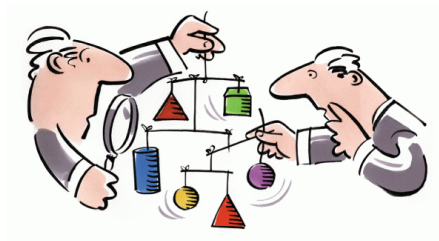


ArchiMate in de praktijk

Modelleren volgens ArchiMate aan de hand van een verzameling good practices



Colofon

Titel : ArchiMate in de Praktijk
Datum : 30 april 2006
Versie : 1.0
Verandering :
Projectreferentie : ArchiMateForum
TI-referentie : TI/RS/2006/019
URL : <https://doc.telin.nl/dscgi/ds.py/Get/File-62468>
Toegangsrechten : Project
Status : Definitief
Redacteur : Hans van Drunen, Egon Willemsz
Bedrijf : ArchiMate Forum
Auteurs : Saco Bekius, Harmen van den Berg, Hans Bosma, Hans van Drunen,
Gerard van Eerdt, Henk Jonkers, Ger Oosting, Robert Slagter, Maarten de
Vos, Egon Willemsz

Inhoudsopgave

1	Introductie	1
1.1	Aanleiding	1
1.2	Doel en doelgroep	1
1.3	Leeswijzer	2
1.4	Oproep voor nieuwe good practices	3
2	Modelleren in de bedrijfslaag	5
2.1	Overzicht good practices	5
2.2	Bedrijfsservice, bedrijfsfunctie en bedrijfsproces	5
2.3	Afbakening bedrijfsservice	7
2.4	Afbakening bedrijfsproces	7
2.5	Afbakening bedrijfsfunctie	8
2.6	Bedrijfsproces en bedrijfsinteractie	9
2.7	Viewpoints voor procesmodellering	10
3	Modelleren in de applicatielaag	12
3.1	Overzicht good practices	12
3.2	Applicatielandschap	12
3.3	Koppeling tussen applicaties	14
3.4	Functionaliteit van een applicatie	15
4	Modelleren in de infrastructuur- / technologielaag	17
4.1	Overzicht good practices	17
4.2	Infrastructuurservices	17
4.3	Infrastructuurlandschap	18
5	Modelleren over de grenzen van lagen heen	20
5.1	Overzicht good practices	20
5.2	Veelgebruikte relaties	20
5.3	Product en applicatieservices	25
5.4	Ondersteuning van batch en online processen	25
5.5	Servicebroker	26
	Referenties	29

1 Introductie

1.1 Aanleiding

Hoe breng ik een applicatielandschap in beeld? Hoe leg ik relaties tussen logische (inrichtingsonafhankelijk) en fysieke bedrijfsprocessen (inrichtingsafhankelijk)? Hoe breng ik in beeld welke afdelingen functioneel eigenaar zijn van welke applicaties? Hoe beeld ik de applicatieportfolio af op de technische infrastructuur? Hoe geef ik aan welke gegevens benodigd zijn bij de levering van welke producten en diensten aan klanten?

Dit zijn allemaal modelleervragen waar je als enterprise architect in de praktijk mee geconfronteerd wordt. Er zijn dan ook veel manieren die worden gehanteerd om antwoord te geven op deze vragen. Veel manieren leidt tot wildgroei en problemen om met de gemaakte platen goed te kunnen communiceren. Achter elk plaatje zit een verhaaltje, maar een plaatje gaat vaak een eigen leven leiden en het verhaaltje raakt verloren, zeker als de oorspronkelijk maker met andere werkzaamheden bezig gaat of zelfs weg is bij een organisatie.

In de praktijk van 'werken onder architectuur' wordt bij steeds meer organisaties gebruik gemaakt van ArchiMate als beschrijvingstaal voor het vastleggen van enterprise architecturen met daarin domeinen als: producten/diensten, processen, organisatie, gegevens, applicaties en technische infrastructuur. Met deze taal is het mogelijk om de globale structuur *binnen* een domein te modelleren en de relaties *tussen* domeinen te modelleren. Een andere belangrijke eigenschap van ArchiMate is de mogelijkheid om modellen op verschillende manieren te visualiseren, gericht op verschillende belanghebbenden. Bovendien levert de taal een formele onderbouwing zodat modellen geanalyseerd kunnen worden (bijvoorbeeld op een aspect als performance). Daarbij is het expliciet niet de bedoeling dat ArchiMate de bestaande talen, zoals BPMN, UML e.d., zal vervangen; het vormt hierop een aanvulling, waarbij wel zo veel mogelijk aangesloten wordt op bestaande standaarden en praktijken.

Ondertussen zijn er veel ervaringen opgedaan met ArchiMate in Nederland en is te zien dat steeds meer herkenbaardere en beter communiceerbare modellen worden opgeleverd. En de populatie van enterprise architecten dat gebruikmaakt van ArchiMate is nog steeds groeiende.

Hierbij is echter ook te zien dat dezelfde typen modelleervraagstukken op verschillende manieren met ArchiMate worden ingevuld, waarbij kan worden gesteld dat niet alle manieren een goede bijdrage leveren aan de communiceerbaarheid van de modellen richting de belanghebbenden. De praktijk leert ook dat je van goede huize moet komen om de taal goed te kunnen vastpakken en toepassen. De drempel is toch nog vrij hoog om er direct mee aan de slag te kunnen gaan.

1.2 Doel en doelgroep

De praktijk heeft behoorlijk wat vragen opgeleverd richting het ArchiMate Forum over de wijze van toepassing van de ArchiMate concepten.

Doel van voorliggend document is om een begin te maken met de beantwoording van deze vragen door middel van het aanreiken van beproefde oplossingen in de vorm van good practices, waarmee de drempel voor de toepassing van ArchiMate in de praktijk aanzienlijk kan worden verlaagd en de effectiviteit van de modellen die worden gemaakt kan worden vergroot.

Dit document is geschreven voor enterprise architecten die ArchiMate in de praktijk willen toepassen als taal voor het vastleggen van modellen van organisaties, de ICT-ondersteuning en de onderlinge relaties daartussen.

Dit document moet worden gezien als aanvulling op de bestaande ArchiMate-resultaten, waaronder (zie de bijlage voor een verwijzing naar deze referenties):

- Enterprise Architecture At Work.
- Inleiding in de ArchiMate-taal.
- Architecture Language Reference Manual.
- Viewpoints Functionality and Examples.
- ArchiMate Quick Reference.

1.3 Leeswijzer

De hierna volgende hoofdstukken bevat een verzameling van good practices die uit de praktijk zijn verzameld. In de beschrijving van elke good practice wordt steeds de volgende structuur gebruikt:

- **Naam in de paragraaftitel:** een herkenbare naam waaraan de inhoud van de good practice te herkennen is.
- **Vraagstelling:** een beschrijving van de vraag die aanleiding is voor het expliciet maken van deze good practice.
- **Oplossing:** een beschrijving en visualisatie van de oplossing die de voorkeur verdient om de vraag te kunnen beantwoorden.
- **Gevolgen:** een beschrijving van de gevolgen van de oplossing voor bijvoorbeeld de communicatie met belanghebbenden, de gevolgen voor de implementatie, e.d.
- **Alternatieven:** een beschrijving van alternatieve oplossingen met de afwegingen om hiervoor te kiezen.
- **Relaties met andere good practices:** een beschrijving van de relaties die er zijn met andere good practices.

In dit document zijn twee soorten good practices verzameld:

- Good practices die antwoord geven op ArchiMate-concept vragen (hoe gebruik je een bepaald concept in de praktijk?).
- Good practices die antwoord geven op modelleervragen.

De good practices zijn op de volgende wijze geordend:

- Modelleren in de bedrijfslaag: hoofdstuk 2.
- Modelleren in de applicatielaag: hoofdstuk 3.
- Modelleren in de infrastructuur-/technologielaag: hoofdstuk 4.
- Modelleren over de grenzen van lagen heen: hoofdstuk 5.

1.4 Oproep voor nieuwe good practices

De inhoud van dit document is zeker niet volledig en laat nog een hoop vragen onbeantwoord. Dit document is bedoeld als groeidocument. Het bevat een eerste overzicht van good practices die uit de praktijk zijn verzameld.

We hopen dat de lezer hiermee in de praktijk aan de slag kan, en dat vervolgens op basis van dit gebruik nieuwe inzichten ontstaan die vervolgens in dit document verwerkt zullen worden.

Heb je modelleervragen waar je graag antwoord op zou willen krijgen? Heb je zelf ook good practices die je wil delen met vakgenoten? Of wil je ook deelnemen aan de werkgroep Gebruik/Tools binnen het ArchiMate Forum, waar we de good practices verzamelen, bespreken en verwerken in dit document? Meld dit dan aan de voorzitters van de werkgroep via het mailadres: usage@archimate.nl. Of telefonisch bij Robert Slagter (053-4850488), Harmen van den Berg (06 5119 8282) of Egon Willemsz (06 2952 6723).

2 Modelleren in de bedrijfslaag

2.1 Overzicht good practices

In dit hoofdstuk zijn de onderstaande good practices opgenomen:

1. Bedrijfsservice, bedrijfsfunctie en bedrijfsproces.
2. Afbakening bedrijfsservice.
3. Afbakening bedrijfsfunctie.
4. Afbakening bedrijfsproces.
5. Bedrijfsproces en bedrijfsinteractie.
6. Viewpoints voor procesmodellering.

2.2 Bedrijfsservice, bedrijfsfunctie en bedrijfsproces

Vraagstelling: In de bedrijfslaag worden ondermeer de concepten bedrijfsservice, bedrijfsfunctie en bedrijfsproces gehanteerd. Het precieze onderscheid tussen deze concepten is niet altijd even duidelijk, dat wil zeggen dat bij toepassing in de praktijk er vaak verwarring ontstaat of iets nu een bedrijfsservice, een bedrijfsfunctie of een bedrijfsproces is. Hieronder geven we definities en omschrijvingen van deze concepten.

