



# Service Oriented Architecture & Logistieke Dienstverleners

De mogelijkheden van SOA voor logistieke dienstverleners

**Master thesis Informatica**

Variant Management & Toepassing (MT)

Radboud Universiteit Nijmegen

Nijmeegs Instituut voor Informatica en Informatiekunde (NIII)

Afstudeernummer: 585

Chris Jager

Mei 2008

**Afstudeerbegeleiders:**

Prof. Dr. Mario van Vliet, Informatica

Dr. Paul Ligthart, Management Sciences

Rik Laurens, Capgemini Nederland

**Referent:**

Dr. Patrick van Bommel, Informatica

## **Colofon**

Radboud Universiteit Nijmegen  
Nijmeegs Instituut voor Informatica en Informatiekunde (NIII)  
Toernooiveld 1  
6525 ED Nijmegen

Capgemini Nederland B.V.  
practice P42  
Papendorpseweg 100  
3528 BJ Utrecht

## Voorwoord

Dit document is mijn master thesis van de studie Informatica aan de Radboud Universiteit Nijmegen met de mastervariant Management en Toepassing. Dit betekent dat de master thesis management en informatica moet combineren en relateren. Het onderzoek is uitgevoerd bij Capgemini Nederland, te Utrecht, bij practice P42, van september 2007 tot april 2008. Het onderzoek is in combinatie met de Radboud Universiteit en Capgemini tot stand gekomen.

Het onderzoek is een verkenning geweest voor het software paradigma Service Oriented Architecture (SOA), een onderwerp dat de laatste paar jaar veel aandacht heeft gekregen. Het ontbreekt nog aan implementaties van een systeem dat zich houdt aan de regels en principes van SOA. Desondanks zijn de verwachtingen erg hoog. Het onderzoek verkent de potentie van SOA bij logistieke dienstverleners. Logistieke dienstverleners zijn in een markt actief die een steeds grotere rol inneemt bij het aaneenschakelen van bedrijfsprocessen van verschillende bedrijven. Dit maakt deze markt potentieel erg geschikt voor een raamwerk als SOA.

Capgemini heeft een grote bijdrage geleverd bij het tot stand komen van dit onderzoek. Ik wilde daarom Frank Wammes bedanken voor het verschaffen van de mogelijkheden en de middelen voor het onderzoek. Ook wilde ik de rest van de afdeling van P42 bedanken voor alle hulp in de afgelopen maanden.

Het onderzoek bestond uit een aantal interviews en gesprekken. Graag wil ik mijn dankbaarheid betuigen aan Jaap Blankevoort, Erik van Dort, Michael van der Dungen, Rik Laurens, Pino Melis, André Rottier, Luc Scheidel, Ron Tolido, Joost van der Vlies, Frank Wammes en Jasper van der Wulp (allen Capgemini) en Frans van der Velden (Verkooijen Transport) voor het verschaffen van zeer waardevolle informatie.

Ook wilde ik in het bijzonder Rik Laurens, Paul Ligthart en Mario van Vliet bedanken voor hun begeleiding, inzet, ondersteuning en opbouwende kritiek tijdens het gehele proces. Tot slot wilde ik mijn medeafstudeerders bij Capgemini Annemaaïke en Sefan bedanken voor de mentale steun, en mijn ouders en mijn vriendin Martine, die me tijdens mijn hele studie hebben gesteund.

Chris Jager

## Samenvatting

Organisaties besteden steeds meer logistieke bedrijfsprocessen uit aan logistieke dienstverleners om verschillende redenen. Organisaties vinden bijvoorbeeld dat logistiek niet tot hun kernactiviteiten behoort, willen sneller kunnen reageren op veranderingen in de markt, of zijn op zoek naar meer expertise op het gebied van logistiek. Waar voorheen vooral de processen zoals transport en opslag werden uitbesteed, worden tegenwoordig steeds meer processen die daaraan gerelateerd zijn uitbesteed. In een supply chain is de logistieke dienstverlener daarom niet alleen steeds meer verantwoordelijk voor de fysieke goederenstroom, maar ook voor de aansturing van deze stroom door middel van IT. IT krijgt een steeds prominentere plek bij logistieke dienstverleners. Een aantal problemen die hierbij komen kijken kunnen mogelijk worden opgelost door de IT in te richten volgens de regels en principes van Service Oriented Architecture (SOA). SOA pretendeert bedrijfsprocessen en applicaties die ondersteuning bieden bij deze processen (deels) te ontkoppelen, waardoor bedrijfsprocessen meer centraal in een organisatie komen te staan. Hierdoor wordt een bedrijf flexibeler.

In deze scriptie zijn drie zaken onderzocht: wat zijn de knelpunten in de IT bij logistieke dienstverleners, welke karakteristieken van SOA zijn er te identificeren, en hoe kunnen deze karakteristieken ingezet worden om de knelpunten bij logistieke dienstverleners op te lossen. Dit wordt bekeken vanuit een technologisch perspectief. De knelpunten zijn verkregen door te kijken naar de implicaties van drie generieke strategieën voor logistieke dienstverleners. De karakteristieken van SOA komen voort uit de uitgangspunten van SOA.

Als grootste knelpunten in de IT van logistieke dienstverleners zijn geïdentificeerd het gebrek aan overeenstemming bij open standaarden (zowel syntactisch als semantisch), de beveiliging van de informatiestroom, het gebrek aan een geschikt raamwerk om informatie te delen tussen bedrijven, de inflexibiliteit van IT systemen en het feit dat vooral de kleinere logistieke dienstverleners geen geld hebben om een investering te doen in IT. Als karakteristieken van SOA zijn interoperabiliteit van het systeem, de herbruikbaarheid van services en de flexibiliteit van het systeem geïdentificeerd. Hier staat tegenover dat SOA vooral effect heeft op de lange termijn, en dat er veel extra sturing (governance) nodig is om SOA te doen laten slagen.

Uit dit onderzoek is gebleken dat SOA de complexiteit van de IT-huishouding van logistieke dienstverleners reduceert. Dit scheelt in de kosten voor onderhoud van systemen. Door de interoperabiliteit van het systeem is het gemakkelijker om samen te werken met andere organisaties. Het uitbesteden van bedrijfsprocessen wordt hierdoor aantrekkelijker. Er spelen echter een aantal problemen van minder technologische aard. Deze verhinderen dat SOA op korte termijn al succesvol is. Het vaststellen van de te gebruiken standaarden is lastig op te lossen, net als de invoering van een systeem dat zich houdt aan de regels en principes van SOA. De onvoorziene problemen hebben allemaal te doen hebben met de kenmerkende, pragmatische instelling van logistieke dienstverleners.

Organisaties die klanten met diensten op maat willen bedienen kunnen veel voordeel behalen door over te stappen op SOA. Het levert echter ook een aantal risico's op, met name op het gebied van tijd en geld. Logistieke dienstverleners die standaard diensten aanbieden en zich daartoe beperken hebben weinig baat bij SOA. Meer onderzoek op basis van de ervaringen met SOA is nodig. Vooralsnog zijn de mogelijkheden hierin helaas beperkt door het (te) kleine aanbod van systemen die zich houden aan de regels en principes van SOA.

## Inhoudsopgave

|                                                            |    |
|------------------------------------------------------------|----|
| 1. Inleiding .....                                         | 7  |
| 1.1. Onderzoeksaanleiding .....                            | 7  |
| 1.2. Theoretisch kader .....                               | 7  |
| 1.3. Probleemstelling .....                                | 8  |
| 1.4. Onderzoeksmethode .....                               | 8  |
| 1.5. Stage .....                                           | 9  |
| 1.6. Opbouw scriptie .....                                 | 9  |
| 2. Context .....                                           | 10 |
| 2.1. Processen .....                                       | 10 |
| 2.2. Uitbesteding van processen .....                      | 11 |
| 2.3. Supply Chain Management .....                         | 13 |
| 2.4. Supply chain strategieën .....                        | 13 |
| 2.5. Koppeling processen aan applicaties .....             | 14 |
| 3. Logistieke dienstverleners .....                        | 16 |
| 3.1. Definitie .....                                       | 16 |
| 3.2. Strategieën .....                                     | 18 |
| 3.3. Implicaties en knelpunten van strategieën .....       | 20 |
| 4. Service Oriented Architecture .....                     | 23 |
| 4.1. Opkomst bedrijfssystemen .....                        | 23 |
| 4.2. Software paradigma's .....                            | 25 |
| 4.3. Enterprise architecture .....                         | 29 |
| 4.4. SOA model .....                                       | 32 |
| 4.5. Service concepten .....                               | 34 |
| 4.6. Definities .....                                      | 36 |
| 4.7. Karakteristieken SOA .....                            | 37 |
| 5. Mogelijkheden SOA voor logistieke dienstverleners ..... | 40 |
| 5.1. Model .....                                           | 40 |
| 5.2. Koppelingen .....                                     | 41 |
| 5.3. Implicaties voor strategieën .....                    | 45 |
| 6. Resultaten .....                                        | 48 |
| 6.1. Methodologie .....                                    | 48 |
| 6.2. Koppelingen .....                                     | 48 |
| 6.3. Extra knelpunten .....                                | 53 |
| 6.4. Vergelijking .....                                    | 54 |
| 7. Conclusie .....                                         | 57 |
| 7.1. Resultaten .....                                      | 57 |
| 7.2. Discussie en implicaties .....                        | 58 |
| Referenties .....                                          | 60 |
| Bijlagen .....                                             | 63 |
| A. Begrippen .....                                         | 63 |
| B. Afkortingen .....                                       | 67 |
| C. Interviewronde .....                                    | 68 |

## Lijst van figuren

|                                                                         |    |
|-------------------------------------------------------------------------|----|
| figuur 1: Grafische representatie onderzoeksmethode .....               | 9  |
| figuur 2: Voorbeeld van processen en functionele domeinen [BRU05] ..... | 10 |
| figuur 3: Uitbesteden van processen .....                               | 12 |
| figuur 4: Procesintegratie en applicatieintegratie .....                | 15 |
| figuur 5: Megaproces logistiek [VLI98] .....                            | 17 |
| figuur 6: Diensten matrix [MAK90] .....                                 | 20 |
| figuur 7: Legenda figuren hoofdstuk 4 .....                             | 26 |
| figuur 8: Software paradigma (algemene weergave) [DOB07] .....          | 26 |
| figuur 9: Procedureel ontwikkelen [DOB07] .....                         | 26 |
| figuur 10: Object georiënteerd ontwikkelen [DOB07] .....                | 27 |

|                                                                                                                    |    |
|--------------------------------------------------------------------------------------------------------------------|----|
| figuur 11: <i>Component gebaseerd ontwikkelen</i> [DOB07]                                                          | 28 |
| figuur 12: <i>Service georiënteerd ontwikkelen</i> [DOB07]                                                         | 29 |
| figuur 13: <i>Integrated Architecture Framework</i> [CAP08]                                                        | 30 |
| figuur 14: <i>5-lagen architectuur</i> [DOB07]                                                                     | 31 |
| figuur 15: <i>SOA model</i> [SPE05]                                                                                | 33 |
| figuur 16: <i>Compleet SOA model</i> [DOB07]                                                                       | 34 |
| figuur 17: <i>De servicelaag toegevoegd</i>                                                                        | 39 |
| figuur 18: <i>Onderzoeksmodel</i>                                                                                  | 40 |
| figuur 19: <i>Koppeling knelpunten met karakteristieken, theoretisch</i>                                           | 45 |
| figuur 20: <i>Koppelingen bij logistieke dienstverleners met directe strategie en routine klussen</i>              | 46 |
| figuur 21: <i>Koppelingen bij logistieke dienstverleners met stop-over strategie en standaard diensten</i>         | 46 |
| figuur 22: <i>Koppelingen bij logistieke dienstverleners met assembled-stop-over strategie en diensten op maat</i> | 47 |
| figuur 23: <i>Koppeling knelpunten met karakteristieken, praktijk</i>                                              | 53 |
| figuur 24: <i>Verschillen tussen theorie en praktijk</i>                                                           | 55 |

## Lijst van tabellen

|                                                                              |    |
|------------------------------------------------------------------------------|----|
| tabel 1: <i>Strategieën en de bijbehorende implicaties</i>                   | 40 |
| tabel 2: <i>Implicaties voor IT en bijbehorende knelpunten</i>               | 41 |
| tabel 3: <i>Uitgangspunten SOA en bijbehorende karakteristieken</i>          | 41 |
| tabel 4: <i>Knelpunten gekoppeld aan interoperabiliteit, theoretisch</i>     | 42 |
| tabel 5: <i>Knelpunten gekoppeld aan herbruikbaarheid, theoretisch</i>       | 42 |
| tabel 6: <i>Knelpunten gekoppeld aan flexibiliteit, theoretisch</i>          | 43 |
| tabel 7: <i>Knelpunten gekoppeld aan governance, theoretisch</i>             | 44 |
| tabel 8: <i>Knelpunten gekoppeld aan lange termijn effecten, theoretisch</i> | 44 |
| tabel 9: <i>Knelpunten gekoppeld aan interoperabiliteit, praktijk</i>        | 49 |
| tabel 10: <i>Knelpunten gekoppeld aan herbruikbaarheid, praktijk</i>         | 50 |
| tabel 11: <i>Knelpunten gekoppeld aan flexibiliteit, praktijk</i>            | 51 |
| tabel 12: <i>Knelpunten gekoppeld aan governance, praktijk</i>               | 52 |
| tabel 13: <i>Knelpunten gekoppeld aan lange termijn effecten, praktijk</i>   | 52 |

## Lijst van definities

|                                             |    |
|---------------------------------------------|----|
| Definitie 1: proces                         | 10 |
| Definitie 2: Supply Chain Management        | 13 |
| Definitie 3: applicatie                     | 14 |
| Definitie 4: logistieke dienstverlener      | 18 |
| Definitie 5: entiteit                       | 25 |
| Definitie 6: behoefte                       | 25 |
| Definitie 7: vermogen                       | 25 |
| Definitie 8: architectuur                   | 29 |
| Definitie 9: service provider               | 32 |
| Definitie 10: service consumer              | 32 |
| Definitie 11: service participanten         | 32 |
| Definitie 12: service directory             | 32 |
| Definitie 13: service                       | 36 |
| Definitie 14: Service Oriented Architecture | 36 |

## 1. Inleiding

Dit hoofdstuk vormt een introductie op het onderwerp van deze scriptie en het onderzoek. In deze scriptie wordt gekeken wat de mogelijkheden van *Service Oriented Architecture* (SOA) zijn voor het uitbesteden van processen aan logistieke dienstverleners. In het theoretische kader worden enkele begrippen uitgelegd die essentieel zijn voor deze scriptie. Hierna volgt de probleemstelling en de onderzoeksmethode. Vervolgens wordt ingegaan op de stage bij Capgemini. Tot slot volgt een leeswijzer voor deze scriptie.

### 1.1. Onderzoeksaanleiding

Het onderzoek wordt uitgevoerd in het kader van de afstudeerscriptie aan de Radboud Universiteit Nijmegen. De afstudeerscriptie vormt het afsluitende onderdeel van de masteropleiding Informatica, variant Management en Toepassing, aan de faculteit Natuurkunde, Wiskunde en Informatica (NWI). Het onderzoek is uitgevoerd bij Capgemini Nederland. De probleemstelling is tot stand gekomen in samenwerking met de Radboud Universiteit Nijmegen en Capgemini Nederland.

### 1.2. Theoretisch kader

Het afstemmen van de activiteiten van samenwerkende organisaties wordt steeds belangrijker. Hierbij is het van groot belang dat ook de informatiesystemen goed op elkaar aansluiten. De integratie van de systemen wordt steeds belangrijker. SOA kan hierin mogelijk een grote rol vervullen. Als inleiding op het onderwerp van deze scriptie en de probleemstelling, worden enkele begrippen uitgelegd. In volgende hoofdstukken zal nader worden ingegaan op de begrippen.

#### Service Oriented Architecture

*Service Oriented Architecture* (SOA) is een *software paradigma* (concept of abstractie, zie hoofdstuk 4.2). Een groot gedeelte van deze scriptie gaat hierover. SOA helpt bij het inrichten van de IT-huishouding van een organisatie. Men biedt IT-functionaliteit aan op een manier dat deze herbruikbaar, interoperabel en flexibel is. Dit wordt bereikt door de IT-functionaliteit aan te bieden in de vorm van zo onafhankelijk mogelijke *services*. Deze services kunnen binnen de IT-huishouding van een bedrijf, maar ook daarbuiten gebruikt worden om applicaties te vormen die op hun beurt bedrijfsprocessen ondersteunen. De services kunnen hergebruikt worden en werken volgens afgesproken standaarden.

#### Bedrijfsprocessen

Een *bedrijfsproces* (afgekort *proces*) is een verzameling van logisch gerelateerde taken met een gedefinieerd bedrijfsresultaat (zie hoofdstuk 2.1). Processen worden met regelmaat gewijzigd. Dit komt omdat de markten waarin bedrijven opereren aan veel invloeden onderhevig zijn. De bedrijfsstructuur is vaak zo ingericht, dat de IT-ondersteuning bij een proces mee moet veranderen als er een wijziging optreedt bij het proces. Door de IT-huishouding zo in te richten dat deze gemakkelijker af te stemmen is op de processen, wordt dit eenvoudiger. SOA pretendeert hiervoor een oplossing te bieden.

#### Supply Chain Management

*Supply Chain Management* (SCM) (zie hoofdstuk 2.3) is het coördineren van inkoop-, productie- en distributieprocessen in een netwerk van bedrijven. Deze bedrijfsprocessen volgen elkaar op. Als de processen ondergebracht zijn in verschillende bedrijven, ontstaat er een keten van aan elkaar geschakelde bedrijven. Dit wordt een *supply chain* genoemd. In een supply chain bestaat een spanningsveld tussen de efficiëntie en de reactietijd van een organisatie. SCM moet deze wegnemen.

#### Logistieke dienstverleners

*Logistieke dienstverleners* (LSP's, *Logistics Service Providers*) (zie hoofdstuk 3) zijn bedrijven die zich gespecialiseerd hebben in logistieke processen. Logistieke dienstverleners nemen deze processen over van externe bedrijven. De bedrijven die gebruik maken van de logistieke diensten worden *verladers* genoemd. De aangeboden diensten kunnen bestaan uit het vervoeren of opslaan van goederen. Maar ook aanverwante zaken zoals planning, het verpakken van goederen of zelfs assemblage van producten worden uitbesteed aan logistieke dienstverleners.

### **Applicaties**

*IT applicaties* (voortaan kortweg *applicaties*) (zie ook 2.5) betekent letterlijk 'toepassing'. Applicaties zijn programma's die een reeks taken uitvoeren onder controle van een gebruiker. Dit in tegenstelling tot systeemprogramma's, die veelal op de achtergrond draaien. In een bedrijf worden applicaties gebruikt om een bedrijfsproces te ondersteunen.

### **1.3. Probleemstelling**

In het onderzoek wordt antwoord gegeven op drie onderzoeksvragen. Per onderzoeksvraag zijn enkele deelvragen geformuleerd, die helpen antwoord te geven op de onderzoeksvraag.

- Welke trends zijn er op het gebied van uitbesteding aan logistieke dienstverleners en welke IT problemen komen kijken bij het realiseren hiervan?
  - o Wat zijn logistieke dienstverleners?
  - o Waarom wordt er uitbesteed aan logistieke dienstverleners?
  - o Wat zijn de implicaties voor de IT-huishouding bij het uitbesteden van processen aan logistieke dienstverleners?
- Hoe kan een SOA een organisatie ondersteunen?
  - o Hoe is SOA ontstaan?
  - o Wat is een SOA?
  - o Wat zijn de karakteristieken van een SOA?
- Waar kan een SOA ondersteuning bieden bij het uitbesteden van processen aan logistieke dienstverleners en wat zijn de tekortkomingen?
  - o Wanneer biedt een SOA toegevoegde waarde voor de relatie verlader en logistieke dienstverlener en op welke manier?
  - o Wat zijn de beperkingen van SOA bij het uitbesteden van processen aan logistieke dienstverleners?
  - o Wat is de ervaring met SOA in de praktijk bij het uitbesteden van processen aan logistieke dienstverleners?

### **1.4. Onderzoeksmethode**

De volgende activiteiten zijn uitgevoerd om de onderzoeksvragen te beantwoorden. De grafische representatie van de onderzoeksmethode is weergegeven in figuur 1.

#### **Literatuurstudie**

Er is een literatuurstudie uitgevoerd over het uitbesteden van diensten aan logistieke dienstverleners en het kader daar omheen, SCM. Hierin zijn de laatste inzichten uit de wetenschap verwerkt. Dit heeft geresulteerd in een lijst van knelpunten voor de IT-huishouding, gezien vanuit het perspectief van de logistieke dienstverlener ter bevordering van het uitbesteden van logistieke processen in de supply chain.

Een tweede studie is uitgevoerd naar het software paradigma SOA. In het bijzonder zijn de uitgangspunten en de hieruit voortkomende karakteristieken van SOA besproken.

#### **Modelvorming**

Een theoretisch model is opgesteld waarin de knelpunten van de IT-huishouding van logistieke dienstverleners gekoppeld zijn aan de karakteristieken van SOA. Dit is gedaan aan de hand van de twee literatuurstudies.

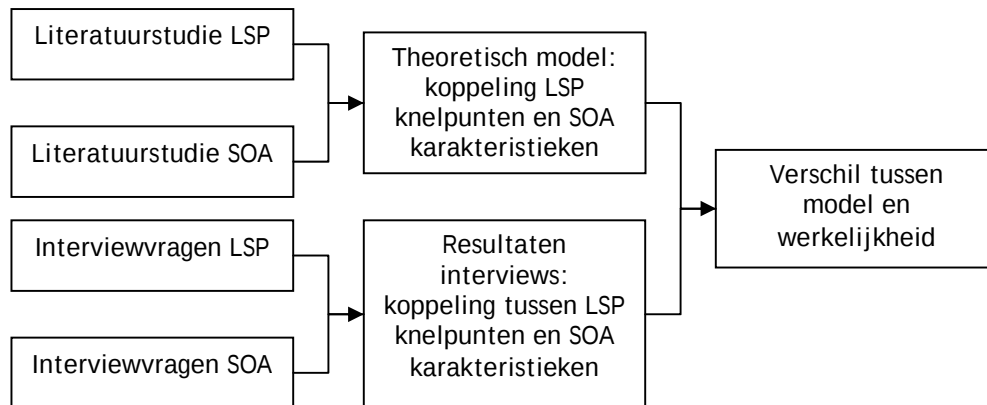
#### **Interviews**

Er zijn drie zaken met behulp van interviews onderzocht. Het eerste dat is onderzocht zijn de knelpunten die bestaan bij logistieke dienstverleners ten aanzien van de IT-huishouding bij het

overnemen van uitbestede processen. Als tweede is onderzocht wat de karakteristieken zijn van SOA in de praktijk. Het derde en laatste wat is onderzocht, is de koppeling tussen de knelpunten bij logistieke dienstverleners ten aanzien van de IT-huishouding en de karakteristieken van SOA.

### Vergelijking

Tot slot zijn de resultaten van de interviews vergeleken met het theoretische model. Hierdoor is een helder beeld ontstaan of het gebruik van SOA mogelijkheden biedt voor logistieke dienstverleners. Ook zijn de risico's bij het gebruik van SOA in beeld gebracht.



figuur 1: Grafische representatie onderzoeksmethode

### 1.5. Stage

Het onderzoek is uitgevoerd bij Capgemini Nederland, een bedrijf dat gespecialiseerd is in consultancy, outsourcing en technology. Capgemini heeft veel consultants in dienst die experts zijn op het gebied van SOA, van logistieke dienstverleners en van software die door deze logistieke dienstverleners wordt gebruikt. Vooral hun kennis uit de praktijk was nodig voor het doen slagen van het onderzoek. Hierdoor kon het theoretische model (zie ook 1.4) getoetst worden aan de werkelijkheid door middel van interviews (zie hoofdstuk 6). Capgemini is met een aantal SOA projecten bezig. Onderzoek naar de toepassing van SOA in (delen van) supply chains staat hoog op de agenda. Verwacht wordt dat veel winst te behalen is voor supply chains bij het gebruik van SOA. Hierdoor is dit onderzoek zowel interessant voor informatica als voor management.

### 1.6. Opbouw scriptie

De scriptie bestaat uit zeven hoofdstukken. In het eerste hoofdstuk wordt een inleiding gegeven op het onderwerp en het onderzoek. In hoofdstuk 2 wordt de context van het onderwerp besproken. Hierin wordt SCM behandeld en de koppeling van processen aan applicaties. In hoofdstuk 3 wordt de rol van logistieke dienstverleners in de supply chain verder uitgediept. Op basis van strategieën voor logistieke dienstverleners wordt een lijst van knelpunten gegeven voor wijzigingen in de IT-huishouding. In het vierde hoofdstuk wordt ingegaan op software paradigma's. SOA wordt hierin behandeld. Op basis van de uitgangspunten van SOA worden de karakteristieken van SOA in kaart gebracht. In hoofdstuk 5 wordt een theoretisch model gevormd van de knelpunten voor de IT-huishouding van logistieke dienstverleners gekoppeld aan de karakteristieken van SOA. In hoofdstuk 6 worden de resultaten van de interviews besproken. De resultaten worden vergeleken met het model uit hoofdstuk 5. En ten slotte worden in hoofdstuk 7 de resultaten van de scriptie besproken. De scriptie bestaat uit twee delen: het theoretische gedeelte en het praktische gedeelte.

## 2. Context

In dit hoofdstuk wordt een kader beschreven waarin we logistieke dienstverleners kunnen plaatsen. Logistieke dienstverleners maken deel uit van een hele keten van organisaties, die allemaal aan elkaar gekoppeld zijn. Door dit raamwerk eerst te beschrijven, wordt de rol van logistieke dienstverleners in de keten duidelijker. Het hoofdstuk begint met een uitleg over processen, en met het uitbesteden daarvan. Vervolgens wordt de sprong naar SCM gemaakt. Het hoofdstuk eindigt met de koppeling van applicaties aan processen.

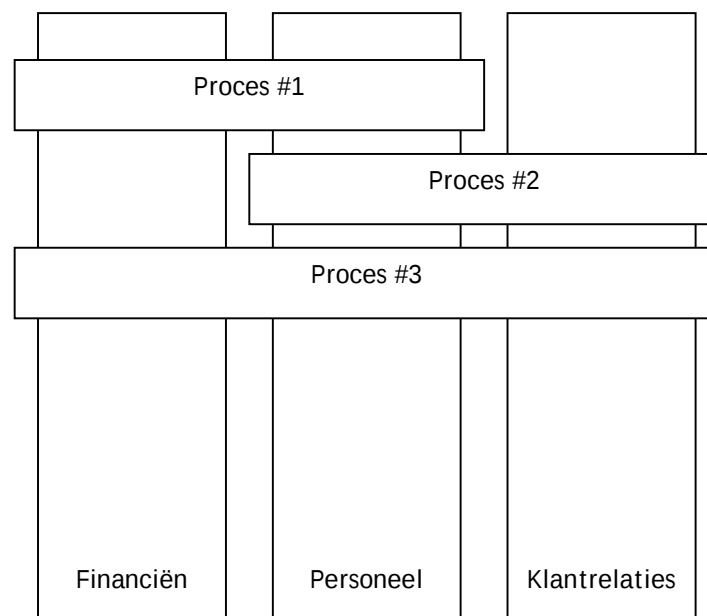
### 2.1. Processen

Een *proces* kunnen we als volgt definiëren:

Definitie 1: proces

*“Een aantal sequentieel en/of parallel uitgevoerde activiteiten die in hun samenhang input verwerken tot output.” - [VLI98]*

Een proces kan door verschillende individuen en applicaties uitgevoerd worden. De samenhang tussen alle individuen of applicaties dient goed op elkaar afgestemd te zijn. De volgorde van processen binnen een organisatie wordt *procesorkestratie* genoemd. Processen bepalen vaak de dagelijkse werkzaamheden van medewerkers. Organisaties zijn echter veelal afgestemd op basis van bedrijfsfuncties. Er zijn aparte onderdelen voor financiën en personeel te onderscheiden, die allen hun eigen hiërarchie en applicaties hebben. Processen lopen over de functionele domeinen van een organisatie heen (zie figuur 2).



figuur 2: Voorbeeld van processen en functionele domeinen [BRU05]

Processen zijn er op verschillende niveaus. Hoe dieper het niveau van een proces, hoe gedetailleerder de beschrijving van het proces wordt. In deze scriptie gebruiken we vier niveaus [VLI98]. Gekozen is voor vier niveaus omdat dit voor de juiste abstractie zorgt die nodig is bij SOA, wat later aan bod komt (zie hoofdstuk 4.3).

### **Megaprocesniveau**

Het *megaprocesniveau* beschrijft processen op het hoogste niveau in een onderneming. De processen op dit niveau worden *megaprocessen* genoemd. Typische megaprocessen zijn beleid, marketing en verkoop, productontwikkeling en logistiek. Deze processen overschrijden de functionele domeinen van een organisatie (zie figuur 2). Ieder megaproces is opgedeeld in *hoofdprocessen*.

### **Hoofdprocesniveau**

Het *hoofdprocesniveau* beschrijft de subprocessen van een megaproces. De processen op dit niveau worden *hoofdprocessen* genoemd. In deze scriptie overschrijden de hoofdprocessen de functionele domeinen niet, in tegenstelling tot de megaprocessen. Hoofdprocessen worden ingedeeld in *primaire* en *ondersteunende* processen. De ondersteunende processen zijn processen die sturing geven aan de primaire processen. De hoofdprocessen zijn op te delen in *deelprocessen*.

### **Deelprocesniveau**

Het *deelprocesniveau* beschrijft de subprocessen van een hoofdproces. De processen op dit niveau worden *deelprocessen* genoemd. De deelprocessen zijn weer veel gedetailleerder dan hoofdprocessen. Deelprocessen zijn iets abstracter dan een activiteit. Hiermee kan over activiteiten gesproken worden zonder de technische details van een proces in beschouwing te nemen.

### **Activiteitsniveau**

Het *activiteitsniveau* beschrijft de *activiteiten* die worden uitgevoerd bij een deelproces. Het activiteitsniveau is het laagste niveau van processen. De activiteiten kunnen worden uitgevoerd door een persoon, maar ook kunnen deze activiteiten uitgevoerd worden door een applicatie. Het grote verschil met het activiteitsniveau en de overige niveaus van procesbeschrijving is de *volgorde* en de *opvolging* van de activiteiten. De volgorde van de activiteiten staat vast. Dit in tegenstelling tot veel processen uit hogere niveaus.

## **2.2. Uitbesteding van processen**

Ten tijden van de Industriële Revolutie produceerde men grote hoeveelheden goederen met behulp van machines. De machines produceerden goederen vele malen sneller dan mensen. Goederen werden drastisch goedkoper. De machines uit die tijd waren vaak wel gebouwd voor slechts één type product. De bedrijfsprocessen in die tijd veranderden dan ook niet veel. Ford Motor Company (tegenwoordig Ford) was een Amerikaans bedrijf dat auto's produceerde op grote schaal. Ford was in bijna alle opzichten zelfvoorzienend [HUG06]. Ford delfde zijn eigen grondstoffen zoals ijzer, hout en katoen. Vele processen later rolde dan een auto uit de fabriek. En dat zonder interventie van een ander bedrijf. Door zelf alles te regelen werd een grote mate van efficiëntie bereikt. Auto's waren voortaan ook betaalbaar voor het normale volk.

