

# **Stakeholders, concerns, principes en patronen in data- architectuur**

*Bert Dingemans*

## Abstract

Veranderingen in en rond organisatie zijn van invloed op de rol van de data-architect. Door deze veranderingen wisselen stakeholders en hun concerns. Dit whitepaper gaat in op de belangrijkste stakeholders, hun belangen en hoe de architect dit kan gebruiken bij het uitvoeren van zijn rol.

Naast de concerns en requirements wordt ingegaan op een aantal andere concepten zoals kwaliteiten, principes en patronen. Deze worden uitgewerkt gericht op het invullen van de drie data functies en worden gerelateerd aan de genoemde concerns

## Inhoud

Abstract .....	2
Inleiding .....	4
ISO 42010 model .....	4
Stakeholders .....	5
Stakeholders analyse .....	5
“Onzichtbare stakeholders” .....	7
Stakeholder management .....	8
Concerns of doelen .....	8
Inventarisatie .....	9
Concerns modelleren .....	10
Acceptatiecriteria .....	12
Kwaliteiten .....	13
Data kwaliteiten .....	13
Softwarekwaliteiten .....	13
Organisatievolwassenheid .....	14
Principes .....	14
Patronen .....	15
Datamodel patronen .....	15
Semantische patronen .....	16
Integratie en SOA patronen .....	16

Software en Interactiepatronen .....	17
Literatuur.....	17

## **Inleiding**

Data-architectuur is een bijzondere manier van kijken naar organisaties, verandering en informatievoorziening. Dat “kijken naar” deze verschillende aspecten van een verband van personen, belangen en modellen is iets wat alle betrokkenen bij architectuur in zijn algemeenheid en data-architectuur in het bijzonder op eigen wijze doen.

Reeds in het whitepaper over de rol van de architect in data architectuur en organisatieverandering is ingegaan op het belang om rekening te houden met de belangen, doelen en behoeften van de betrokkenen. Hierbij is stakeholder management een middel waarmee de architect zorg kan dragen voor een goede inbedding van de belangenbehartiging van de verschillende stakeholder. Dat dient op een dusdanige wijze te gebeuren dat alle betrokkenen het idee hebben dat hun belangen en wensen voldoende gehonoreerd worden.

Om dat laatste mogelijk te maken is een analyse van de verschillende stakeholders en een inventarisatie van hun belangen een minimale vereiste. Zeker in grote organisaties waarbij je te maken hebt met veel stakeholders en een veelheid van belangen is het vertalen van al deze specifieke belangen en behoeften naar meer generieke beschrijvingen zoals principes en patronen een hulpmiddel gericht op de beheersbaarheid. In dit whitepaper gaan we in op dit aspect van het werkveld van de data-architect

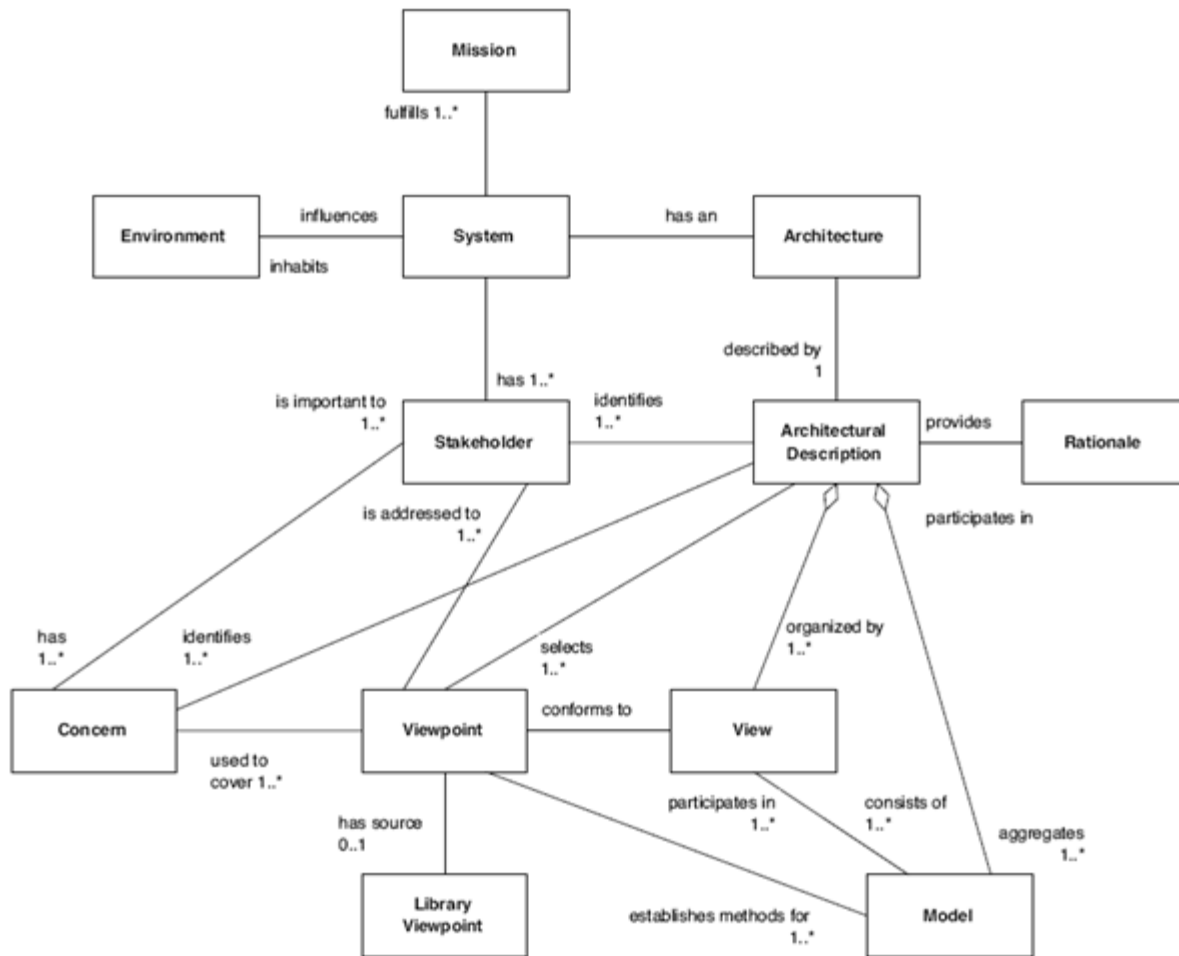
## **ISO 42010 model**

Op het vlak van enterprise architectuur is door een aantal standaardisatie organisaties een architectuurmodel opgesteld dat de belangrijkste elementen rond stakeholders en concerns in een model.