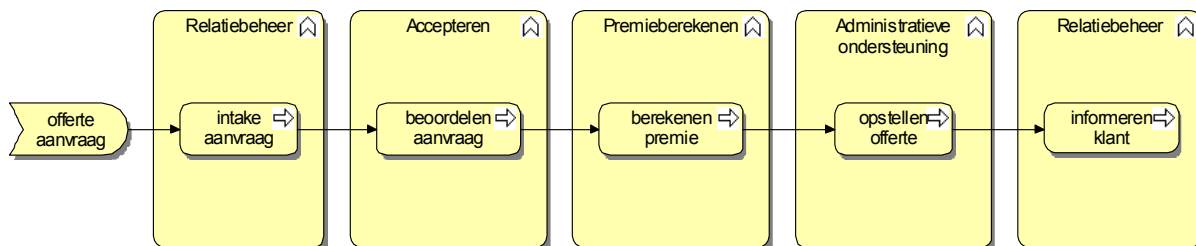
Oplossing: Een bedrijfsservice is de bijdrage die vanuit de organisatie wordt geleverd aan de omgeving. Er kan onderscheid worden gemaakt tussen interne en externe services. Interne services zijn de bijdragen die worden geleverd binnen het domein waar de service onderdeel van uitmaakt. Externe services zijn de bijdragen die beschikbaar worden gesteld aan andere domeinen of aan de omgeving (bijvoorbeeld klanten).

Een bedrijfsfunctie is een aandachtsgebied waaraan het bedrijf structureel aandacht wil besteden (= energie in wil stoppen, structureel middelen voor wil inzetten) om zijn bedrijfsdoelstelling te realiseren. Een bedrijfsfunctie kan daarom ook gezien worden als een groepering van intern gedrag op basis van een bepaald criterium (bijvoorbeeld plaats (dezelfde afdeling), communicatie, benodigde competenties, gedeelde bronnen en gedeelde kennis). Een bedrijfsfunctie representeert een stuk toegevoegde waarde van de organisatie.

Een bedrijfsproces is een eenheid van intern gedrag of een verzameling van causaal (volgorde, afhankelijkheid) gerelateerde eenheden van intern gedrag, met als doel een voorgedefinieerde verzameling van producten en diensten te produceren. Een bedrijfsproces kan bestaan uit deelprocessen of activiteiten. Een bedrijfsproces wordt getriggerd (opgestart) door één of meerdere business events (gebeurtenissen) of andere bedrijfsprocessen.

Informeel zou dus gezegd kunnen worden dat een bedrijfsproces bestaat uit een aantal activiteiten of deelprocessen die in een bepaalde volgorde worden uitgevoerd. Elke activiteit maakt onderdeel uit, of hoort bij, een bedrijfsfunctie. Met andere woorden, een proces rijgt een keten van activiteiten aan elkaar die elk horen bij bedrijfsfuncties, zoals ook gevisualiseerd in onderstaand schema. Overigens is het niet zo dat één proces slechts bij

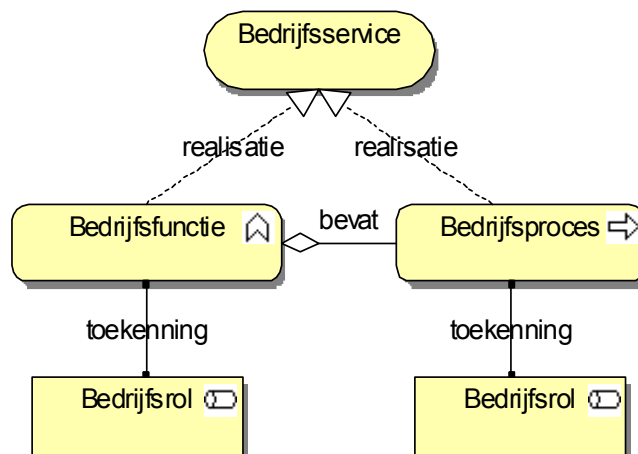
één bedrijfsfunctie hoort (zoals in dit voorbeeld): een bedrijfsfunctie omvat vrijwel altijd meerdere activiteiten en processtappen, en een proces zal vaak door meerdere bedrijfsfuncties worden gerealiseerd.



Figuur 2-1: Voorbeeld van bedrijfsfuncties in relatie tot processen

Zowel bedrijfsfuncties als processen beschrijven de interne gang van zaken binnen een aandachtsgebied of organisatie. Een bedrijfsservice beschrijft juist datgene dat van buitenaf zichtbaar en bruikbaar is van processen en functies. Het beschrijft de dienst die gebruikt kan worden, zonder dat duidelijk is hoe die dienst met behulp van processen of bedrijfsfuncties gerealiseerd wordt.

In onderstaande figuur zijn de belangrijkste relaties tussen deze concepten aangegeven (een deel van het metamodel):



Figuur 2-2: Belangrijkste relaties tussen concepten

Consequenties: Niet van toepassing.

Alternatieven: Niet van toepassing.

Relaties met andere good practices: Een soortgelijk onderscheid als tussen bedrijfsservice en bedrijfsproces / bedrijfsfunctie geldt tussen applicatieservice en applicatiefunctie. Soortgelijke criteria en heuristieken kunnen daarom ook op de applicatielaag gebruikt worden.

2.3 Afbakening bedrijfsservice

Vraagstelling: In de bedrijfslaag wordt ondermeer het concept bedrijfsservice gehanteerd. Maar hoe onderken je een bedrijfsservice, en hoe baken je deze af?

Oplossing: Als iets als een bedrijfsservice wordt gedefinieerd moet de service in principe door meerdere gebruikers af te nemen zijn. Services kunnen gegroepeerd worden met behulp van aggregatierelaties. Dit betekent dat het alleen zinvol is om een service te decomponeren (via aggregatie) in deelservices als deze deelservices ook zelfstandig als services af te nemen zijn.

Hanteer de volgende heuristieken om bedrijfsservices te onderkennen:

- Onderken een service vanuit het perspectief van de serviceafnemer. De service moet herkenbaar en bruikbaar zijn voor de afnemer. De naamgeving van services moet ook vanuit het perspectief van de serviceafnemer gebeuren.
- Onderken services op basis van de activiteiten die onderkend worden in de bedrijfslaag, en op basis van de producten die geleverd worden.
- Onderken verschillende services voor verschillende belangen (concerns).
- Vermijd overlap in geboden services: verschillende services bieden verschillend gedrag. Overlap in gedrag is een indicator om het overlappende gedrag als aparte service te beschouwen.
- Een service wordt gerealiseerd door één of meer bedrijfsfuncties of processen, die concreet intern gedrag van de organisatie representeren. Een bedrijfsfunctie mag meerdere bedrijfsservices realiseren.
- Modelleer van een bedrijfsservice altijd door welke bedrijfsfuncties of processen deze gerealiseerd wordt en door welke bedrijfsprocessen (in het geval van interne services) deze gebruikt wordt.
- Een interne bedrijfsservice wordt altijd gebruikt door minimaal één bedrijfsfunctie of bedrijfsproces.
- Houd services consistent: zorg dat soortgelijk gedrag op een soortgelijke manier in services aangeboden wordt.
- Gebruik services om implementatiedetails af te schermen. Het moet voor een serviceafnemer voldoende zijn om te weten *dat* een bepaalde service aangeboden wordt en hoe de afnemer hiervan gebruik moet maken. Een serviceafnemer hoeft niet te weten *hoe* een service gerealiseerd wordt.

Consequenties: Niet van toepassing.

Alternatieven: Niet van toepassing.

Relaties met andere good practices: Soortgelijke criteria en heuristieken kunnen ook op de applicatielaag voor applicatieservices en op de infrastructuurlaag voor infrastructurele services gebruikt worden.

2.4 Afbakening bedrijfsproces

Vraagstelling: In de bedrijfslaag wordt ondermeer het concept bedrijfsproces gehanteerd. Maar hoe onderken je een bedrijfsproces, en hoe baken je deze af?

Oplossing: Hanteer de volgende heuristieken om bedrijfsprocessen te onderkennen:

- Een bedrijfsproces wordt getriggerd door een bedrijfsevent, en levert uiteindelijk een dienst of product op voor een klant, of deelproducten of deeldiensten die gebruikt worden als onderdeel voor een dienst of product voor een klant.
- Verdeel een proces in fasen die achtereenvolgens doorlopen worden. Voorbeelden van fasen zijn aanvraag, afhandeling en after sales. Zo'n fase komt vaak overeen met een werkproces of functie binnen de organisatie. Een veel voorkomende vorm van fasering is de waardeketen (value chain). U herkent fasen vaak aan de toewijzing van acties aan verschillende actoren, gecombineerd met een tijdsvolgorde tussen die acties.
- Groepeer acties op grond van het tijdstip waarop zij plaatsvinden (online- of batchverwerking, overdag of 's nachts).
- Deel een proces op in stukken op grond van de kennis en vaardigheden die zijn vereist om bepaalde acties uit te voeren. Dit blijkt onder meer uit de benodigde functies die zijn aangegeven bij elke taak.
- Deel een proces in stukken op grond van het geografisch gebied (fysieke locatie) waar de activiteiten plaatsvinden, bijvoorbeeld naar regio.
- Verdeel een proces in deelprocessen die (grotendeels) onafhankelijk uitgevoerd kunnen worden. Het gaat hierbij vaak om verschillende (tussen)producten die worden opgeleverd.
- Bakken expliciet de deelprocessen af die vaker voorkomen. Deze algemene deelprocessen kunt u vervolgens apart definiëren, en dan meermaals gebruiken.
- Bakken expliciet de deelprocessen af die als geheel herhaald worden.