De vraag naar meer gedifferentieerde producten is tegenwoordig vele malen groter dan toen, en neemt alleen maar toe [JAC06]. Doordat de markten waarin organisaties opereren snel veranderen, is het noodzakelijk processen snel aan te kunnen passen. Maar een verandering in een proces kan verstrekende gevolgen hebben voor een proces ergens anders. Op deze manier wordt een kettingbeweging in gang gezet. Het kan daarom lang duren voordat een rimpel in de schakeling van processen is uitgevlakt. Stel dat Ford Motor Company andere modellen auto's ging bouwen die veel kleiner waren dan het ene type die ze voorheen produceerden. De hoeveelheid verf voor een auto neemt dan af, de stoelen moeten misschien wel andere maten hebben, de bekleding van de stoelen moet anders, etc. Al deze wijzigingen moeten doorgevoerd worden. Als een organisatie alle processen van grondstof tot eindproduct beheerst kan dit heel lang duren. De organisatie kan hierdoor minder snel inspelen op veranderingen van de markt [HUG06]. Daarom is een organisatie zoals Ford Motor Company tegenwoordig ondenkbaar.

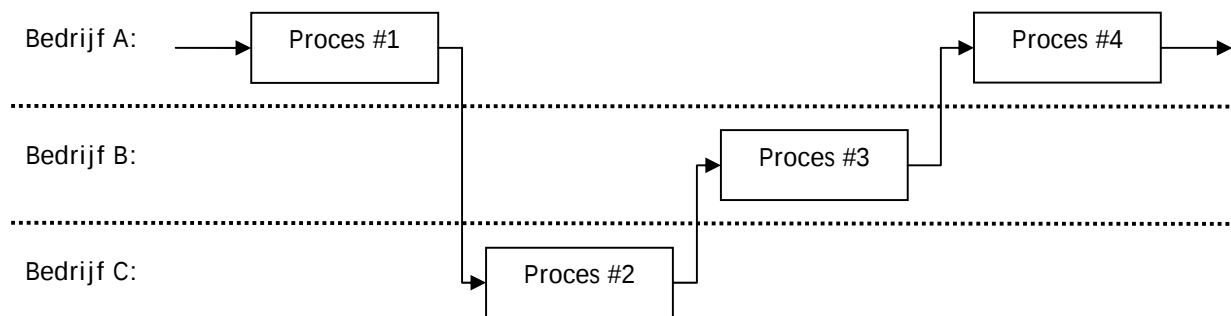
De snelheid waarmee een organisatie kan reageren op veranderingen in de markt noemt men de *flexibiliteit* van een organisatie. Om als organisatie flexibel te zijn, zijn kennis en vaardigheden nodig. Een organisatie heeft drie mogelijkheden om aan extra kennis en vaardigheden te komen [RAZ98]:

- A. De kennis en vaardigheden zelf ontwikkelen;
- B. Een organisatie overnemen die de kennis en vaardigheden al bezit;
- C. De processen, die de kennis en vaardigheden nodig hebben, uitbesteden aan een dienstverlenend bedrijf.

De kennis en vaardigheden die zelf zijn ontwikkeld (optie A) kunnen meteen worden geïntegreerd in bestaande processen. Daarbij bepaalt de organisatie zelf welke kant ze opgaat met deze kennis en vaardigheden. Het grote nadeel is echter dat het ontwikkelen van kennis en vaardigheden erg duur kan zijn en lang kan duren. Ook kan men kiezen om zich te ontwikkelen in kennis en vaardigheden die op een dood spoor terecht komen. De HD DVD standaard van Toshiba is daar een voorbeeld van. Bij snelle veranderingen in de markt moet een organisatie snel zijn bedrijfsprocessen kunnen wijzigen. Als de kennis of vaardigheid die nodig is voor een bedrijfsproces nog niet aanwezig is en door de organisatie zelf ontwikkeld moet worden, kan dit lang duren. De flexibiliteit van het bedrijf blijft hierdoor laag.

Het voordeel van een organisatie overnemen die de kennis en vaardigheden al wel bezit (optie B) is dat de kennis en vaardigheden meteen toegepast kunnen worden. De twee organisaties moeten echter wel met elkaar geïntegreerd worden. Ook moet een investering gedaan kunnen worden om het andere bedrijf over te nemen.

De laatste optie is om een (deel van een) bedrijfsproces *uit te besteden*. Hierbij wordt een deel (hoofdproces) of zelfs het gehele proces (megaproces) overgedragen aan een andere organisatie.



figuur 3: *Uitbesteden van processen*

Bij het uitbesteden is het van belang dat er, op de uitbesteding na, geen nadelige gevolgen optreden voor de eindconsument. Van buitenaf beschouwt lijkt het alsof de stroom van goederen (*goederenstroom*) hetzelfde is gebleven. Het uitbesteden van processen aan dienstverleners heeft als nadeel dat er geen directe invloed uitgeoefend kan worden op de koers van de dienstverlener. Echter, de handen blijven vrij om te focussen op de overige processen van de organisatie [PLO02, RAZ98, BOL03]. Men kan zich richten op de bedrijfsprocessen waarin de organisatie zich onderscheidt van zijn concurrenten. Men focust zich op de *kernactiviteiten* van een organisatie. Bij uitbesteden is een organisatie daarbij vrij om een andere partner te kiezen (op contractuele voorwaarden na). Hierdoor kan er gemakkelijk worden overgeschakeld op andere of efficiëntere diensten, die meer waarde toevoegen. Het uitbesteden van processen vergroot dus de flexibiliteit van een bedrijf doordat het bedrijf sneller kan reageren op veranderingen in de markt. Verder kan het uitbesteden van processen vele malen goedkoper zijn dan zelf het proces te ontwikkelen, te onderhouden of het hele proces te kopen en te integreren met bestaande processen [RAZ98].

### 2.3. Supply Chain Management

Supply Chain Management (SCM) kunnen we als volgt definiëren:

Definitie 2: Supply Chain Management

*“De coördinatie van verwerven, verwerken en verzenden van goederen in een netwerk van bedrijven met als doel de beste combinatie te vinden tussen de efficiëntie en de reactietijd van het netwerk.” - [HUG06]*

SCM richt zich op een hele keten van bedrijven, van grondstof tot de levering aan de eindconsument. SCM zou daarom eigenlijk geen megaproces moeten zijn, maar een overtreffende trap daarvan. In deze scriptie beschouwen we SCM echter als een megaproces. De hoofdprocessen binnen SCM zijn het *verwerven* van grondstoffen of onderdelen voor een product, het *verwerken* van deze grondstoffen of onderdelen tot een (deel)product en het *verzenden* van dit (deel)product [VLI98]. Deze drie hoofdprocessen worden aangestuurd op basis van informatie. Alle informatie die betrekking kan hebben op één van deze drie hoofdprocessen is relevant. SCM heeft om die reden ook raakvlakken met andere megaprocessen zoals financiën, productontwikkeling en marketing. Al deze megaprocessen genereren informatie die van invloed kan zijn op de hoofdprocessen binnen SCM.

SCM heeft erg veel raakvlakken met het megaproces *logistiek*. Logistiek wordt toegespitst op een enkele organisatie. Hierdoor hoeft er minder rekening gehouden te worden met allerlei trends in de markt die mogelijk van invloed kunnen zijn op partnerorganisaties. Echter, de risico's zijn ook groter wanneer slechts de organisatie wordt beschouwd. SCM wordt gezien als de uitgebreide versie van logistiek.

Het soepel laten verlopen van een supply chain krijgt een steeds hogere prioriteit [AKK03]. Organisaties proberen zo min mogelijk goederen op voorraad te hebben. Dit heeft voordelen omdat er dan minder opslagruimte nodig is. Tevens hoeven er minder artikelen geretourneerd te worden op het moment een artikel uit de verkoop wordt gehaald. Om zo min mogelijk artikelen op voorraad te hebben, is het belangrijk dat goederen op het juiste moment worden geleverd (*Just-in-Time management*). Dit vereist een hoge mate van integratie tussen de bedrijven in de supply chain om processen goed op elkaar af te stemmen. Tegelijkertijd vereist dit ook een IT-huishouding die goed kan insprijngen op veranderingen in de procesorkestratie. Het verschaffen van de juiste informatie aan de leden in de supply chain is van groot belang.

### 2.4. Supply chain strategieën

Een *strategie* van een bedrijf is een plan waarin staat vastgelegd welke weg het bedrijf op wil gaan. Het doel is om een betere positie op de markt te verwerven [WIT04]. Strategieën zijn er op meerdere niveaus. Zo kan een strategie voor een afdeling worden geformuleerd, een strategie op het niveau van een organisatie, maar ook voor een hele supply chain. In de volgende paragraaf behandelen we de achterliggende ideeën voor supply chain strategieën.

Een meubelmaker maakt tafels. Hij maakt tafels die drie kleuren hebben: groen, rood en blauw. Omdat hij niet weet hoeveel tafels van elke kleur hij zal verkopen, maakt hij van elk 100 stuks. Eenmaal in de verkoop, blijkt de blauwe tafel erg in trek te zijn. De meubelmaker moet daarom nog 50 extra blauwe tafels maken. De groene tafels daarentegen verkopen minder goed. Na een evaluatie blijkt dat er 150 blauwe tafels zijn verkocht, 100 rode en 50 groene. De meubelmaker blijft met 50 groene tafels zitten. Om dit in het vervolg te voorkomen, gaat de meubelmaker proberen te *voorspellen* hoeveel van elke kleur tafel er verkocht gaat worden. Dit doet hij aan de hand van meningen van zijn kopers, op basis van verkoop cijfers, etc.

Hierboven is natuurlijk slechts een voorbeeld. Maar dit is wel hoe een supply chain zonder goede afstemming tussen de betrokken organisaties werkt. Men probeert aan de hand van allerlei indicatoren een voorspelling te maken van het aantal goederen dat verkocht gaat worden. De kans dat een voorspelling niet precies uitkomt, is natuurlijk groot. Het gevolg is dat de bedrijven in de supply chain met veel goederen blijven zitten. Daarbij zijn bedrijven vaak geneigd om net iets meer te produceren

dan gevraagd wordt, om te kunnen voldoen aan een mogelijke piek in de vraag. Als de eindconsument om 10 producten vraagt, zal de leverancier aan het eind van de keten 12 producten produceren. Hiervoor zijn onderdelen nodig voor 12 producten van de leverancier van de leverancier. Maar die maakt ook een overschatting: die produceert onderdelen voor maar liefst 15 producten. En hoe verder deze order teruggaat in de keten, hoe meer onderdelen er geproduceerd worden. Als de eindconsument uiteindelijk zijn 10 producten koopt, blijft de hele keten met onderdelen zitten. Dit wordt het *bullwhip-effect* [PAG98, YAN03] genoemd.

Een manier om deze inefficiëntie tegen te gaan, is om pas goederen te produceren als er een order binnenkomt. Hierdoor wordt altijd de juiste hoeveelheid producten geproduceerd. Echter, op het moment dat de eindconsument een order plaatst, moet deze order eerst helemaal teruglopen door de supply chain tot aan het begin. Daarna pas komt de order weer via de hele supply chain terug bij de eindconsument. Hierdoor duurt het lang voordat een order geleverd wordt. Door de orders niet via alle tussenliggende bedrijven te laten gaan, maar meteen bij de juiste producent neer te leggen, kan er aan tijd worden gewonnen. Hiervoor dienen afspraken met alle betrokken leden in de supply chain te worden gemaakt.

Een andere manier is het produceren van *halffabricaten*. Als voorbeeld nemen we de meubelmaker. De meubelmaker kan ook kiezen om de tafels te bouwen, maar nog niet te verven. Op het moment er een blauwe tafel wordt besteld, hoeft de meubelmaker slechts de tafel blauw te verven. Hierdoor blijft de meubelmaker niet met een grote hoeveelheid groene tafels zitten. Het uitstellen van het geven van een aantal unieke kenmerken aan producten wordt ook wel *postponement* (vertaald: uitstellen) genoemd. Duidelijke voorbeelden hiervan zijn specifieke kleuren verf (worden meestal op maat gemengd in de winkel met basiskleuren) en computers (de hardware onderdelen worden allemaal pas bij elkaar gezet bij een bestelling). Op het gebied van transport bestaat ook een dergelijk concept. Men kan van te voren bedenken in welk gebied hoeveel van een bepaald product wordt afgenomen. Ook kan men op een centrale plaats grote hoeveelheden opslaan en pas verschepen bij een order. Dit laatste duurt langer, maar heeft als voordeel dat producten niet retour hoeven als er geen vraag meer naar een bepaald product is.

De meubelmaker ontkomt er echter niet aan om een schatting te maken van het totale aantal te verkopen tafels. Dit is in het begin van de supply chain. De hoeveelheid halffabricaten zal worden geschat, het aantal eindproducten wordt op basis van orders gemaakt. Het omslagpunt, waar wordt overgeschakeld van voorspellen naar vraag van klanten, wordt het *klant order ontkoppelingspunt* genoemd.

Het uitstellen van productieprocessen, zoals het voorbeeld van de ongeverfde tafels, wordt door Pagh & Cooper [PAG98] *manufacturing postponement* genoemd. Het uitstellen van transport door gebruik te maken van centrale opslagplaatsen wordt door Pagh & Cooper *logistics postponement* genoemd.

## **2.5. Koppeling processen aan applicaties**

Veel activiteiten op het activiteitsniveau (zie paragraaf 2.2) binnen processen worden ondersteund of zelfs uitgevoerd door *applicaties*. De definitie van applicatie die in deze scriptie gehanteerd wordt, is de volgende:

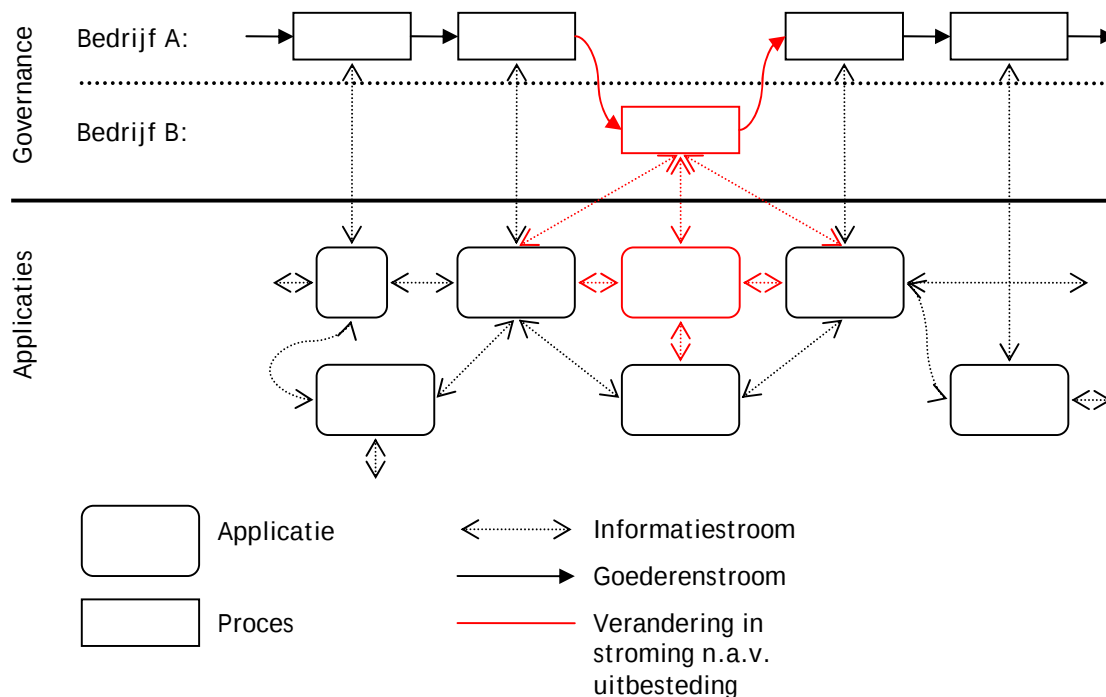
Definitie 3: applicatie

*“Een geheel of arrangement van elementen die zo zijn georganiseerd om door middel van het verwerken van informatie een bepaald doel te bereiken”* - [PRE01]

In deze scriptie beschouwen we ‘systeem’ als een synoniem voor applicatie. Als processen worden uitbesteed aan dienstverleners, dan moeten processen van beide bedrijven vaak met elkaar worden geïntegreerd. Kijkend naar het activiteitsniveau van processen, dan moeten ook de activiteiten geïntegreerd worden. Als een activiteit informatie vereist over een andere activiteit die door de uitbesteding door een andere organisatie wordt uitgevoerd, moeten de applicaties, die ondersteuning bieden aan de activiteiten, ook met elkaar geïntegreerd worden (zie figuur 4). Doordat er erg diverse

applicaties bestaan, is het koppelen van de applicaties niet triviaal. Applicaties wisselen onderling berichten aan elkaar uit. De stroom van berichten die op deze manier ontstaat, wordt de *informatiestroom* genoemd. In de supply chain wordt de informatie uit de informatiestroom gebruikt om de goederenstroom te sturen. De berichten die tussen de applicaties worden uitgewisseld, kan een heel specifieke vorm hebben. Applicaties van andere softwarebedrijven kunnen deze berichten vaak niet lezen. Hiervoor dient eerst weer een oplossing geschreven te worden. Dit duurt vaak weer een tijd en kost geld, wat niet ten goede komt aan de flexibiliteit. Ook komt het regelmatig voor dat systemen die min of meer dezelfde processen ondersteunen naast elkaar draaien. Dit is een gevolg van overnames en klantspecifieke eisen. In de praktijk komen deze knelpunten vaak voor [AKK03, CHA04, DAM01].

In figuur 4 is te zien dat bedrijf A een proces uitbesteed aan bedrijf B (het proces is rood gekleurd). Stel dat bedrijf B een vervoersbedrijf is dat zich niet bezighoudt met opslag, dan moet bedrijf A informatie sturen waarin staat waar welke goederen heen moeten. Bedrijf A wil graag zijn goederen kunnen volgen. Een ondersteunende applicatie bij bedrijf B moet daarom ook informatie over de locatie van de goederen geven aan de applicaties van bedrijf A. Op het moment bedrijf A om welke reden dan ook een wijziging in onderstaand schema wil aanbrengen, moeten alle koppelingen die rood zijn gekleurd in ieder geval verlegd worden. Als de applicaties dan niet communiceren via berichten waarvan de vorm bekend is, moet weer een extra oplossing gemaakt worden per koppeling.



figuur 4: *Procesintegratie en applicatieïntegratie*

Een terugkerende vraag bij het uitbesteden van processen is hoe de applicaties aan de processen gekoppeld worden en welke applicaties gebruikt worden. Worden de bestaande applicaties van bedrijf A gebruikt, worden de applicaties van bedrijf B gebruikt of wordt er een geheel nieuwe applicatie gebouwd ter ondersteuning van processen? In hoofdstuk 3 wordt dit uitgewerkt voor processen die uitbesteed worden aan logistieke dienstverleners.

### 3. Logistieke dienstverleners

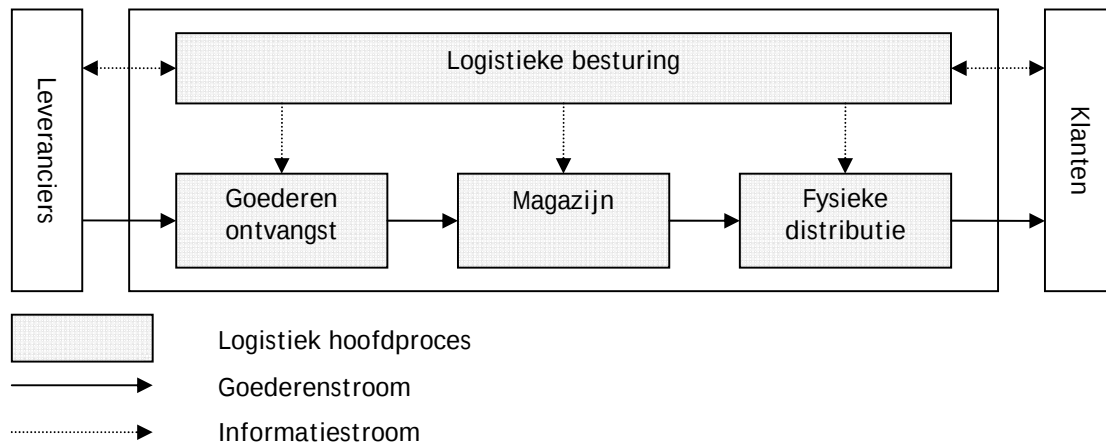
In het volgende hoofdstuk wordt de rol van logistieke dienstverleners in de supply chain duidelijk gemaakt. Dit wordt gedaan door de processen van een logistieke dienstverlener te beschrijven. Hieruit volgt een definitie. Vervolgens komen strategieën voor logistieke dienstverleners aan bod. Deze strategieën zijn gebaseerd op de supply chain strategieën (zie hoofdstuk 2.4). Door de strategieën te beschrijven worden de richtingen die logistieke dienstverleners op kunnen gaan duidelijk gemaakt. Uit de strategieën volgen een aantal implicaties voor veranderingen van het IT landschap van logistieke dienstverleners. Bij deze implicaties zijn een aantal knelpunten te identificeren. De knelpunten belemmeren de richting een logistieke dienstverlener op wilt gaan. Deze worden behandeld in de laatste paragraaf, en worden in hoofdstuk 5 gebruikt om het theoretische model op te stellen. De knelpunten zullen gekoppeld worden aan de karakteristieken van SOA.

#### 3.1. Definitie

Logistiek vindt zijn oorsprong in de krijgsmacht. Al ten tijden van de Grieken, maar ongetwijfeld ook daarvoor, werd de bevoorrading van een leger als belangrijk onderdeel gezien die de uitslag van een veldslag kon bepalen. Uit geschriften blijkt dat Alexander de Grote een groot deel van zijn overwinningen te danken had aan een goede bevoorrading van zijn leger. Als het leger na een overwinning verder reisde naar een ander gebied, was het van groot belang dat de bevoorrading daarop werd aangepast. Het bedrijfsleven heeft dit concept in de twintigste eeuw opgepakt. Logistieke dienstverleners zijn organisaties die zijn gespecialiseerd in het leveren van diensten met betrekking tot logistieke processen. Oorspronkelijk was dit vooral vervoer en opslag, maar deze twee hoofdprocessen worden steeds meer gecombineerd en ook aanverwante processen worden uitbesteed aan logistieke dienstverleners.

Een logistieke dienstverlener heeft te maken met één of meerdere partijen. Dit is de reden dat logistiek in het Engels ook wel *Third-Party Logistics (TPL, 3PL of contract logistics)* wordt genoemd [RAZ98]. De logistieke dienstverlener produceert zelf niets om te vervoeren of om op te slaan. Zodoende heeft een logistieke dienstverlener een opdrachtgever nodig. Deze opdrachtgever wordt *verlader* genoemd. De verlader is een bedrijf dat goederen naar zijn klant wil brengen. Hiervoor heeft de verlader zelf niet de middelen of kennis. Ook kan het zijn dat de verlader andere prioriteiten heeft gesteld. De verlader maakt daarom gebruik van de diensten van de logistieke dienstverlener. De verlader kan overigens ook zelf zijn eigen klant zijn. Zo kan een oliemaatschappij olie winnen in de buurt van Zuid-Amerika en een logistieke dienstverlener inschakelen om de ruwe olie te vervoeren naar zijn eigen raffinaderij in Rotterdam. Hoe nauwer de samenwerking tussen de logistieke dienstverlener en de andere betrokken organisaties is, hoe meer de logistieke dienstverlener bij de supply chain betrokken raakt. Dit wordt verder uitgediept in paragraaf 3.3.

De hoofdprocessen van logistieke dienstverleners verschillen van organisatie tot organisatie. Dit ligt aan de specialisatie van de organisatie. Als een logistieke dienstverlener alleen het opslaan van goederen als dienst aanbiedt, dan zal er geen hoofdproces *fysieke distributie* zijn. Voor deze scriptie hanteren we echter de meest algemene vorm. Als *primaire hoofdprocessen* (processen die direct invloed hebben op de goederenstroom) worden de processen *goederenontvangst*, *magazijn* en *fysieke distributie* gebruikt [VLI98]. Als *ondersteunend hoofdproces* wordt *logistieke besturing* gebruikt.



figuur 5: Megaproces logistiek [VLI98]

### Goederenontvangst

Onder het hoofdproces *goederenontvangst* worden alle fysieke en administratieve activiteiten die voortvloeien uit de aanvoer van goederen verstaan. Pas als de goederen in de voorraad zijn opgenomen, kunnen de goederen worden geleverd aan de klant. De deelprocessen *lossen/ uitpakken*, *controleren* en *ompakken* vallen onder het hoofdproces goederenontvangst. Het komt nogal eens voor dat logistieke dienstverleners de *inkoop van goederen* van de verlader overnemen. Het hoofdproces inkoop gaat vooraf aan goederenontvangst. Deze twee samen kunnen gerekend worden tot het *verwerven* van goederen.

### Magazijn

Onder het hoofdproces *magazijn* worden alle uitvoerende activiteiten die samenhangen met *inslag van goederen in de voorraad*, *inventarisatie van aanwezige voorraad* en *orderverzamelen* verstaan. In dit hoofdproces wordt gezorgd dat duidelijk is waar welke goederen liggen zodat deze op de juiste locatie gebracht kunnen worden voor verscheping naar de klant. Extra diensten, verzameld onder de naam *value-added services*, vallen ook onder het hoofdproces magazijn. Diensten als assemblage, producten in landspecifieke verpakkingen doen en reparatie van goederen zijn voorbeelden van deze value-added services. Deze diensten zitten al heel dicht tegen het produceren van goederen aan. Het kan namelijk goed zijn dat voor het verpakken van goederen of het assembleren van halffabrikaten tot een eindproduct grondstoffen nodig zijn die niet worden aangeleverd door de verlader, maar afkomstig zijn uit een andere bron. Een producent van computermuizen zal zich hoogstwaarschijnlijk niet bezig houden met het karton waarin de muizen verpakt worden. Dit karton is afkomstig uit een andere supply chain, waar de logistieke dienstverlener ook onderdeel van uitmaakt. Voor een logistieke dienstverlener is het hoofdproces magazijn met aanverwante processen het *verwerken* van goederen.

### Fysieke distributie

Het hoofdproces *fysieke distributie* wordt gedefinieerd als alle uitvoerende activiteiten met als doel het afleveren van het eindproduct bij de klant. In dit hoofdproces worden juiste orders samengesteld en de goederen naar de klant gebracht. De deelprocessen *opslaan eindproduct*, *samenstellen orders/zendingen*, *verzendgereed maken orders* en *extern transport* vallen onder het hoofdproces fysieke distributie. Ook processen als custom services vallen onder dit hoofdproces. Fysieke distributie wordt ook wel *verzenden* genoemd.

### Logistieke besturing

Het hoofdproces *logistieke besturing* is een ondersteunend proces voor de overige drie hoofdprocessen. Door middel van informatie worden de primaire hoofdprocessen aangestuurd. De deelprocessen *voorraadbeheer*, *vraagvoorspelling*, *inkoopplanning*, *productieplanning*, *distributieplanning* en

*magazijnplanning* vallen onder het hoofdproces logistieke besturing. Afhankelijk van de diensten die de logistieke dienstverlener levert worden deze deelprocessen uitgevoerd.

In de literatuur zijn veel verschillende definities te vinden over logistieke dienstverleners en wat ze precies doen. Een belangrijke oorzaak is dat er onduidelijkheid over het woord logistiek bestaat. De Amerikaanse betekenis van *logistics* gaat uit van voornamelijk transport, opslag en aanverwante zaken. De Nederlandse definitie van logistiek is veel breder. Het produceren van goederen wordt ook vaak tot logistiek gerekend. In deze scriptie beperken we ons tot de hoofdprocessen die eerder deze paragraaf zijn beschreven als taak van een logistieke dienstverlener. Dit neigt meer naar de Amerikaanse definitie van *logistics*. We gaan uit van de volgende definitie:

Definitie 4: logistieke dienstverlener

*“Een organisatie die zijn werkzaamheden primair heeft liggen in het verwerven, verwerken en verzenden van goederen, en dit in de vorm van diensten aanbiedt aan andere organisaties.”*

### 3.2. Strategieën

De strategieën van logistieke dienstverleners zijn in grote mate geïnspireerd door de ideeën uit de supply chain. Als een verlader besluit om halffabrikaten te produceren, en deze pas bij bestelling te assembleren tot een eindproduct, dan heeft dit grote invloed op de inrichting van de logistieke processen. Als deze logistieke processen worden uitbesteed aan een logistieke dienstverlener, dan heeft dit ook invloed op de handelswijze van de logistieke dienstverlener, en daarmee ook op de strategieën die de logistieke dienstverlener hanteert. Zoals gesteld in hoofdstuk 2.4, maken we in deze scriptie onderscheid tussen *manufacturing postponement* en *logistics postponement*. Bask [BAS01] heeft drie strategieën voor logistieke dienstverleners afgeleid uit het concept *logistics postponement*. Definitie 4 sluit goed aan bij deze strategieën, omdat hierbij het produceren van goederen buiten beschouwing wordt gelaten. Deze drie strategieën heeft Bask gekoppeld aan drie type diensten die logistieke dienstverleners kunnen leveren. Deze nieuwe strategieën tonen gelijkenissen met de drie generieke strategieën van Porter [WIT04]. Dankzij de strategieën ontstaat er een typering voor logistieke dienstverleners. Dit stelt ons in staat om uitspraken te doen over de relatie tussen de verschillende types logistieke dienstverleners en SOA.

#### Directe strategie

De *directe strategie* wordt gekenmerkt door transport rechtstreeks van de verlader naar winkels of klanten, met eventueel een tijdelijke opslag. Dit kan het geval zijn met eindproducten, of met halffabrikaten die in de winkel geassembleerd worden. Een voorbeeld hiervan is de levering van groentes aan supermarkten, of het aanleveren van basiskleuren verf die in de doe-het-zelfzaak in de juiste verhouding gemengd worden voor de klant.

#### Stop-over strategie

De *stop-over strategie* kenmerkt zich door gebruik te maken van gespecialiseerde gedecentraliseerde opslagcentra, of juist van gecentraliseerde opslagcentra waarbij het uitgebreide netwerk van een logistieke dienstverlener kan zorgen voor kostenverlaging. Gedecentraliseerde opslagcentra zijn minder efficiënt dan gecentraliseerde opslagcentra, doordat transport op basis van voorspellingen worden uitgevoerd. Deze inefficiëntie kan gecompenseerd worden door de opslagcentra voor meerdere verladers/klanten te gebruiken. In gecentraliseerde opslagcentra kunnen goederen over een grotere regio worden verspreid. Hierdoor wordt de voorraad bij bijvoorbeeld winkels verkleind zonder dat risico wordt gelopen dat een product niet meer te verkrijgen is.

#### Assembled-stop-over strategie

De *assembled-stop-over strategie* kenmerkt zich door een sterk gecentraliseerd distributienetwerk. Dit heeft als voordeel dat op een centrale plaats een heleboel value-added services uitgevoerd kunnen worden. Processen zoals labelen, assemblage en verpakken vinden eerst plaats, voordat het eindproduct verzonden wordt. In een gedecentraliseerd netwerk van opslagcentra is dit erg inefficiënt,

omdat men van te voren niet precies weet hoeveel en welke goederen waar heen moeten. De kans dat goederen, eventueel al verpakt en geassembleerd, retour gaan is dan erg groot.

De strategieën van logistieke dienstverleners worden ook sterk beïnvloed door het type diensten dat logistieke dienstverleners aanbieden. Als een organisatie louter en alleen goederen vervoerd van locatie A naar locatie B, ziet de strategie er compleet anders uit dan van een bedrijf die het gehele megaproces logistiek overneemt van een andere organisatie, met hoofdprocessen zoals inkoop, opslag, assemblage, verpakking en vervoer. Ieder type dienst heeft een bepaalde *klanthechtheid*. Het voorbeeld van simpel goederen vervoeren is gebaat bij een kleine klanthechtheid. Dit omdat een nauwe verbondenheid met de verlader/klant extra geld kost, maar geen extra waarde toevoegt aan de dienst. Op het moment een organisatie het hele megaproces logistiek uitbested aan een logistieke dienstverlener, is samenwerking juist van groot belang om het uitbesteden te doen laten slagen. Anders wordt de relatie gekenmerkt door kwaliteitsproblemen. Een logistieke dienstverlener zal een keuze moeten maken wat voor een soort diensten ze aan wil bieden om niet te verzanden in inefficiëntie.