Deze standaard geeft enerzijds een de definitie van architectuur. Anderzijds geeft het een aantal definities van wat er onder verschillende entiteiten verstaan wordt. Dat is geen overdreven luxe gezien de homoniemen en synoniemen problemen die in het vakgebied voorkomen. Denk hierbij aan termen als system, model en view.

In onderstaande afbeelding zie je een uitwerking van het model. In dit model neemt de stakeholder en zijn concerns een centrale plaats aan. Daarnaast vervullen de views en viewpoints een belangrijke functie.

Voor data architectuur is dit ISO model goed toepasbaar. Wel is het belangrijk om te beseffen dat een data-architect andere stakeholders, concerns en viewpoints heeft dan een andersoortige architect. In dit whitepaper gaan we in op de specifieke data-architectuur stakeholders en hun concerns.



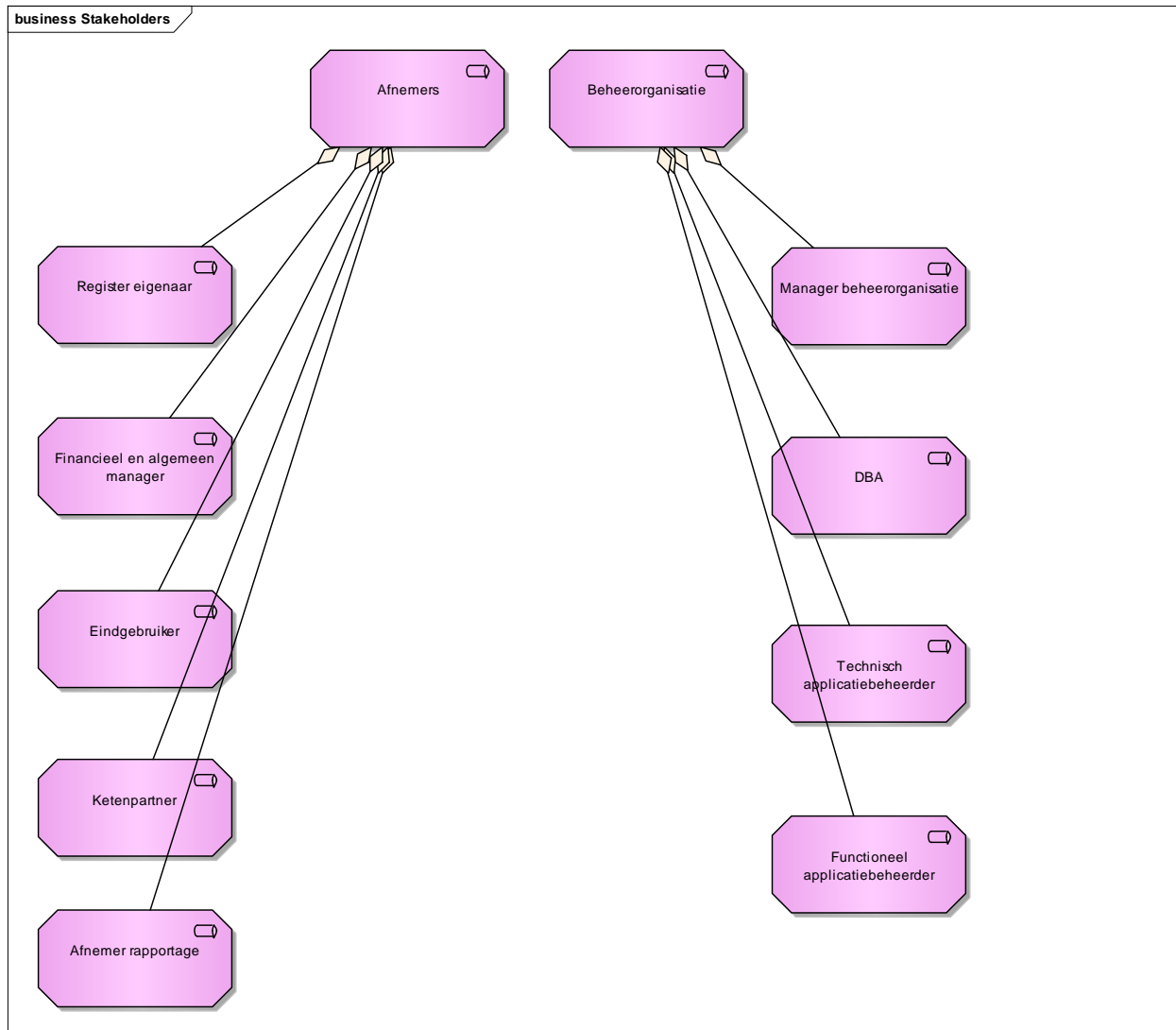
Afbeelding: ISO achitectuurmodel

## Stakeholders

In de definitie van ISO 42010 is een stakeholder een actor in het “system of interest” van wie de concerns of belangen van fundamenteel belang zijn voor de architectuur. Met andere woorden de belangen die architectonisch belangrijk zijn.