Consequenties: Niet van toepassing.

Alternatieven: Niet van toepassing.

Relaties met andere good practices: Niet van toepassing.

2.5 Afbakening bedrijfsfunctie

Vraagstelling: In de bedrijfs slaag wordt ondermeer het concept bedrijfsfunctie gehanteerd. Maar hoe onderken je een bedrijfsfunctie, en hoe bakken je deze af?

Oplossing: Hanteer de volgende heuristieken om bedrijfsfuncties te onderkennen:

- Maak een aparte functie voor de *interacties met elke omgevingsactor*. Bijvoorbeeld een functie voor toeleveranciers, afnemers en de overheid. Deze heuristiek kan worden verbijzonderd naar:
 - *Scheiden relatiebeheer van overige functies:* Maak een aparte functie voor de activiteiten die direct te maken hebben met het contact met de klant. In principe gaat het hier om de externe klant van het bedrijf, maar voor grotere bedrijven kan het ook toepasbaar zijn voor een interne klant (bijvoorbeeld een andere afdeling).
 - *Scheiding naar markt:* Maak een aparte functie voor elke klantgroep/doelgroep van het bedrijf. Voorbeelden van dergelijke groepen zijn zakelijke klanten en particulieren.
- Maak aparte functies voor processen met verschillende soorten *triggers* (business events). Zo kunnen processen worden geïdentificeerd met periodieke triggers (maandelijks ontvangen van declaraties) en met incidentele triggers (vragen om offerte). Een verbijzondering hiervan is:

- *Gevalsonderscheid*: Maak aparte functies voor activiteiten die betrekking hebben op verschillende 'gevallen'. Zo kan een bedrijf onderscheid maken tussen verschillende schadegevallen: schades kleiner respectievelijk groter dan f 1000,-. In dit geval zou er dus een functie voor kleine en een functie voor grote schadegevallen onderscheiden kunnen worden.
- Maak een aparte functie voor elke *productgroep*, bijvoorbeeld voor activiteiten rond schadeverzekeringen en rond pensioenverzekeringen. Algemener kan een onderscheid worden gemaakt rondom een willekeurig bedrijfsobject:
 - *Scheiding naar bedrijfsobject*: Maak aparte functies voor een groep activiteiten, indien ze werken op een bepaald bedrijfsobject. Zo kunnen er functies zijn die betrekking hebben op facturen, op geldstromen, op offertes, op klantdossiers, etc.
- Maak een aparte functie voor elke fase of toestand waarin een product zich kan bevinden. In geval van een schadeverzekering zouden dit bijvoorbeeld aparte functies kunnen opleveren voor aanvragen, premieheffing en afhandeling van schadeclaims, beëindiging en beheer.
- Maak een aparte functie voor een groep activiteiten, indien deze activiteiten speciale 1) vaardigheden, 2) expertise of 3) verantwoordelijkheden vereisen. Voorbeelden hiervan zijn juridische kennis, actuariële expertise, communicatieve vaardigheden of autorisaties voor het nemen van bepaalde beslissingen (fiattering).
- Maak aparte functies voor activiteiten die het primaire proces *besturen*. Het gaat daarbij in het bijzonder om planning en controle.
- Maak aparte functies voor activiteiten die *veranderingen* aanbrengen aan het primaire proces of de implementatie daarvan. Dit leidt tot aparte functies voor bijvoorbeeld marketing, productontwikkeling en systeemontwikkeling.

Consequenties: Niet van toepassing.

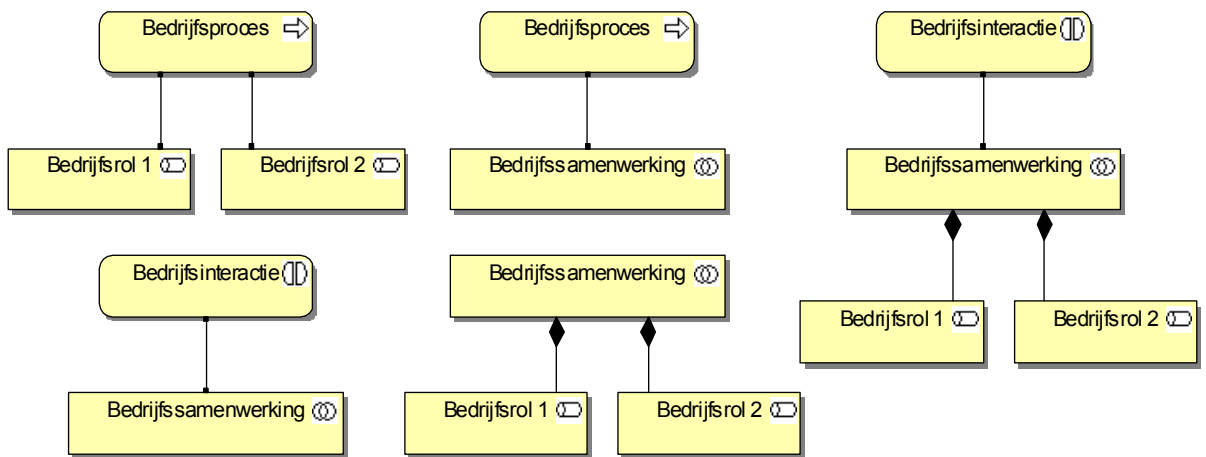
Alternatieven: Niet van toepassing.

Relaties met andere good practices: Soortgelijke criteria en heuristieken kunnen ook op de applicatielaag voor applicatiefuncties gebruikt worden.

2.6 Bedrijfsproces en bedrijfsinteractie

Vraagstelling: ArchiMate kent naast het concept bedrijfsproces ook het concept bedrijfsinteractie. Een bedrijfsinteractie is een proces dat door twee of meer actoren (een collaboratie of samenwerking van actoren) wordt uitgevoerd. Waarom is het concept bedrijfsinteractie nodig naast het concept bedrijfsproces? Is het niet voldoende om een proces te koppelen aan een collaboratie (samenwerking) van actoren, en daarmee uit te drukken dat het proces eigenlijk een interactie is?

Oplossing: Een bedrijfsinteractie is gedefinieerd als een specialisatie van bedrijfsproces (of preciezer van het generieke concept gedrag). Door een bedrijfsinteractie te gebruiken voeg je dus informatie toe, namelijk dat het gedrag door een samenwerking van rollen of actoren wordt uitgevoerd. Door een bedrijfsproces te koppelen aan een bedrijfssamenwerking geef je dezelfde informatie weer, namelijk dat het gedrag door een samenwerking van rollen of actoren wordt uitgevoerd. In bijgaande figuur zijn enkele varianten afgebeeld:



Figuur 2-3: Varianten van bedrijfsproces en bedrijfsinteractie

Linksboven is gemodelleerd dat een bedrijfsproces is toegekend aan twee rollen, middenboven dat een bedrijfsproces is toegekend aan een bedrijfssamenwerking, linksonder dat een interactie is toegekend aan een bedrijfssamenwerking, middenonder dat een bedrijfssamenwerking bestaat uit twee rollen, en geheel rechts de combinatie dat een interactie is toegekend aan een bedrijfssamenwerking die weer bestaat uit twee rollen.

Deze redundantie heeft echter wel een goede reden. Soms wil je niet de bedrijfssamenwerking expliciet tekenen (bijvoorbeeld als de nadruk ligt op het gedrag, en niet op de actoren), maar toch aangeven dat het om een interactie door meerdere rollen gaat. Een ander voorbeeld is dat je wel wilt aangegeven dat het een samenwerking is, maar niet door welke partijen of actoren de samenwerking wordt vormgegeven; dit kan bijvoorbeeld zinvol zijn in ontwerptraject waar nog niet bepaald is welke partijen uiteindelijk zullen bijdragen aan een proces of dienst.

Gevolgen: Het doel dat een architect heeft met het modelleren of visualiseren van een bedrijfsinteractie of -samenwerking bepaalt dus in grote mate welke oplossing hiervoor gekozen wordt en welk aspect de nadruk krijgt.

Alternatieven: Niet van toepassing.

Relaties met andere good practices: Een soortgelijk onderscheid als hiervoor beschreven speelt ook op de applicatielaag, namelijk rond applicatie-interactie en applicatiesamenwerking. De hiervoor beschreven voorbeelden en richtlijnen zijn ook hierop van toepassing.

2.7 Viewpoints voor procesmodellering

Vraagstelling: Hoe modelleer je in ArchiMate een proces? Het ArchiMate boek geeft een aantal viewpoints, maar welke combinatie van viewpoints heb je nodig om een proces in opeenvolgende stappen te beschrijven?

Oplossing: Het ArchiMate boek geeft twee viewpoints voor procesmodellering: het Bedrijfsprocessamenwerkings viewpoint en het Bedrijfsproces viewpoint. Het verschil

tussen deze viewpoints is dat in het Bedrijfsprocessamenwerkings viewpoint de relatie *tussen* processen en de relatie met de omgeving wordt weergegeven, terwijl in het Bedrijfsproces viewpoint één bedrijfsproces wordt uitgelicht. We concentreren ons hier op het Bedrijfsproces viewpoint; het Bedrijfsprocessamenwerkings viewpoint is analoog.

Bij het modelleren van een proces in ArchiMate kunnen de volgende aspecten worden weergegeven: de onderdelen van het proces, hun samenhang, de toewijzing van procesdelen aan rollen, actoren of bedrijfsonderdelen en de ondersteuning door applicaties. Je kunt dit doen in de volgende stappen:

1. *Modelleer het bedrijfsproces op hoofdlijnen*: Decomponeer het bedrijfsproces en leg verbanden door bedrijfsevents en triggeringrelaties. Daaraan toegevoegd kan worden de wijze waarop het proces bedrijfsservices verleent aan zijn omgeving.
2. *Informatie-overdracht*: Modelleer de informatie-overdracht tussen processen door het toevoegen van flowrelaties tussen processen, of het lezen en schrijven van bedrijfsobjecten.
3. *Toewijzing aan organisatie*: Modelleer welk organisatie-onderdeel het bedrijfsproces uitvoert. Dit kan gevisualiseerd worden door toekenningsrelaties of door gebruik te maken van procesbanen.
4. *Ondersteuning door applicaties*: Modelleer de wijze waarop applicaties de bedrijfsprocessen ondersteunen. Hiervoor kan het Applicatiegebruik viewpoint worden gebruikt.