In deze scriptie gaan we uit van drie type diensten van logistieke dienstverleners. Deze komen tot stand door de klanthechtheid tussen logistieke dienstverlener en verlader (en eventueel de klant van de verlader) uit te zetten tegen de complexiteit van de diensten die logistieke dienstverlener aanbiedt (zie figuur 6).

### **Routine klus**

*Routine klussen* zijn diensten waarover geen speciale afspraken gemaakt hoeven te worden. De diensten zijn winstgevend juist vanwege dit routinematige karakter. De redenen om gebruik te maken van deze diensten zijn de lage prijzen, de betrouwbaarheid en de eenvoud om een overeenkomst aan te gaan. Onder routine klussen vallen simpele diensten zoals transport en opslag.

### **Standaard dienst**

*Standaard diensten* zijn diensten die iets meer afspraken vergen als routine klussen, maar nog steeds op redelijk grote schaal uitgevoerd kunnen worden. Diensten zoals het vervoeren van gekoelde producten, producten die warm moeten blijven, vervoer in gepantserde vrachtwagens of het uitsorteren van producten in een distributiecentrum voor een verlader/klant vallen hieronder.

### **Dienst op maat**

*Diensten op maat* zijn diensten die in samenspraak met de verlader ontwikkeld zijn. Vanzelfsprekend is de klanthechtheid hierbij erg groot. Ook dient er veel openheid te bestaan wat betreft het uitwisselen van informatie. Dit type diensten is meestal erg complex. De meeste processen kunnen niet op een standaard manier worden uitgevoerd, en hangen nauw samen met andere processen. Processen zoals product assemblage, verpakking (van bijvoorbeeld de juiste computeronderdelen in de juiste doos stoppen), reparatieservices en after-sales services horen bij dit type diensten. De overeenkomst tussen logistieke dienstverlener en verlader/klant is hierbij vaak van langere duur dan het geval bij standaard diensten en routine klussen. Dit omdat de samenwerking een grote investering vergt.

|                |       | Complex             | Simpel           |
|----------------|-------|---------------------|------------------|
| Klanthechtheid | Groot | Dienst op maat      | Te hoge prijzen  |
|                |       |                     | Standaard dienst |
|                | Klein | Kwaliteitsproblemen | Routine klus     |

figuur 6: *Diensten matrix* [MAK90]

De strategieën afgeleid uit het concept postponement zijn door Bask gekoppeld aan de type services van Makelin & Vepsalainen [MAK90]. Hierdoor ontstaan *drie generieke strategieën voor logistieke dienstverleners*: de *assembled-stop-over strategie met diensten op maat*, de *stop-over strategie met standaard diensten* en de *directe strategie met routine klussen*. Op het moment dat een logistieke dienstverlener diensten op maat levert terwijl de strategie zich vooral focust op directe levering aan winkels en klanten (directe strategie), zullen er kwaliteitsproblemen ontstaan. Routine klussen gecombineerd met een gecentraliseerde opslagplaats leveren vooral hoge transactiekosten op, en een lange leveringstijd. Het is dus belangrijk dat het juiste type dienst en de juiste strategie worden gecombineerd.

### 3.3. Implicaties en knelpunten van strategieën

De drie generieke strategieën voor logistieke dienstverleners van Bask hebben gevolgen voor de inrichting van de processen van logistieke dienstverleners, en daarmee ook op de applicaties die de processen ondersteunen. In de volgende paragraaf wordt per strategie bekeken wat voor implicaties de strategie heeft voor de IT-huishouding van de logistieke dienstverlener, en welke knelpunten hierbij optreden.

#### Implicaties directe strategie met routine klussen

De directe strategie met routine klussen kenmerkt zich door de kleine klanthechtheid en de eenvoudige routine klussen. Logistieke dienstverleners die deze diensten aanbieden focussen zich vooral op het zo *goedkoop mogelijk leveren* van diensten, *betrouwbaarheid* op het leveren van goederen en de *eenvoud om een overeenkomst aan te gaan*. Door de geringe loyaliteit maakt het verladers weinig uit van welke logistieke dienstverlener zij gebruik maken, als het maar onder de beste voorwaarden gebeurt. Om deze reden willen verladers de overeenkomsten zo kort mogelijk houden, zodat snel gewisseld kan worden naar een logistieke dienstverlener met betere voorwaarden. Hierdoor zijn de marges ook erg klein. Logistieke dienstverleners doen niet teveel moeite aansluiting te vinden met de IT van de verlader, ze kunnen immers ieder moment weer aan de kant gezet worden.

In het meest ideale geval zou een verlader niet op zoek hoeven te gaan naar een logistieke dienstverlener. Door middel van een applicatie zou automatisch een logistieke dienstverlener gekozen kunnen worden. Hierbij is geen menselijke handeling nodig. Dit scheelt aanzienlijk in de kosten. Communicatie waarbij geen menselijke handelingen aan te pas komen, wordt *machine-to-machine (m2m)* genoemd. Voor m2m is het essentieel dat de applicaties van de participerende organisaties met elkaar kunnen communiceren. De koppeling van meerdere applicaties is vaak een lastige zaak (zie hoofdstuk 2.5). Om dit eenvoudiger te maken, zijn *gestandaardiseerde interfaces* en het gebruik van *open standaarden* nodig. De *interface* van een applicatie is het gedeelte waarmee de applicatie bestuurd kan worden. Door de interface op een standaard manier aan te bieden, kunnen verschillende logistieke dienstverleners bij verschillende verladers hun diensten aanbieden. Een open standaard is een specificatie van in dit geval een bericht dat voor iedereen in te zien is. Door gebruik te maken van

open standaarden, is iedereen in staat om berichten naar een applicatie te sturen, mits geautoriseerd. Als dit het geval is, spreekt men van een gestandaardiseerde interface van de applicatie. Bij het gebruik van open standaarden en gestandaardiseerde interfaces is het van belang dat zowel de *vorm* van het bericht (syntax) als de *betekenis* (semantiek) eenduidig is. M2m communicatie is niet alleen van belang bij het vinden en selecteren van een partner. Ook voor het opereren met partners kan m2m uitkomst bieden. De communicatie via m2m verloopt veel sneller als communicatie waarbij menselijke handelingen vereist zijn. Feedback over de routine klussen kan veel sneller worden verstrekt bij het gebruik van m2m. Hierdoor is het voor logistieke dienstverleners mogelijk om realtime informatie te verstrekken. Open standaarden en gestandaardiseerde interfaces zijn nog lang niet overal geïntroduceerd. De integratie van applicaties van verlader en logistieke dienstverlener is een groot knelpunt bij deze strategie.

Door de geringe klanthechtheid, worden bij deze strategie de applicaties van de logistieke dienstverlener gebruikt ter ondersteuning van de uitbestede processen. Voor de m2m communicatie zijn aanpassingen nodig in de IT zowel aan de kant van logistieke dienstverleners als aan de kant van de verladers. Verladers hebben in deze samenwerking een grote macht. Ze kunnen gemakkelijk overstappen op een concurrent. Door de geringe klanthechtheid zal de logistieke dienstverlener niet heel happig zijn om zijn applicaties af te stemmen op een enkele verlader. Dit is immers een grote investering. Daarnaast zullen ze het aangepaste deel van hun systeem ook nog moeten onderhouden. Voor logistieke dienstverleners die routine klussen leveren zijn de kosten om te investeren in IT daarom ook een knelpunt.

#### **Implicaties stop-over strategie met standaard diensten**

De stop-over strategie met standaard diensten valt in tweeën uiteen: het gebruik van centrale opslagcentra en van decentrale opslagcentra voor goederen. In het eerste geval wordt een hoge mate van efficiëntie bereikt doordat de voorraden van bijvoorbeeld winkels klein gehouden kunnen worden. Bij het gebruik van decentrale opslagcentra kan een grote mate van efficiëntie worden bereikt door goederen van verschillende verladers op te slaan in hetzelfde (decentrale) opslagcentrum. Door de value-added services algemeen in te richten kunnen deze voor meerdere verladers gebruikt worden. Dit geldt zowel voor centrale als decentrale opslagcentra. De marges van logistieke dienstverleners die zich concentreren op standaard diensten zijn niet heel groot. De marges worden verdiend door de value-added services voor meerdere verladers in te zetten. Als knelpunten voor logistieke dienstverleners die standaard diensten leveren kunnen we net als bij de logistieke dienstverleners met routine klussen de integratie van applicaties en de hoge investering die nodig is in de IT voor het verbeteren van hun applicaties als knelpunt aanmerken.

Het volgen van goederen is een belangrijke value-added service. Systemen die dit kunnen worden *trackingsystemen* genoemd. Om trackingsystemen goed te laten werken, is het noodzakelijk dat zoveel mogelijk organisaties in de supply chain aangesloten zijn op het systeem. Ook dient de informatie zo actueel mogelijk te zijn om realtime tracking te benaderen. Hiermee kan de goederenstroom beter voorspeld en bestuurd worden. Bij dit systeem is het vooral van belang dat de logistieke dienstverlener realtime kan aangeven waar goederen zich bevinden. Dus ook hier kunnen we het knelpunt van integratie van applicaties identificeren. Ook is het van belang om *passief* informatie aan de rest van de supply chain te verschaffen, over de locatie van de goederen. Passief informatie verschaffen wil zeggen dat deze informatie alleen te lezen is, en niet te muteren. Het verschaffen van informatie moet op een discrete manier gebeuren. Alleen geautoriseerde organisaties mogen de informatie lezen. De manier waarop deze informatiestroom aangeboden wordt aan geautoriseerde organisaties of organisatieonderdelen is daarom ook een knelpunt.

#### **Implicaties assembled-stop-over strategie met diensten op maat**

De assembled-stop-over strategie met diensten op maat kenmerkt zich door de hechte relatie met de verlader. Door processen nauwkeurig op elkaar af te stemmen kunnen processen heel efficiënt uitgevoerd worden. De logistieke dienstverlener en de verlader worden in dit geval echt partners voor langere periode. De logistieke dienstverlener zorgt niet alleen voor de logistieke processen, maar kan ook waardevolle informatie verschaffen voor de indeling van andere processen. Door informatie tussen de organisaties open uit te wisselen (*transparantie*), kunnen organisaties van elkaar leren en waar

mogelijk processen bijsturen om de supply chain in zijn geheel te optimaliseren. In tegenstelling tot een trackingsysteem kunnen ook bijvoorbeeld orders geplaatst worden door de logistieke dienstverlener namens de verlader. Logistieke dienstverleners kunnen in dit geval dus ook *actief* mutaties doorvoeren. Doordat er bedrijfsgevoelige informatie verstuurd wordt die essentieel is voor het succes van een supply chain en van organisaties in het bijzonder, is de *beveiliging* van deze informatiestroom erg belangrijk. Het lekken van deze informatie kan desastreus zijn voor de concurrerende positie van de betrokken organisaties. Ook moeten de samenwerkende organisaties het vertrouwen van elkaar hebben om informatie te delen. Wantrouwen kan ontstaan als een logistieke dienstverlener twee klanten heeft die allebei in dezelfde branche opereren. De manier waarop de informatiestroom wordt aangeboden, wordt beveiligd (op zowel technologisch vlak als op het vlak van menseninteractie) en gemuteerd kan worden vormt een knelpunt bij logistieke diensten met diensten op maat.

Zoals al eerder gesteld kunnen markten erg veranderlijk zijn. Hierdoor dienen processen aangepast te worden (zie hoofdstuk 2.2), wat weer gevolgen heeft voor de ondersteunende applicaties (zie hoofdstuk 2.5). Door processen uit te besteden aan logistieke dienstverleners, kunnen verladers en klanten zich meer specialiseren in de overige processen. Echter, als er erg nauw wordt samengewerkt, en de processen van de verlader of klant worden aangepast aan de veranderende markt, heeft dit ook gevolgen voor de logistieke processen die worden uitbesteed aan de logistieke dienstverlener. Als de vorm van een product drastisch wordt gewijzigd, moet het transport van dat product ook worden gewijzigd. Men kan misschien minder goederen, of juist meer goederen tegelijk vervoeren. Dit heeft ook weer zijn uitwerking op de applicaties. De applicaties moeten mogelijk aangepast worden, anders worden ingericht of er moeten zelfs nieuwe applicaties geschreven worden. Het voordeel van het ontwikkelen van nieuwe applicaties is dat er met een schone lei begonnen kan worden. Het nadeel is dat er wel weer een nieuwe applicatie beheerd moet worden. Doordat er een heleboel nieuwe technologieën zijn geïntroduceerd in korte tijd [CHA04] en ook vanwege overnames van concurrenten [PLO02], kampen veel organisaties al met een uiterst complexe applicatiestructuur. Nieuwe applicaties toevoegen aan het landschap maakt de situatie alleen maar complexer. Als er veel verschillende applicaties zijn, moeten deze ook allen onderhouden worden. Wijzigingen doorvoeren in veel verschillende systemen is erg foutgevoelig, en kost veel tijd. Tevens kost het veel geld om de kennis over deze systemen in de organisatie te houden. Niet zelden is deze kennis ook verdwenen door uitbesteding of gewoon doordat werknemers weg zijn gegaan. Hierop wordt vaak besloten om een systeem niet uit te breiden, maar een heel nieuw systeem te bouwen, wat de complexiteit wederom laat toenemen. De redundantie van systemen is een groot knelpunt voor logistieke dienstverleners die diensten op maat bieden. Zowel voor het onderhoud alsook voor de kosten van het bouwen van een nieuw systeem. Vanzelfsprekend duurt het langer om een nieuw systeem van niets op te bouwen dan uit bestaande onderdelen. De ontbrekende flexibiliteit, het vermogen van een logistieke dienstverlener om zich aan te passen op veranderende processen, is een groot knelpunt.

Per strategie zijn er andere eisen en aandachtspunten voor het ondersteunende systeem. In hoofdstuk 4 gaan we verder in op de huidige bedrijfssystemen en op SOA, en wat SOA onderscheidt van de huidige bedrijfssystemen. In hoofdstuk 5 wordt gekeken hoe SOA wel of juist geen invulling geeft aan de knelpunten van logistieke dienstverleners.

## 4. Service Oriented Architecture

Het volgende hoofdstuk gaat over het software paradigma *Service Oriented Architecture (SOA)*. SOA is een concept dat is voortgevloeid uit verschillende stromingen. Deze stromingen worden in paragraaf 4.1 behandeld. Hierna worden verschillende *software paradigma's* behandeld. Dit deel sluit af met het paradigma *service-oriented*. Om te zien hoe het paradigma service-oriented in de praktijk gebruikt kan worden, wordt een model beschreven. Hieruit volgt een definitie van SOA. Het hoofdstuk sluit af met de karakteristieken van SOA zoals in de literatuur wordt verondersteld.

### 4.1. Opkomst bedrijfssystemen

Deze paragraaf behandelt de verschillende stromingen en ideeën waaruit het concept SOA ontstaan is. Eerst wordt ingegaan op de bedrijfskant van informatiesystemen, gevolgd door de ontwikkelingen in software. Hierna worden de ontwikkelingen in de hardware beschreven.

#### Bedrijfskant

In de jaren 60 van de vorige eeuw ontstonden de eerste bedrijfssystemen. Deze systemen werden vooral gebruikt voor het bijhouden van de voorraad. Met wat simpele berekeningen kon er een schatting worden gemaakt hoeveel materiaal voor een bepaalde order nodig was. Stel, men wil 20 fietsen maken. Dan zijn er 40 wielen nodig. Per wiel zijn er 20 spaken nodig. Voordat de wielen in elkaar gezet kunnen worden, zijn er 800 spaken nodig. Door de planning van het produceren van goederen te koppelen aan de hoeveelheid onderdelen die nodig zijn voor een product, kon men met behulp van deze simpele bedrijfssystemen berekenen wanneer en hoeveel onderdelen binnen moesten komen. Deze systemen noemde men *Material Requirements Planning (MRP) systemen* [JAC06]. In de jaren 60 en begin jaren 70 waren de markten erg stabiel [WIT04]. De MRP systemen voldeden goed aan de eisen van toen. In de jaren 70 ontstonden aparte systemen voor inkoop, productie en verkoop. De markt begon te veranderen. Er ontstond behoefte naar meer gedifferentieerde producten. Het bedrijfsleven wilde daarom sneller in kunnen springen op de vraag uit de markt. Het idee ontstond om de losse systemen voor inkoop, productie en verkoop aan elkaar te koppelen. Hierdoor werd het mogelijk om bijvoorbeeld meteen te kijken of een product op voorraad was op het moment een order binnenkwam. Zo niet, dan werd dit doorgegeven aan het systeem van inkoop. Deze systemen werden *Manufacturing Resource Planning (MRP II) systemen* genoemd [JAC06]. MRP II systemen werden rond de jaren 80 steeds verder verrijkt met allerlei andere onderdelen, met als bekendste voorbeeld *Human Resource Management (HRM)*. Ook deze nieuwe onderdelen werden gekoppeld aan de MRP II systemen. Hierdoor ontstonden grote onoverzichtelijke systemen, die bovendien veel onderhoud vergden. Alle losse systemen, weliswaar aan elkaar gekoppeld, hadden allemaal hun eigen database. Hierdoor was het niet vanzelfsprekend dat informatie over bepaalde processen realtime opgezocht konden worden. Besloten werd om te werken vanuit één database. Veranderingen gemaakt door het systeem dat bijvoorbeeld het proces verkoop ondersteunde werd hierdoor meteen zichtbaar voor het proces inkoop. Met een enkele database werd het mogelijk om een hele organisatie te voorzien van de informatie realtime. Deze nieuwe systemen werden *Enterprise Resource Planning (ERP) systemen* genoemd [JACOBS06]. Vooral in de jaren 90 nam het gebruik van ERP systemen toe, niet in de minste plaats vanwege de *millenniumbug* die in veel systemen zat [JAC06].

In het nieuwe millennium bestaat steeds meer behoefte om processen te integreren die over de muren van een organisatie heen lopen [AKK03]. Dit is een gevolg van het uitbesteden van veel processen aan meer gespecialiseerde bedrijven of naar lage lonen landen. Veel systemen worden weer aan elkaar geknoopt. Het probleem dat bestond voor de komst van ERP systemen (het gebruik van meerdere databases die niet synchroniseren), lijkt opnieuw te ontstaan. Alleen in dit geval speelt het probleem zich af op een andere schaal: namelijk die van de supply chain en niet van een enkele organisatie [AKK03]. Bedrijfssystemen die in staat zijn processen te ondersteunen die over de grenzen van een organisatie heen lopen worden gezien als de volgende stap naar ERP systemen.

### Software

In de jaren 60 waren de bedrijfssystemen allemaal maatwerk. Informatica stond nog in de kinderschoenen, dus heel vreemd was dat dan ook niet. Halverwege de jaren 70 begonnen bedrijven in te zien dat bedrijfsprocessen van verschillende bedrijven niet heel veel van elkaar verschilden. Door systemen iets algemener op te zetten, kon een systeem voor twee totaal verschillende branches gebruikt worden. Grote bedrijven als SAP, Oracle, Peoplesoft en JD Edwards ontstonden in deze periode. Bedrijf verkochten *pakketten* met daarin verschillende onderdelen om bedrijfsprocessen te ondersteunen [JAC06]. Omdat deze pakketten slechts eenmaal geschreven hoefden te worden en daarna konden worden hergebruikt, waren deze pakketten vele malen goedkoper als het schrijven van software op maat. Het gevolg was dat de bedrijfsprocessen aangepast werden aan het pakket. Dit was opmerkelijk, want hierdoor werd de software leidend in plaats van de bedrijfsprocessen, waarmee organisaties zich kunnen onderscheiden van concurrenten. Processen die afweken van het pakket werden of aangepast aan het pakket of met een hoop geld en tijd toch op maat gemaakt en geïntegreerd. Dit kwam echter niet vaak voor [AKK03]. In de jaren 90 wonnen ERP pakketten meer terrein. Deze bleven *monolithisch* van opzet, opdat de verschillende onderdelen goed op elkaar afgestemd konden worden. Organisaties kregen vaak een heleboel functionaliteit waar ze niet om gevraagd hadden, maar waar ze wel voor moesten betalen [AKK03].

Tegenwoordig zijn er al wel mogelijkheden om gedeeltes van een pakket te kopen. Hierdoor zitten bedrijven niet meer vast aan een heleboel overbodige functionaliteit. Organisaties zien echter liever dat deze gedeeltes van pakketten worden opgesplitst in nog kleinere, onafhankelijke basisonderdelen. Door te schuiven in deze basisonderdelen kan een pakket precies op maat worden gemaakt. Dat de basisonderdelen in een dergelijke configuratie dan van verschillende softwarehuizen afkomstig zijn, zou niet uit moeten maken [AKK03].

### Hardware

In de jaren 60 waren computers erg groot en had de software nog veel weg van machinetaal. IBM was destijds een grote speler van het bouwen van bedrijfssystemen. IBM bouwde computers (hardware), en software lag in het verlengde hiervan [JAC06]. In de jaren 70 werden de gegevens losgekoppeld van de applicaties. Men begon *databases* te ontwikkelen. Het grote voordeel van een database is dat de programmacode niet meer aangepast hoefde te worden als er veel dezelfde soort gegevens verwerkt moesten worden. Met de komst van databases werd het *mainframe model* geïntroduceerd. Op een mainframe stonden alle gegevens, en door middel van een terminal konden de gegevens bewerkt worden. In de jaren 80 deed de *Personal Computer (PC)* zijn intrede. Hiermee werd het mogelijk om berekeningen waar minder kracht voor nodig was op een kleinere, en goedkopere computer uit te voeren [JAC06]. Dit was het begin van de *Client Server Technology (CST)*. CST gaat uit van een mainframe waarop programma's draaien, de *server*. Een andere computer, meestal een minder krachtige, wordt de *client* genoemd. De client verstuurt een verzoek aan de server, de server verwerkt dit verzoek, en de client krijgt antwoord terug. De server kan meerdere clients bedienen. Door CST werden programma's gesplitst in een client- en serverdeel. Veel ERP pakketten zijn gebouwd op CST. De clientcomputers werden echter steeds krachtiger, waardoor de mainframes niet altijd meer nodig waren. Ook was het niet altijd even gemakkelijk om een programma op te splitsen in een client- en een serverdeel.

In de jaren 90 deed Internet zijn intrede. Geïnspireerd door het Internet, werd er steeds meer naar een Internet-achtige architectuur (voor architectuur, zie paragraaf 4.3) gestreefd. Dit wordt ook wel een *Peer-To-Peer (p2p) architectuur* genoemd. Iedere computer in het p2p netwerk kan zowel als server als client dienen. Koppelingen tussen computers kunnen ontstaan overal in het netwerk. Hierdoor is het gemakkelijker om applicaties te koppelen en te ontkoppelen. Een vereiste is wel dat alle applicaties gemakkelijk met elkaar informatie kunnen uitwisselen, iets wat nog niet vanzelfsprekend is. De interfaces van de applicaties verschillen vaak nog. Dit komt omdat de meeste bestaande applicaties, veelal een ERP pakket, door organisaties zijn gebouwd die hun eigen standaarden gebruiken.

Een stap de goede richting op is een *Enterprise Service Bus (ESB)* [SCH05]. Dit kan gezien worden als een grote snelweg die over de bestaande applicaties (de monolithische systemen) heen ligt. Iedere applicatie die nieuwe informatie tot zijn beschikking heeft, stuurt de informatie over deze snelweg.

Applicaties die geautoriseerd zijn om deze informatie te lezen, krijgen de informatie via deze snelweg binnen. Het is vanzelfsprekend dat de informatie op een afgesproken en herkenbare manier over deze snelweg wordt gestuurd. Het aanleggen van een ESB ziet men ook wel als het *ver-service-en* of het *SOA-enabled* maken van bedrijfssystemen. Een ESB neemt echter niet de redundantie van applicaties weg die veelal ontstaat als er meerdere pakketten gebruikt worden. In de volgende paragraaf gaan we verder in op het paradigma *service-oriented*, dat mogelijk een oplossing biedt voor dit probleem.

#### 4.2. Software paradigma's

Een *paradigma* kan omschreven worden als een overkoepelend denkraamwerk. Hierdoor is men in staat op een abstracte manier tegen problemen aan te kijken en daarvoor een oplossing te bedenken, binnen dit raamwerk. Een paradigma kan gezien worden als een model van de werkelijkheid, waarin details weggelaten worden voor de duidelijkheid. Wanneer er oplossingen worden gevonden die niet overeenkomen met de regels van het raamwerk, is er sprake van een *paradigmaverschuiving*. Een bekend voorbeeld van een paradigmaverschuiving is de constatering van Galileo Galilei, die de Zon centraal stelde in het zonnestelsel en niet de Aarde.

Software is vaak uiterst complex. Dit is het gevolg van een dynamische omgeving, de vele factoren waarvan de output van software afhankelijk is en de omvang van systemen. In de historie van de software ontwikkeling zien we daarom ook dat er verschillende paradigma's zijn ontstaan. Deze zijn bedoeld om de complexiteit te reduceren door software componenten opnieuw te gebruiken [DOB07]. In een software paradigma is er altijd een *behoefte* die door middel van een *entiteit* gekoppeld wordt aan een *oplossing*. De entiteit heeft een bepaald *vermogen* waarmee invulling gegeven kan worden aan de behoefte.

Definitie 5: entiteit

*"Een entiteit is iets dat wezenlijk bestaat. In dit geval een persoon, een organisatie(deel) of een systeem(deel)."* – [DOB07]

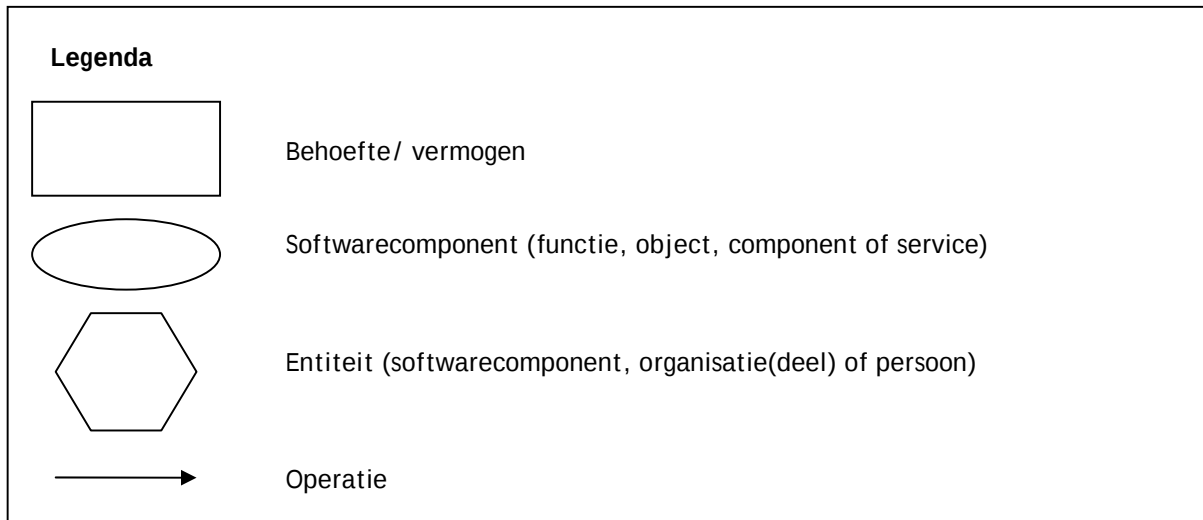
Definitie 6: behoefte

*"Een behoefte is iets dat een entiteit nodig heeft"* – [DOB07]

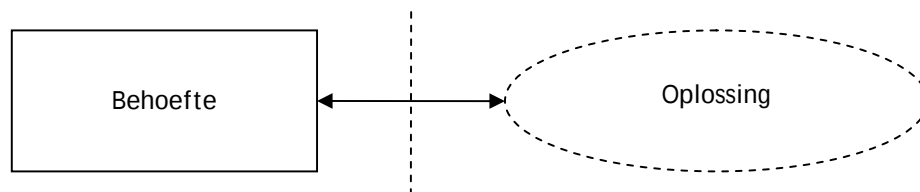
Definitie 7: vermogen

*"Een vermogen is dat waartoe een entiteit in staat is, een bekwaamheid"*  
– [DOB07]

Een software paradigma kan schematisch weergegeven worden als in figuur 8. In de volgende figuren staan de behoeftes steeds links van de verticale gestippelde streep, rechts de elementen die een oplossing representeren. De pijlen geven weer hoe de behoeftes en oplossingen van elkaar afhangen. Een pijl uit entiteit A richting een entiteit B geeft aan dat entiteit A gebonden is aan entiteit B. De legenda geldt voor alle overige figuren in dit hoofdstuk, tenzij anders aangegeven.