### *Stakeholders analyse*

Op basis van bovenstaande definitie kunnen de stakeholders in kaart gebracht worden voor een data-architectuur. Hierbij kan de motivation extensie van ArchiMate hulp bieden bij het categoriseren van de stakeholders en hun onderlinge relaties. Onderstaande afbeelding toont een aantal algemeen geldende data-architectuur stakeholders. Houdt er rekening mee dat in de eigen organisatie en omgeving een andere rolbenaming en –indeling gebruikt kan worden. Stel daarom voor de eigen situatie een specifiek model op.



Afbeelding: Stakeholders.

Bij de uitwerking van de stakeholders is er in deze afbeelding een tweedeling gemaakt in afnemers en beheerders. Afnemers zijn de uiteindelijke gebruikers van de data en hebben daardoor specifieke belangen. Vaak zijn die belangen sterk gerelateerd op de software die tussen de stakeholder en de gegevens in zit. Toch zijn er ook voor de data architectuur een aantal afnemers concerns te benoemen zoals zal blijken in de volgende paragraaf.

De beheerorganisatie is voor de data-architectuur een ander belangrijk cluster. Zij stellen eisen aan andere aspecten dan de afnemers, ook al zijn zij vaak zelf ook afnemer van specifieke gegevenssets. Hierbij ontstaat een soort van functionele- en niet-functionele rubricering die in latere paragrafen verder uitgewerkt wordt.

## Afnemers

- **Register eigenaar**, is de belangrijkste stakeholder. Hij heeft enerzijds belang bij een goede kwaliteit van de gegevens, zal er zorg voor dragen dat de gegevens effectief gebruikt worden

door de betrokkenen. Vaak is de register eigenaar ook eigenaar van de processen die gerelateerd zijn aan de gegevens. In die situatie is het belang van een goede aansluiting van de processen op de gegevens en omgekeerd goed geïmplementeerd is.

- **Financieel of algemeen manager**, heeft belangen op het vlak van de organisatie, de continuïteit en de omzet en winst. Deze aspecten kunnen van invloed zijn op de gegevens, bijvoorbeeld is dit een kostenpost of kunnen met de gegevens nieuwe diensten en markten gerealiseerd worden.
- **Ketenpartner**, gegevens worden steeds vaker over de grenzen van afdelingen en organisaties heen ingezet. Hiermee zijn ketenpartners, zowel afnemers als producenten belangrijke stakeholders geworden. In hoeverre is bijvoorbeeld een gegevensleverancier bereid om aanpassingen te doen aan gegevensleveringen. Maar ook afnemers zijn in deze van belang, is het doen van kwalitatieve gegevensleveringen naar deze afnemers eenvoudig realiseerbaar
- **Eindgebruiker**, is de register eigenaar een stakeholder, ook de uiteindelijke gebruikers van de gegevens zijn van belang. In hoeverre ondersteunen de gegevens op efficiënte wijze de activiteiten van de eindgebruikers. Zeker bij een diverse gebruikerspopulatie of bij verschillende wensen kunnen de belangen van deze eindgebruikers lastig te realiseren zijn
- **Afnemer rapportages**, dit is een groep van bijzondere eindgebruikers. Vaak stellen zij specifieke eisen aan de datamodellering, maar ook eisen op het vlak van mutaties, transactielogging en historie van gegevens. Gevolg van de concerns rond rapportages vraagt bijzonder vormen van gegevensinrichting. Bijvoorbeeld de inzet van datawarehouses e.d.

## Beheerorganisatie

- **Manager beheerorganisatie**, beheren van informatiesystemen en de bijbehorende gegevens set is complex en zeker bij een suboptimale inrichting van het landschap duur. Dat brengt met zich mee dat de manager van een beheerorganisatie veelal concerns heeft op het vlak van complexiteitsreductie en onderhoudbaarheid.
- **DBA**, is een bijzonder beheerder, met name omdat hij gericht is op de inrichting van de gegevensverzamelingen in de fysieke infrastructuur. Zeker bij grote en complexe gegevensverzamelingen worden de concerns van deze stakeholders belangrijk. Denk bijvoorbeeld aan situaties waarbij een hoge beschikbaarheid van de gegevens bijzondere eisen stelt aan de inrichting
- **Technisch applicatiebeheerder**, is gericht op een aantal data- of softwarekwaliteiten. Denk hierbij aan continuïteit en risicobeheersing maar ook aan onderhoudbaarheid, uitbreidbaarheid en aanpasbaarheid van gegevenssets en de bijbehorende software.
- **Functioneel applicatiebeheerder**, hebben vergelijkbare concerns als de technisch applicatiebeheerders maar de focus is veel meer gericht op de semantiek van de gegevens. Beschrijving van de datasets in combinatie met begrijpelijkheid en analyseerbaarheid zijn voorbeelden van concerns

*“Onzichtbare stakeholders”*

Naast de stakeholders die evident een belang hebben bij de data-architectuur zijn er ook zogenaamde “onzichtbare stakeholders”. Dit zijn personen of organisaties die zeker een rol

spelen bij ontwikkelingen op het werkveld maar als belanghebbende niet direct aanspreekbaar zijn. Denk hierbij bijvoorbeeld aan:

- **Wetgever**, de overheid als wetgevend macht stelt eisen aan de inrichting van de data-architectuur. Bijvoorbeeld op het vlak van de bescherming van persoonsgegevens en privacy maar ook op het vlak van traceerbaarheid van bewerkingen en dergelijke.
- **Concurrentie**, voor commerciële organisaties hebben de activiteiten van de concurrerende bedrijven direct invloed op de interne ontwikkelingen. Veelal zal dit gesignaleerd worden door het management van de eigen organisatie maar ook de data-architect kan hierin een rol vervullen.
- **Technologische ontwikkelingen**, zowel op ICT vlak als op het vlak van data gaan de ontwikkelingen stormachtig. Denk aan ontwikkelingen als big-data, datavirtualisatie, servicetechnologie. Hiermee hebben deze ontwikkelingen indirecte belangen op het vlak van de interne data-architectuur
- **Leveranciers**, leveranciers zijn enerzijds een directe stakeholders bij de uitvoering van ontwikkel- of beheerwerkzaamheden. Anderzijds hebben zij vaak andere belangen dan de opdrachtgever. Denk hierbij bijvoorbeeld aan continuïteit van de werkzaamheden, het inzetten van producten en diensten die lucratiever zijn etcetera.
- **Overkoepelende organisaties**, bijvoorbeeld binnen de Nederlandse overheid zijn er allerlei samenwerkingsverbanden maar ook beleid ontwikkelende organisaties die van invloed zijn op de data-architectuur. Denk bijvoorbeeld aan de overheidsreferentie architecturen, of het overheid informatiebeveiligingsinstituut die mede bepalend zijn voor de ontwikkelingen in het werkveld

### Stakeholder management

Het in kaart brengen van de stakeholders door een lijst te definiëren is een goed begin van stakeholdermanagement. Vervolgstap is het categoriseren van deze stakeholders. Togaf heeft daar een magisch kwadrant voor ontwikkeld waarbinnen de stakeholders over twee assen ingedeeld kunnen worden.

		Interesseniveau	
		Laag	Hoog
Macht	Hoog	Tevreden houden	Sleutelfiguur
	Laag	Minimale inspanning	Informereren

Andere optie is het inzetten van een RACI diagram in combinatie met een aantal concerns, RACI staat voor:

- **Responsible**, is verantwoordelijk
- **Accountable**, dient te betalen
- **Consulted**, wordt om advies gevraagd
- **Informed**, wordt geïnformeerd.

### Concerns of doelen



ISO 42010 identificeert concerns als volgt: de toepassing van het systeem, de geschiktheid van de architectuur voor het bereiken van doelen van het systeem, de haalbaarheid van de bouw en invoering van het systeem, de mogelijke risico's en gevolgen van het systeem en haar belanghebbenden gedurende hun levenscyclus. Bijvoorbeeld onderhoudbaarheid en evolueerbaarheid van het systeem.

### *Inventarisatie*

Hieruit blijkt dat er per stakeholder heel veel concerns zullen zijn en dat de concerns divers kunnen zijn. Echter een inventarisatie zal toegevoegde waarde hebben. Rubriceren van concerns biedt hierbij structuur.

Over werkvormen bij concern analyse kan veel gezegd worden. Er zijn vele creatieve werkvormen mogelijk, zeker op het vlak van interactieve analyse. De werkvormen kennen veelal een herkenbare opbouw bestaande uit de volgende stappen:

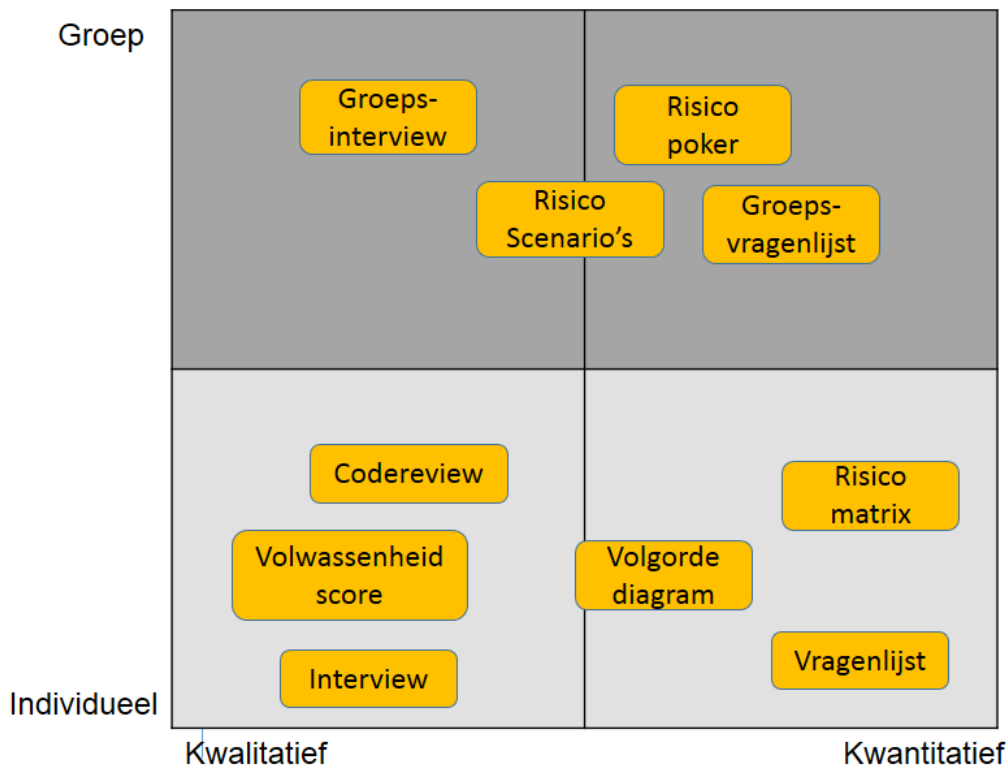
- Bepaal de relevante stakeholders
- Betrek betrokkenen bij de scope en bepaal scope bijvoorbeeld informatiebeveiliging, virtualisatie, beheer of ontwikkeling
- Bepaal de mate van detail
- Benoem het te analyseren element (bijvoorbeeld risico(effect/frequentie/maatregel))
- Bepaal de inventarisatiewijze

We geven hier geen uitputtende lijst van de inventarisatiewijze, maar een aantal voorbeelden:

- **Groepsinterview;** de architect gaat met een groep betrokkenen in gesprek en bepaalt op basis hiervan zelf een score.
- **Risicoscenario's;** in samenspraak met betrokkenen een als-dan scenario uitwerken rond een bepaald risico, is veelal kwalitatief van opzet maar kan ook op basis van kansberekeningen uitgevoerd worden.
- **Risicopoker;** een vorm afkomstig uit een agile ontwikkelmethode waarbij met een groep betrokkenen een bepaalde risico beschreven wordt waarbij vervolgens de betrokkenen een cijfer geven voor de kans van optreden. De hoogste en laagste score lichten hun keuze toe waarna de groep opnieuw scoort.
- **Groepsvragenlijst;** met een groep worden de vragen besproken en vervolgens wordt een gezamenlijk antwoord geformuleerd.
- **Codereview;** veelal gebaseerd op non-functionele aspecten wordt gekeken hoe de technische onderdelen zijn opgebouwd, ingericht en geïmplementeerd. Wordt individueel of door kleine teams uitgevoerd.
- **Volwassenheidsscore;** een bijzondere en krachtige wijze om de kwaliteit binnen een bepaald aspect te bepalen. Voordeel van deze opzet is dat met een beperkt aantal vragen over specifieke kenmerken een score bepaald kan worden.

- **Volgordediagram;** plaats vergelijkbare elementen in een volgorde voor het bepalen van de voorkeur en vertaal deze in een score. Kan goed toegepast worden om bij grote aantal dezelfde elementen snel een beeld te bepalen.
- **Concernmatrix;** indeling van concerns over twee dimensies en vervolgens indelen van elementen binnen deze dimensie en op basis daarvan een score bepalen of elementen uit de architectuur indelen over deze twee assen.

Welke werkvorm dient voor welk doel geselecteerd te worden? Voor **een** economische analyse is meestal een kwantitatieve individuele analyse mogelijk, voor stakeholderparticipatie zijn kwalitatieve interactieve analyses goed inzetbaar. Voor concerninventarisaties is dit afhankelijk van het onderwerp. In onderstaande figuur een indeling op basis van twee analysevormen.



Afbeelding: Concernmatrix

### *Concerns modelleren*

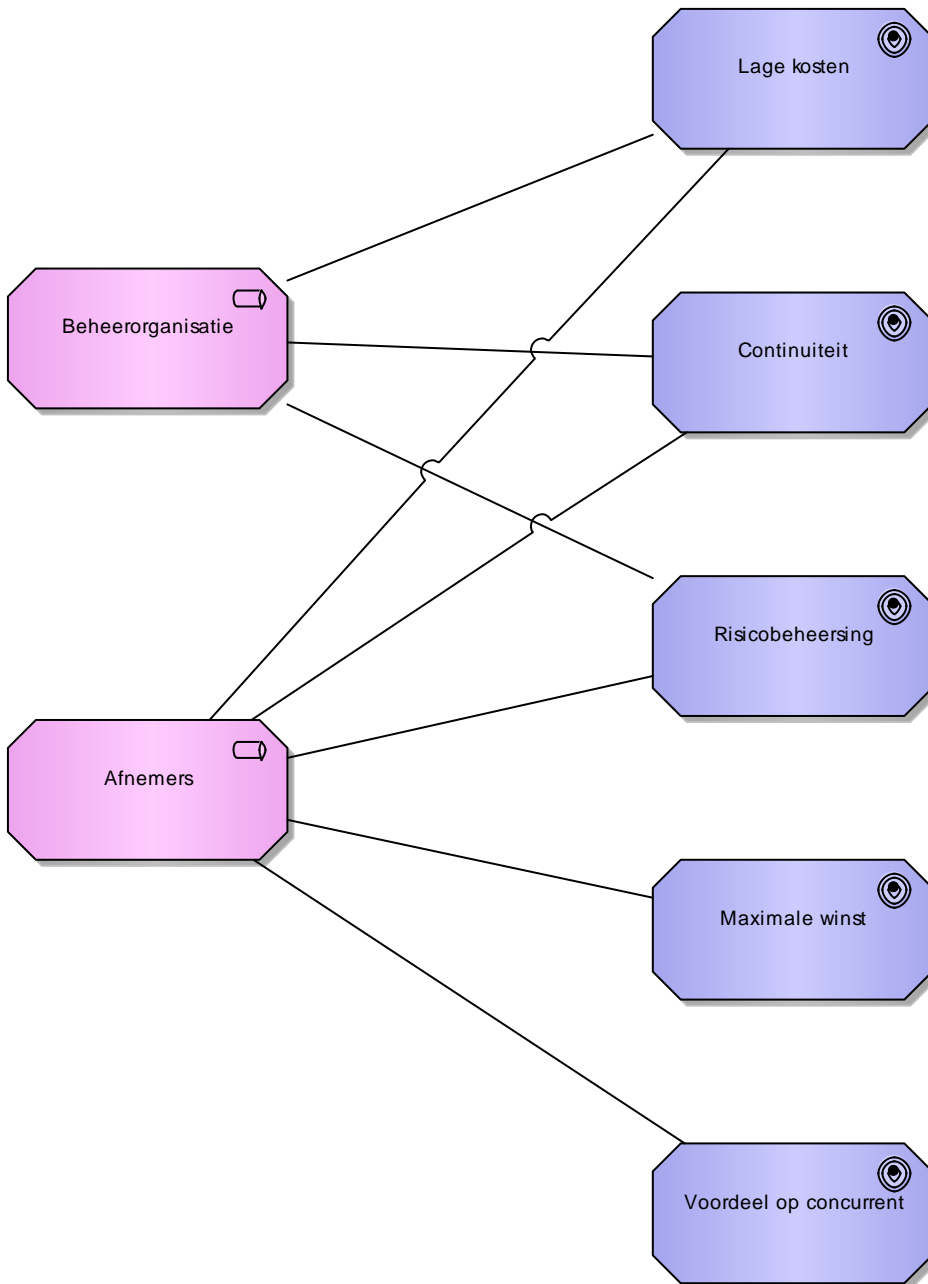
Een belangrijk aspect van concern inventarisatie is het inzichtelijk maken van de verbanden van de concerns onderling en met andere concepten zoals stakeholders en componenten., Dit geldt zowel voor de huidige (baseline) als de gewenste doel (target) architectuur. Dit modelleren

bestaat uit een tekstuele beschrijving waarbij de analyseresultaten veelal op eenvoudige wijze toegevoegd en geïntegreerd kunnen worden met de tekstuele inhoud.