Eventueel kan nog een algemeen onderscheid gemaakt worden tussen een 'logische' en een 'concreet' niveau van modelleren. Een *logisch model* geeft de functionele structuur en de logische samenhang van bedrijfsprocessen weer. Er wordt niet ingegaan op tijd, plaats, mensen en machines. Een *inrichtingsmodel* komt sterk overeen met wat je in de werkelijkheid ziet of wilt realiseren. Het beschrijft *fysieke* zaken zoals mensen en machines, met daaraan toegewezen concrete taken. Indien een organisatie een standaard typering van processen kent / wil onderkennen, moet hiervoor een specifieke afspraak worden gemaakt bij de opzet van het modelleren met ArchiMate hiervoor. De relatie tussen 'logisch proces' en 'fysiek proces' is van het type compositie.

Tenslotte kun je een proces specialiseren in een aantal typen processen. Bijvoorbeeld kan een proces 'Aanvragen verzekering' gespecialiseerd worden in een drietal subprocessen 'Aanvragen reisverzekering' en 'Aanvragen schadeverzekering'.

Gevolgen: Niet van toepassing.

Alternatieven: Niet van toepassing.

Relaties met andere good practices: Zie ook de good practices 'Bedrijfsservice, bedrijfsfunctie en bedrijfsproces' en 'Afbakening bedrijfsproces'.

3 Modelleren in de applicatielaag

3.1 Overzicht good practices

In dit hoofdstuk zijn de onderstaande good practices opgenomen:

1. Applicatielandschap.
2. Koppeling tussen applicaties.
3. Functionaliteit van een applicatie.

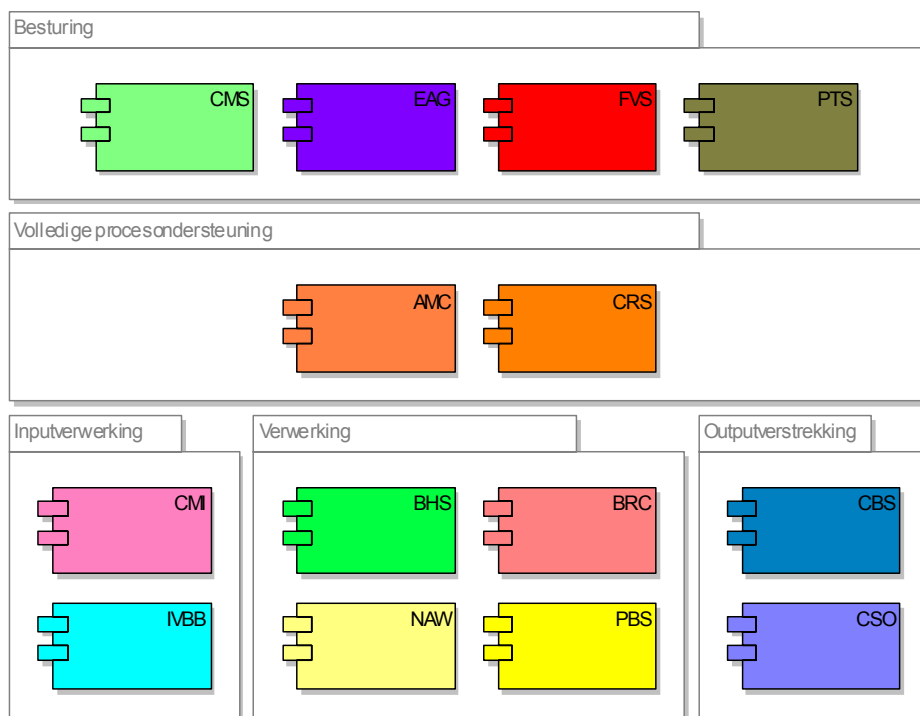
3.2 Applicatielandschap

Vraagstelling: Mijn applicatielandschap is zeer complex, hoe krijg ik daar grip op, hoe modelleer ik dit?

Vaak wordt een overzicht van applicaties getoond in een applicatielandschap. Dit om een overzicht te verschaffen van de relaties tussen de applicaties. Vaak geeft dit een groot aantal verbindingslijnen en eigenlijk geen enkel overzicht. Dergelijke overzichten geven alleen aan hoe complex het applicatielandschap eruit ziet. Het geeft geen houvast voor het onderhouden van je applicatieportfolio.

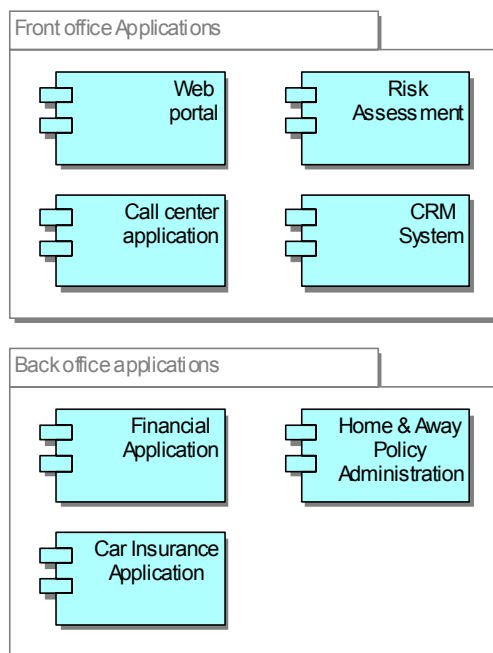
Oplossing: Het is belangrijk om de overzichten leesbaar te houden. Houd een regel aan van maximaal 10 a 15 applicaties per overzicht. Zijn het er meer, dan is het aan te bevelen om gebruik te maken van een algemeen model met daarin de hoofdgroepering plus een detailmodel per groep. Kies een viewpoint die voor de doelgroep van belang is. De vorm is altijd een Applicatiestructuur viewpoint. Een classificatie kan worden gedaan via proces, organisatie, gegevens, type functionaliteit of infrastructuur. Groepeer de applicaties per gebied en verbind alleen de gebieden. Bij grotere aantallen is het tonen van relaties via een matrix veel overzichtelijker.

Hieronder is een voorbeeld opgenomen van een applicatielandschap ingedeeld naar de type functionaliteit van de applicaties (besturing, inputverwerking, verwerking, outputverstrekking en volledige procesondersteuning).



Figuur 3-1: Voorbeeld van applicatielandschap ingedeeld naar type functionaliteit van applicaties

Hieronder is een voorbeeld opgenomen van een applicatielandschap ingedeeld naar het gebruik door de frontoffice en backoffice.



Figuur 3-2: Voorbeeld van applicatielandschap ingedeeld naar gebruik door frontoffice en backoffice

Gevolgen: Voor een overzicht van alle relaties zijn meerdere diagrammen nodig. Gebruik hiervoor een matrix. Een tool kan in dit geval zeer waardevol zijn om deze informatie vast te leggen en bijvoorbeeld een kruistabel mee te maken.

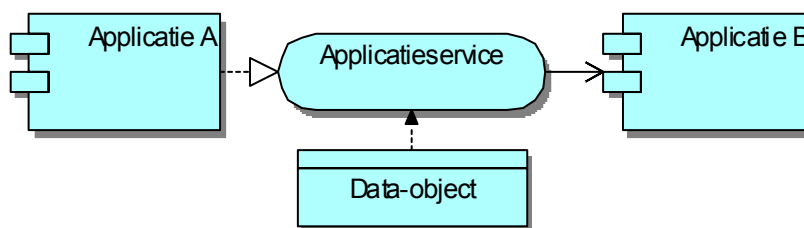
Alternatieven: Het alternatief is de grote 'plaat' met alle applicaties en de vele lijnen. Deze is alleen bruikbaar om aan te geven hoe complex het landschap is, verder nut is er weinig van een dergelijk overzicht.

Relaties met andere good practices: Hier ligt een belangrijke relatie met de good practice 'Koppeling tussen applicaties'.

3.3 Koppeling tussen applicaties

Vraagstelling: Het is wenselijk dat koppelingen tussen applicaties in beeld worden gebracht. Welke concepten binnen ArchiMate kunnen hiervoor worden gebruikt?

Oplossing: Een applicatie wordt gemodelleerd als applicatiecomponent en een koppeling tussen twee applicaties wordt gemodelleerd via een applicatieservice. De applicatieservice levert gegevens door middel van een toegangsrelatie met een data-object. In het onderstaande voorbeeld is gemodelleerd dat applicatie A een applicatieservice realiseert die door B wordt gebruikt. De applicatieservice levert de gegevens die in het data-object zijn opgenomen.



Figuur 3-3: Applicatiekoppeling middels applicatieservice en gekoppeld data-object

Gevolgen: Als via deze oplossing de huidige koppelingen tussen applicaties binnen een bedrijf in beeld worden gebracht, dan kan de suggestie worden gewekt dat er al sprake is van een servicegeoriënteerde architectuur terwijl daar in werkelijkheid nog geen sprake van is.

Alternatieven: Een andere wijze van modelleren die in de praktijk veelal wordt toegepast, is het leggen van een gebruikt door-relatie tussen twee applicaties. In het onderstaande voorbeeld is gemodelleerd dat applicatie A wordt gebruikt door applicatie B. Hierin is niet meer te zien via welke applicatieservice deze relatie loopt, maar de view wordt hiermee wel eenvoudiger en voor bepaalde doelgroepen beter communiceerbaar. Door middel van de juiste toolondersteuning is Figuur 3-4 af te leiden van Figuur 3-3.