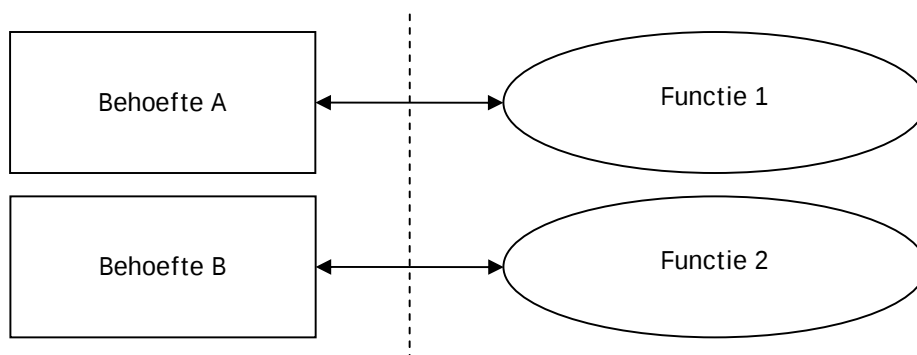
figuur 7: Legenda figuren hoofdstuk 4



figuur 8: Software paradigma (algemene weergave) [DOB07]

**Procedureel**

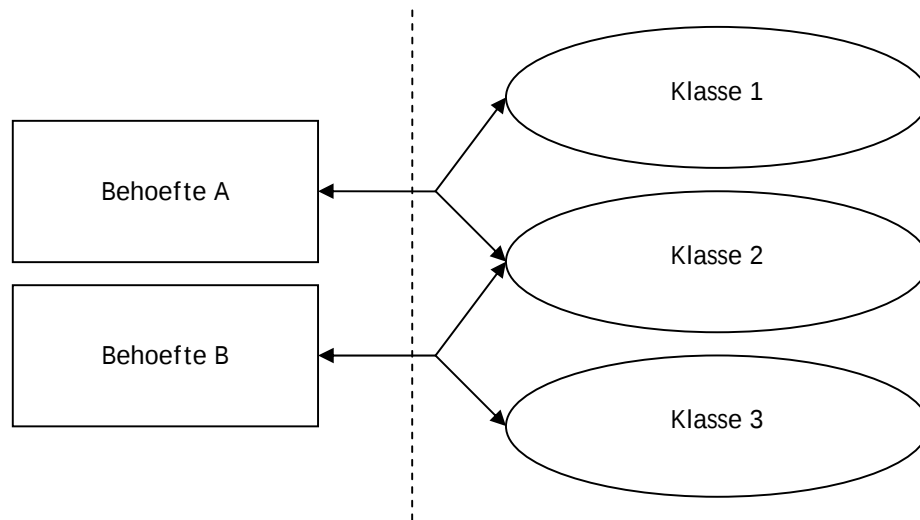
In *procedureel* ontwikkelen (zie figuur 9) wordt hergebruik gerealiseerd door een *procedure* of *functie* aanroepbaar te maken. In een programma kunnen de procedures of functies opnieuw aangeroepen worden. In dit geval is software een verzameling van instructies en aanroepen naar procedures of functies. De oplossing, in dit geval de procedure, is nauw verbonden (Engels: *tightly coupled*) met de behoefte. Vaak is er sprake van een één-op-één relatie tussen de oplossing en de behoefte.



figuur 9: Procedureel ontwikkelen [DOB07]

### Object georiënteerd

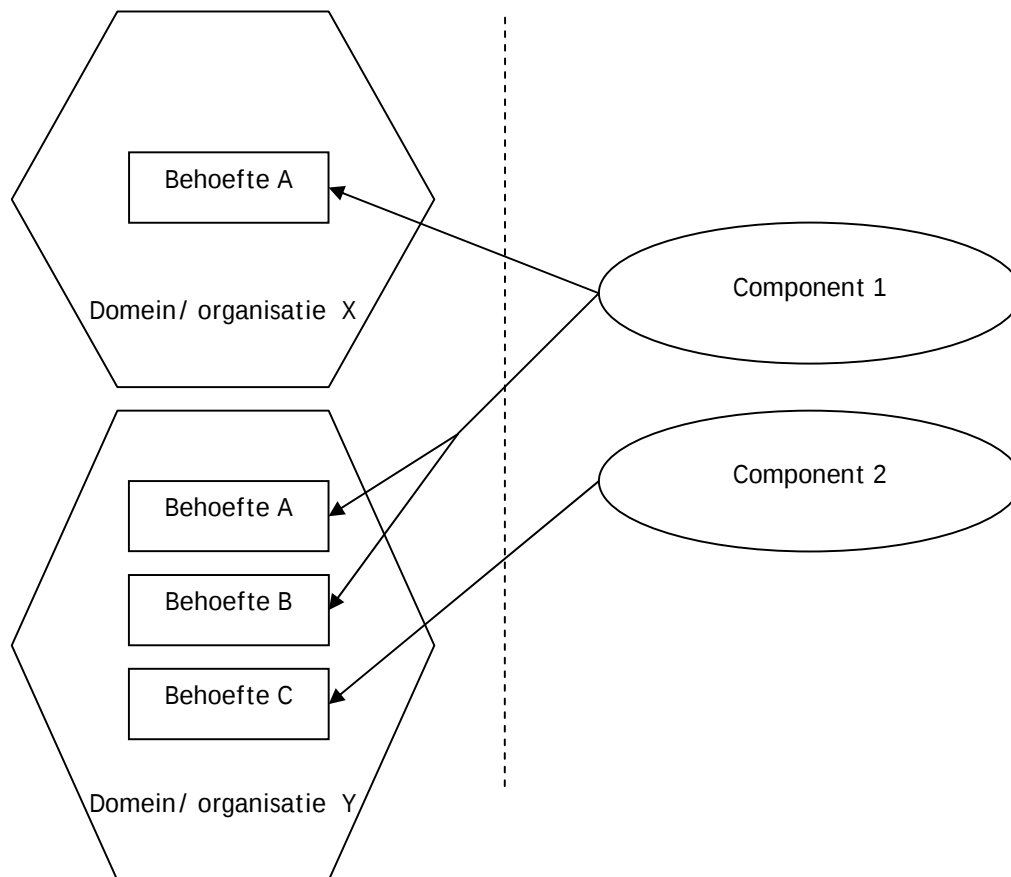
Bij *object georiënteerd* ontwikkelen (zie figuur 10) worden klassen ontworpen. Met de klassen kunnen objecten worden aangemaakt waar specifieke methoden bij horen. De methoden zijn te vergelijken met de functies uit procedureel ontwikkelen. Het is mogelijk om een nieuwe klasse te maken die bepaalde functionaliteiten overerft van een andere klasse. Hierdoor wordt hergebruik van klassen bevorderd. Echter, de behoefte blijft bepalend voor de oplossing. De klassen blijven verbonden aan een programmeeromgeving.



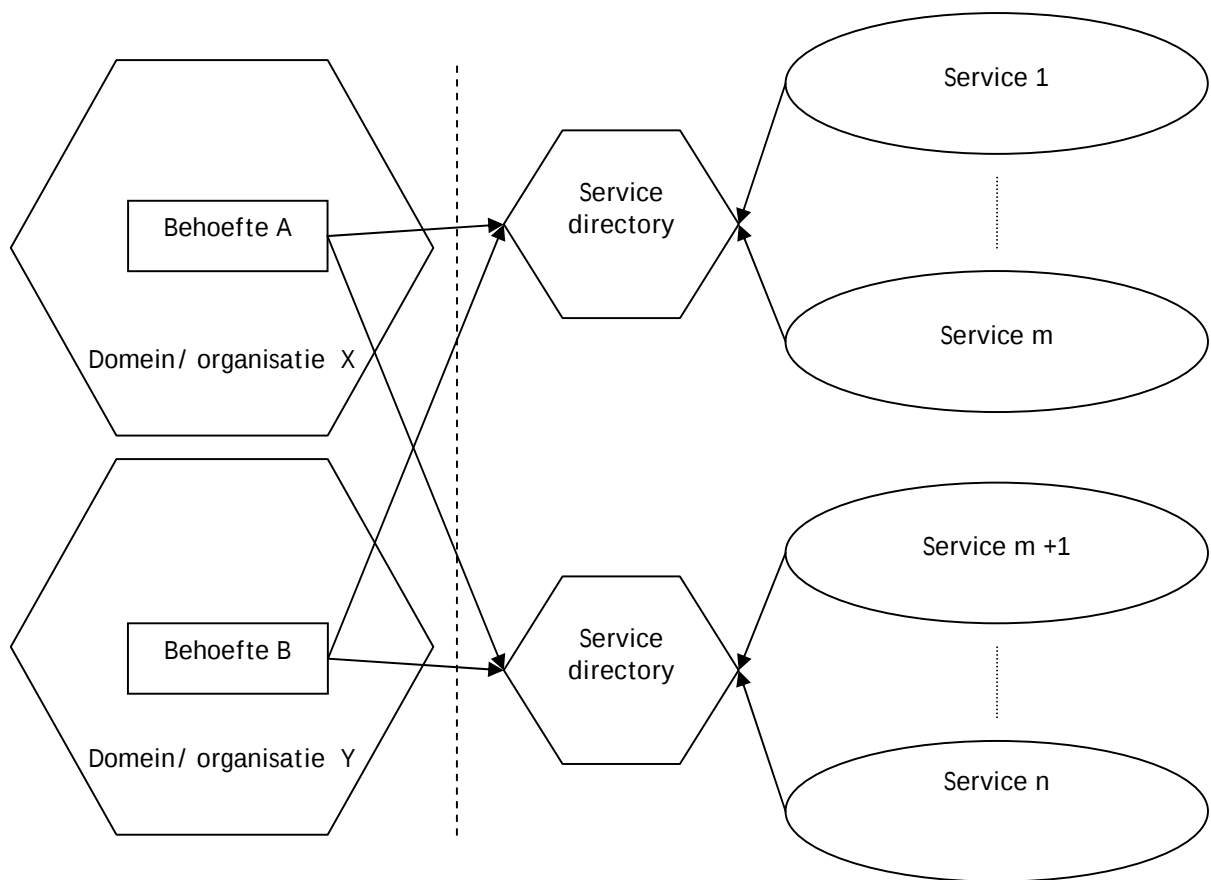
figuur 10: *Object georiënteerd ontwikkelen* [DOB07]

### Component gebaseerd

*Component gebaseerd* ontwikkelen (zie figuur 11) gaat uit van software onderdelen die (in een ideale omgeving) helemaal onafhankelijk van elkaar kunnen opereren. Een component is een softwareonderdeel met een hoger abstractieniveau dan een object. Deze componenten zijn al kant-en-klaar, en kunnen slechts op details bijgesteld worden. Ook hoeven niet alle functionaliteiten die een component biedt benut te worden. Een leverancier van componenten levert vaak alleen maar componenten die voor meerdere organisaties ingezet kunnen worden. De componenten kunnen op deze manier opnieuw gebruikt worden. Gevolg is wel dat de entiteit die een behoefte heeft, minder oplossingen heeft om uit te kiezen. Stond bij procedureel en object georiënteerd ontwikkelen een individuele behoefte centraal, bij component gebaseerd ontwikkelen staat een behoefte die in meerdere organisaties of domeinen voorkomt centraal. Een *domein* is een verzameling van samenhangende entiteiten, zoals een organisatie of een branche. De relatie van entiteiten kan van functionele aard zijn, zoals marketing of logistiek. Een voorbeeld van een component kan een Human Resources component zijn. Deze component biedt mogelijkheden voor behoeftes als het beheren van personeelgegevens en contracten. De leverancier bepaalt in grote mate hoe een component eruit ziet. De pijltjes zijn daarom hier ook niet dubbel. Pakketten (zie paragraaf 4.1) zijn typisch component gebaseerd opgezet.

figuur 11: *Component gebaseerd ontwikkelen* [DOB07]**Service georiënteerd**

Bij *service georiënteerd* ontwikkelen (zie figuur 12) worden functionaliteiten aangeboden in de vorm van services. Een oplossing voor een behoefte kan uit meerdere services bestaan. De services kunnen hergebruikt worden. De diensten worden door middel van een *service directory* openbaar gemaakt aan entiteiten. Als een entiteit op zoek is naar een oplossing, wordt eerst de service directory doorzocht. Als de gewenste service niet wordt gevonden, kan de service alsnog door de entiteit zelf ontwikkeld worden. Een service maakt gebruik van open standaarden. Hierdoor is het idealiter ook mogelijk om gebruik te maken van services buiten een domein om. Het verschil tussen services en componenten zit in het gebruik van open standaarden. Bij service georiënteerd is dit een vereiste, terwijl dit bij component gebaseerd in het midden wordt gelaten. Daarbij zijn services in staat veel specifiek op de behoeftes van een entiteit in te gaan dan componenten. Dit komt omdat binnen services een hiërarchie bestaat, waardoor services gebruik kunnen maken van andere services. Hierdoor kunnen gemakkelijk bepaalde functionaliteiten, die bij componenten vastzitten, gebruikt worden van services die in eerste instantie gebouwd zijn voor een ander domein. De structuur die op deze manier ontstaat wordt Service Oriented Architecture genoemd. In paragraaf 4.4 wordt hier verder op in gegaan.



figuur 12: *Service georiënteerd ontwikkelen* [DOB07]

Software paradigma's kunnen goed naast elkaar bestaan. Object georiënteerd en procedureel ontwikkelen worden gebruikt om op laag niveau code te structureren. Component gebaseerd en service georiënteerd ontwikkelen daarentegen worden gebruikt om op een abstracter niveau te kunnen spreken over de samenhang in software. Een organisatie kan componenten op een service georiënteerde manier aan elkaar koppelen. Ook kunnen services eigenlijk componenten zijn. Op dit niveau is dus overlap mogelijk.

#### 4.3. Enterprise architecture

Omdat SOA een *architectuur* is, definiëren we eerst wat dat precies is. In de digitale wereld wordt onder een architectuur het volgende verstaan:

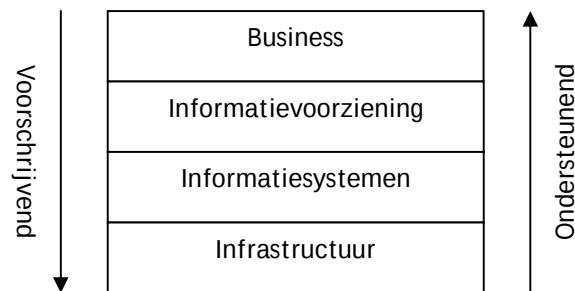
Definitie 8: architectuur

*"Architectuur is de fundamentele organisatie van een systeem, uitgedrukt in zijn componenten, hun relaties met elkaar en met de omgeving en de principes die ontwerp en evolutie leiden"* – [IEE00]

Architectuur wordt in een organisatie gebruikt om structuur aan te brengen in de IT-huishouding. Hierdoor wordt de complexiteit verkleind, waardoor de samenhang tussen verschillende componenten gemakkelijker kan worden gecreëerd en beheerd. Een architectuur werkt als abstractie om met complexe systemen te kunnen werken. Als middel om deze abstractie helder te presenteren wordt vaak gewerkt met een raamwerk van *lagen*. Iedere laag staat voor een onderdeel in een bedrijf. Een laag bevat soortgelijke componenten. De lagen onderling kunnen relaties met elkaar hebben. Deze samenhang is erg belangrijk. Een verandering in de hoogste laag kan gevolgen hebben voor andere

lagen. Alle lagen bij elkaar genomen kan gezien worden als een *holistisch systeem*. Het geïntegreerde geheel van alle onderdelen in een bedrijf ondergebracht in de verschillende lagen noemt men een *enterprise architecture* [DOB07].

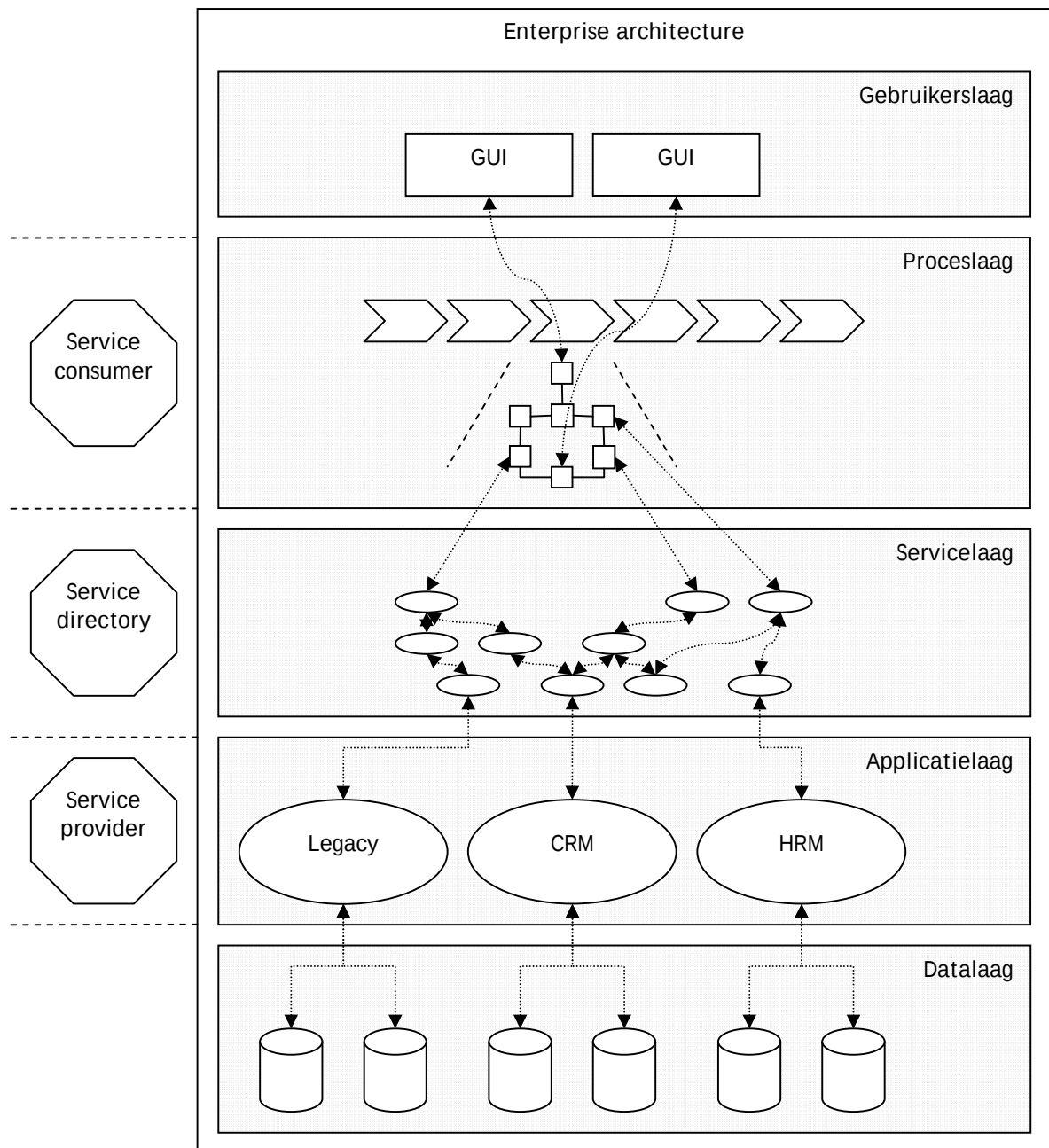
In der loop der jaren zijn er veel verschillende systemen bedacht met bijbehorende enterprise architectures. Capgemini gebruikt al jaren het *Integrated Architecture Framework (IAF)*, een raamwerk dat bij erg veel organisaties gebruikt kan worden door de brede opzet. IAF gaat uit van een 4-lagen architectuur. Zie ook figuur 13.



figuur 13: *Integrated Architecture Framework* [CAP08]

Met de laag *business* wordt de laag bedoeld waarin besluiten worden genomen over welke koers een organisatie gaat voeren. Dit wordt ook wel *governance* genoemd. In deze laag worden besluiten genomen over welke producten er worden geproduceerd, welke diensten er worden aangeboden, en hoe de bedrijfsprocessen, die nodig zijn om deze producten te maken of de diensten te leveren, worden aangestuurd. De laag van *informatievoorziening* wordt ook wel de *proceslaag* genoemd. Hierin bevindt zich de informatiestroom (zie hoofdstuk 2.5), documentenstroom, informatiebehoefte, informatiebronnen en de informatie-uitwisseling buiten de organisatie om. Dit geldt voor bestuurders, maar ook voor medewerkers die van elkaar willen leren. De laag *informatiesystemen* bevat alle applicaties en mechanismen om verschillende applicaties met elkaar te integreren. De infrastructuur ten slotte bevat de fundering voor de applicaties. Denk hierbij aan besturingssystemen, hardware, netwerken maar ook gemeenschappelijke applicaties zoals tekstverwerkers en emailprogramma's. Van boven naar beneden zijn de lagen *voorschrijvend*. Dat wil zeggen dat hogere laag voor een lagere laag bepaalt wat er gedaan wordt (voorschrijven wat te doen). Van beneden naar boven zijn de lagen *ondersteunend*. Een lagere laag helpt een hogere laag bij het uitvoeren van zijn taken.

IAF is breed toepasbaar en daarom ook erg abstract. Het IAF zegt echter niets over de inrichting van informatiesystemen, alleen maar hoe de laag informatiesystemen in verhouding staat tot andere lagen uit het IAF. Daarom gebruik we een *5-lagen architectuur*, gebaseerd op de 7-lagen architectuur van Dobbe [DOB07]. Deze 5-lagen architectuur heeft betrekking op de informatiesystemenlaag van het IAF.



figuur 14: 5-lagen architectuur [DOB07]

**Gebruikerslaag**

De bovenste laag van de 5-lagen architectuur is de gebruikerslaag. Alle componenten die te maken hebben met interactie met gebruikers, zitten in deze laag. Voorbeelden hiervan zijn *Graphical User Interfaces (GUI's)*. Gebruikers bepalen hoe processen doorlopen worden. Niet bij alle processen is een gebruikersinput nodig. Er zijn verschillende manieren om gegevens te presenteren en om input te vragen. De manier waarop dit wordt gedaan is erg belangrijk.

### Proceslaag

Processen in een organisatie kunnen op abstracte manier worden ingevoerd, door middel van een procestaal. Deze taal beschrijft hoe een proces verloopt, welke stappen na elkaar komen en welke input nodig is. De activiteiten die bij ieder proces horen worden in deze laag beschreven.

### Servicelaag

De servicelaag bevat services die gebruikt worden bij de executie van processen. In de servicelaag is een hiërarchie van services mogelijk. Services kunnen andere services weer gebruiken. Op deze manier kunnen services worden gebouwd die precies een processtap ondersteunen, of slechts een gedeelte hiervan. In de servicelaag bevindt zich de service directory (zie paragraaf 4.4). De daadwerkelijk executie van processtappen wordt gedaan in de applicatielaag.

### Applicatielaag

De applicatielaag bevat applicaties voor verschillende functionele onderdelen van een organisatie. De componenten van de applicaties zorgen door middel van open standaarden dat ze de juiste informatie in de juiste vorm kunnen leveren aan de services. Vaak wordt er nog een extra laag bovenop de applicatielaag geplaatst, de integratielaag. De integratielaag is verantwoordelijk voor de integratie van de applicaties onderling. In deze scriptie is alleen een applicatielaag met daarin de integratielaag voldoende.

### Datalaag

In de datalaag bevinden zich gegevens, opgeslagen in databases. Applicaties gebruiken deze gegevens.

## 4.4. SOA model

Bij SOA staan services centraal. Een service is een mechanisme dat een vermogen koppelt aan een behoefte. Een service biedt zelf geen oplossing voor een behoefte. Voor een bepaalde behoefte kunnen ook meerdere vermogens nodig zijn. En een vermogen kan hergebruikt worden, net als een service. Een entiteit die services aanbiedt, wordt een *service provider* genoemd. Een service provider biedt vermogens aan, en kan zelf ook over een vermogen beschikken. In de praktijk is dit minder interessant, omdat de gebruiker van de service niet hoeft of wil weten hoe een service werkt. Een entiteit die behoefte heeft aan een service, wordt de *service consumer* genoemd.

Definitie 9: service provider

*“Een service provider is een entiteit die een service aanbiedt.” – [DOB07]*

Definitie 10: service consumer

*“Een service consumer is een entiteit die een service gebruikt.” – [DOB07]*

Definitie 11: service participanten

*“De service consumers en de service providers worden gezamenlijk aangeduid als service participanten.” – [DOB07]*

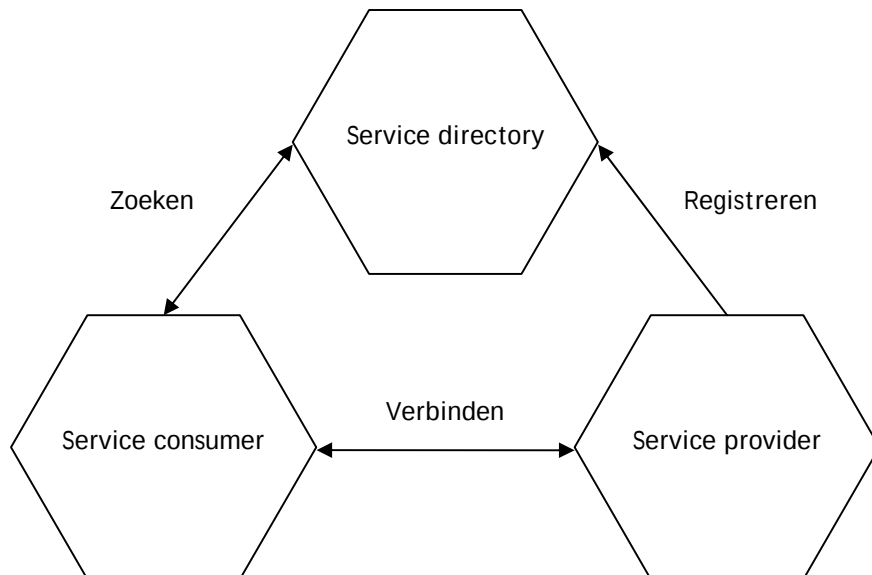
Een service consumer en een service provider moeten elkaar kunnen vinden, ook als ze in verschillende domeinen zijn ondergebracht. Dit wordt gedaan door de service die de service provider aanbiedt te registreren bij de *service directory* (ook wel service repository, service catalog of service registry genoemd). In de service directory is van iedere geregistreerde service een *service description* te vinden. In de service description staat waarvoor de service bedoeld is en hoe de interactie plaatsvindt. Op deze manier kan een service consumer op zoek gaan naar de juiste service.

Definitie 12: service directory

*“Een service directory is een verzameling van service descriptions, waarin service consumers kunnen zoeken.” – [DOB07]*

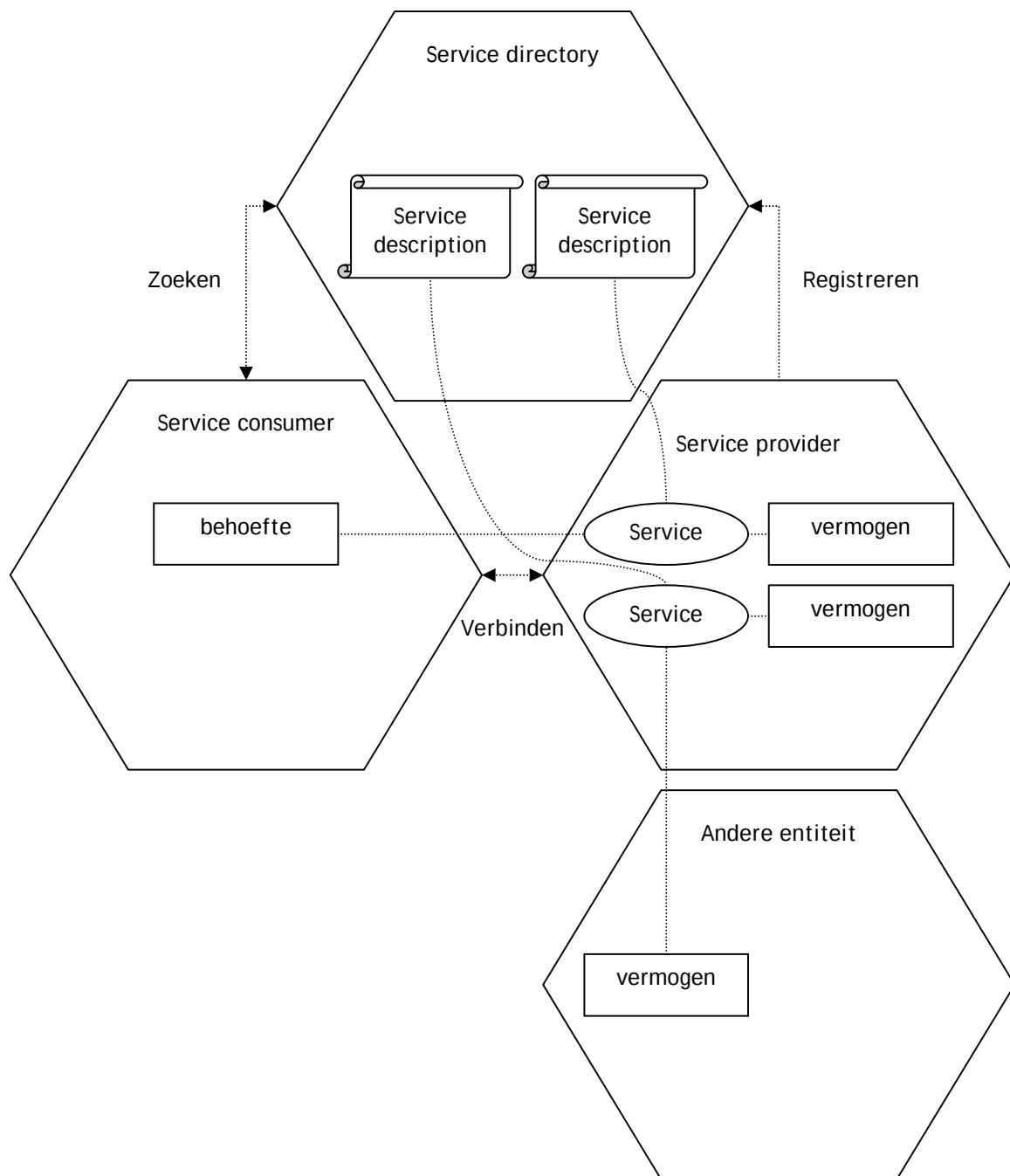
De relatie tussen de service provider, consumer en directory is in figuur 15 weergegeven. De service provider biedt een service aan. Om dit kenbaar te maken, wordt de service geregistreerd bij de service

directory. De service consumer heeft een behoefte, en zoekt in de service directory naar een geschikte service. Als deze gevonden wordt, verbindt de service consumer zich met de service provider. Deze levert de service aan de service consumer.



figuur 15: SOA model [SPE05]

Om het plaatje compleet te maken, volgt een compleet model van SOA in figuur 16. Een service provider biedt twee services aan. De service provider maakt daarbij gebruik van twee vermogens binnen zijn eigen domein, maar ook van een vermogen buiten zijn domein. De service consumer maakt gebruik van een service aangeboden door de service provider. De service directory bevat twee service descriptions van de twee services die de service provider aanbiedt.



figuur 16: Compleet SOA model [DOB07]

#### 4.5. Service concepten

Omdat services een centrale rol spelen in SOA, wordt in de volgende paragraaf verder ingegaan op enkele eigenschappen en concepten van services. Er zijn een drietal concepten belangrijk om de interactie met services plaatsvindt: de *zichtbaarheid* tussen service providers en service consumers, de *interactie* ertussen en het *effect* van de interactie [DOB07].

### Zichtbaarheid

Het is van groot belang dat de services zichtbaar zijn voor de service consumers. Doordat services ook vanuit andere domeinen of andere organisaties aangeboden kunnen worden, is dit zeker niet triviaal. Drie voorwaarden zijn van belang om zichtbaarheid mogelijk te maken.

- *Bewustzijn*. De service consumer moet zich bewust zijn van het bestaan en de toepassing van een service provider. Door gebruik te maken van een service directory en een service description is dit mogelijk.
- *Bereidheid*. De service participanten moeten bereid zijn om met elkaar te communiceren. Het gaat er hier om of de afspraken die over de communicatie bestaan worden nagekomen. De afspraken zijn vastgelegd in de service description.
- *Bereikbaarheid*. Communicatie tussen de service participanten moet mogelijk zijn.

### Interactie

De interactie tussen de service participanten verloopt via berichten. De volgende begrippen zijn hierbij relevant.

- *Informatiemodel*. Het informatiemodel zegt iets over de vorm en de betekenis van een bericht. De betekenis van een bericht wordt *semantiek* genoemd. De regels waarmee de vorm van een bericht wordt bepaald wordt de *syntax* genoemd.
- *Gedragmodel*. Het gedragmodel bevat de acties een service kan uitvoeren en de manier waarop berichten worden uitgewisseld. Dit wordt vastgelegd in een protocol.

### Effect

Het effect van een service is het leveren van informatie, het veranderen van de staat van een entiteit of een combinatie van beiden. Zo kan er een service zijn die een overzicht geeft van alle productiecijfers van een bepaalde fabriek. Deze service geeft alleen een overzicht. Ook kan er een service zijn die een product besteld. Hierbij wordt de staat van inkoop gewijzigd.

Een aantal andere relevante termen die betrekking hebben op SOA zijn *granulariteit*, *ontkoppeling*, *service interface* en *service description*.

### Service description

Een service description bevat elementaire informatie over een service. Deze informatie wordt door de service consumer gebruikt om te bepalen of de service voldoet aan de behoefte van de service consumer. Er is niet een eenduidige invulling welke informatie een service description hoort te bevatten. De volgende vier punten behoren er in elk geval wel in te staan:

1. Dat de service bestaat en hoe deze bereikbaar is;
2. Welke functies of set van functies de service uitvoert;
3. Aan welke voorwaarden voldaan moet worden;
4. Hoe de interactie plaatsvindt met de service provider (de service interface).

### Service interface

De service interface is de manier waarop interactie met een service kan plaatsvinden. De service interface maakt onderdeel uit van de service description. De service interface bevat de protocollen voor het versturen van berichten en welke commando's gegeven kunnen worden die resulteren in een effect. Bij SOA is het van groot belang dat de interfaces met eenzelfde syntax worden beschreven.

### Ontkoppeling

Om een service zo onafhankelijk mogelijk te laten wezen, is het van belang dat de service interface zo los mogelijk staat van de implementatie van de service. Deze scheiding van implementatie en interface wordt *ontkoppeling* genoemd (Engels: *loose coupling*). Doordat de interface los staat van de implementatie, hoeft de service consumer zich alleen maar te bekommeren om de interface van de service, dat volgens open standaarden gaat. De processen, die gebruik maken van de services, worden op deze manier ontkoppeld van de applicaties, die de functionaliteit van de services aanbieden.