Naast deze tekst zal een visuele weergave van de beschrijving van de architectuur toegevoegde waarde hebben in de zin van het bieden van overzicht. Voor deze visuele weergave zijn er meerdere mogelijkheden. Denk bijvoorbeeld aan matrices en bomen. Daarnaast biedt ArchiMate een aantal relevante concepten.

Sinds versie 2.0 van ArchiMate is het mogelijk om met deze taal binnen een aantal extensies te modelleren, waaronder de motivatie of de aanleiding. Hierin kunnen diverse aspecten gemodelleerd worden, zoals concerns, requirements, stakeholders en principes. Deze indeling is een goed hulpmiddel om de verschillende uitwerkingen van kwalitatieve en kwantitatieve analysemethoden te categoriseren en toe te wijzen aan de basiselementen zoals processen, functies en componenten binnen de architectuurbeschrijving. In onderstaande afbeelding een eenvoudig voorbeeld van concerns en stakeholders.

## motivation Stakeholders en concerns



Afbeelding: Archimate concerns en stakeholders

### *Acceptatiecriteria*

Een werkwijze die nauwe raakvlakken heeft met het inventariseren van concerns is het inzetten van acceptatiecriteria. Deze werkwijze wordt in beheerorganisaties steeds vaker ingezet. Krachtig aan acceptatiecriteria is dat deze getoetst kunnen worden. De werkwijze is dat voorafgaand aan een veranderingstraject de acceptatiecriteria opgesteld worden. Dit gebeurt op

een dusdanige wijze dat de verandering voorafgaand aan de daadwerkelijke implementatie in de (beheer)organisatie getoetst kan worden op basis van deze acceptatiecriteria.

Wanneer geïnvesteerd wordt in de uitwerking van acceptatiecriteria ontstaat een register van specifieke en generieke acceptatiecriteria. Daarmee worden acceptatiecriteria herbruikbaar en wordt het opstellen van een lijst van acceptatiecriteria voor toekomstige verandertrajecten eenvoudiger realiseerbaar.

## **Kwaliteiten**

Een specifieke vorm van concerns zijn kwaliteiten. Kwaliteiten zijn wat meer gericht op de implementatie en beheer, maar zijn zeker ook relevant vanuit een afnemersperspectief.

Kwaliteiten kunnen ingezet worden bij een vraag- en aanbod inventarisatie. Kwaliteiten worden voornamelijk gebruikt bij non-functionele requirements.

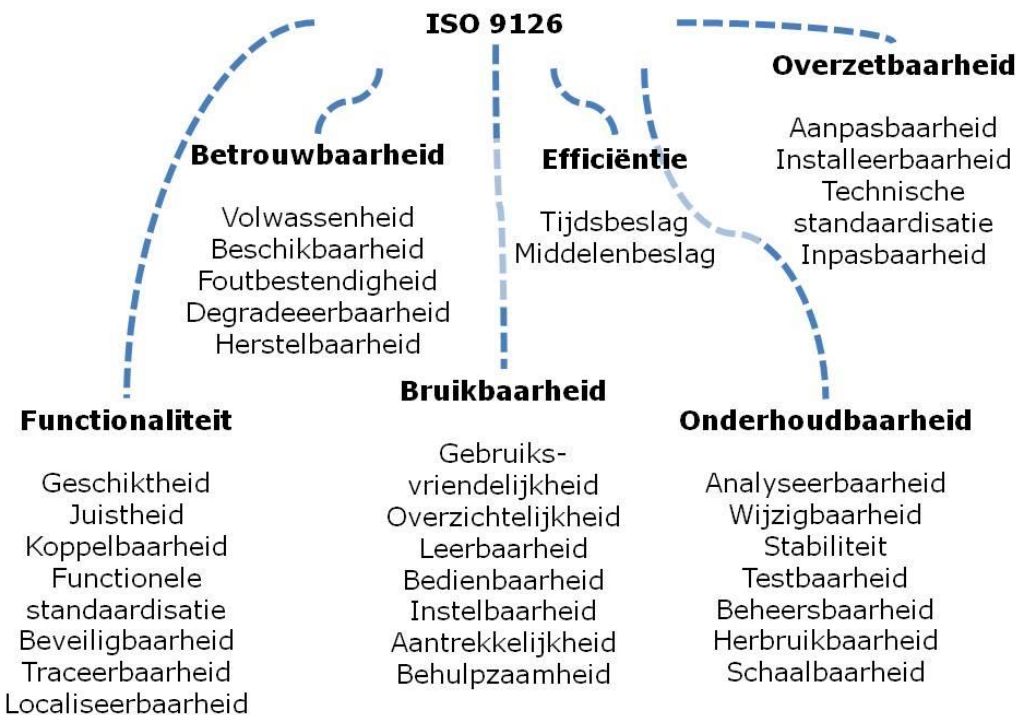
Voor meerdere vakgebieden zijn kwaliteiten uitgewerkt, niet alle zijn relevant. In onderstaande paragrafen wordt ingegaan op een aantal voor data-architectuur relevante kwaliteiten.

### *Data kwaliteiten*

Door DaMa International een internationaal consortium op het gebied van data management zijn een aantal data kwaliteiten geformuleerd die bij het in kaart brengen van concerns een goed hulpmiddel zijn. Hiervoor is een whitepaper geschreven en beschikbaar op [www.architecuurassistent.nl](http://www.architecuurassistent.nl) [datakwaliteiten].