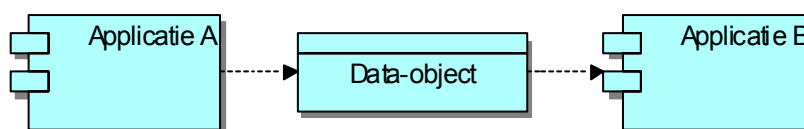
Figuur 3-4: Applicatiekoppeling middels rechtstreekse gebruikt door-relatie

Een andere mogelijkheid is gebruik te maken van een applicatieservice tussen twee applicaties zonder gekoppeld data-object. In het onderstaande voorbeeld is gemodelleerd dat applicatie A een applicatieservice realiseert die door B wordt gebruikt.



Figuur 3-5: Applicatiekoppeling middels applicatieservice

Een andere mogelijkheid is gebruik te maken van een data-object tussen twee applicaties, waarbij gebruik wordt gemaakt van een toegangsrelatie tussen een applicatie en een data-object. In het onderstaande voorbeeld is gemodelleerd dat applicatie A een data-object creëert of wijzigt, dat door applicatie B wordt geraadpleegd.



Figuur 3-6: Applicatiekoppeling middels data-object

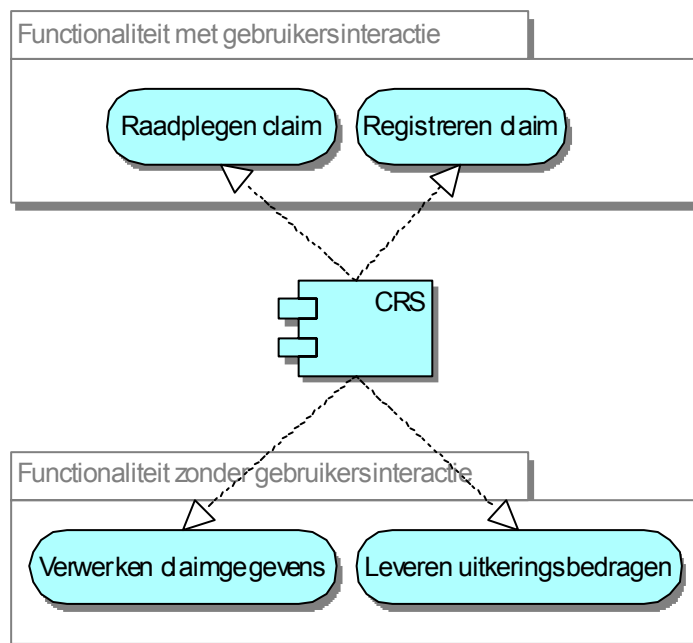
Relaties met andere good practices: Dit kan worden gebruikt bij het opbouwen van een applicatielandschap via de good practice 'Applicatielandschap'.

3.4 Functionaliteit van een applicatie

Vraagstelling: Hoe breng je met ArchiMate de extern zichtbare functionaliteit van applicaties in beeld?

Oplossing: Een applicatie kan zichtbare functionaliteit (met gebruikersinteractie) leveren richting een eindgebruiker, maar kan ook functionaliteit (zonder gebruikersinteractie) leveren richting andere applicaties. Het ArchiMate-concept dat voor beide vormen dient te worden gebruikt is de applicatieservice dat via een realisatierelatie is gekoppeld aan een applicatiecomponent dat de applicatie representeert. Om in het model onderscheid te maken in de soorten functionaliteit kan gebruik worden gemaakt van het groepconcept. Een gebruikelijk onderscheid is functionaliteit met of zonder gebruikersinteractie. Een ander onderscheid kan worden gemaakt via proces, organisatie, gegevens, type functionaliteit of infrastructuur.

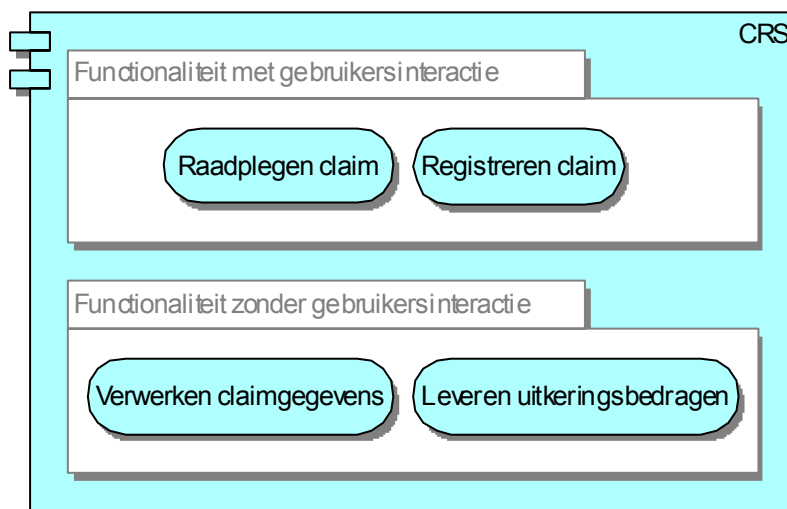
In het onderstaande voorbeeld is gemodelleerd dat de applicatie 'CRS' twee keer een functionaliteit met gebruikersinteractie realiseert: 'Raadplegen claim' en 'Registreren claim'. En daarnaast twee keer een functionaliteit zonder gebruikersinteractie: 'Verwerken claimgegevens' en 'Leveren uitkeringsbedragen'.



Figuur 3-7: Voorbeeld van functionaliteit van een applicatie

Gevolgen: Als een applicatie veel functionaliteit levert, dan kan een dergelijk model onoverzichtelijk worden door de veelheid aan realisatierelaties.

Alternatieven: Een andere wijze van modelleren is het visualiseren van de applicatieservices als visuele compositie binnen een applicatiecomponent. In het onderstaande voorbeeld zijn de realisatierelaties tussen de applicatie 'CRS' en de functionaliteit gemodelleerd als visuele compositie.



Figuur 3-8: Voorbeeld van functionaliteit van een applicatie

Relaties met andere good practices: Voor niet interactieve functionaliteit ligt een relatie met de good practice 'Koppeling tussen applicaties'. Het in beeld brengen van functionaliteit van een applicatie lijkt veel op het in beeld brengen van de applicaties binnen een

applicatieportfolio. De good practice 'Applicatielandschap' is ook toepasbaar voor de functionaliteit van een applicatie.

4 Modelleren in de infrastructuur- / technologielaag

4.1 Overzicht good practices

In dit hoofdstuk zijn de onderstaande good practices opgenomen:

1. Infrastructuurservices.
2. Infrastructuurlandschap.

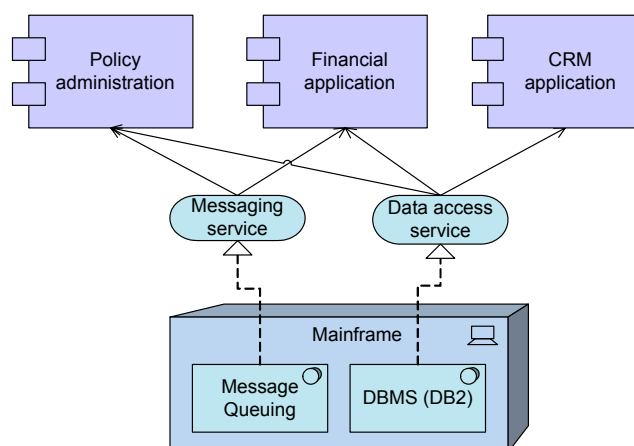
4.2 Infrastructuurservices

Vraagstelling: Welke services zou je op de infrastructuurlaag willen modelleren?

Vanuit het oogpunt van beheer is op de infrastructuurlaag vaak veel detailinformatie nodig: welke systemen zijn er en hoe zijn die precies met elkaar verbonden? De stap naar het beschrijven van voor de omgeving betekenisvolle services is niet triviaal: hoe begin je met het modelleren van services op de infrastructuurlaag en welke typen services moet je hierbij onderscheiden?

Oplossing: Beschrijf op de infrastructuurlaag de services waarmee de infrastructuur applicaties ondersteunt. Het is aan te raden hierbij onderscheid te maken tussen verwerkingsservices, opslagservices en communicatieservices. Essentieel bij het modelleren van de infrastructuurlaag is het abstraheren van interne details, bijvoorbeeld over implementatiedetails. Hiervoor kunnen domeinspecifieke talen gebruikt worden.

Maak hierbij gebruik van het Infrastructuurgebruik viewpoint uit het ArchiMateboek (Lankhorst et al., 2005, p. 187). Dit viewpoint geeft je de mogelijkheid te laten zien hoe applicaties ondersteund worden door software en hardware infrastructuur.



Figuur 4-1: Voorbeeld van infrastructuurservices

Hanteer de definitie zoals beschreven in ArchiMate (Lankhorst et al., 2005): een infrastructuurservice is een zichtbare eenheid van functionaliteit, geleverd door één of meerdere nodes, aangeboden via goed gedefinieerde interfaces en betekenisvol voor de omgeving.

Gevolgen: Niet van toepassing.

Alternatieven: Niet van toepassing.

Relaties met andere good practices: Niet van toepassing.