Ontkoppeling kan ook doelen op de context van de service. Als de context ontkoppelt is van de service zelf, is de service breder toepasbaar. Ook de locatie van de service en de service zelf kunnen ontkoppeld worden. Hierdoor is de service ongeacht de locatie altijd bruikbaar.

#### **Granulariteit**

Granulariteit kan omschreven worden als de omvang van de functionaliteit van een service, en de mate van detaillering.

### **4.6. Definities**

De definitie van service die gehanteerd wordt in deze scriptie, is als volgt:

Definitie 13: service

*“Een service is een logische representatie van een herhaalbaar bedrijfsproces, dat een specifieke uitkomst heeft. Typische kenmerken van een service zijn:*

- *Een service opereert zelfstandig en is onafhankelijk (self-contained);*
- *Een service kan zijn opgebouwd uit andere services of kan hiervan gebruik maken;*
- *Een service is een ‘black box’ voor gebruikers van de service.” – [OPE06]*

Impliciet wordt hier gezegd dat een service vermogens gebruikt om te kunnen voldoen aan behoeftes in een organisatiecontext. Een service is een representatie van een herhaalbaar proces. Dit maakt de service geschikt voor hergebruik. Dat de service zelfstandig opereert en zelfstandig is, benadrukt dat de service onafhankelijk van andere services en onderliggende techniek is. Ook heeft dit tot gevolg dat de gebruiker zich geen zorgen hoeft te maken over nieuwere versies van services. Dit komt ook in de definitie tot uitdrukking, namelijk dat een service als black box gezien kan worden door de gebruikers. Doordat een service gebruik kan maken van andere services, is het mogelijk een hiërarchie aan te brengen in de services. We spreken van *composite services* en *single services* [DOB07]. Als een service gebruik maakt van één of meer services, wordt dit een composite service genoemd. Anders een single service.

De definitie van SOA die in deze scriptie gebruikt wordt, is de volgende.

Definitie 14: Service Oriented Architecture

*“SOA is een paradigma dat kan worden toegepast op een organisatie, waarbij:*

- *Services worden gebruikt om bedrijfsprocessen te ondersteunen;*
- *Services uit verschillende domeinen kunnen worden aangeboden;*
- *Services herbruikbaar zijn;*
- *De service interface ontkoppeld is van de implementatie van de service. Dit betekent dat de implementatie op ieder platform en in iedere taal mag, maar dat de interactie verloopt volgens standaarden.” – [DOB07]*

De servicelaag neemt de plaats in van de applicatielaag ten opzichte van de proceslaag. Voorheen ondersteunden de applicaties de processen. Het is daarom logisch dat services bedrijfsprocessen ondersteunen. In plaats van dat applicaties de bedrijfsprocessen ondersteunen, zijn het nu de services of de composite services die de ondersteuning bieden. De services maken op hun beurt weer gebruik van het vermogen van applicaties om aan de behoeftes van de processen te kunnen voldoen. De services dienen herbruikbaar te zijn. Hierdoor kunnen services voor verschillende processen worden gebruikt. Doordat de interactie via open standaarden verloopt, wordt gezorgd dat de services ook daadwerkelijk voor verschillende processen gebruikt kunnen worden. Ook kunnen services uit verschillende domeinen worden aangeboden. Het staat services in een SOA dus vrij waar ze de gegevens vandaan halen en hoe. Daarbij verloopt de communicatie van deze gegevens op een manier die iedereen zou moeten kunnen interpreteren. Het gebruik van open standaarden heeft ook tot gevolg dat de services gemakkelijk gebruik kunnen maken van applicaties in de applicatielaag.

#### 4.7. Karakteristieken SOA

SOA is niet een heel nieuw concept. Bepaalde uitgangspunten zijn al bijna even oud als het vakgebied informatica zelf. De mogelijkheden waren echter altijd nog beperkt door een gebrek aan de juiste hardware en infrastructuur. Maar mede door de opkomst van het Internet zijn de ontwikkelingen op het gebied van SOA in een stroomversnelling gekomen. Hierbij zijn de meest uiteenlopende definities ontstaan. Vaak wordt SOA ook verward met *Web Services (WS)* of met *Service Oriented Computing (SOC)*. Deze twee begrippen hebben zeker wel wat te maken met SOA, maar kunnen niet gelijk aan elkaar gesteld worden. WS's zijn implementaties van de definitie van een service. Er zijn echter talloze manieren om services te implementeren. SOC is de manier van programmeren van services, waarbij rekening gehouden moet worden met de granulariteit van services. Ook dienen de services asynchroon van elkaar te functioneren, omdat ze anders eindeloos op elkaar gaan zitten wachten.

In de inleiding werd gesteld dat SOA *pretendeert* een oplossing te bieden voor het gemakkelijker afstemmen van de IT-huishouding op processen. Dit omdat er nog maar weinig literatuur beschikbaar is gebaseerd op de praktijk. Het gaat vooral om verwachtingen die SOA schept. Deze verwachtingen lopen nogal uiteen, mede door het gebruik van verschillende definities. De meest genoemde karakteristieken van SOA zijn *interoperabiliteit*, *herbruikbaarheid* en *flexibiliteit*.

##### Interoperabiliteit

Met interoperabiliteit wordt de mate van uitwisselbaarheid van gegevens, informatie, documenten, etc. bedoeld. Een systeem met een lage interoperabiliteit wordt gekenmerkt doordat het lastig te vinden is, er geen berichten mee uitgewisseld kunnen worden, etc. Een systeem met een hoge interoperabiliteit is gemakkelijk te integreren met andere systemen, en maakt gebruik van gestandaardiseerde interfaces en open standaarden. Van SOA wordt beweerd dat het de interoperabiliteit van het systeem verhoogt. Dit komt doordat services dienen te communiceren via een afgesproken standaard (het liefst met brede ondersteuning van andere organisaties) en ook aanspreekbaar dienen te zijn uit andere domeinen. Dit leidt tot gemakkelijkere integratie van systemen [HAG01, VAU02, FER03, LAN03, IYE03, CUR06, JAM05, MOI05, BIC06, CUR03]. Een vereiste is wel dat beide domeinen gebruik maken van SOA (of in elk geval SOA *enabled* zijn, zie paragraaf 4.1). Hierdoor wordt het uitbesteden van processen ook gemakkelijker gemaakt (zie ook hoofdstuk 2.5) [BIC06, HAG01].

##### Herbruikbaarheid

Een belangrijke eis aan een SOA is dat services hergebruikt kunnen worden. Hierdoor hoeven er uiteindelijk minder services ontwikkeld te worden. Dit hergebruik wordt bereikt door services te beschouwen als een black box. Zowel de implementatie, context als de locatie van de service is ontkoppeld van de interface. Hierdoor worden services in staat gesteld gemakkelijker hergebruikt te worden, of andere services opnieuw te gebruiken. Dezelfde functionaliteit wordt daarom minder vaak opnieuw bedacht, wat resulteert in minder redundante code en minder fouten in het systeem [CHA04, CHE06, HAG01, JAM05, CUR06]. Doordat er minder code geschreven hoeft te worden, zijn (composite) services ook sneller ontwikkeld en gemakkelijker te onderhouden. Alleen voor de ontbrekende functionaliteiten dient een (composite) service geschreven te worden. Dit resulteert in lagere ontwikkelingskosten [IYE03, ELF04, JAM05, HAG01, CHA04], snellere aanpassing van systemen bij wijzigingen in processen [CHA04, IYE03, MOI05] en een systeem dat sneller operationeel is [CHA04, ELF04, CHE06].

##### Flexibiliteit

Organisaties in snel veranderende markten hebben behoefte aan flexibele systemen. Dit betekent dat als de processen gewijzigd worden, de ondersteunende applicaties gemakkelijk aan de processen aan te passen zijn. Bij SOA staan de services centraal. De services maken gebruik van het vermogen van applicaties. Dus het zijn nu niet de applicaties die aangepast moeten worden bij wijzigingen van processen, maar de services. In tegenstelling tot applicaties hebben services wel de eigenschap dat ze gemakkelijk te configureren zijn door de gestandaardiseerde interfaces en het gebruik van open standaarden [CHA04, HAG01, LAN03, IYE03, JAM05, ZHA05]. Dit zorgt er ook voor dat processen meer centraal komen te staan in een organisatie [CHA04]. In plaats van dat een organisatie zijn processen aanpast aan softwarepakketten (zie paragraaf 4.1), kunnen de services gemakkelijk aangepast worden

aan de processen. Hierdoor kunnen processen specifiekere worden ondersteund, wat resulteert in betere prestaties voor de organisatie [CUR06, JAM05, HAG01, CHE06].

In de literatuur komen ook een aantal implicaties naar voren die onvermijdelijk zijn bij het implementeren en het gebruiken van een systeem dat zich houdt aan de principes en eisen van SOA. We scharen deze implicaties ook onder karakteristieken, opdat we deze later op eenzelfde manier kunnen behandelen als de bovengenoemde drie karakteristieken. We beschouwen *governance* en *lange termijn effecten*.

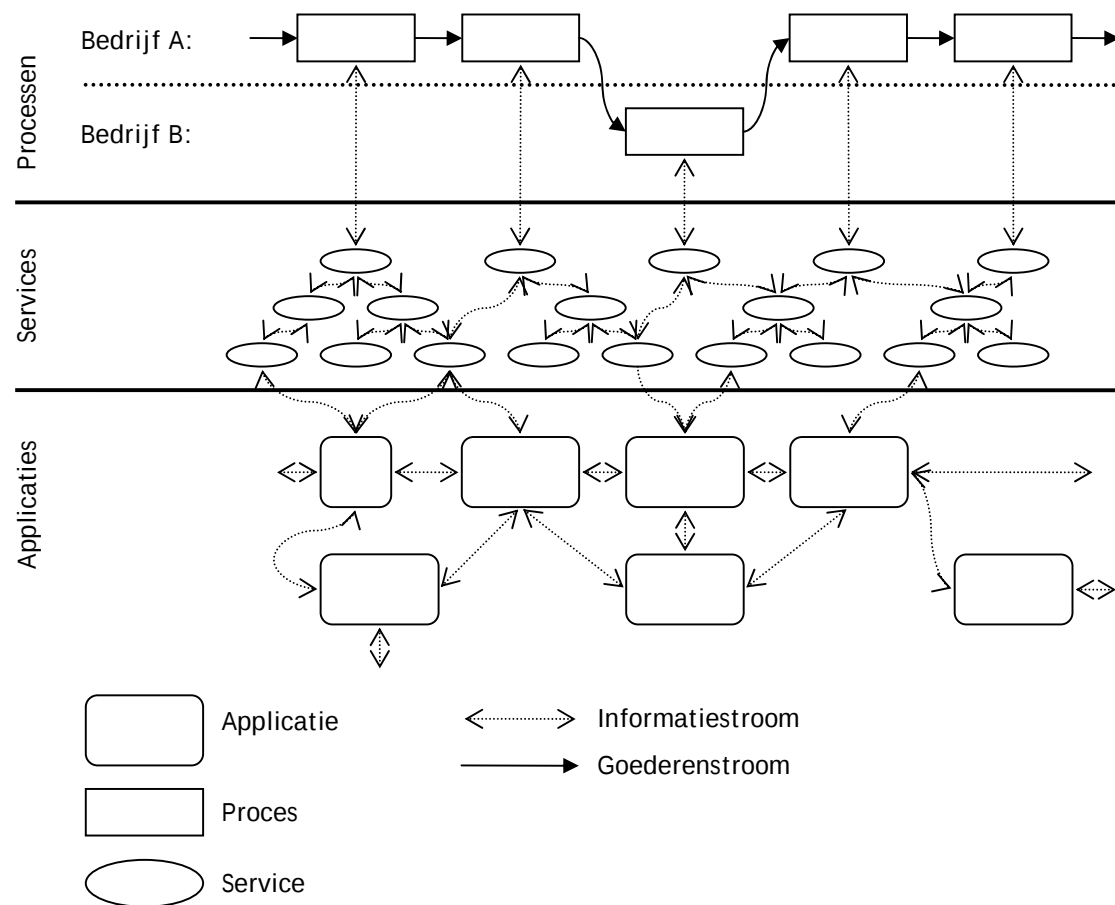
### **Governance**

Bij het ontwikkelen en toepassen van SOA is het van belang dat de regels en principes worden nageleefd [DOB07, LEA04, NEZ06]. Verder dienen services beschikbaar gesteld te worden. Dit wordt *governance* of *service management* genoemd. De kwaliteit van de services moet gegarandeerd worden [FRE02, KRE03, PEL03, CAS03, ELF04, BIR06, BIC06]. Het is van belang dat er goede afspraken worden gemaakt over de verantwoordelijkheid van de services, juist omdat services zich niet meer in een bepaald domein bevinden [DOB07]. Ook is het van belang na te gaan welke informatie wel en welke niet wordt aangeboden via een service. Privacy gevoelige informatie moet beschermd worden [VAU02, SHI02, KRE03, PEL03, LEA04, BIR06, CHE06].

### **Lange termijn effecten**

Het opzetten van een SOA is een grote investering. Alle processen en applicaties moeten worden aangepast of zelfs worden vervangen. Ook moeten alle services geïmplementeerd worden, en de service directory moet onderhouden worden. Services zullen niet meteen de juiste granulariteit hebben voor hergebruik. Services moeten bijgeschaafd worden, een proces dat jaren kan duren [DOB07]. Dit vormt een hoge drempel voor organisaties. Pas op de lange termijn zal SOA vruchten afwerpen [VAU02, CHE06, BUS07]. Daarbij zijn er nog veel verschillende standaarden voor dezelfde type berichten. Software bedrijven introduceren vaak hun eigen standaard, wat weer problemen oplevert met de integratie van andere systemen [LAN03, LEA04, NEZ06, PEL03, SHI02].

In figuur 4 van hoofdstuk 2.5 hebben we gezien dat applicaties vaak tightly coupled zijn aan processen. In hoofdstuk 3.3 hebben we uit de drie generieke strategieën voor logistieke dienstverleners afgeleid dat er behoefte is aan systemen die loosely coupled zijn. Met SOA hoeven de koppelingen tussen de applicaties en processen niet meer direct gelegd te worden.



figuur 17: De servicelaag toegevoegd

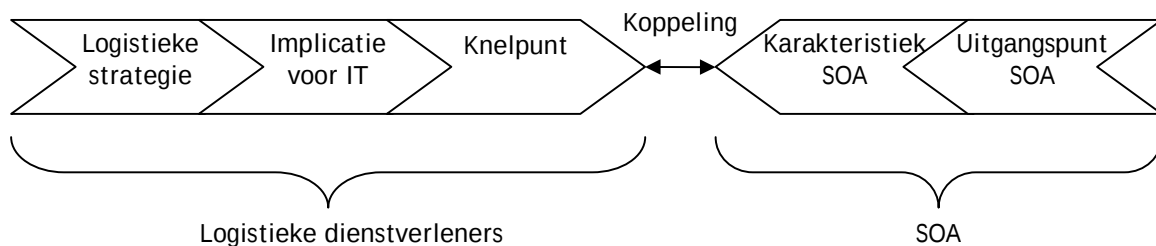
De vraag welk systeem gebruikt wordt (van bedrijf A, van bedrijf B of een nieuwe systeem) is lang niet zo belangrijk meer met het gebruik van SOA. Dit komt door de interoperabiliteit van SOA. Services kunnen buiten het domein van een organisatie worden gebruikt. Hierdoor zijn ze voor meer dan één organisatie te gebruiken. Er kunnen zelfs services worden gebruikt die van bedrijf A noch van bedrijf B zijn. Doordat de services gemakkelijk te configureren zijn en niet opnieuw geschreven hoeven te worden (herbruikbaarheid), zijn composite services gemakkelijk te creëren. Hierdoor kunnen wijzigingen in processen snel ondersteund worden door applicaties, wat resulteert in meer flexibiliteit. In hoofdstuk 5 kijken we wat SOA specifiek kan betekenen voor logistieke dienstverleners.

## 5. Mogelijkheden SOA voor logistieke dienstverleners

In het volgende hoofdstuk wordt bekeken welke mogelijkheden SOA biedt voor het uitbesteden van processen aan logistieke dienstverleners. Hiervoor gebruiken we de resultaten van de literatuurstudie in hoofdstuk 3 en hoofdstuk 4. Van deze resultaten is een model gebouwd waarin de knelpunten van logistieke dienstverleners gekoppeld worden aan de karakteristieken van SOA. Elke koppeling wordt besproken. Tot slot volgt de conclusie, waarin wordt besproken wanneer SOA wel en wanneer geen oplossing kan bieden.

### 5.1. Model

In hoofdstuk 3 hebben we een drietal strategieën voor logistieke dienstverleners besproken. Deze strategieën hebben implicaties voor de koers die een organisatie gaat volgen. De implicaties hebben betrekking op de inrichting voor de IT-huishouding. Hier zijn een aantal (potentiële) knelpunten te identificeren. In hoofdstuk 4 is het software paradigma SOA besproken. SOA heeft een aantal uitgangspunten, waaruit karakteristieken van SOA zijn af leiden. Door de karakteristieken van SOA te koppelen aan de knelpunten van de logistieke dienstverleners, maken we inzichtelijk hoe en wanneer SOA mogelijk een oplossing biedt voor het uitbesteden van processen aan logistieke dienstverleners. Dit is schematisch weergegeven in figuur 18.



figuur 18: Onderzoeksmodel

In tabel 1 staat weergegeven welke implicaties voor de IT van logistieke dienstverleners volgen uit de strategieën van logistieke dienstverleners. In tabel 2 staan de knelpunten die geïdentificeerd zijn bij de implicaties. Daarmee is het linker gedeelte van figuur 18 ingevuld. De strategieën met de bijhorende implicaties en knelpunten zijn beredeneerd in hoofdstuk 3.3, gebaseerd op de literatuur.

| Strategie voor logistieke dienstverlener           | Implicatie volgend uit strategie               |
|----------------------------------------------------|------------------------------------------------|
| Directe strategie met routine klussen              | Integratie van applicaties is essentieel (m2m) |
|                                                    | Kleine marges voor geleverde diensten          |
| Stop-over strategie met standaard diensten         | Integratie van applicaties is essentieel       |
|                                                    | Grotere transparantie van applicaties          |
|                                                    | Kleine marges voor geleverde diensten          |
| Assembled-stop-over strategie met diensten op maat | Integratie van applicaties is essentieel       |
|                                                    | Grotere transparantie van applicaties          |
|                                                    | Specifieke eisen van verlader of klant         |

tabel 1: Strategieën en de bijbehorende implicaties

| Implicatie volgend uit strategie         | Potentieel knelpunt                                                                                                    |
|------------------------------------------|------------------------------------------------------------------------------------------------------------------------|
| Integratie van applicaties is essentieel | Geen overeenstemming over syntax berichten                                                                             |
|                                          | Geen overeenstemming over semantiek berichten                                                                          |
| Kleine marges voor geleverde diensten    | Geen geld voor grote investeringen in IT                                                                               |
| Grotere transparantie van applicaties    | De beveiliging van de informatiestroom                                                                                 |
|                                          | Geen geschikt raamwerk (zowel op technologisch vlak als op organisatorisch vlak) voor aanbieden/muteren van informatie |
| Specifieke eisen van verlader of klant   | IT systemen zijn niet flexibel genoeg                                                                                  |

tabel 2: Implicaties voor IT en bijbehorende knelpunten

In tabel 3 staan de uitgangspunten van SOA met de bijbehorende karakteristieken. Hiermee is ook het rechter gedeelte van figuur 18 ingevuld. De uitgangspunten van SOA met de bijbehorende karakteristieken komen voort uit de definities van service en SOA (zie hoofdstuk 4.6) en zijn behandeld in hoofdstuk 4.7.

| Uitgangspunten SOA                                                                                                                         | Karakteristieken       |
|--------------------------------------------------------------------------------------------------------------------------------------------|------------------------|
| Services communiceren via afgesproken interface, vanuit meerdere domeinen                                                                  | Interoperabiliteit     |
| Implementatie, locatie en context van service zijn ontkoppeld van interface                                                                | Herbruikbaarheid       |
| Services worden benaderd als black box, ze kunnen gebruikt worden zonder dat de werking ervan bekend is                                    | Flexibiliteit          |
| Services moeten voldoen aan een hoop eisen                                                                                                 | Governance             |
| Huidige IT systemen moeten anders worden opgezet, er moeten afspraken worden gemaakt over open standaarden, services moeten gebouwd worden | Lange termijn effecten |

tabel 3: Uitgangspunten SOA en bijbehorende karakteristieken

## 5.2. Koppelingen

In de volgende paragraaf koppelen we de potentiële knelpunten bij logistieke dienstverleners aan de karakteristieken van SOA. Per koppeling wordt aangegeven in hoeverre de oplossing die SOA biedt gebruikt kan worden om het knelpunt weg te nemen. Aan het eind van deze paragraaf volgt een overzicht van de koppelingen. Alles wordt bekeken vanuit het perspectief van SOA: wat betekent een karakteristiek van SOA voor de knelpunten uit de logistiek. Per karakteristiek van SOA wordt een tabel gegeven. In de tabel staan de koppelingen schematisch weergegeven. Een plus wil zeggen dat de karakteristiek van SOA een oplossing biedt voor het knelpunt, een min wil zeggen dat het juist het knelpunt vergroot, plus/min wil zeggen dat het geen oplossing biedt maar ook niet het knelpunt vergroot, en de nul wil zeggen dat er geen relatie tussen karakteristiek en knelpunt bestaat. De tabellen worden gebundeld in figuur 19. In de tekst die volgt op de tabel met koppelingen, worden de koppelingen beredeneerd. Als een koppeling wordt gelegd, wordt het betreffende knelpunt tussen haakjes genoemd met een plus, min of plus/min (die uiteraard corresponderen met de bijbehorende tabel).

### Interoperabiliteit

Interoperabiliteit is de mate van uitwisselbaarheid van gegevens en informatie van een systeem. Een hoge interoperabiliteit betekent dat informatie en gegevens gemakkelijk uitgewisseld kunnen worden. In het volgende stukje wordt gekeken welke potentiële knelpunten baat hebben bij een hoge interoperabiliteit.

|                              | Interoperabiliteit |
|------------------------------|--------------------|
| Syntax van berichten         | +                  |
| Semantiek van berichten      | +/-                |
| Geen geld voor investering   | 0                  |
| Beveiliging informatiestroom | -                  |
| Geen geschikt raamwerk       | +                  |
| Inflexibiliteit systemen     | +                  |

tabel 4: *Knelpunten gekoppeld aan interoperabiliteit, theoretisch*

Services communiceren via open standaarden en over verschillende domeinen heen. Hiervoor zijn gestandaardiseerde interfaces nodig. Logistieke dienstverleners hebben juist te kampen met veel verschillende interfaces, wel of niet gestandaardiseerd. Logistieke dienstverleners willen wel applicaties met elkaar integreren, maar doordat er geen of weinig afspraken zijn gemaakt over de communicatie, gaat dit langzaam. Afspraken moeten alsnog gemaakt worden. Als logistieke dienstverleners zich houden aan de principes van SOA, vormt de syntax van de berichtcommunicatie geen probleem meer (syntax van berichten, +). Hoe een bericht precies wordt geïnterpreteerd, dus de semantiek, is een ander knelpunt. Het is mogelijk om bij de service description uitleg te geven over de semantiek van de berichten, maar daar moet dan wel op toegezien worden. Mogelijk moeten daarover ook extra afspraken worden gemaakt. Dus een interoperabel systeem levert niet geen duidelijke oplossing voor de semantiek van berichten (semantiek van berichten, +/-). Doordat applicaties met open standaarden en gestandaardiseerde interfaces gemakkelijker geïntegreerd kunnen worden, kunnen processen transparanter worden gemaakt. Interoperabiliteit is essentieel voor een raamwerk voor het gemakkelijk uitwisselen van gegevens (inflexibiliteit systemen, +) en eventueel voor het muteren van gegevens, over domeinen heen (geen geschikt raamwerk, +). De beveiliging van deze informatiestromen is wel erg belangrijk. Door het gebruik van open standaarden en gestandaardiseerde interfaces is het voor potentiële kwaadwillende entiteiten al wel duidelijk hoe een service aangesproken moet worden, wat ten koste gaat van de veiligheid (beveiliging informatiestromen, -).

### Herbruikbaarheid

Een aantal problemen in de IT ontstaan doordat er veel verschillende systemen naast elkaar draaien die vrijwel hetzelfde doen. Herbruikbaarheid kan deze complexiteit reduceren. In het volgende stukje kijken we welke invloed de herbruikbaarheid heeft op de potentiële knelpunten van logistieke dienstverleners.

|                              | Herbruikbaarheid |
|------------------------------|------------------|
| Syntax van berichten         | +                |
| Semantiek van berichten      | +/-              |
| Geen geld voor investering   | +                |
| Beveiliging informatiestroom | -                |
| Geen geschikt raamwerk       | +                |
| Inflexibiliteit systemen     | +                |

tabel 5: *Knelpunten gekoppeld aan herbruikbaarheid, theoretisch*

Services dienen zo opgezet te worden dat ze zo vaak mogelijk hergebruikt kunnen worden. Hierdoor hoeft weinig code geschreven te worden. De redundantie en het aantal fouten in systemen af, met tot gevolg lagere ontwikkelingskosten etc. (zie ook hoofdstuk 4.7). Dit draagt bij aan de flexibiliteit van een informatiesysteem (inflexibiliteit systemen, +). De investering in een nieuwe applicatie is met het gebruik van services veel minder groot (geen geld voor investering, +). De implementatie, de locatie en context van de service zijn ontkoppeld van interface. Hierdoor maakt het niet uit hoe de service eruit ziet, zolang de communicatie maar via een afgesproken manier gaat. De herbruikbaarheid wordt vergroot door een standaard te gebruiken die breed wordt gesteund door de branche waarin de logistieke dienstverlener actief is. De syntax van berichten is nauw verbonden met de herbruikbaarheid

van services. Applicaties worden vaker met een gestandaardiseerde interface uitgerust als services meer hergebruikt kunnen worden, juist om gebruik te kunnen maken van de service (syntax van berichten, +). De betekenis van de berichten staat hier echter helemaal los van. Juist doordat de context van services ontkoppeld is van de service zelf, kunnen services heel breed worden ingezet. Er moeten wel duidelijke afspraken worden gemaakt over de semantiek van berichten. Men wordt hierdoor ook gedwongen over de semantiek na te denken. Dit is daarom niet voor- of nadelig te noemen voor het knelpunt semantiek van berichten (semantiek van berichten, +/-). Door de herbruikbaarheid wordt er wel een goed raamwerk neergezet om processen transparanter te maken (geen geschikt raamwerk, +). Het is eenvoudig om een trackingsysteem dat voor de ene verlader wordt gebruikt, te hergebruiken voor een andere verlader. Dit kan echter ook leiden tot beveiligingsproblemen op grote schaal. Het inbreken in een service kan mogelijk invloed hebben op alle composite services die gebruik maken van de service waarop is ingebroken (beveiliging informatiestroom, -).

### Flexibiliteit

De flexibiliteit van een organisatie is de snelheid waarmee de organisatie zich kan aanpassen aan veranderende omstandigheden. Dit kan een veranderende markt zijn, maar ook politieke invloeden kunnen ten grondslag liggen van deze veranderde omstandigheden. Een organisatie past hierop zijn bedrijfsprocessen aan, wat ook gevolgen heeft voor de IT-huishouding. In het volgende stukje kijken we of flexibiliteit de potentiële knelpunten van logistieke dienstverleners beïnvloed. Triviaal is de koppeling met 'inflexibiliteit systemen' (inflexibiliteit systemen, +).

|                              | Flexibiliteit |
|------------------------------|---------------|
| Syntax van berichten         | +             |
| Semantiek van berichten      | 0             |
| Geen geld voor investering   | -             |
| Beveiliging informatiestroom | 0             |
| Geen geschikt raamwerk       | +             |
| Inflexibiliteit systemen     | +             |

tabel 6: *Knelpunten gekoppeld aan flexibiliteit, theoretisch*

De flexibiliteit van services wordt veroorzaakt doordat services als een black box beschouwd kunnen worden. Hierdoor weet de gebruiker van de service eigenlijk niet hoe de service werkt, op welk platform deze draait en in welke taal de service is geschreven. De gebruiker hoeft alleen maar te weten hoe de service gebruikt kan worden. Dit kan gedaan worden door berichten in de juiste vorm (syntax) te sturen naar de service. Dit maakt services extreem flexibel. Een flexibel systeem garandeert eigenlijk dat de syntax goed wordt afgesproken (syntax berichten, +). Een logistieke dienstverlener kan veel meer verladers met diensten bedienen, doordat de ondersteunende IT gemakkelijk opgezet kan worden. Dit flexibele raamwerk hoeft maar eenmalig opgezet te worden, en hoeft daarna alleen nog maar bijgeschaafd te worden (geen geschikt raamwerk, +). De eenmalige (grote) investering in het nieuwe raamwerk is natuurlijk wel nodig (geen geld voor investering, -).

### Governance

Governance wordt letterlijk vertaald als 'besturing'. In de context van SOA wil dit zeggen dat er toezicht en sturing wordt uitgeoefend om te garanderen dat alle principes en regels van SOA worden nageleefd. In het volgende stukje kijken we welke invloed governance heeft op de potentiële knelpunten van logistieke dienstverleners.

|                              | Governance |
|------------------------------|------------|
| Syntax van berichten         | 0          |
| Semantiek van berichten      | -          |
| Geen geld voor investering   | -          |
| Beveiliging informatiestroom | 0          |
| Geen geschikt raamwerk       | 0          |
| Inflexibiliteit systemen     | 0          |

tabel 7: *Knelpunten gekoppeld aan governance, theoretisch*

Het is van groot belang dat services onderhouden worden, waarbij rekening wordt gehouden met de ont koppeling van de interface en de implementatie, de context van de services en de locatie van de services. Het is van belang om afspraken te maken over de betekenis van de berichten die worden uitgewisseld, en dat deze ook worden nageleefd. Bedrijven kunnen vast blijven houden aan hun eigen standaarden, waardoor er alsnog afspraken gemaakt dienen te worden over de betekenis van standaarden (semantiek van berichten, -). Een nieuwe versie van een service uitbrengen is relatief eenvoudig, doordat de service ont koppeld is van de interface. De functionaliteiten moeten echter wel gecontinueerd blijven, ook wanneer de context van de service wordt uitgebreid of de granulariteit wordt vergroot. Een service is geen applicatie die, eenmaal opgeleverd, geen aandacht meer behoeft. Hierdoor kunnen de kosten oplopen (geen geld voor investering, -).

