betrekking tot wijzigingen in de eisen van de casus, bijvoorbeeld in het geval van een wetswijziging.

Om deze tekortkoming op te lossen, is het COMPAS-project geïnitieerd. Met het SOA-raamwerk kan de organisatie het gehele SOA-spectrum ondersteunen op een gecentraliseerde wijze. Bovendien moet het met het raamwerk en de bijhorende tools eenvoudiger worden om complianceoplossingen op een integrale manier voor SOA-omgevingen te ontwikkelen en onderhouden.

## 8. Tot slot

SOA is een softwarearchitectuur waarmee organisaties gedistribueerde informatiesystemen kunnen ontwerpen en koppelen met behulp van services. Webservices is op dit moment de meest prominente technologie om platform-onafhankelijke services te ontwikkelen. Dit gaat echter niet gepaard zonder risico's. Het gebrek aan strategie kan leiden tot een wildgroei aan services. Deficiënties in de interne beheersing van een SOA kunnen daarnaast leiden tot bedrijfsrisico's op het gebied van informatiebeveiliging en informatiebetrouwbaarheid. SOA vereist daarom net als andere IT-ontwikkelingen behoedzaamheid, beheersing en een duidelijk visie over wat men wil bereiken met SOA.