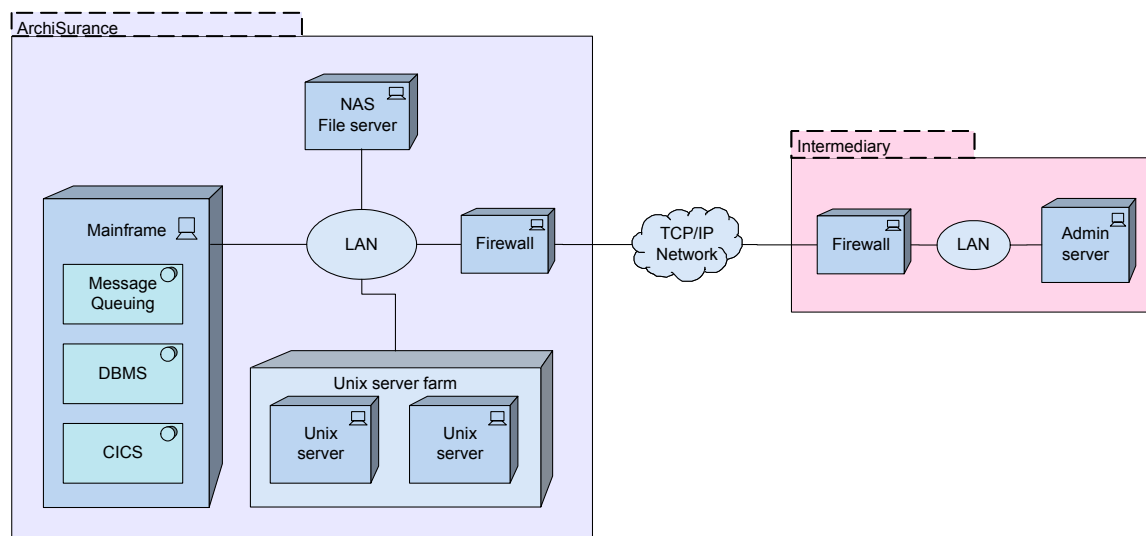
4.3 Infrastructuurlandschap

Vraagstelling: Hoe modeller je een infrastructuurlandschap in ArchiMate? Enterprise architectuur gebruik je om de essentie van architectuurdomeinen vast te leggen en de relaties tussen die domeinen expliciet te maken; echter op het niveau van infrastructuur zijn vaak implementatie details belangrijk. Welke aspecten moet je wel en niet modelleren in een infrastructuurlandschap in ArchiMate?

Oplossing: Zoals gebruikelijk is het essentieel om bij het modelleren van een infrastructuurlandschap uit te gaan van het doel en de doelgroep. In de meeste enterprise architectuur trajecten is het doel van een infrastructuurlandschap het in beeld brengen van de belangrijkste elementen (hardware en software) van de infrastructuur, hoe deze via netwerken verbonden zijn en wat hun geografische locatie is (zoals het hoofdkantoor en regiokantoren).

Laat in een ArchiMate infrastructuurlandschap alleen de belangrijkste fysieke systemen en netwerken zien en specificeer hierbij de essentiële ondersteunende software zoals operating systems, database management systemen en middleware.

Maak hierbij gebruik van het Infrastructuur viewpoint uit het ArchiMateboek (Lankhorst et al., 2005, p. 186). Dit viewpoint geeft je de mogelijkheid te laten zien welke essentiële hardware en software het infrastructuur landschap bepalen en hoe deze via netwerken verbonden zijn. Het is aan te raden in dit viewpoint infrastructuur elementen op basis van geografische locatie te groeperen en deze groepen expliciet te maken.



Figuur 4-2: Voorbeeld van een infrastructuurlandschap

Gevolgen: Niet van toepassing.

Alternatieven: Niet van toepassing.

Relaties met andere good practices: Niet van toepassing.

5 Modelleren over de grenzen van lagen heen

5.1 Overzicht good practices

In dit hoofdstuk zijn de onderstaande good practices opgenomen:

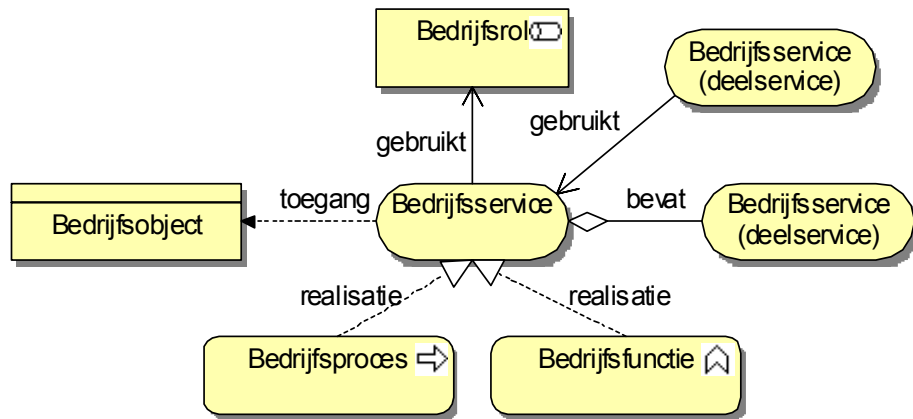
1. Veelgebruikte relaties.
2. Product en applicatieservices.
3. Ondersteuning van batch en online processen.
4. Servicebroker.

5.2 Veelgebruikte relaties

Vraagstelling: ArchiMate onderkent een behoorlijk aantal concepten en relaties. De vraag welke relatie gemodelleerd mag worden tussen een tweetal concepten is weliswaar precies gedefinieerd in ArchiMate, maar in de praktijk blijkt het lastig om de juiste keuze te maken binnen de toegestane mogelijkheden. De vraag is dan ook welke relaties tussen concepten in de meest voorkomende gevallen gemodelleerd zouden moeten worden. Hieronder geven we voor de meest voorkomende situaties aan welke relaties gemodelleerd zouden moeten worden. Om misverstanden te voorkomen: in onderstaande opsomming worden niet alle ArchiMate-relaties tussen concepten besproken, alleen de relaties die naar onze inschatting het vaakst gebruikt worden. Een compleet overzicht van alle relaties die mogelijk zijn tussen verschillende concepten is te vinden in het referentiemanual van ArchiMate.

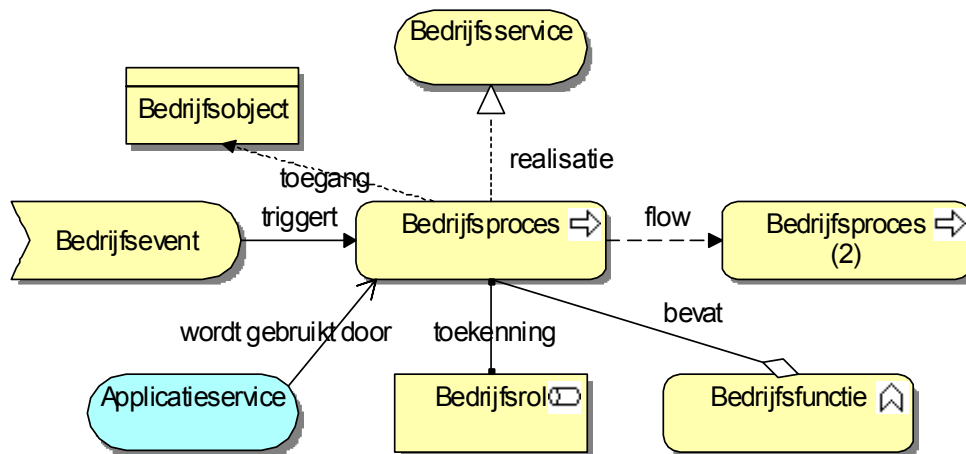
Oplossing: Hieronder volgt voor een aantal concepten (met name uit de bedrijfslaag en de applicatielaag) een overzicht van de meest gebruikte relaties vanuit of naar dat concept met andere concepten. Niet alle concepten worden hieronder opgesomd, alleen de meest gebruikte.

- **Bedrijfsservice:**
 - een bedrijfsservice wordt gerealiseerd door een bedrijfsproces of een bedrijfsfunctie;
 - een bedrijfsservice wordt gebruikt door een bedrijfsrol;
 - een bedrijfsservice heeft toegang tot een bedrijfsobject (een bedrijfsservice creëert, leest, wijzigt of vernietigt een bedrijfsobject)
 - een bedrijfsservice kan bestaan uit andere bedrijfsservices., en kan andere bedrijfsservices gebruiken.



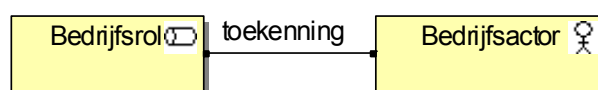
Figuur 5-1: Belangrijkste relaties tussen concepten

- **Bedrijfsproces:**
 - Een bedrijfsproces realiseert een bedrijfsservice;
 - Een bedrijfsproces wisselt gegevens uit met andere bedrijfsprocessen (via de flowrelatie);
 - Een bedrijfsproces wordt getriggerd door of triggert een bedrijfsevent, een bedrijfsfunctie of andere bedrijfsprocessen;
 - Een bedrijfsproces wordt toegekend aan een bedrijfsrol;
 - Een bedrijfsproces maakt onderdeel uit van een bedrijfsfunctie;
 - Een bedrijfsproces heeft toegang tot een bedrijfsobject (een bedrijfsproces creëert, leest, wijzigt of vernietigt een bedrijfsobject);
 - Een bedrijfsproces gebruikt applicatieservices.



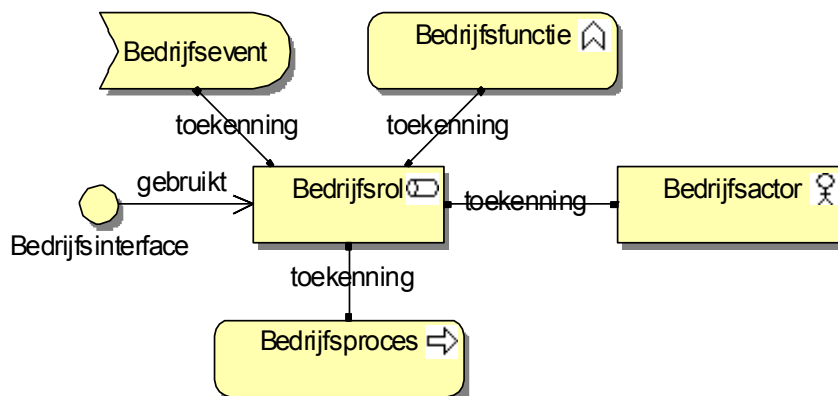
Figuur 5-2: Belangrijkste relaties tussen concepten

- **Bedrijfsactor:**
 - Een bedrijfsrol kan toegekend worden aan een bedrijfsactor.