#### Lange termijn effecten

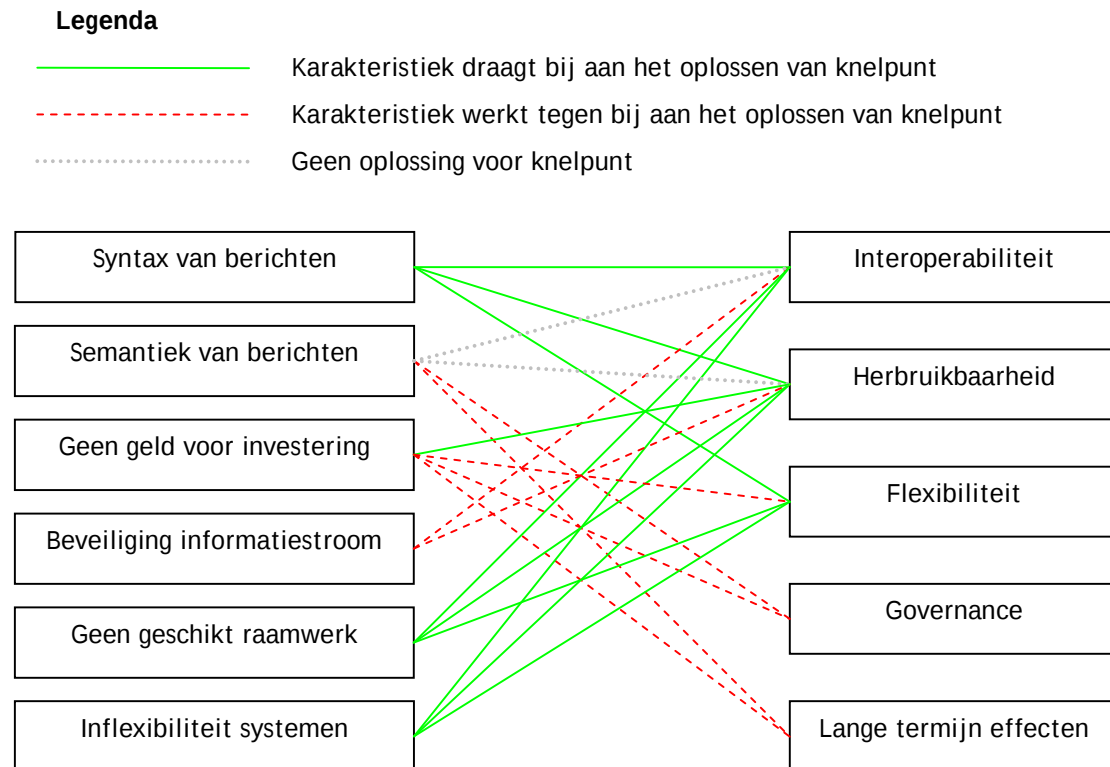
Er is een hoop tijd nodig voordat de vruchten geplukt kunnen worden van systemen die volgens de principes en regels van SOA werken (zie ook hoofdstuk 4.7). Het volgende stukje bekijkt welke invloed de lange termijn effecten hebben op de potentiële knelpunten van logistieke dienstverleners.

|                              | Lange termijn effecten |
|------------------------------|------------------------|
| Syntax van berichten         | 0                      |
| Semantiek van berichten      | -                      |
| Geen geld voor investering   | -                      |
| Beveiliging informatiestroom | 0                      |
| Geen geschikt raamwerk       | +/-                    |
| Inflexibiliteit systemen     | +/-                    |

tabel 8: *Knelpunten gekoppeld aan lange termijn effecten, theoretisch*

Het maken van afspraken over de semantiek van berichten kan erg lang duren. Er moet overeenstemming worden bereikt met heel veel verschillende bedrijven, juist doordat logistieke dienstverleners in zoveel verschillende supply chains actief zijn. Zolang hier geen afspraken over bestaan, zijn logistieke dienstverleners minder geneigd hun huidige systemen op de schop te gooien (semantiek van berichten, -). Dit kost immers een hoop geld. Diensten moeten in de tussentijd geleverd blijven worden, waardoor er een periode ontstaat dat het nieuwe systeem naast het oude systeem blijft draaien. Het gewenste flexibele raamwerk is pas na een langere periode bruikbaar en dan pas kunnen de vruchten worden geplukt (inflexibiliteit systemen/geen geschikt raamwerk, +/-). Deze periode is vooral voor minder kapitaalkrachtige logistieke dienstverleners, die moeten overleven met kleine marges, niet te overbruggen (geen geld voor investering, -).

In figuur 19 zijn de koppelingen tussen de knelpunten van logistieke dienstverleners met de karakteristieken van SOA schematisch weergegeven. Voor een oplossing van het knelpunt is de groene lijn gebruikt, rood is gebruikt voor een negatief effect van een karakteristiek op het knelpunt. Als er geen oplossing wordt geboden voor het knelpunt, is de lijn grijs gestippeld. Als er geen relatie bestaat tussen knelpunt en voor- of nadeel, dan is er geen lijn getrokken.



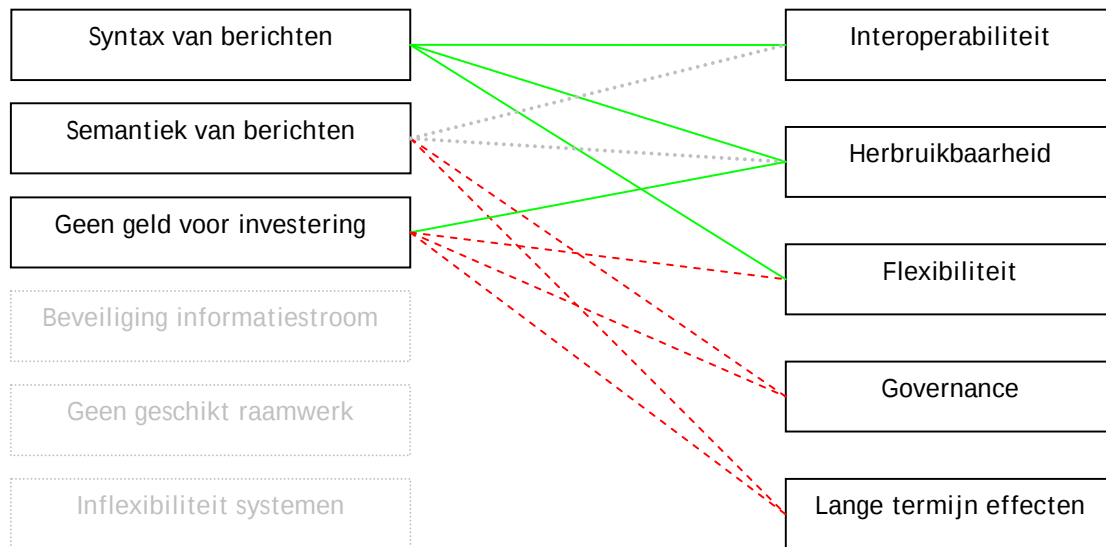
figuur 19: Koppeling knelpunten met karakteristieken, theoretisch

### 5.3. Implicaties voor strategieën

In figuur 19 zijn alle knelpunten afgeleid uit de verschillende strategieën van logistieke dienstverleners op een hoop gegooid. In deze paragraaf zullen we kijken wat de implicaties zijn van de koppelingen uit figuur 19 voor de verschillende generieke strategieën van logistieke dienstverleners van Bask [BAS01].

#### Logistieke dienstverleners met een directe strategie en routine klussen

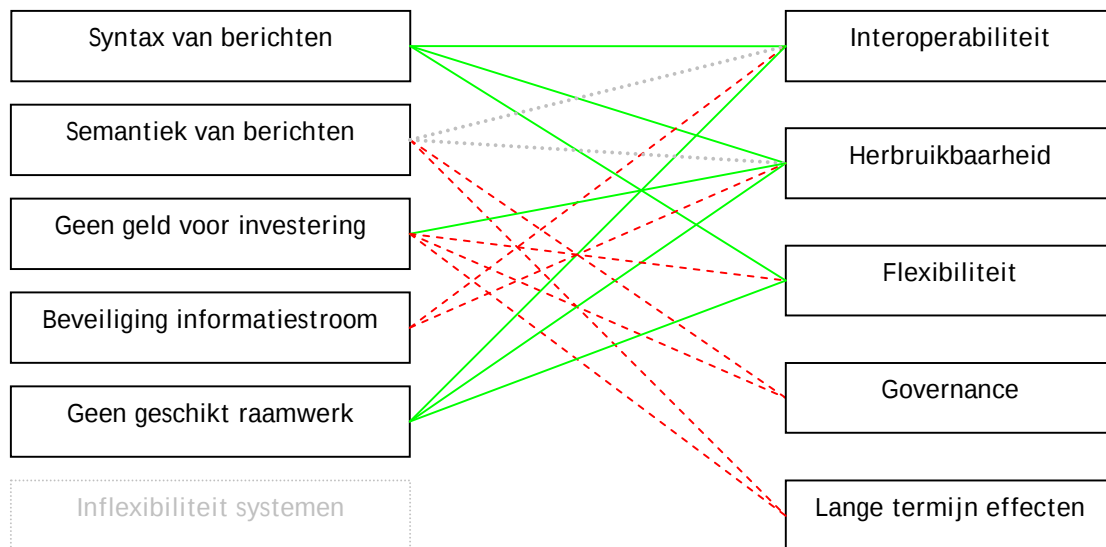
Logistieke dienstverleners met een directe strategie en routine klussen hebben als knelpunten dat er geen overeenstemming over de syntax en semantiek van berichten bestaat, en dat ze maar weinig geld te besteden hebben voor investeringen in de IT (zie hoofdstuk 3.3). In figuur 20 staan de koppelingen die alleen relevant zijn voor dit type logistieke dienstverleners. De vraag is hoe happig logistieke dienstverleners zijn om gemakkelijk applicaties te kunnen integreren met hun verlader. Het biedt vooral voordelen voor de verlader, want die wordt in staat gesteld nog beter te vergelijken en te selecteren. Logistieke dienstverleners moeten echter uiteindelijk vanzelf mee veranderen als alle verladers eisen dat er gecommuniceerd wordt via open standaarden. Hiervoor is echter geen flexibel systeem nodig. Daarbij komt nog dat door de kleine marges deze logistieke dienstverleners geen geld hebben om te investeren in een flexibeler systeem. Een SOA systeem is op basis hiervan voor deze logistieke dienstverleners niet aan te raden.



figuur 20: Koppelingen bij logistieke dienstverleners met directe strategie en routine klussen

#### Logistieke dienstverleners met stop-over strategie en standaard diensten

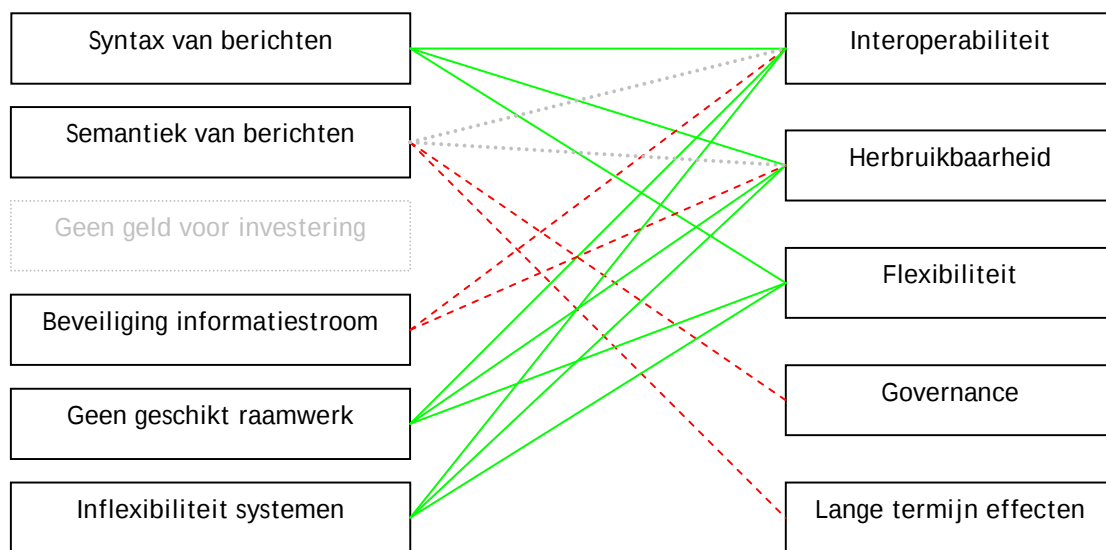
Voor logistieke dienstverleners met een stop-over strategie en standaard diensten is ook de interoperabiliteit van SOA aantrekkelijk. Met name om bepaalde processen inzichtelijk te maken, is de interoperabiliteit erg wenselijk. Doordat deze logistieke dienstverleners standaard diensten leveren (zie hoofdstuk 3.2), is een grote mate van flexibiliteit van het systeem niet heel erg noodzakelijk. Ook de herbruikbaarheid van de services die de standaarddiensten leveren zullen niet snel gewisseld worden. Per verlader zal waarschijnlijk wel het totale dienstenpakket anders zijn, maar dit hoeft niet perse ondersteund te worden door SOA. Een uitgebreid pakket (bij voorkeur ERP) waarin externe diensten aan- en uitgeschakeld kunnen worden met een ESB bijvoorbeeld voldoet ook. Dat is tevens een minder grote investering dan een SOA systeem.



figuur 21: Koppelingen bij logistieke dienstverleners met stop-over strategie en standaard diensten

**Logistieke dienstverleners met assembled-stop-over strategie en diensten op maat**

Logistieke dienstverleners met een assembled-stop-over strategie en diensten op maat kampen met grote onderhoudsproblemen doordat ze regelmatig aanpassingen moeten maken in hun IT-huishouding (zie ook hoofdstuk 3.3). Veel kennis over oudere systemen is verdwenen uit de organisatie, en de wensen van de verlader zijn zo specifiek dat er regelmatig gekozen wordt om een nieuw systeem te bouwen. SOA kan goed bijdragen aan het snel en gemakkelijk in elkaar zetten van een nieuw systeem. Het succes is echter wel van een aantal factoren afhankelijk. Een grote investering is nodig om de huidige IT-huishouding te veranderen, verladers moeten meewerken en de gebruikte standaarden moeten breed gedragen worden (zie hoofdstuk 4.7). Ook zijn er verschillende dilemma's op vertrouwensvlak, die overwonnen moet worden voor sommige diensten. Dit speelt zich vooral af op het gebied van beveiliging, en kunnen zowel technologisch als sociaal van aard zijn. Om een einde te maken aan de grote onderhoudsproblemen, lijkt SOA echter een duurzame oplossing te bieden voor op de lange termijn.



figuur 22: Koppelingen bij logistieke dienstverleners met assembled-stop-over strategie en diensten op maat

Zoals in hoofdstuk 4.1 werd beschreven, is SOA ontstaan uit verschillende stromingen in zowel het bedrijfsleven als een stromingen meer van technologische aard. SOA kan gezien worden als een logisch vervolg op ERP. ERP wordt gezien als een systeem dat alle onderdelen binnen een organisatie of enterprise overkoepeld. In dat licht kan SOA beschouwd worden als de overkoepeling van meerdere bedrijven, of een aaneengesloten keten van bedrijven, een supply chain. Bovenstaande figuren geven een bevestiging van het verband tussen de klanthechtheid van de logistieke dienstverlener en de mate waarop SOA een oplossing kan bieden voor de knelpunten. Hoe nauwer de logistieke dienstverlener betrokken is bij de activiteiten van de verlader of de supply chain van de verlader, hoe krachtiger de oplossingen worden die SOA biedt. De koppelingen zijn echter gebaseerd op literatuur en op eigen bevindingen. In het volgende hoofdstuk zullen de verwachte koppelingen geverifieerd worden.

## 6. Resultaten

In het volgende hoofdstuk kijken we naar de resultaten van het onderzoek en vergelijken we deze met het theoretisch model uit hoofdstuk 5. Hierdoor krijgt het model extra empirische waarde. Eerst wordt de methode besproken hoe de gegevens zijn verzameld. Hierna volgen de koppelingen tussen de knelpunten uit hoofdstuk 3 en de voor- en nadelen van hoofdstuk 4. Deze worden wederom tegen elkaar uitgezet. Het verschil met hoofdstuk 5 is dat de koppelingen nu zijn gelegd op basis van gesprekken, interviews en voorbeelden uit de praktijk. Tot slot volgt een vergelijking van het theoretische model en het model dat is voortgekomen uit de interviews. Ontbrekende knelpunten die in de interviews aan het licht kwamen komen eerst aan bod. Daarna wordt gekeken naar de verschillen in de twee modellen.

### 6.1. Methodologie

Het theoretische model is getest in de praktijk door goed te kijken naar praktijkvoorbeelden, het voeren van gesprekken en het afnemen van interviews. Bij interviews zijn een aantal aandachtspunten van belang. Deze staan in Appendix C uitgelegd. Het belangrijkste is dat de geïnterviewden niet beïnvloed worden door de manier waarop de vraag gesteld wordt door de interviewer, en dat de vragen open zijn. Hierdoor praat de geïnterviewde helemaal uit zijn of haar eigen ervaringen en wordt de geïnterviewde gedwongen om meer dan alleen met “ja en nee” te antwoorden. Voor dit onderzoek is het namelijk vooral belangrijk dat de koppelingen worden genoemd. Gezien SOA nu pas langzaam doordringt in het bedrijfsleven, zijn de ervaringen met SOA pril. Harde cijfers over de werking van SOA zijn nog niet bekend, waardoor geïnterviewden vooral spreken over hun verwachtingen. Een verkennend onderzoek is hierbij beter op zijn plaats.

Interviews berusten vooral op meningen van een groep mensen. De representatie van deze groep moet overeenkomen met de werkelijkheid. Het gevaar kan zijn dat deze groep een afwijkende mening heeft over de gevraagde onderwerpen dan modaal. Ook kan het zijn dat men vooroordelen heeft over bepaalde onderwerpen, of dat men beïnvloed wordt door een externe partij. Als onderzoeker moet daarom per gesprek worden afgewogen hoe relevant de opgedane bevindingen zijn en of deze niet te gekleurd zijn [HAV08].

Hier staat tegenover dat het afnemen van interviews een heel geschikt middel is om kennis op te doen over zaken die de onderzoeker zelf niet kan waarnemen of om een visie van de informanten te vormen. Het is daarbij relatief gemakkelijk en goedkoop [HAV08].

De groep van geïnterviewden is in twee delen op te splitsen. De eerste groep heeft vooral veel kennis van technologische zaken, en in dit geval vooral van SOA. De tweede groep heeft zijn kennis vooral liggen bij logistieke dienstverleners en in Supply Chain Management. De thema's van de interviews sluiten hier op aan. De thema's bestaan grofweg uit vier delen: een gedeelte over de huidige situatie wat betreft de IT van logistieke dienstverleners, een gedeelte over de knelpunten op het gebied van IT bij logistieke dienstverleners, wat SOA is en wat de karakteristieken zijn, en hoe deze karakteristieken wel of geen oplossingen kunnen bieden voor de knelpunten van logistieke dienstverleners. Ook hierin is de lijn die in figuur 18 is uitgezet terug te vinden. In plaats van dat er expliciet gevraagd is naar de verschillende strategieën van logistieke dienstverleners, is er gevraagd naar de diensten die de logistieke dienstverleners aanbieden. Hierdoor is toch een goed beeld ontstaan over de complexiteit van de diensten zonder dat de strategieën die in deze scriptie zijn gehanteerd besproken hoefden te worden tijdens de interviews. De oorzaken die ten grondslag liggen aan de knelpunten worden wel behandeld als deze afwijken van implicaties van hoofdstuk 3.3. Dit wordt gedaan in paragraaf 6.3.

### 6.2. Koppelingen

In de volgende paragraaf worden de koppelingen uitgelegd zoals deze uit de interviews naar voren zijn gekomen. Hierbij gaan we uit van de knelpunten en de voor- en nadelen zoals deze ook in het theoretische model van hoofdstuk 5.2 aan bod kwamen. Hierdoor is het mogelijk om het nieuwe model dat in deze paragraaf gevormd wordt te vergelijken op dezelfde punten als het theoretische model. Het vergelijken van de theorie met de praktijk wordt gedaan in paragraaf 6.4. Extra knelpunten bij

logistieke dienstverleners die bij de interviews naar voren kwamen, komen aan bod in paragraaf 6.3. In de tabel staan de koppelingen schematisch weergegeven, net als in hoofdstuk 5.2. Een plus wil zeggen dat de karakteristiek van SOA een oplossing biedt voor het knelpunt, een min wil zeggen dat het juist het knelpunt vergroot, plus/min wil zeggen dat het geen oplossing biedt maar ook niet het knelpunt vergroot, en een nul wil zeggen dat er geen relatie tussen karakteristiek en knelpunt bestaat. In de tekst die volgt op de tabel met koppelingen, worden de koppelingen beredeneerd. Als een koppeling wordt gelegd, wordt het betreffende knelpunt tussen haakjes genoemd met een plus, min of plus/min (die uiteraard corresponderen met de bijbehorende tabel).

### Interoperabiliteit

Interoperabiliteit is de mate van uitwisselbaarheid van gegevens en informatie van een systeem. Een hoge interoperabiliteit betekent dat informatie en gegevens gemakkelijk uitgewisseld kunnen worden. In het volgende stukje wordt gekeken welke knelpunten baat hebben bij een hoge interoperabiliteit.

|                              | Interoperabiliteit |
|------------------------------|--------------------|
| Syntax van berichten         | +                  |
| Semantiek van berichten      | +/-                |
| Geen geld voor investering   | 0                  |
| Beveiliging informatiestroom | -                  |
| Geen geschikt raamwerk       | +                  |
| Inflexibiliteit systemen     | +                  |

tabel 9: *Knelpunten gekoppeld aan interoperabiliteit, praktijk*

Interoperabiliteit speelt voor kleinere bedrijven een kleinere rol dan bij grotere bedrijven. Dit heeft vooral te maken met de betrokkenheid van de kleinere logistieke dienstverleners bij de verladers. Zo werken kleinere logistieke dienstverleners nog vaak met eenvoudige ERP systemen, die op maat zijn gemaakt voor de administratie van de logistieke dienstverlener. Zij hebben hun prioriteiten ergens anders liggen dan bij IT.

Grote bedrijven kunnen eigenlijk niet meer om het gebruik van open standaarden heen. De case van Océ is hier een mooi voorbeeld van. Océ wilde graag alle logistieke processen uitbesteden. Océ verkoopt printers over de hele wereld. Maar er was nog geen logistieke dienstverlener met een distributienetwerk dat zich spande over alle uithoeken van de wereld. Inmiddels hebben een aantal grote logistieke dienstverleners de ambitie en steeds meer mogelijkheden om dit voor elkaar te krijgen. Logistieke dienstverleners bereiken dit door ook uit te besteden aan meer gespecialiseerde logistieke dienstverleners. Het netwerk is dus erg belangrijk voor logistieke dienstverlener, omdat de logistieke dienstverleners onmogelijk in alle uithoeken van de wereld actief kan zijn. Interoperabiliteit is hierbij essentieel, het is ondoenlijk om rekening te houden met ieder systeem van elke partner. Syntax speelt hierbij een erg grote rol. Ondanks allerlei ontwikkelingen op dit gebied (syntax van berichten, +), geeft de syntax van berichten soms toch nog problemen. Ook de semantiek moet goed worden afgestemd, en is van groot belang. Zo zijn er tijdens de interviews verschillende voorbeelden gegeven van gevallen waarbij een geldbedrag verkeerd werd geïnterpreteerd. Er bestaan nog steeds veel problemen met de semantiek van berichten (semantiek van berichten, +/-). De interpretatie van een standaard is minstens zo belangrijk als de vorm.

De geïnterviewden zagen de interoperabiliteit van SOA vooral als een erg groot voordeel voor het uitbesteden van processen. Koppelingen kunnen gemakkelijk gelegd worden. Dit heeft zowel voordelen aan de aanbod- als aan de vraagkant. Doordat logistieke dienstverleners met open standaarden werken, kunnen ze gemakkelijk extra verladers bedienen. Logistieke dienstverleners kunnen misschien wel 10 verschillende systemen draaien, maar als dat allemaal schuil gaat achter één interface, wordt het voor verladers veel aantrekkelijker gebruik te maken van logistieke dienstverleners. Dit zorgt ervoor dat de systemen flexibeler worden (inflexibiliteit systemen, +). Processen worden gemakkelijker meetbaar, vooral ook voor verladers die graag de prestaties van de processen die zij uitbesteden willen meten. Afspraken worden ondubbelzinnig vastgelegd, juist doordat alles via standaarden verloopt. Partners

kunnen elkaar afrekenen op deze afspraken. Hierdoor ontstaat een goed raamwerk voor het transparanter maken van processen (geen geschikt raamwerk, +). Er ligt wel een beveiligingsrisico op de loer. Door de openheid van de communicatie wordt het bedrijf kwetsbaarder voor aanvallen [GOO08]. Dit is ook door de geïnterviewden beaamd (beveiliging informatiestroom, -).

### Herbruikbaarheid

Een aantal problemen in de IT ontstaan doordat er veel verschillende systemen naast elkaar draaien die vrijwel hetzelfde doen. Herbruikbaarheid kan deze complexiteit reduceren. In het volgende stukje kijken we welke invloed de herbruikbaarheid heeft op de knelpunten van logistieke dienstverleners.

|                              | Herbruikbaarheid |
|------------------------------|------------------|
| Syntax van berichten         | +                |
| Semantiek van berichten      | +                |
| Geen geld voor investering   | +                |
| Beveiliging informatiestroom | 0                |
| Geen geschikt raamwerk       | +                |
| Inflexibiliteit systemen     | +/-              |

tabel 10: *Knelpunten gekoppeld aan herbruikbaarheid, praktijk*

Kleinere organisaties zijn meer gericht op een bepaalde doelgroep of handelen hun opdrachten nog gewoon per telefoon, email of fax af. Bij grotere logistieke dienstverleners is het beeld heel anders. Die opereren op zo een grote schaal, dat ze niet om automatisering heen kunnen. Door overnames, door groeispurten, en door in te gaan op hele specifieke eisen van klanten zijn logistieke dienstverleners niet altijd in staat geweest om hun eigen IT goed af te stemmen. De aangeboden diensten zijn vaak niet ontwikkeld vanuit een duidelijke strategie. Het gevolg hiervan is dat logistieke dienstverleners met een heel groot aantal systemen te maken heeft. De grote verscheidenheid in applicaties is een heel groot knelpunt bij (vooral) de grote logistieke dienstverleners. Ter illustratie: bij DHL hebben ze over de 100 verschillende inventory management systemen. De kennis over dergelijke systemen moet behouden blijven om onderhoud te kunnen plegen. Op den duur droogt de kennis toch langzaam op doordat werknemers vertrekken of op andere posities terecht komen. Onderhoud aan deze systemen kost daarom veel geld en tijd. Door applicaties te hergebruiken kan de complexiteit van de IT-huishouding teruggedrongen worden. Bijkomend voordeel is dat dan ook vaker dezelfde standaarden worden gebruikt. De kennis over de vorm en de betekenis van de standaarden wordt hiermee vergroot (syntax van berichten/ semantiek van berichten, +). Ook kunnen nieuwe applicaties snel opgezet worden, hoewel dit minder vaak voorkomt in de praktijk dan werd aangenomen.

Het aanschaffen van softwarepakketten kost veel geld. Bij SAP en Oracle, twee grote softwarebedrijven op het gebied van bedrijfssystemen, zijn ze op het moment bezig met het bouwen van service repositories. Hierdoor kunnen organisaties bepaalde functionaliteiten gebruiken, zonder aan een heel pakket vast te zitten. Dit kan geld besparend zijn, ook voor de kleinere logistieke dienstverleners (geen geld voor investering, +). Het raamwerk waarin de service wordt gebruikt moet dan wel juist zijn.

Logistieke dienstverleners moeten de diensten die ze aanbieden, met daarbij de ondersteunende services, leidend maken. Hierdoor wordt de herbruikbaarheid van de services optimaal benut. Dit is de enige manier om de complexiteit van hun eigen IT-huishouding terug te dringen. Door de services zo *modulair* mogelijk te maken, kan er heel specifiek ingegaan worden op de wensen van een klant (geen geschikt raamwerk, +). Gevolg is wel dat als de klant toch iets anders wil, de logistieke dienstverlener een keuze moet maken. Maken ze de klant weer leidend, of zal de klant water bij de wijn moeten doen? De flexibiliteit van het uiteindelijke systeem is dus wel afhankelijk van de strategie die de logistieke dienstverlener voert (inflexibiliteit systemen, +/-).

**Flexibiliteit**

De flexibiliteit van een organisatie is de snelheid waarmee de organisatie zich kan aanpassen aan veranderende omstandigheden. In het volgende stukje kijken we of flexibiliteit de knelpunten van logistieke dienstverleners beïnvloed.

|                              | Flexibiliteit |
|------------------------------|---------------|
| Syntax van berichten         | +             |
| Semantiek van berichten      | 0             |
| Geen geld voor investering   | -             |
| Beveiliging informatiestroom | 0             |
| Geen geschikt raamwerk       | +             |
| Inflexibiliteit systemen     | +             |

tabel 11: *Knelpunten gekoppeld aan flexibiliteit, praktijk*

Logistieke dienstverleners kunnen worden beschouwd als organisaties die de goederenstromen, maar ook steeds vaker de informatiestromen van een supply chain aan elkaar knopen. DHL heeft voor Philips het gehele logistieke proces spare parts voor consumer electronics overgenomen. Hiervoor is een heel nieuw SAP systeem gebouwd. Dit systeem is alleen zo nauw verweven met dat van Philips, dat het systeem niet gebruikt kan worden ter ondersteuning van logistieke processen van andere bedrijven. Daarvoor is het systeem niet flexibel genoeg. Het ministerie van defensie heeft ook te maken met allerlei logistieke zaken. Dit varieert van het voorzien van eten en munitie aan gelegerde troepen tot het periodiek onderhouden van voertuigen en materiaal. Een missie in Bosnië en in Uruzgan hebben echter hele verschillende doelen: in Bosnië is de missie het land helpen op te bouwen, in Uruzgan wordt nog hevig gevochten. De gebruikte IT van defensie moet flexibel genoeg zijn om beide missies te kunnen ondersteunen.

Door een systeem op te zetten dat met services werkt, kan de context van de service losgekoppeld worden van de interface. Hierdoor maakt het voor de service niet uit of er ontwikkelingsgoederen worden vervoerd of munitie en tanks. Ook kunnen de services die gebruikt worden voor Philips zo opgezet worden dat ze met een kleine aanpassing kunnen worden hergebruikt voor bijvoorbeeld Sony (inflexibiliteit systemen, +). SOA zet de gebruiker van het systeem centraal. Hierdoor wordt het gemakkelijker om processen te monitoren en te sturen. De transparantie van het systeem wordt door de flexibiliteit vergroot (geen geschikt raamwerk, +). Een dergelijk flexibel systeem zorgt ervoor dat er geen onduidelijkheden meer mogen bestaan over de syntax (syntax van berichten, +). Een dergelijk raamwerk opzetten kost wel een hoop geld (geen geld voor investering, -).

Voor kleinere bedrijven heeft deze flexibiliteit meestal niet de hoogste prioriteit. Stand-alone applicaties of eenvoudige ERP software voldoen voorlopig. De IT-huishouding is niet complex genoeg om deze mate van flexibiliteit in te voeren. Diensten zijn nog gemakkelijk te overzien, hiervoor is nog geen complexe IT nodig.

**Governance**

Governance wordt letterlijk vertaald als 'besturing'. In de context van SOA wil dit zeggen dat er toezicht en sturing wordt uitgeoefend om te garanderen dat alle principes en regels van SOA worden nageleefd. In het volgende stukje kijken we welke invloed governance heeft op de knelpunten van logistieke dienstverleners.

|                              | Governance |
|------------------------------|------------|
| Syntax van berichten         | +          |
| Semantiek van berichten      | +          |
| Geen geld voor investering   | -          |
| Beveiliging informatiestroom | -          |
| Geen geschikt raamwerk       | +/-        |
| Inflexibiliteit systemen     | 0          |

tabel 12: *Knelpunten gekoppeld aan governance, praktijk*

Een hoop extra maatregelen zijn nodig om een systeem gebaseerd op regels en principes van SOA draaiende te houden. Ontwikkelaars van services moeten bereid zijn om de services te delen, de services moeten beschikbaar zijn en de services moeten bereikbaar zijn. Dit kost een hoop geld om op te zetten (geen geld voor investering, -). De kwaliteit van de services moeten worden gegarandeerd zoals de bereikbaarheid via het Internet. Daarbij kan het beschikbaar stellen van services op verzet stuiten van ontwikkelaars die de services vooral voor zichzelf willen houden. Bij de AIVD bijvoorbeeld mogen werknemers nog steeds niet naar buiten toe mailen, omdat men bang is dat informatie wordt gelekt (beveiliging informatiestroom, -). De transparantie, die voor veel organisaties erg belangrijk is, kan dus op veel wantrouwen rekenen. In sommige gevallen is SOA dus niet de beste oplossing voor het ondersteunen van diensten. Dit is puur afhankelijk hoe progressief de koers is een logistieke dienstverlener wil varen. UPS bijvoorbeeld heeft een heel goed trackingsysteem dankzij deze progressieve houding. Maar voor bedrijven die liever eerst de kat uit de boom kijken, kan SOA misschien nog een te grote stap zijn (geen geschikt raamwerk, +/-).