### *Softwarekwaliteiten*

Niet alleen op het vlak van data zijn kwaliteiten inzetbaar. Op het vlak van datagebruik en data-integratie zijn softwarekwaliteiten van belang. Hiervoor is een ISO standaard opgesteld namelijk ISO 9126. Deze is later uitgebreid in het Quint 2 model dat in Nederland meer wordt toegepast dan de ISO 9126 indeling. In onderstaande afbeelding een overzicht van de Quint 2 kwaliteitseisen.



Afbeelding: Quint 2

Ten behoeve van het inventariseren van softwarekwaliteiten is een checklist opgesteld op basis waarvan twee scenario's met elkaar vergeleken kunnen worden. Zie de link voor de deze softwarekwaliteiten checklist: <http://assistent.interactory.nl/frmQuint.aspx>.

### *Organisatievolwassenheid*

Een bijzondere vorm van het inventariseren van kwaliteiten is het kijken naar de volwassenheid van een organisatie of een organisatie-onderdeel. Het toepassen van bepaalde concerns en requirements is te relateren aan het volwassenheidsniveau van een organisatie. Zo zal het inventariseren van datakwaliteiten in een omgeving met een lage volwassenheid weinig toegevoegde waarde hebben. Dat komt omdat de relevante stakeholders hele andere dagelijkse problemen hebben en daardoor geen beeld hebben bij kwaliteit van data.

Het inventariseren van deze volwassenheid kan beschouwd worden als een specifieke vorm van kwaliteitsanalyse. Een dergelijke werkwijze kan helpen bij het zoeken van de juiste hulpmiddelen om in te zetten bij het inventariseren van stakeholders en concerns [Rijn].

### **Principes**

Het aantal concerns bij een inventarisatie zal snel toenemen. Gevolg is dat werken met concerns al snel complex en tijdrovend wordt. Principes zijn dan een hulpmiddel om concerns te combineren en te herformuleren naar principes.

Principes zijn algemene regels en richtlijnen, gericht op langdurig inzet, worden zelden gewijzigd en geven invulling aan hoe de organisatie is ingericht om haar doelen te realiseren

[Togaf]. Hiermee zijn principes een goed hulpmiddel om op een samenvattende en krachtige manier doelen en belangen aan de architectuur te relateren. Over data-principes is daarom een whitepaper beschikbaar dat principes in detail uitwerkt [data-principes].

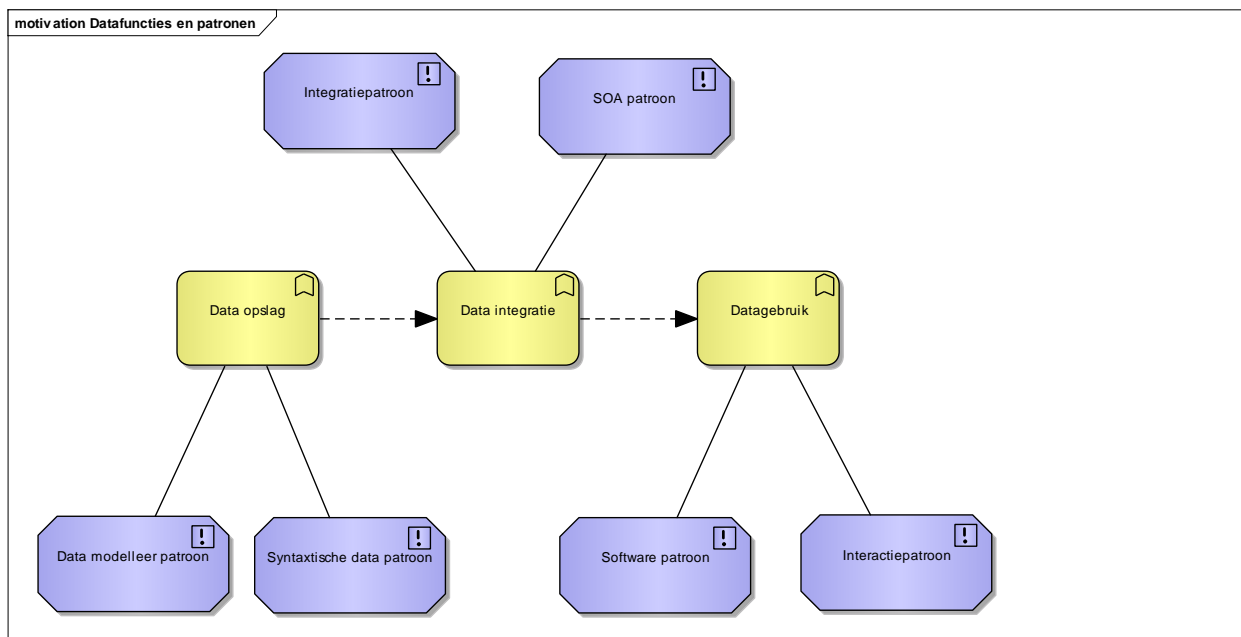
## Patronen

Volgens de definitie van ontwerppatronen, “Een **ontwerppatroon** of **patroon** in de informatica is een generiek opgezette softwarestructuur, die een bepaald veelvoorkomend type software-ontwerpprobleem oplost.” zijn het oplossingen voor veelvoorkomende ontwerpproblemen.

Daarmee worden patronen vanuit het oogpunt van stakeholders en concerns interessant, problemen in het algemeen en ontwerpproblemen in het bijzonder hebben altijd de aandacht van één of meerdere betrokkenen.

Toch hebben patronen een ander karakter dan de voorgaande concepten als principes. Met name de oplossingsgerichtheid maakt ze interessant. Door inzet van patronen is de sturing van de architectuur een veel natuurlijker proces. Er is een probleem in het werkveld en door inzet van één of meerdere patronen wordt hieraan invulling gegeven. Uitwerken van verschillende patroonregisters is daarmee vanuit de rol van de architect een krachtig sturingsmiddel.