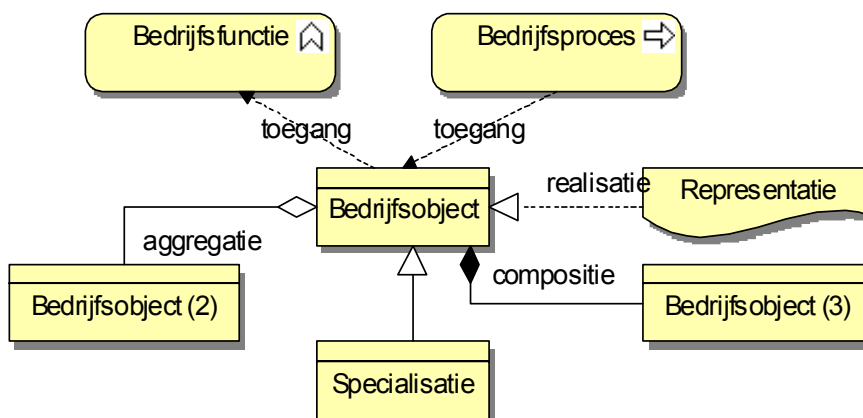
Figuur 5-3: Relatie tussen bedrijfsrol en bedrijfsactor

- Bedrijfsrol:
 - Een bedrijfsrol kan toegekend worden aan een bedrijfsactor;
 - Een bedrijfsrol kan toegekend worden aan een bedrijfsproces, bedrijfsfunctie of bedrijfsevent.
 - Een bedrijfsrol kan gebruik maken van een bedrijfsinterface.



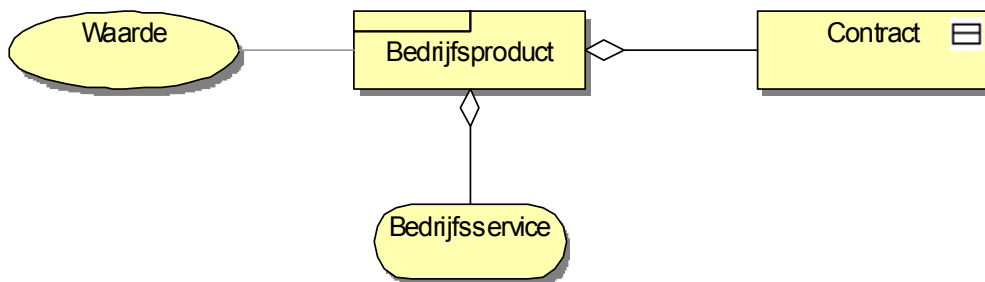
Figuur 5-4: Belangrijkste relaties tussen concepten

- Bedrijfsobject:
 - Een bedrijfsobject wordt gecreëerd, gelezen, gewijzigd of verwijderd door een bedrijfsproces of bedrijfsfunctie (via toegangsrelatie);
 - Een bedrijfsobject kan specialisaties hebben;
 - Een bedrijfsrepresentatie realiseert een bedrijfsobject;
 - Een bedrijfsobject kan verwijzen naar andere objecten (aggregatierelatie);
 - Een bedrijfsobject kan andere objecten bevatten (compositierelatie).



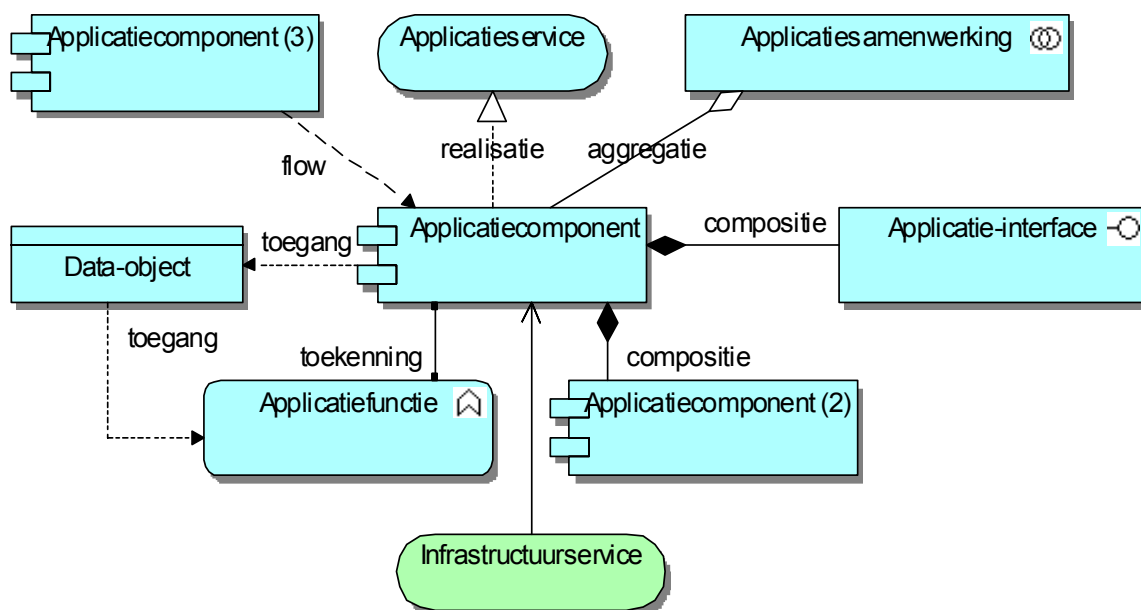
Figuur 5-5: Belangrijkste relaties tussen concepten

- Bedrijfsproduct:
 - Een bedrijfsproduct bestaat uit services en contract(en);
 - Een bedrijfsproduct representeert een waarde.



Figuur 5-6: Belangrijkste relaties tussen concepten

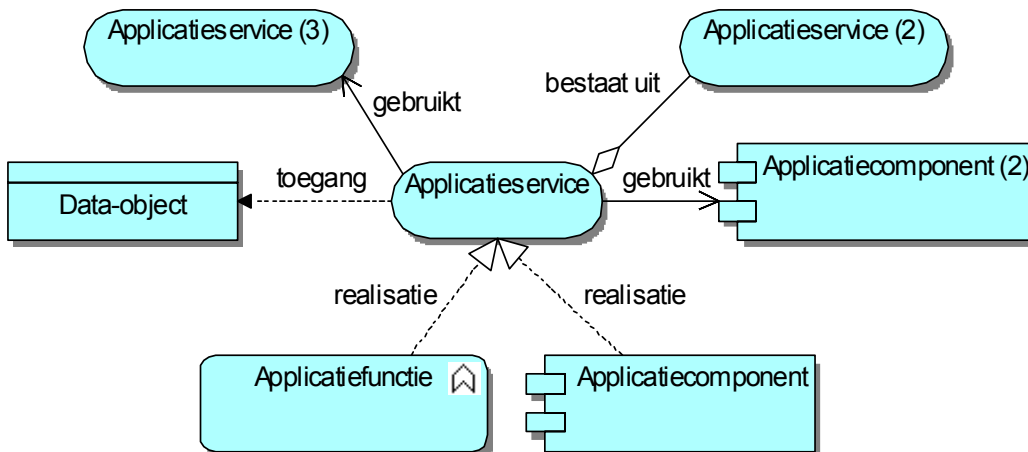
- Applicatiecomponent:
 - Een applicatiecomponent realiseert een applicatieservice;
 - Een applicatiecomponent heeft een of meerdere applicatie-interfaces;
 - Een dataobject wordt gecreëerd, gelezen, gewijzigd of verwijderd door een applicatiecomponent of applicatiefunctie;
 - Een applicatiecomponent kan onderdeel zijn van een applicatiesamenwerking (via aggregatierelatie);
 - Een applicatiecomponent kan bestaan uit meerdere applicatiecomponenten (via compositierelatie);
 - Een applicatiecomponent kan toegekend worden aan een applicatiefunctie;
 - Een applicatiecomponent gebruikt infrastructuurservices;
 - Tussen applicatiecomponenten kunnen gegevenstromen bestaan (flowrelatie).



Figuur 5-7: Belangrijkste relaties tussen concepten

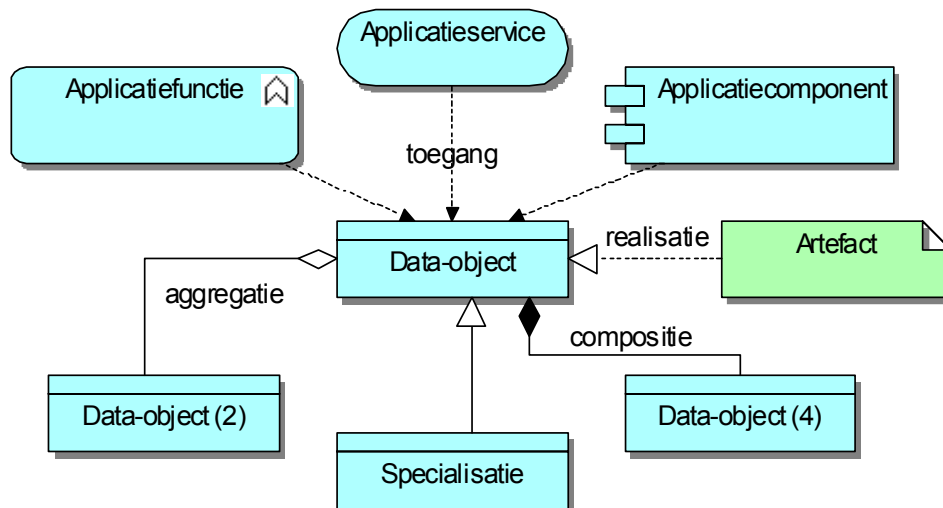
- Applicatieservice:
 - een applicatieservice wordt gerealiseerd door een applicatiecomponent of een applicatiefunctie;
 - een applicatieservice wordt gebruikt door een applicatiecomponent;
 - een applicatieservice heeft toegang tot een dataobject (een applicatieservice creëert, leest, wijzigt of vernietigt een dataobject)

- een applicatieservice kan bestaan uit andere applicatieservices., en kan andere applicatieservice gebruiken.