Governance zorgt er ook voor dat services op een andere manier beoordeeld worden dan traditionele software. De herbruikbaarheid van services staat hoog in het vaandel. Hierdoor moet de toepassing van de service zo breed mogelijk zijn. Als een service eenmaal breed toe te passen is, kan er ook meer gebruik van worden gemaakt. Dit zorgt ervoor dat de beste services het meest worden gebruikt. Dit draagt bij aan de standaardisatie van interfaces (syntax van berichten, +), en zorgt ervoor dat men meer bekend raakt met de semantiek van berichten (semantiek van berichten, +).

### Lange termijn effecten

Er is een hoop tijd nodig voordat de vruchten geplukt kunnen worden van systemen die volgens de principes en regels van SOA werken (zie ook hoofdstuk 4.7). Het volgende stukje bekijkt welke invloed de lange termijn effecten hebben op de knelpunten van logistieke dienstverleners.

|                              | Lange termijn effecten |
|------------------------------|------------------------|
| Syntax van berichten         | -                      |
| Semantiek van berichten      | -                      |
| Geen geld voor investering   | -                      |
| Beveiliging informatiestroom | -                      |
| Geen geschikt raamwerk       | +/-                    |
| Inflexibiliteit systemen     | 0                      |

tabel 13: *Knelpunten gekoppeld aan lange termijn effecten, praktijk*

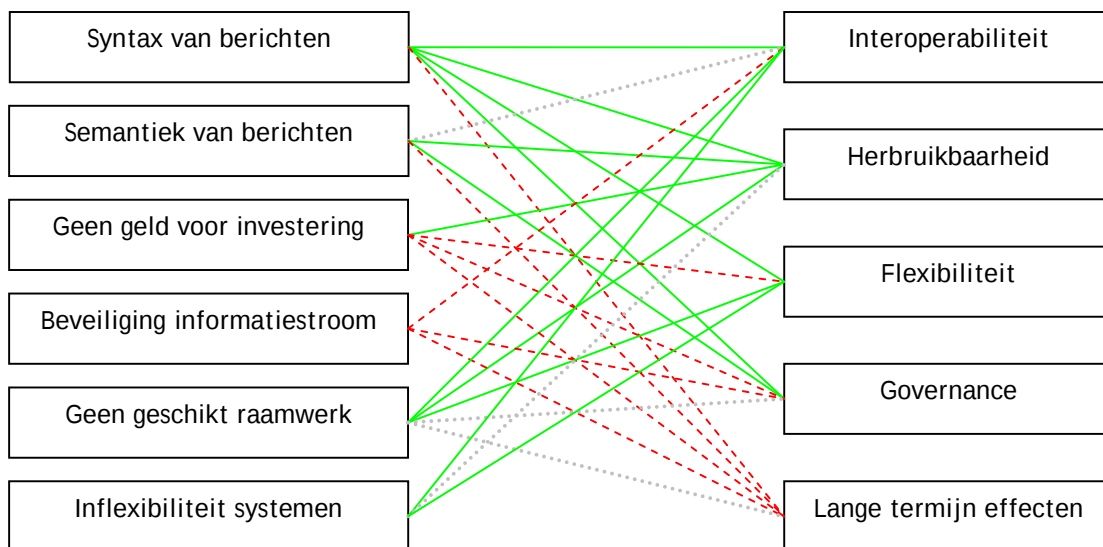
De geïnterviewden geven aan dat het lastig blijft om tot een compromis te komen wat betreft de open standaarden (syntax van berichten, -). De integratie van applicaties gaat hierdoor ook lastig, en dus ook de integratie van processen. In het verleden is al een poging gedaan om een standaard te introduceren voor het uitwisselen van berichten, genaamd *EDIFACT (Electronic Data Interchange For Administration, Commerce, and Transport)*. EDIFACT was echter lastig in de omgang, mede doordat het lastig te lezen was voor mensen. Daardoor werd het aan elkaar knopen van applicaties niet vergemakkelijkt. De standaard *XML (Extended Markup Language)* lijkt meer succes te hebben, maar nog steeds blijft het lastig om de semantiek te vatten (semantiek van berichten, -). Gebruik op grote schaal kan de doorslag geven. Een grote verlader kan dit initiatief nemen, of onder druk van overheden. Zo is het bijvoorbeeld

gelukt om bij veevervoer, met name voor de pluimvee industrie, tot een algemeen geaccepteerde standaard te komen door druk van overheden.

De implementatie van SOA is nog een groot knelpunt. Er is geen applicatielandschap tot nog toe die de principes van SOA helemaal heeft weten te benutten. Voor alsnog zijn het heel veel ideeën en theorieën. Vragen als hoe services het beste kunnen worden ingericht en hoe repositories moeten worden gevuld zijn nog steeds niet helder te beantwoorden (geen geschikt raamwerk, +/-). Verladere willen al wel dat logistieke dienstverleners aan al hun eisen kunnen voldoen (voorbeeld Océ), maar willen daar niet in investeren. Logistieke dienstverleners willen wel een flexibel systeem gebruiken, maar willen dat verladers meebetalen aan de ontwikkeling daarvan. Om uit deze impasse te raken moeten logistieke dienstverleners nadenken hoe zij service oriëntatie willen toepassen (geen geld voor investering, -).

Zoals al aangegeven onder het kopje *Interoperabiliteit*, is beveiliging iets waar nog veel in geïnvesteerd moet worden. Deze problemen zullen pas op de lange termijn verholpen kunnen worden (beveiliging informatiestroom, -). Een mogelijkheid is het *Jericho initiatief*. Het idee achter dit concept is dat het onmogelijk is om een groot systeem dat over meerdere organisaties loopt helemaal te beveiligen, en dat dan ook gewoon niet gebeurt. Alleen de belangrijke informatie wordt beveiligd, de rest van het systeem wordt gewoon helemaal opengegooid zodat men niet oneindig meer bezig is met dichten van gaatjes.

In figuur 23 zijn de koppelingen tussen de knelpunten van logistieke dienstverleners met de karakteristieken van SOA schematisch weergegeven. Voor een oplossing van het knelpunt is de groene lijn gebruikt, rood is gebruikt voor een negatief effect van een karakteristiek van SOA op het knelpunt. Als er geen oplossing wordt geboden voor het knelpunt, is de lijn grijs gestippeld. Als er geen relatie bestaat tussen knelpunt en karakteristiek, dan is er geen lijn getrokken.



figuur 23: Koppeling knelpunten met karakteristieken, praktijk

### 6.3. Extra knelpunten

Tijdens de interviews kwamen extra knelpunten bij logistieke dienstverleners aan het licht. Opvallend vaak werd de pragmatische instelling van logistieke dienstverleners aangehaald tijdens de interviews, die vooral bij de kleinere organisaties terug te vinden is. Zij zien IT vooral als een administratieve last dat verplicht wordt gesteld door de overheid. Liever besteden ze hun geld aan navigatieapparatuur of aan nieuwe vrachtwagens dan aan complexe informatiesystemen. Een heel typisch voorbeeld werd genoemd tijdens de interviews. Een logistieke dienstverlener zonder activa (*4PL, Forth-Party Logistics*)

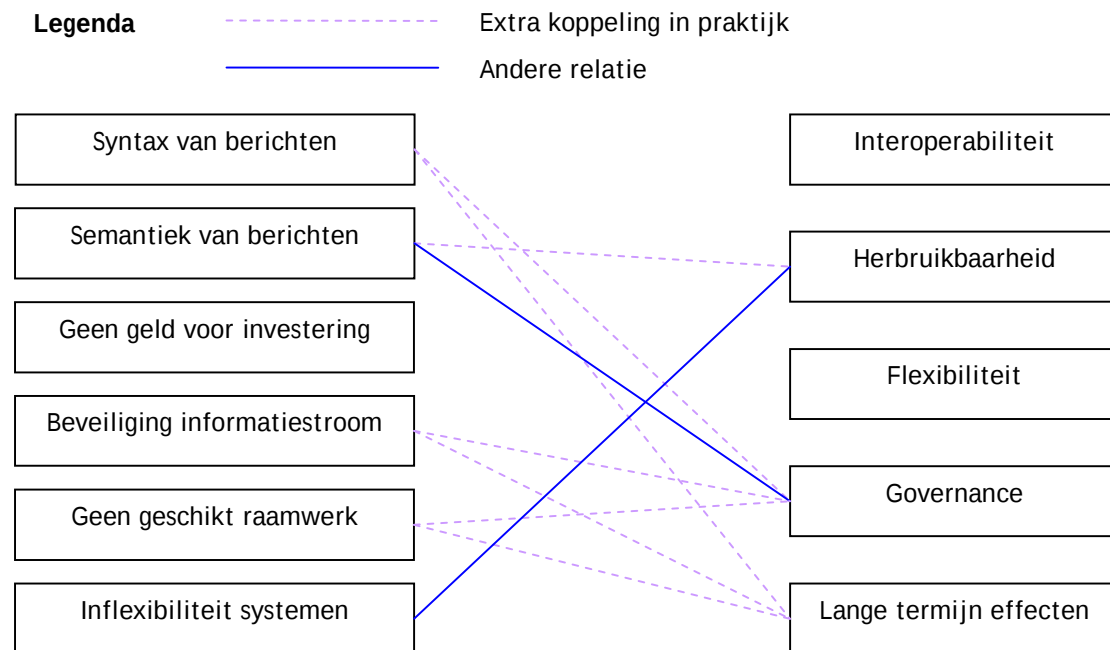
nam een opdracht aan van een verlader voor een bepaalde prijs. De 4PLer stuurde deze opdracht automatisch door naar al zijn partners uit zijn netwerk, in de veronderstelling dat er vast wel iemand zou zijn die de opdracht op zou pakken. De logistieke dienstverleners die waren aangesloten bij het netwerk van de 4PL wilden echter niet rijden voor de afgesproken prijs, en dus kwam uiteindelijk niemand opdagen. De 4PLer had zelf geen vrachtwagens om het pakketje te vervoeren, en dus werd de opdracht teruggegeven. Uit tijdsnood belde de verlader snel rond en vond een plaatselijke transportbedrijf (dat Internet vooral als middel zag om email te versturen) bereid om het pakketje alsnog te vervoeren. Het plaatselijke transportbedrijf voelde zich vervolgens gesterkt in zijn opvatting dat al die IT nergens voor nodig was.

Bij de grotere logistieke dienstverleners wordt inmiddels al wel volop geprofessionaliseerd. Ze moeten mee groeien met de verwachtingen van hun klanten, en kunnen daarom niet meer om complexe informatiesystemen heen. Desondanks bestaat er veel weerstand bij de invoering van nieuwe systemen, die bedoeld zijn om de complexiteit te verkleinen. Veel afdelingen bij logistieke dienstverlener zijn kleine eilandjes. De afdelingen denken vooral aan zichzelf, en deze opvatting is kenmerkend voor de logistieke dienstverlening. Als gevolg van die eilandjes is een overkoepelend systeem introduceren heel erg lastig. Als er een nieuw systeem moet komen voor het traceren van goederen van een distributiecentrum in Nederland naar Spanje, is niet de discussie hoe dat systeem eruit moet komen te zien, maar wie opdraait voor de kosten. Is dat de Spaanse of de Nederlandse afdeling? Daarbij moeten lopende contracten uitgediend kunnen worden, dus gaat er sowieso een hele tijd overheen voordat de oudere systemen definitief kunnen worden afgeschaft.

SOA lost deze knelpunten niet op. SOA is puur iets technologisch, terwijl dit allemaal knelpunten zijn met een grote menselijke factor. Hiervoor moet men de pragmatische instelling van een organisatie of zelfs sector veranderen. Er moet ruimer gedacht worden, over grenzen van organisaties heen. Zolang dit niet gebeurt, zet ook SOA weinig zoden aan de dijk. In het theoretische gedeelte zijn deze knelpunten niet of nauwelijks aan bod gekomen, omdat we ons vooral hebben gefocust op het technologische aspect van de logistieke dienstverleners, en niet op de menselijke aspecten.

#### **6.4. Vergelijking**

In deze paragraaf kijken we naar het verschil tussen figuur 19 en figuur 23, en proberen de verschillen te verklaren. Een aantal verschillen zijn opmerkelijk. Deze verschillen staan schematisch afgebeeld in figuur 24. De paarse lijnen zijn koppelingen die extra zijn gelegd in het empirische model. De blauwe lijnen zijn koppelingen die in beide modellen zijn gelegd, maar van elkaar verschillen. In het theoretische model zijn geen koppelingen gelegd die niet voorkwamen in het empirische model.

figuur 24: *Verschillen tussen theorie en praktijk*

Als we figuur 18 uit hoofdstuk 5.1 bekijken zien we dat de knelpunten voortkomen uit implicaties van drie generieke strategieën voor logistieke dienstverleners. Deze strategieën zijn niet genoemd in de interviews. Het is dus goed mogelijk dat er hele andere redenen ten grondslag liggen aan de koppelingen uit figuur 23 die wel overeenkomen met de koppelingen uit figuur 19. Deze koppelingen laten we in deze vergelijking buiten beschouwing, en we richten ons op de verschillen.

In tegenstelling tot de theorie blijkt 'governance' een oplossing te bieden voor het knelpunt 'semantiek van berichten' bij IT systemen van logistieke dienstverleners. Door het extra management dat nodig is om SOA tot een succes te maken, wordt er aangestuurd op het gebruik van open standaarden en het hergebruik van services. Als bepaalde services vaker hergebruikt worden, wordt de semantiek van de berichten waarmee de service communiceert steeds duidelijker. Dit is de verklaring waarom in het empirische model een koppeling is gelegd tussen de 'semantiek van berichten' en 'herbruikbaarheid'. Door deze extra sturing kan het zijn dat er juist minder open standaarden worden gebruikt, wat ook ten gunste komt aan de syntax van berichten. Hierdoor ontstaan een soort van best-practice open standaarden. Dit verklaart meteen de extra koppeling tussen de 'syntax van berichten' en 'governance' die in het theoretische model niet is gelegd. In het theoretische model werd uitgegaan dat governance juist negatief zou zijn door de extra uitgaven die gepaard gaan met de controle en onderhoud van de services. Ook werd ervan uitgegaan dat bedrijven hun eigen standaarden door wilden drukken, waardoor alsnog afspraken over de betekenis van berichten gemaakt moesten worden.

Volgens de literatuur bevordert de herbruikbaarheid van de services de flexibiliteit van het systeem. Het idee hierachter is dat services deelprocessen ondersteunen. Men hoeft deze deelprocessen alleen maar in elkaar te schuiven, en de juiste service eraan te koppelen. Volgens de geïnterviewden is deze gedachtegang juist, ware het niet dat verladers altijd wel een aanmerking hebben op een service. Hierdoor ontstaan er heel veel verschillende services die nauwelijks van elkaar verschillen, en wordt de herbruikbaarheid van de services niet benut. Logistieke dienstverleners moeten verladers meer dwingen om gebruik te maken van dezelfde services. Hierdoor neemt de flexibiliteit weer af.

In het theoretische model werd uitgegaan dat het vooral lastig zou zijn om de semantiek van berichten op elkaar af te stemmen. In de praktijk blijkt echter dat er ook een hoop open standaarden zijn. Het

blijkt een lastige klus te zijn om met al deze open standaarden rekening te houden of om ze überhaupt op elkaar afgestemd te krijgen. Het kan erg lang duren een overeenstemming te bereiken. Dit verklaart de extra koppeling die is gelegd tussen 'lange termijn effecten' en 'syntax van berichten'.

Er is nog geen werkend systeem op de markt dat zich strikt aan de uitgangspunten van SOA houdt. Dus vooralsnog is SOA niet meer dan een samenraapsel van ideeën en theorieën. Een oorzaak hiervan is dat er nog geen definitieve overeenstemming is bereikt over hoe een SOA er in de praktijk nou uitziet tussen alle betrokken partijen. Het kan lang duren voordat alle partijen op dezelfde lijn zitten en ervaringen met een implementatie van SOA gedeeld kunnen worden. Dit verklaart de koppeling tussen 'geen geschikt raamwerk' en 'lange termijn effecten'.

Sommige organisaties hebben nog steeds hun vraagtekens over de beveiliging van systemen. Vandaar dat er in het empirische model een negatieve koppeling is gelegd tussen governance en de beveiliging van de informatiestroom. Ook kunnen organisaties hun bedenkingen hebben of de transparantie van SOA wel zo goed is voor hun organisaties. Dit verklaart de koppeling tussen 'geen geschikt raamwerk' en 'governance'. De ontwikkelingen op het gebied van beveiliging is nog steeds iets waaraan gewerkt moet worden. Regelmatig duiken er weer virussen op, er worden regelmatig systemen gehackt, etc. De meeste kraken zijn een oorzaak van menselijke fouten, door paswoorden onbeveiligd op te sturen bijvoorbeeld. In de geraadpleegde literatuur ging men uit van het beveiligen van informatiestromen met behulp van SSL en dergelijke technologieën, maar aan menselijke fouten is vooralsnog weinig te doen. Om systemen goed beveiligd te krijgen is een andere aanpak vereist, zoals bijvoorbeeld het al eerder genoemde Jericho initiatief. Het zal nog een tijd duren voordat het wantrouwen van een aantal logistieke dienstverleners ten aanzien van technologie is opgehelderd. Dit verklaart de extra koppeling in het empirische model tussen lange termijn effecten en de beveiliging van de informatiestroom.

De twee modellen verschillen niet heel veel. De verwachte koppelingen sluiten redelijk aan bij de empirische bevindingen. Bij het empirische model had de menselijke factor alleen wel veel extra koppelingen tot gevolg. De bedrijfskant van het verhaal is dus een erg belangrijke, waar rekening mee gehouden moet worden bij de afweging of SOA wel of geen oplossing kan bieden voor een logistieke dienstverlener. In het laatste hoofdstuk wordt gekeken naar de resultaten van het onderzoek, en naar de implicaties daarvan.

## 7. Conclusie

In het volgende hoofdstuk wordt gekeken naar de resultaten van het onderzoek, met betrekking tot de onderzoeksvragen. Hierna kijken we naar de praktische implicaties van deze resultaten. Waar mogelijk worden aanbevelingen gedaan. Tot slot wordt gekeken naar de gehanteerde onderzoeksmethode, en waar meer onderzoek vereist is.

### 7.1. Resultaten

In de volgende paragraaf worden antwoorden geformuleerd op de onderzoeksvragen (zie hoofdstuk 1.3). Het eerste gedeelte gaat over de trends die bestaan bij logistieke dienstverleners, en de knelpunten die hierbij komen kijken. Het tweede gedeelte gaat over het software paradigma SOA en hoe SOA in een organisatie ingezet kan worden. Het derde gedeelte gaat over de inzetbaarheid van SOA bij logistieke dienstverleners specifiek.

#### Logistieke dienstverleners

Het onderzoek begon met de vraag wat logistieke dienstverleners nou eigenlijk zijn. Op basis van literatuur is een brede definitie gedefinieerd waar uit wordt gegaan van een organisatie die zich bezig houdt met logistieke processen, maar zelf geen goederen produceert voor de eindconsument.

Er zijn een aantal redenen aan te wijzen waarom gebruik wordt gemaakt van logistieke dienstverleners. Verladers willen zich vaak richten op hun kernactiviteiten. Hiermee willen ze hun concurrentiepositie op de markt verstevigen. Logistieke processen horen daar niet altijd bij. Een andere reden is dat organisaties vaak niet de expertise hebben op het gebied van logistiek, wat logistieke dienstverleners wel hebben. Ook kan het zijn dat een verlader wil kunnen inspringen op veranderingen in de markt waarin de verlader opereert. Door logistieke processen uit te besteden worden verladers flexibeler.

Aan de hand van een drietal generieke strategieën voor logistieke dienstverleners geformuleerd door Bask (zie hoofdstuk 3.2) zijn implicaties voor de IT-huishouding van logistieke dienstverleners geformuleerd. Hieruit blijkt dat de integratie van systemen essentieel is om processen die over meerdere organisaties heen lopen te ondersteunen. De mogelijkheden om applicaties snel en goedkoop te integreren zijn nu nog te beperkt. Omdat processen worden uitbesteed, willen organisaties in staat zijn deze processen te monitoren. De uitbesteede processen zullen dus transparanter moeten worden. Hiervoor is nog geen goed raamwerk beschikbaar. Dit geldt voor zowel op organisatorisch als op technologisch vlak. De beveiliging van de processen is erg belangrijk. Een aantal verladers stellen hele specifieke eisen aan de dienstverlening van logistieke dienstverleners. De gebruikte systemen van logistieke dienstverleners zijn vaak te inflexibel om aan deze wensen te kunnen voldoen, waardoor er een uiterst complexe IT-huishouding ontstaat. De marges van kleinere logistieke dienstverleners zijn zo klein, dat investeringen in IT een lage prioriteit hebben.

#### Service Oriented Architecture

Al sinds het begin van de computerindustrie hebben organisaties gebruik gemaakt van de mogelijkheden van computers. In eerste instantie werd per bedrijfsproces een applicatie gebouwd ter ondersteuning. Al deze systemen binnen een organisatie werden langzaam geïntegreerd met elkaar. Ook werd de functionaliteit van dergelijke systemen vergroot. Deze systemen voldoen echter niet meer aan de huidige eisen van organisaties. Organisaties willen graag bedrijfssystemen die processen ondersteunen die over de muren van een organisatie heen lopen.

SOA vormt een raamwerk om applicaties op een flexibele manier aan elkaar te koppelen. Voorheen werden applicaties vaak rechtstreeks gekoppeld aan processen. Bij het uitbesteden van deze processen ontstond de vraag wat te doen met de applicaties. SOA ontkoppelt de applicaties van de bedrijfsprocessen. Daarvoor in de plaats worden services aan deelprocessen gekoppeld. Door de services op de juiste manier te schakelen, kan een heel hoofdproces of zelfs megaproces ondersteund worden door services. De services onttrekken hun informatie uit bestaande applicaties. Doordat de services communiceren met open standaarden en breed beschikbaar zijn, is een schakeling van services

veel sneller gemaakt dan dat applicaties met elkaar geïntegreerd kunnen worden. Hierdoor ontstaat een loosely coupling van bedrijfsprocessen en applicaties, waardoor bedrijven veel flexibeler worden.

SOA zorgt voor meer flexibiliteit van een organisatie, interoperabiliteit van de IT en herbruikbaarheid van software. De indirecte voordelen hiervan zijn dat organisaties sneller kunnen reageren op veranderingen in de markt, en dat men minder tijd kwijt is aan het onderhouden van systemen. Er bestaan wel zorgen over de invoering van SOA. SOA zou pas vruchten afwerpen nadat een systeem een tijdje draait, en ook de investering in een dergelijk systeem is groot. Daarbij moet gezorgd worden dat de services de juiste granulariteit krijgen. Ook moet actief gezorgd worden dat de services beschikbaar zijn, dat organisaties bereid zijn de services te delen en dat de services vindbaar zijn.

### **SOA bij logistieke dienstverleners**

Uit de interviews en op basis van de literatuur blijkt SOA voor een aantal knelpunten van logistieke dienstverleners een oplossing te kunnen bieden. Met name de interoperabiliteit van systemen wordt drastisch verbeterd. Organisaties verplichten zich tot het gebruik van open standaarden voor communicatie tussen de services. De herbruikbaarheid van services draagt bij aan het gebruik van best-practice open standaarden, en kan zorgen voor een grote reductie in complexiteit van interne bedrijfssystemen van logistieke dienstverleners. Services kunnen snel aan andere services gekoppeld worden, waardoor een organisatie veel flexibeler wordt. Dit is een oplossing van het grootste knelpunt dat momenteel bij logistieke dienstverlener speelt. Door de juiste services te implementeren, kunnen processen veel transparanter worden gemaakt.

SOA biedt technologisch gezien wel een goed raamwerk voor het ondersteunen van processen die over de grenzen van een organisatie heen lopen. De beveiliging van de informatiestroom is nog wel een puntje van aandacht. Op organisatorisch vlak moet er bij logistieke dienstverleners veel veranderd worden. Door de pragmatische instelling van veel logistieke dienstverleners stuit de introductie van nieuwe IT op veel verzet. Voordat SOA een succes kan worden, moet de instelling van een organisatie veranderen. Men moet meer denken in schaalgroottes als van een supply chain. Door de huidige organisatiestructuur denkt men vooral nog in afdelingen.

Grote software bedrijven als SAP en Oracle zijn op het moment druk bezig met het maken van een service directory. Nog maar weinig bedrijfssystemen werken volgens de principes van SOA, laat staan bij logistieke dienstverleners. SOA is vooralsnog een verzameling van ideeën en theorieën. De vraag hoe SOA er in de praktijk precies moet komen uit te zien, blijft onbeantwoord.

## **7.2. Discussie en implicaties**

In de volgende paragraaf kijken we wat de implicaties zijn van de resultaten van het onderzoek. Dit wordt gedaan voor logistieke dienstverleners, maar kan ook door Capgemini gebruikt worden. Hierna behandelen we het onderzoek, en behandelen we mogelijke vervolgonderzoeken.

### **Reflectie**

Waar logistieke dienstverleners nog niet zo lang geleden vooral processen zoals transport en opslag voor hun rekening namen, worden steeds meer aanverwante diensten ook aangeboden. Het dirigeren van de goederenstroom in de supply chain wordt steeds meer de verantwoordelijkheid van logistieke dienstverleners. Twee richtingen waarin logistieke dienstverleners divergeren zijn te identificeren: de richting waar simpele diensten worden aangeboden, en de richting waarin complexe diensten worden aangeboden die op maat gesneden kunnen worden voor de verlader. Zie ook hoofdstuk 3.2.

SOA is een raamwerk dat vooral veel voordelen biedt voor bedrijven die flexibel willen kunnen inspelen op de vraag vanuit de markt. In het geval van logistieke dienstverleners zijn dit de verladers. Voor logistieke dienstverleners die zich vooral richten op de simpele en standaard diensten, biedt SOA wel een aantal voordelen. Maar deze wegen niet op tegen de nadelen, van met name de kosten. Er zijn andere alternatieven die minstens even goed werken.

De vraag is of SOA een geschikt raamwerk biedt voor logistieke dienstverleners die diensten op maat leveren. SOA is ontstaan vanuit de technologiekant, en biedt in ieder geval oplossingen voor onderhoudsproblemen. Op functioneel gebied is SOA nog erg nieuw, en organisaties worstelen met de vraag wat service oriëntatie nou precies kan betekenen voor de organisatie (zie hoofdstuk 6.2). Vooruitstrevende organisaties kunnen veel winst behalen door nu al over te schakelen op SOA. Het risico dat SOA niet de gouden bergen brengt die zijn beloofd is echter ook aanwezig. Ten eerste moeten duidelijke en heldere afspraken komen over de te gebruiken open standaarden, branchebreed. Dit is zowel een taak van de logistieke dienstverleners, maar ook van de verladers. Vervolgens moet er goed worden nagedacht welke services gebruikt moeten worden. En ten derde, en dat is waarschijnlijk de grootste drempel, moet de instelling van de logistieke dienstverlener, die zich in eerste instantie vooral op operationele activiteiten richt, worden gewijzigd. De beoordeling van services kan niet hetzelfde werken als bij traditionele software, omdat services voor een veel bredere context inzetbaar dienen te zijn. Software ontwikkelaars moeten software anders creëren. Maar bovenal moeten logistieke dienstverleners een ander beleidsplan hebben, die meer gefocust is op de organisatie binnen een supply chain, in plaats van allerlei kleine eilandjes van afdelingen (zie hoofdstuk 6.3). Iedereen moet de voordelen van SOA inzien, niet alleen de mensen die bovenin de organisatie staan. Hier ligt de grootste uitdaging.

SOA heeft ook een grote impact op softwarehuizen en de zogenaamde software integrators, de organisaties die de IT inrichten voor bijvoorbeeld logistieke dienstverleners. Grote softwarehuizen die van oudsher oppermachtig waren wat betreft bedrijfssystemen zullen zelf ook moeten vernieuwen. De softwarehuizen onderling zullen intensief met elkaar moeten gaan samenwerken om SOA te doen laten slagen. Zolang ieder softwarehuis met zijn eigen standaarden aan komt zetten, blijven dezelfde integratieproblemen zoals die er nu zijn bestaan. De kracht van SOA schuilt juist in het feit dat open standaarden door een hele branche worden gesteund. Hier liggen ook kansen voor software integrators. Software integrators zijn in staat om op grote schaal open standaarden in te voeren bij zowel verladers als logistieke dienstverleners, omdat zij in het verleden altijd al zaken deden met de eindgebruikers van de bedrijfssystemen. Hiermee kunnen zij klanten binden aan hele specifieke services, bijvoorbeeld hun eigen services.

### **Vervolgonderzoek**

In het onderzoek zijn een paar zaken naar voren gekomen die meer onderzoek nodig hebben. In de volgende paragraaf worden deze benoemd, en worden er suggesties gegeven voor eventueel vervolgonderzoek op dit onderzoek.

Toen dit onderzoek startte, werd bewust gekozen om SOA te benaderen vanuit een technologische inslag, gezien dit onderzoek is uitgevoerd onder toezicht van het Nijmeegs Instituut voor Informatica en Informatiekunde. Achteraf gezien had in het theoretische model meer rekening gehouden mogen worden met (pragmatische) instelling van logistieke dienstverleners. Deze heeft veel invloed op het succesvol introduceren van SOA in een organisatie. Als vervolgonderzoek is een onderzoek naar de adaptie van SOA bij logistieke dienstverleners erg nuttig.

Omdat er nog geen werkende systemen op de markt zijn die zich zuiver houden aan de principes van SOA, worden veel eigenschappen met daaruit voortvloeiende voor- en nadelen van SOA veronderstelt. Problemen met de werking van SOA of het ontwikkelen van services kunnen daarom nog niet getest worden. Het vergelijken van SOA systemen met elkaar zou een erg waardevolle bijdrage zijn aan de discussie hoe de implementatie van SOA eruit zou moeten zien of hoe services ontwikkeld dienen te worden.

Een interessant aspect dat naar voren kwam in deze scriptie is het tot stand komen van open standaarden. Het gereedschap is aanwezig, maar blijkbaar speelt een grote politieke factor een rol bij het tot stand komen van open standaarden in de logistieke dienstverlening. Een onderzoek hoe open standaarden tot stand komen, en hoe druk van bijvoorbeeld overheden invloed kan hebben op het gebruik ervan, zou een waardevolle bijdrage leveren hoe problemen rondom syntax en semantiek opgelost kunnen worden.