Ontwerp- en architectuurpatronen zijn er voor meerdere werkgebieden, denk aan software, integratie of datapatronen. In onderstaande afbeelding is een indeling gemaakt op basis van de data functies met een aantal relevante patroonvormen. Deze worden vervolgens in onderstaande paragrafen toegelicht



Afbeelding: Data functies en patronen

### *Datamodel patronen*

Datamodelpatronen zijn een beschrijving van best practices voor het uitwerken van datamodellen. Ze zijn met name interessant voor de data opslag maar komen ook voor binnen integratie en SOA patroontalen.

Idee achter deze patronen is dat bij het modelleren van data er grote delen van het datamodel generiek zijn. Bijvoorbeeld bij het modelleren van een persoon zijn bijna altijd voornaam, tussenvoegsel en achternaam relevant. Door modellen op te stellen van deze generieke entiteiten, eigenschappen en onderlinge relaties ontstaat een register van initiele modellen die ingezet kunnen worden bij het uitwerken van het eigen datamodel. Het is hierbij niet ongebruikelijk om meerdere van deze patronen met elkaar te combineren.

Datamodelpatronen bestaan al vele tientallen jaren, daardoor is literatuur rond dit onderwerp veelal al enige jaren oud. Dat is echter geen enkel bezwaar zie bijvoorbeeld de “Data model patterns, convention of thoughts” van David Hay wat nog steeds waardevolle modellen bevat. Een andere interessante bron zijn catalogie van registers zoals de stelselcatalogus van de Nederlandse overheid. Ook vermeldenswaardig zijn diverse open standaarden die voor de uitwisseling van gegevens in diverse branches datamodellen hebben ontwikkeld. Bijvoorbeeld UBL voor business entiteiten of HRXML voor HRM modellen.

### *Semantische patronen*

In de voorgaande paragraaf zijn de concrete datapatronen aan de orde gekomen. Deze patronen geven een gedetailleerde uitwerking van de entiteiten en veelal ook de eigenschappen. Er is echter ook een meer abstract niveau van modelleren waarbij je minder over datamodelpatronen maar meer over data-ontwerppatronen spreekt.

Bij data-ontwerppatronen worden meer abstracte patronen gebruikt die een oplossing geven voor een bepaald veelvoorkomend modelleerprobleem zonder dat dit concreet is toe te wijzen aan een werkveld. In onderstaande opsomming een drietal voorbeelden

- **Veel op veel relatie implementatie**, bij conceptuele modellering kunnen entiteiten op basis van veel op veel relaties aan elkaar gerelateerd zijn. Bijvoorbeeld een leraar heeft veel studenten en een student veel leraren. Dit wordt bij het uitwerken van een fysiek model omgezet in een koppelentiteit vanwege het feit dat data opslag niet met veel op veel relaties kan omgaan.
- **Ouder-kind of bestaansafhankelijkheid relatie**, hierbij modelleer je een entiteit dat gerelateerd is aan een andere entiteit op een dusdanige wijze dat de eerste niet kan bestaan zonder de andere. Denk bijvoorbeeld aan een orderregel die niet kan bestaan zonder een order maar veelal ook niet zonder een artikel of een dienst.

Deze conceptuele patronen zijn minder zichtbaar in een model of een probleem maar bieden veelal een krachtige oplossing voor problemen, echter registers rond dit onderwerp zijn schaars.

### *Integratie en SOA patronen*

Integratie en SOA patronen richten zich onder andere op data integratie, veelal zijn deze patronen gericht op het transformeren van data. Daarnaast zijn er patronen gericht op verschillende transport- en opslagmechanismen. Een goed voorbeeld is de Enterprise Service Bus, een concept dat bij veel organisaties conceptueel of fysiek wordt ingezet. Een ESB is de implementatie van samengestelde SOA patronen (zie:

[http://soapatterns.org/compound\\_patterns/enterprise\\_service\\_bus](http://soapatterns.org/compound_patterns/enterprise_service_bus))

Op het vlak van applicatie integratie en SOA zijn meerdere patronen en patroonregisters aanwezig. Zie bijvoorbeeld <http://www.eaipatterns.com> of <http://www.soapatterns.org>. Deze



registers kennen patronen op meerdere integratiewerkvelden waaronder data transformatie en – representatie.

### *Software en Interactiepatronen*

Binnen datagebruik worden met name de softwarepatronen toegepast. Veelal gericht op transformatie- en validatie van data. Interessant en minder bekend zijn interactiepatronen. Dit zijn patronen voor de grafische gebruikersinterface en voor Human Computer Interaction. Op het vlak van interactiepatronen is voor een architect veel te bereiken. Het uitwerken van een beperkte set van interactiepatronen die ingezet mogen worden voor de gebruikersinterface van informatiesystemen en websites e.d. vergroot het gebruiksgemak. Op het vlak van interactiepatronen zijn een aantal goede registers (en literatuur) te vinden. Zie bijvoorbeeld het register <http://designinginterfaces.com/patterns/>.

## **Literatuur**

Best, Bart de, *Acceptatiecriteria*, Academic service, 2011.

Datamanagement International, [www.dama.org](http://www.dama.org).

Dingemans, Bert, *Datakwaliteiten*, [www.architectuurassistent.nl](http://www.architectuurassistent.nl), 2013.

Hay, David, *Data model patterns, convention of thoughts*, Pearson Education, 1996.

ISO, <http://www.iso-architecture.org/ieee-1471/>

Rijn, Ria van, *Volwassenheid en data-architectuur*, [www.architectuurassistent.nl](http://www.architectuurassistent.nl), 2013.

The Open Group, *ArchiMate 2.0*, Van Haren Publishing, 2011.

The Open Group, *Togaf 9*, Van Haren Publishing, 2011.