Figuur 5-8: Belangrijkste relaties tussen concepten

- Dataobject:
 - Een dataobject wordt gecreëerd, gelezen, gewijzigd of verwijderd door een applicatiecomponent, applicatieservice of applicatiefunctie (via toegangsrelatie);
 - Een dataobject kan specialisaties hebben;
 - Een artifact realiseert een dataobject;
 - Een dataobject kan verwijzen naar andere dataobjecten (aggregatierelatie);
 - Een dataobject kan andere dataobjecten bevatten (compositierelatie).



Figuur 5-9: Belangrijkste relaties tussen concepten

Gevolgen: Let op, dit zijn een aantal veelvoorkomende situaties. Het gebruik van relaties binnen ArchiMate is breder dan hierboven aangegeven. Een volledig overzicht is te vinden in de Architecture Language Reference Manual.

Alternatieven: Niet van toepassing.

Relaties met andere good practices: Niet van toepassing.

5.3 Product en applicatieservices

Vraagstelling: Is het niet vreemd dat in een product naast bedrijfsservices ook applicatie- en infrastructuurservices zitten? Een klant kan per definitie toch geen applicatieservices afnemen? Dus als een applicatieservice bijna direct wordt gebruikt door een klant, dan zou je daar toch een bedrijfsservice tussen moeten plaatsen? In sommige voorbeelden, bijvoorbeeld in de Architecture Language Reference Manual, wordt een product namelijk samengesteld uit applicatie- en bedrijfsservices.

Oplossing: De Architecture Language Reference Manual bevat inderdaad voorbeelden waar klanten rechtstreeks applicatieservices gebruiken. Het is beter om daar een bedrijfsservice tussen te zetten. In de productbeschrijving van de Architecture Language Reference Manual (p. 26) worden de applicatieservices 'Account status' en 'Money transfer' dan bedrijfsservices.

De koppeling tussen een bedrijfsservice en een applicatieservice verloopt dan via een bedrijfsproces die de bedrijfsservice ondersteunt, en op zijn beurt ondersteund wordt door een applicatieservice. Daarmee geldt de afgeleide relatie dat de bedrijfsservice een applicatieservice gebruikt.

Gevolgen: Overwogen wordt of de aggregatierelatie tussen product en applicatieservice verwijderd wordt.

Alternatieven: Niet van toepassing.

Relaties met andere good practices: Niet van toepassing.

5.4 Ondersteuning van batch en online processen

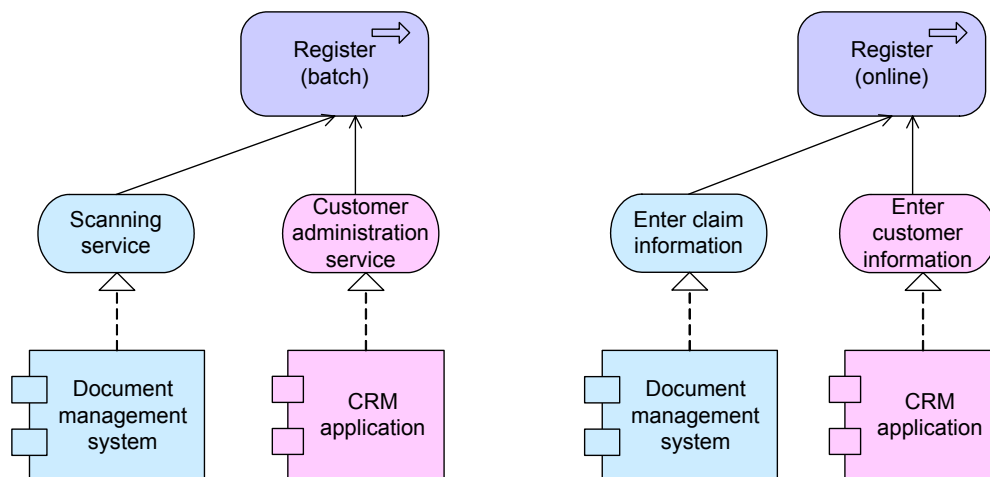
Vraagstelling: Hoe modelleer je in ArchiMate twee varianten van de ondersteuning van hetzelfde proces, waarbij één via een batch verloopt en de ander handmatig (online) verloopt?

Oplossing: Onderscheid twee varianten van het bedrijfsproces: één voor de handmatige (online) verwerking en één voor de batch verwerking. Hoewel de resultaten van de processen hetzelfde kunnen zijn, biedt de handmatige (online) verwerking meer ruimte voor uitzonderingen en kunnen vaak fijnmaziger processtappen onderscheiden worden. In beide procesbeschrijvingen is het verstandig bij het vaststellen van de 'grootte' van acties aan te houden dat acties *eenheden zijn van tijd, plaats en handeling*. Dit zal in de twee varianten van het bedrijfsproces waarschijnlijk verschillend zijn.

Beschrijf op basis van de twee varianten van het bedrijfsproces de applicatieservices die deze processen ondersteunen. Ga hierbij uit van de acties die gemodelleerd zijn in het bedrijfsproces: de applicatieservices voor beide varianten kunnen dus verschillend zijn! Vaak zal het zo zijn dat de applicatieservices voor het ondersteunen van een batch proces meer grofmazig zijn dan applicatieservices voor het ondersteunen van een online proces, omdat in het laatste geval meer fijnmazige acties onderscheiden worden. Let wel, het is goed mogelijk dat de *applicatiefuncties* die de verschillende versies van de

applicatieservices realiseren wél gemeenschappelijk zijn: dit wijst op hergebruik van functionaliteit in de batch en online verwerking.

Maak hierbij gebruik van het Applicatiegebruik viewpoint uit het ArchiMateboek (Lankhorst et al., 2005, p. 184). Dit viewpoint geeft je de mogelijkheid te laten zien hoe bedrijfsprocessen ondersteund worden door applicaties, via applicatieservices.



Figuur 5-10: Voorbeeld van een batch en online (handmatig) variant van hetzelfde proces

Gevolgen: Niet van toepassing.

Alternatieven: Niet van toepassing.

Relaties met andere good practices: Zie ook de good practices 'Bedrijfservice, bedrijfsfunctie en bedrijfsproces', 'Afbakening bedrijfsproces' en 'Viewpoints voor procesmodellering'.

5.5 Servicebroker

Vraagstelling: Hoe modelleren je het gedrag van een servicebroker in relatie tot de applicaties binnen de applicatieportfolio?

Het doel van een servicebroker is het vereenvoudigen van de vele relaties tussen verschillende applicaties.

		demonstratiemodel Profit	Acceptatie systeem	Beheer Applicatie	Beheren klantgegevens	Beoordelings systeem	Betalings systeem	Document Informatie Systeem	Klantbeheer applicatie	Klantgegevens module	Product module	Registratie systeem	Tussenpersoon beheer applicatie	Tussenpersoon beheer module
	demonstratiemodel Profit													
	Acceptatie systeem													
	Beheer Applicatie													
	Beheren klantgegevens													
	Beoordelings systeem													
	Betalings systeem		V											
	Document Informatie Systeem					V								
	Klantbeheer applicatie					V								
	Klantgegevens module													
	Product module													
	Registratie systeem		V			V								
	Tussenpersoon beheer applicatie													
	Tussenpersoon beheer module													

Figuur 5-12: Voorbeeld van een overzicht van koppelingen tussen applicaties

Gevolgen: Om dit concept te kunnen modelleren moeten meerdere views worden gemaakt. Deze worden dan aan elkaar gekoppeld om het volledige concept te kunnen visualiseren.

Alternatieven: Niet van toepassing

Relaties met andere good practices: De servicebroker is een onderdeel van het applicatielandschap. Zie hiervoor de good practice 'Appicatielandschap'.

Referenties

- M. Lankhorst, *Enterprise Architecture At Work – Modelling, Communication, and Analysis*, Telematica Instituut/Springer, 2005, ISBN 3-540-24371-2.
- H. Bosma, H. Jonkers en M. Lankhorst, *Inleiding in de ArchiMate-taal*, ArchiMate/D.1.1.6a, v1.1, Telematica Instituut/Ordina, 24 oktober 2005.
<https://doc.telin.nl/dscgi/ds.py/Get/File-49772>.
- R. van Buuren, S. Hoppenbrouwers, H. Jonkers, M. Lankhorst en G. Veldhuijzen van Zanten, *Architecture Language Reference Manual*, ArchiMate/D2.2.2b, v4.1, Telematica Instituut, 4 april 2006. <https://doc.telin.nl/dscgi/ds.py/Get/File-31626>.
- H. ter Doest, M.-E. Iacob, M. Lankhorst, D. van Leeuwen en R. Slagter, *Viewpoints Functionality and Examples*, ArchiMate/D3.4.1a, v2.6, Telematica Instituut, 18 november 2004. <https://doc.telin.nl/dscgi/ds.py/Get/File-35434>.
- ArchiMate Team, *ArchiMate Quick Reference*, Telematica Instituut, 2005.
<https://doc.telin.nl/dscgi/ds.py/Get/File-52048>.