## Referenties

- [AKK03] Akkermans, H. A. & Bogerd, P. & Yucesan, E. & Wassenhove, L. N. van, *The impact of ERP on supply chain management: Exploratory findings from a European Delphi study*, European Journal of Operational Research 146, 2003, pp. 284 – 301.
- [BAS01] Bask, A. H., *Relationships among TPL providers and members of supply chains – a strategic perspective*, MCB University Press, Journal of Business & Industrial Marketing, Vol. 16 No. 6, 2001, pp. 470-486.
- [BIC06] Bichler, M. & Lin, K.J., *Service-Oriented Computing*, Computer, Vol. 39 No. 3, 2006, pp. 99-101
- [BIR06] Birman, K., *The Untrustworthy Web Services Revolution*, Computer, Vol. 39 No. 2, 2006, pp. 98-100
- [BOL03] Bolumole, Y. A., *Evaluating the Supply Chain Role of Logistics Service providers*, University of North Florida, 2003, The International Journal of Logistics Management, Vol. 14 No. 2, 2003, pp. 93-107
- [BRU05] Bruce Silver Associates, BPMInstitute.org, *The 2006 BPMS Report*, <http://www.bpminstitute.org/bpmsreport.html>, 2005.
- [BUS07] Bussler, C., *The Fractal Nature of Web Services*, Computer, Vol. 40 No. 3, 2007, pp. 93-95
- [CAP07] Capgemini, *Integrated Architecture Framework*, [http://www.capgemini.com/services/soa/ent\\_architecture/iaf/](http://www.capgemini.com/services/soa/ent_architecture/iaf/), geraadpleegd 11 februari 2008.
- [CAS03] Casati, F. & Shan, E. & Dayal, U. & Shan, M. C., *Business-Oriented Management of Web Services*, Communications of the ACM, Vol. 46 No.10, 2003, pp. 55-60
- [CHA04] Channabasavaiah, K. & Holley, K. & Tuggle, E.M., *Migrating to a service-oriented architecture*, IBM whitepaper, 2004.
- [CHE06] Cheng, H. K. & Tang, Q. C. & Zhao, J. L., *Web Services and Service-Oriented Application Provisioning: An Analytical Study of Application Service Strategies*, IEEE Transactions on Engineering Management, Vol. 53 No. 4, 2006, pp.520-533
- [CUR03] Curbera, F. & Khalaf, R. & Muhki, N., *The Next Step in Web Services*, Communications of the ACM, Vol. 46 No. 10, 2003, pp. 29-34
- [CUR06] Currie, W. L. & Parikh, M. A., *Value creation in web services: an integrative model*, Journal of Strategic Information Systems, Vol. 15 No. 2, 2006, pp. 153-174
- [DAM01] Damen, J.T.W., *Service-controlled agile logistics*, Logistics Information Management, Vol. 14 No. 3, 2001, pp 185-195
- [DOB07] Dobbe, T., *Service Oriented Architecture, Een 7 lagen architectuur voor service orientatie*, Master thesis informatiekunde, Radboud Universiteit Nijmegen, 2007.
- [ELF04] Elfatraty, A. & Layzell, P., *Negotiating in Service\_oriented Enviroments*, Communications of the ACM, Vol. 47 No. 8, 2004, pp. 103-108
- [FER03] Ferris, C. & Farrel, J., *What are Web Services?*, Communications of the ACM, Vol. 46 No. 6, 2003, p. 31

[FRE02] Fremantle, P. & Weerawarana, S. & Khalaf, R., *Enterprise Services*, Communications of the ACM, Vol. 45 No. 10, 2002, pp. 77-82

[GOO08] Techzine, *Google Apps misbruikt door oplichters*, <http://www.techzine.nl/nieuws/15693/Google-Apps-misbruikt-door-oplichters.html>, geraadpleegd 16 april 2008

[HAG01] Hagel III, J. & Brown, J. S., *Your next IT Strategy*, Harvard Business Review, Vol. 79 No. 9, 2001, pp. 25-27

[HAV08] Paul ten Have, Interviews, <http://www2.fmg.uva.nl/emca/INT.htm>, geraadpleegd 15 april 2008

[HEC99] Hecker, F., *Setting up shop: The business of open-source software*, IEEE Software, Vol. 16 No. 1, pp. 45-51

[HUG06] Hugos, M., *Essentials of Supply Chain Management, Second edition*, John Wiley & Sons, Inc., 2006.

[IEE00] The Institute of Electrical and Electronics Engineers, *IEEE Standard 1471-2000: IEEE recommended practice for architecture description of software-intensive systems*, ISBN: 0738125180, 2000.

[IYE03] Iyer, B. & Freedman, J. & Gaynor, M. & Wyner, G., *Web Services: Enabling Dynamic Business Networks*, Communications of the AIS(11), 2004, pp. 525-554

[JAC06] Jacobs, F. R. & Weston, F. C., *Enterprise Resource Planning (ERP) – A Brief History*, Elsevier B.V., 2006.

[JAM05] Jammes, F. & Smith, H., *Service-Oriented Paradigms in Industrial Automation*, IEEE Transactions on Industrial Informatics, Vol. 1 No. 1, 2005, pp. 62-70

[KRE03] Kreger, H., *Fulfilling the Web Services promise*, Communications of the ACM, Vol. 46 No. 6, 2003, pp. 29-34

[LAN03] Langdon, C. S., *The State of Web Services*, Computer, Vol. 36 No.7, 2003, pp. 93-94

[LEA04] Leavitt, N., *Are Web Services Ready to Deliver?*, Computer, Vol. 37 No. 11, 2004, pp. 14-18

[MAK90] Makelin, M. & Vepsalainen, A. P. J., *Service Strategies 2 – Services, Channels and Information Technology*, HM&V Research, 1990

[MOI05] Moitra, D. & Ganesh, J., *Web Services and Flexible Business Processes: Towards the Adaptive Enterprise*, Information & Management, Vol. 42 No. 7, 2005, pp. 921-933

[NEZ06] Nezhad, H.R.M. & Benatallah, B. & Casati, F. & Toumani, F., *Web Services Interoperability Specifications*, Computer, Vol. 39 No. 5, 2006, pp. 24-32

[OPE06] The Open Group, *Definition of SOA*, <http://www.opengroup.org/projects/soa/doc.tpl?CALLER=index.tpl&qdid=10632>, geraadpleegd 8 juni 2006.

[PAG98] Pagh, J. D. & Cooper, M. C. & Lambert, D. M., *Supply Chain Management: Implementation Issues and Research Opportunities*, Ohio State University, The International Journal of Logistics Management, Vol. 9 No. 2, 1998, pp. 1-19

- [PAP03] Papazoglou, M. P., *Service-Oriented Computing: Concepts, Characteristics and Directions*, Infolab, Tilburg University, Department of Information Systems and Management, 2003
- [PEL03] Peltz, C., *Web Services Orchestration and Choreography*, Computer, Vol. 36 No. 10, 2003, pp. 46-52
- [PLO02] Ploos van Amstel, W. & Goor, A. R. van, *Van logistiek naar supply chain management*, Ten Hagen Stam, ISBN: 9055770493, 2002
- [PRE01] R. S. Pressman, *Software engineering – A practitioner's Approach*, Fifth edition, McGraw-Hill Publishing Company, 2001
- [RAZ98] M. A. Razzaque & C. C. Sheng, *Outsourcing of logistics functions: a literature survey*, International Journal of Physical Distribution & Logistics Management, Vol. 28 No. 2, 1998, pp. 89-107.
- [SCH05] Schmidt, M. T. & Hutchison, B. & Lambros, P. & Phippen, R., *The Enterprise Service Bus: Making service-oriented architecture real*, IBM Systems Journal, Vol. 44 No. 4, 2005, pp. 787-791
- [SHI02] Shirky, C., *Web Services and Context Horizons*, Computer, Vol. 35 No. 9, 2002, pp. 98-100
- [SPE05] Specht, T., Drawehn, J., Thränert, M., Kühne, S., *Modeling Cooperative Business Processes and Transformation to a Service Oriented Architecture*, Proceedings of the Seventh IEEE International Conference on E-Commerce Technology, 2005
- [VAU02] Vaughan-Nichols, S. J., *Web services: Beyond the hype*, Computer, Vol. 35 No. 2, 2001, pp. 18-21
- [VLI98] Vliet, M. van, *Logistieke benchmarking en best practices*, Kluwer Bedrijfsinformatie, 1998.
- [WIT04] Wit, B. de & Meyer, R., *Strategy – Process, Content, Context*, Thomson Learning, 3rd Revised edition, ISBN: 1861529643, 2004
- [YAN03] Yang, B. & Burns, N., *Implications of postponement for the supply chain*, International Journal of Production Research, Vol. 41 No. 9, 2003, pp. 2075-2090
- [ZHA05] Zhao, J. L. & Cheng, H. K., *Web Services and Process Management: A Union of Convenience or a New Area of Research?*, Decision Support Systems, Vol. 40 No. 1, 2005, pp. 1-8

## Bijlagen

### A. Begrippen

#### **bedrijfsproces**

Een verzameling van logisch gerelateerde taken met een gedefinieerd bedrijfsresultaat.

#### **best-practice**

De beste mogelijkheid gekozen op basis van ervaringen.

#### **bullwhip-effect**

Een overschot van goederen doordat leveranciers steeds teveel inkopen.

#### **Client Server Technology**

Een concept waarbij de functionaliteit van een applicatie is opgedeeld in een rekenintensief gedeelte (serverdeel) en een gebruikers gedeelte (clientdeel).

#### **Component gebaseerd**

Een software paradigma waarbij men uitgaat van onafhankelijk van elkaar functionerende onderdelen van een systeem, die voor verschillende organisaties ongeveer dezelfde taken kunnen verrichten.

#### **composite services**

Een service opgebouwd uit andere services.

#### **custom services**

Diensten met betrekking tot importeren en exporteren over landsgrenzen heen.

#### **domein**

Een verzameling van samenhangende entiteiten, zoals een organisatie of branche.

#### **EDIFACT**

Open standaard voor het versturen van berichten, specifiek bedoelt voor de transport sector, opgesteld door de VN.

#### **Enterprise Resource Planning**

Benaming van type bedrijfssystemen waarbij gebruik wordt gemaakt van een enkele database, zodat wijzigingen meteen zichtbaar zijn voor de hele organisatie.

#### **Enterprise Service Bus**

Een integratielaag bovenop bestaande applicaties, waarin gecommuniceerd wordt via open standaarden. Een bus is een topologie afgeleid van de manier waarop een CPU communiceert met andere hardwareonderdelen.

#### **eXtended Markup Language**

Een taal waarmee op een hele algemene vorm erichten vormgegeven kunnen worden, zonder dat dat heel erg ten koste gaat van de leesbaarheid.

#### **flexibiliteit**

De mate waarop een entiteit in staat is zich aan te passen aan een veranderende omgeving.

#### **forth-party logistics**

Een logistieke dienstverlener zonder activa. Door middel van een netwerk onder zich kan een 4PLer fungeren als een tussenpersoon voor verlader en logistieke dienstverlener.

**governance**

Besturing

**Graphical User Interfaces**

De grafische presentatie van gegevens.

**granulariteit**

De mate van detaillering van services.

**halffabricaten**

Product van verwerking van grondstoffen, gebruikt voor de vervaardiging van gebruiksvoorwerpen .

**Human Resource Management**

Het sturen van personeel en het inplannen van beschikbare werknemers, met alle aanverwante taken.

**interface**

Koppeling tussen verschillende computersystemen.

**Integrated Architecture Framework**

Een abstract raamwerk voor het reduceren van de complexiteit van een organisatie, ontwikkelt door Capgemini.

**interoperabiliteit**

De mate van uitwisselbaarheid van gegevens, informatie, documenten, etc.

**Just-in-Time management**

Het sturen van de goederenstroom zodat de goederen precies op tijd worden geleverd.

**kernactiviteiten**

De activiteiten waardoor een organisatie uitspringt boven zijn concurrenten.

**klant order ontkoppelpunt**

Het punt waarop de hoeveelheid te produceren goederen niet meer wordt voorspelt, maar men zich laat leiden door de vraag vanuit de consument.

**klanthechtheid**

De mate van klantcontact.

**loosely coupled**

Niet één-op-één gekoppeld, maar vrijer en losser.

**machine-to-machine communicatie**

Communicatie tussen twee machines, zonder dat daar een menselijke handeling aan te pas komt.

**mainframe model**

Model waarbij de database is ontkoppeld van de rest van de software.

**Manufacturing Resource Planning**

Benaming van type bedrijfssystemen waarbij inkoop-, verkoop- en productieapplicaties aan elkaar gekoppeld zijn.

**Material Requirements Planning**

De eerste generatie bedrijfssystemen waarbij per bedrijfsproces een aparte applicatie draaide. Deze bedrijfssystemen werden vooral gebruikt voor het bijhouden van voorraden.

**modulair**

Een concept waarbij wordt uitgegaan van de kleinst mogelijke en niet verder deelbare onderdelen, die al dan niet met elkaar geschakeld kunnen worden om een groter geheel te vormen.

**modules**

Systeemonderdelen die onafhankelijk van elkaar kunnen opereren ter ondersteuning van bedrijfsprocessen.

**monolithisch**

Een eenheid die niet uit elkaar te halen is.

**object georiënteerd**

Een software paradigma waarbij klassen worden geprogrammeerd, die hergebruikt kunnen worden.

**ontkoppeling**

Loskoppeling

**open standaard**

Een manier van informatie overdragen, waarbij publiekelijk bekend is hoe dit gaat.

**paradigma**

Een model om complexe problemen te kunnen vereenvoudigen.

**Peer-To-Peer architectuur**

Een architectuur waarbij uit wordt gegaan van een netwerk. In dit netwerk zijn alleen entiteiten aanwezig die zowel als client als server kunnen dienen (zie ook Client Server Technology).

**postponement**

Uitstellen

**procedureel**

Door middel van procedures of functies een oplossing voor een probleem vinden.

**proces**

zie bedrijfsproces.

**semantiek**

Betekenis.

**Service Oriented Computing**

Een manier van programmeren waarbij de specifieke kenmerken van services centraal staan.

**service management**

De aansturing, planning en uitvoering van het beleid rondom het onderhouden en bouwen van services.

**single services**

Een service die niet verder is opgebouwd uit andere services.

**SOA-enabled**

Geschikt om te communiceren met andere systemen die service georiënteerd zijn gebouwd.

**Software as a Service**

Traditioneel wordt er eenmalig voor de aanschaf van een applicatie betaald. SaaS gaat uit van betaling voor het gebruik van een applicatie.

**supply chain**

Een keten van organisaties, waarbij de bedrijfsprocessen op het gebied van logistiek over de grenzen van de organisaties heen lopen.

**syntax**

De vorm van een entiteit.

**transparantie**

De zichtbaarheid van een entiteit, zoals bijvoorbeeld een proces. Door middel van transparantie kunnen processen beter gemeten worden.

**uitbesteden**

Het overdoen van bedrijfsprocessen aan een andere organisatie.

**value-added services**

Diensten die, naast hoofdprocessen, een extra toegevoegde waarde leveren voor de verlader.

**verlader**

Een entiteit die een product of goederen te vervoeren heeft.

**Web Services**

Een programmeerbare op XML gebaseerde service.

## **B. Afkortingen**

|         |                                                                         |
|---------|-------------------------------------------------------------------------|
| 3PL     | Third-Party Logistics                                                   |
| 4PL     | Forth-Party Logistics                                                   |
| CST     | Client-Server Technology                                                |
| EDIFACT | Electronic Data Interchange For Administration, Commerce, and Transport |
| ERP     | Enterprise Resource Planning                                            |
| ESB     | Enterprise Service Bus                                                  |
| GUI     | Graphical User Interface                                                |
| HRM     | Human Resource Management                                               |
| IAF     | Integrated Architecture Framework                                       |
| LSP     | Logistics Service Provider                                              |
| m2m     | machine-to-machine                                                      |
| MRP     | Material Requirements Planning                                          |
| MRP II  | Manufacturing Resource Planning                                         |
| p2p     | peer-to-peer                                                            |
| SaaS    | Software as a Service                                                   |
| SCM     | Supply Chain Management                                                 |
| SOA     | Service Oriented Architecture                                           |
| SOC     | Service Oriented Computing                                              |
| TPL     | Third-Party Logistics                                                   |
| WS      | Web Service                                                             |
| XML     | eXtended Markup Language                                                |

### C. Interviewronde

#### Het interview

Het interview is een geaccepteerde sociale vorm van het verzamelen van informatie voor wetenschappelijk onderzoek [HAV08]. Het onderzoeksinterview is voor de geïnterviewde redelijk belangeloos, terwijl er voor de interviewer waardevolle informatie wordt verschaft. Er bestaat altijd het gevaar dat de geïnterviewde beperkt wordt door zijn positie of door persoonlijke belangen om informatie te verschaffen. De achtergrond van de geïnterviewde speelt daarom een grote rol bij het interview. Bij een interview bestaat er een sterke 'taakverdeling'. De geïnterviewde geeft antwoord, terwijl de interviewer de vraag stelt en anticipeert op het antwoord van de geïnterviewde. Dit stelt de interviewer in staat om dieper in te gaan op onderwerpen of op eventuele onduidelijkheden.

Een interviewer kan een gesprek ingaan op twee manieren: helemaal open zonder een idee te hebben waar hij of zij naar op zoek is, of met de nodige voorkennis. Het eerste geval is vooral geschikt als de interviewer iets te weten wil komen over iets waar hij of zij nog helemaal geen verstand van heeft. Het onderzoek is dan ook meer een verkenning op het onderwerp. Een gesprek ingaan met de nodige voorkennis stelt de interviewer in staat om hele gerichte vragen te stellen over bepaalde onderwerpen en details. Het gevaar dat hier op de loer ligt is dat de geïnterviewde beïnvloed wordt door de interviewer, iets wat voorkomen dient te worden. Een gesprek mag wel een bepaalde richting gestuurd worden, maar een geïnterviewde moet nooit een mening in de mond gelegd krijgen. Om een open gesprek te krijgen, is het van belang om open vragen te stellen waar niet 'ja' of 'nee' op geantwoord kan worden.

Voor dit onderzoek is gebruik gemaakt van een vragenlijst. Deze vragenlijst was vooral bedoeld om ervoor te zorgen dat alle onderwerpen die besproken moesten worden, ook de revue passeerden. Het kwam regelmatig voor dat de vragenlijst niet in de onderstaande volgorde aan bod kwam. In de vragenlijst is rekening gehouden met de verschillende achtergronden en disciplines van mensen. Deze komen in de volgende paragraaf aan bod.

#### Doelgroep

De groep van geïnterviewden bestond grofweg uit twee delen: een groep specialisten op het gebied van logistieke dienstverleners (LSP)/ Supply Chain Management, en een groep specialisten op het gebied van bedrijfssystemen, in het bijzonder SOA. De personen die zijn geïnterviewd voor dit onderzoek, staan in onderstaande tabel.

| Naam                   | Functie/ rol                                                    | LSP | SOA |
|------------------------|-----------------------------------------------------------------|-----|-----|
| Jaap Blankevoort       | Accountmanager DHL & Norfolk Lines, Capgemini                   | X   |     |
| Erik van Dort          | Vice President, Consulting Services/ Logistics, Capgemini       | X   |     |
| Michael van der Dungen | Managing Consultant, Sector Products, Capgemini                 | X   | X   |
| Rik Laurens            | Solution Architect, Sector Products, Capgemini                  |     | X   |
| Pino Melis             | Managing Consultant, Sector Products, Capgemini                 | X   |     |
| André Rottier          | Enterprise Architect, Capgemini                                 |     | X   |
| Luc Scheidel           | Operations & supply mgt, Capgemini                              | X   |     |
| Ron Tolido             | CTO Continental Europe & Asia Pacific, Capgemini                |     | X   |
| Frans van der Velden   | Accountmanager, Verkooijen Transport                            | X   |     |
| Joost van der Vlies    | Solution Architect, Capgemini                                   |     | X   |
| Frank Wammes           | Practice Manager P42, Sector Products, Capgemini                | X   | X   |
| Jasper van der Wulp    | Sales leader voor hightech manufacturing & logistics, Capgemini | X   |     |

**De vragen****Interviewronde**

Chris Jager  
Chris.jager@capgemini.com  
+31 6 127 11 046

Student Radboud Universiteit Nijmegen  
Faculteit FNWI  
Master scriptie informatica, MT variant  
S#: 0118060

Versie 4.2  
Zondag 16 maart 2008

**Introductie**

Mijn naam is Chris Jager, en ik ben op het moment aan het afstuderen bij Capgemini, practice P42. Ik ben student aan de Radboud Universiteit Nijmegen, en studeer daar Informatica. Er zijn verschillende master varianten. Deze zijn bedoeld voor meer verdieping in een bepaald vakgebied. Ik heb gekozen voor de master variant Management & Toepassing.

Mijn afstudeerscriptie onderzoekt 'de mogelijkheden die SOA biedt bij het uitbesteden van bedrijfsprocessen aan logistieke dienstverleners'. Hierbij staan twee vragen centraal: 'wat zijn de voor- en nadelen bij het gebruik van SOA', en 'welke behoeftes bestaan er bij logistieke dienstverleners op het vlak van IT'. De bedoeling is dat koppelingen worden gelegd tussen de antwoorden op deze twee vragen.

Deze interviewronde is de toetsing van de literatuurstudie van beide vragen in de praktijk. In de interviews worden vragen gesteld over het gebruik van SOA met de bijbehorende voor- en. Ook worden vragen gesteld over de behoeftes van logistieke dienstverleners met betrekking op de IT die ondersteuning biedt aan bedrijfsprocessen.

**Interview**

In totaal komen vier thema's aan bod: SOA, uitbesteding van bedrijfsprocessen, behoeftes van logistieke dienstverleners en de koppeling tussen SOA en deze behoeftes. Het interview begint met een introductie en eindigt met een afsluiting. Er staat een half uur voor een interview. Vooraf wordt gevraagd of er geluidsopnames gemaakt mogen worden. De geluidsopnames worden alleen voor deze scriptie gebruikt, en daarna vernietigd. De connectie tussen mijzelf en de geïnterviewde wordt uitgelegd.

De vragen zijn genummerd per thema. Per vraag staat de voorkennis die nodig is aangegeven. Waar nodig zijn ook alternatieve vragen geformuleerd. Hierdoor kan toch een beeld worden gevormd zonder de benodigde voorkennis. Tijdens het interview kan er afgeweken worden van de vragen als dit ten goede komt van de resultaten.

**Introductie**

| Nr. | Voorkennis |     | Vragen                                      |
|-----|------------|-----|---------------------------------------------|
|     | SOA        | LSP |                                             |
| 1.1 |            |     | Wat is uw naam?                             |
| 1.2 |            |     | Wat is uw functie?                          |
| 1.3 |            |     | Hoe bent u op deze functie terecht gekomen? |

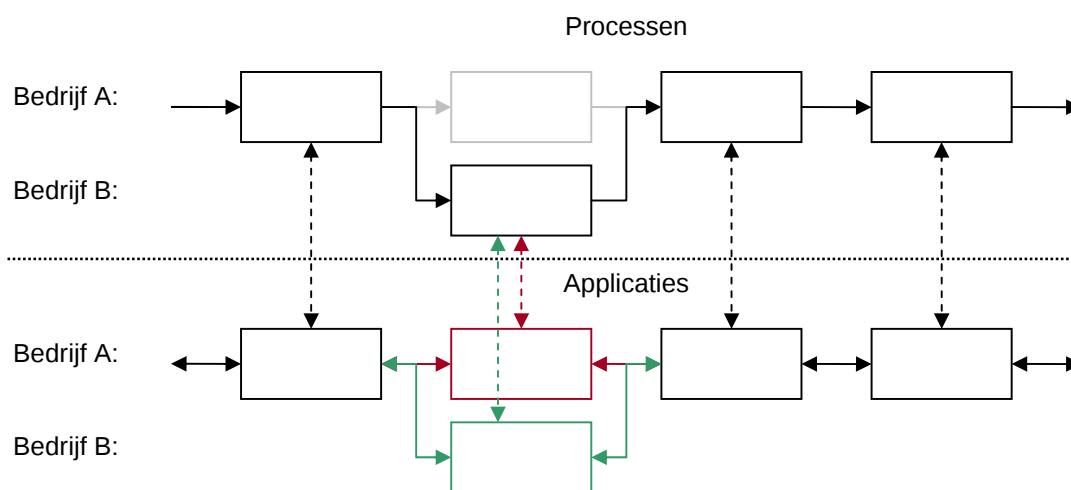
|                                                                                                                                                |            |     |                                                                                                                                  |
|------------------------------------------------------------------------------------------------------------------------------------------------|------------|-----|----------------------------------------------------------------------------------------------------------------------------------|
| 1.4                                                                                                                                            |            |     | Wat is uw ervaring op het gebied van SOA?                                                                                        |
| 1.5                                                                                                                                            |            |     | Wat is uw ervaring op het gebied van uitbesteding van processen aan logistieke dienstverleners?                                  |
| <h2>Service Oriented Architecture</h2>                                                                                                         |            |     |                                                                                                                                  |
| Nr.                                                                                                                                            | Voorkennis |     | Vragen                                                                                                                           |
|                                                                                                                                                | SOA        | LSP |                                                                                                                                  |
| 2.1                                                                                                                                            | X          |     | Wat is SOA volgens u? Kunt u een omschrijving geven van een paar zinnen?                                                         |
| <p>We beschouwen drie dimensies: <i>people</i>, <i>process</i> en <i>technology</i>. Graag wil ik de dimensie <i>technology</i> bespreken.</p> |            |     |                                                                                                                                  |
| 2.2                                                                                                                                            | X          |     | Wat is het grote verschil met systemen die niet op SOA zijn gebaseerd?                                                           |
| 2.3                                                                                                                                            | X          |     | Wat zijn de belangrijkste business benefits/ voordelen van SOA voor het gebruik binnen een organisatie volgens u?                |
| 2.4                                                                                                                                            | X          |     | Wat zijn de belangrijkste business benefits/ voordelen van SOA voor het gebruik buiten de grenzen van een organisatie volgens u? |
| 2.5                                                                                                                                            | X          |     | Welke knelpunten of struikelblokken bij het in gebruik nemen van een SOA verwacht u?                                             |
| 2.6                                                                                                                                            | X          |     | Hoe zou volgens u de integratie van SOA in een bedrijf moeten verlopen?                                                          |
| 2.7                                                                                                                                            | X          |     | Wat voor een invloed heeft SOA volgens u op de andere twee dimensies, <i>people</i> & <i>process</i> ?                           |
| 2.8                                                                                                                                            | X          |     | Hoe ziet het bedrijfsleven er volgens u uit in de toekomst, met betrekking tot SOA?                                              |
| <h2>Logistieke processen</h2>                                                                                                                  |            |     |                                                                                                                                  |
| Nr.                                                                                                                                            | Voorkennis |     | Vragen                                                                                                                           |
|                                                                                                                                                | SOA        | LSP |                                                                                                                                  |
| 3.1                                                                                                                                            |            | X   | Voor welke logistieke dienstverleners heeft u gewerkt, en met wat voor een verladers?                                            |
| 3.3                                                                                                                                            |            | X   | Welke bedrijfsprocessen werden door de verladers veelal uitbesteed aan logistieke dienstverleners?                               |
| 3.4                                                                                                                                            |            | X   | Hoe verlopen deze bedrijfsprocessen?                                                                                             |
| 3.5                                                                                                                                            |            | X   | Welke van de bedrijfsprocessen worden ondersteund door IT-applicaties?                                                           |
| 3.6                                                                                                                                            |            | X   | Hoe en met welke IT-applicaties worden deze bedrijfsprocessen ondersteund?                                                       |

## Behoeftes van logistieke dienstverleners

Bij het uitbesteden van bedrijfsprocessen kunnen verschillende IT-applicaties worden gebruikt. Er zijn 3 scenario's te bedenken:

1. De IT-applicaties van bedrijf A worden gebruikt (rood);
2. De IT-applicaties van bedrijf B worden voortaan gebruikt (groen);
3. Er wordt een heel nieuw systeem gebouwd.

In onderstaand figuur zijn scenario's 1 & 2 weergegeven.



| Nr. | Voorkennis |     | Vragen                                                                                  |
|-----|------------|-----|-----------------------------------------------------------------------------------------|
|     | SOA        | LSP |                                                                                         |
| 4.1 |            | X   | Welke van de drie scenario's wordt het meest toegepast bij logistieke dienstverleners?  |
| 4.2 |            | X   | Welke factoren zijn bepalend voor een dergelijk besluit bij logistieke dienstverleners? |
| 4.3 |            | X   | Waar zitten de knelpunten bij de verschillende scenario's?                              |
| 4.4 |            | X   | Kunt u een voorbeeld geven met een van de bedrijfsprocessen uit 3.5?                    |

## Mogelijkheden SOA voor logistieke dienstverleners

### Alternatief A

| Nr.   | Voorkennis |     | Vragen                                                                                               |
|-------|------------|-----|------------------------------------------------------------------------------------------------------|
|       | SOA        | LSP |                                                                                                      |
| 5.1.a | X          | X   | Hoe voldoen de business benefits/ voordelen van SOA aan de behoeftes van logistieke dienstverleners? |
| 5.2.a | X          | X   | Kunt u een voorbeeld geven hoe SOA voor een bedrijfsproces uit 3.5 voordelen oplevert?               |
| 5.3.a | X          | X   | Welke problemen verwacht u bij logistieke dienstverleners bij het                                    |

|       |   |   |                                                                                      |
|-------|---|---|--------------------------------------------------------------------------------------|
|       |   |   | introduceren van SOA?                                                                |
| 5.4.a | X | X | Wanneer en waarom is het gebruik van SOA af te raden bij logistieke dienstverleners? |
| 5.5.a | X | X | Hoe ziet u de toekomst in van SOA bij logistieke dienstverleners?                    |

### Alternatief B

| Nr.   | Voorkennis |     | Vragen                                                                                                                                                                   |
|-------|------------|-----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|       | SOA        | LSP |                                                                                                                                                                          |
| 5.1.b |            | X   | Hoe belangrijk is het voor logistieke dienstverleners dat de ontwikkelingssnelheid van (composite) applicaties ter ondersteuning van bedrijfsprocessen omhoog gaat?      |
|       |            | X   | Hoe belangrijk is het dat bedrijfsprocessen ontkoppeld zijn van applicaties voor logistieke dienstverleners?                                                             |
|       |            | X   | Kunt u aangeven hoe belangrijk het gebruik van open standaarden en gestandaardiseerde interfaces, voor het uitwisselen van gegevens, is voor logistieke dienstverleners? |
| 5.2.b |            | X   | Kunt u een voorbeeld geven hoe dit voor een proces uit 3.5 voordelen oplevert?                                                                                           |
| 5.3.b |            | X   | Welke problemen verwacht u bij het vaststellen van open standaarden?                                                                                                     |
|       |            | X   | Welke problemen verwacht u bij de introductie van een nieuw systeem bij logistieke dienstverleners?                                                                      |

### Alternatief C

Voor het geval de geïnterviewde niet precies weet wat logistieke dienstverleners voor type processen overnemen, zijn er ook alternatieve vragen geformuleerd. Bij logistieke dienstverleners kunnen drie verschillende soorten diensten geïdentificeerd worden:

1. Simpele routineklussen
2. Standaard diensten met extra waarde toevoeging die ingezet kan worden voor verschillende verladers
3. Complexe diensten die op maat gemaakt moeten worden per verlader

| Nr.   | Voorkennis |     | Vragen                                                                     |
|-------|------------|-----|----------------------------------------------------------------------------|
|       | SOA        | LSP |                                                                            |
| 5.1.c | X          |     | Voor wat voor een soort diensten is SOA in te zetten verwacht u?           |
| 5.2.c | X          |     | Kunt u een voorbeeld geven waarop u zich baseert?                          |
| 5.3.c | X          |     | Wat voor een problemen verwacht u per type dienst met het gebruik van SOA? |
| 5.4.c | X          |     | Bij welk type dienst is het gebruik van SOA af te raden?                   |
| 5.5.c | X          |     | Hoe ziet u de toekomst in van SOA bij logistieke dienstverleners?          |

### Afsluiting

| Nr. | Voorkennis |     | Vragen                                                                                       |
|-----|------------|-----|----------------------------------------------------------------------------------------------|
|     | SOA        | LSP |                                                                                              |
| 6.1 |            |     | Heeft u nog vragen over het onderwerp?                                                       |
| 6.2 |            |     | Heeft u nog vragen over het deze interviewronde?                                             |
| 6.3 |            |     | Bent u beschikbaar voor eventuele aanvullende vragen nadat het interview is uitgewerkt?      |
| 6.4 |            |     | Bent u geïnteresseerd in een (digitale) kopie van de master thesis nadat deze is ingeleverd? |