

WHITE PAPER

DYA | ARCHITECTUURPRINCIPES

DEEL 1 - BASICS

SERGE BOUWENS

WHITE PAPER

DYA | ARCHITECTUURPRINCIPES

DEEL 1 - BASICS

Serge Bouwens

Versie: 1.0
oktober 2009

Copyright Sogeti Nederland B.V. © te Vianen

Niets uit deze uitgave mag worden verveelvoudigd (voor willekeurig welke doeleinden) door middel van druk, fotokopie, microfilm, geluidsband, elektronisch of op welke andere wijze dan ook zonder voorafgaande schriftelijke toestemming van Sogeti Nederland B.V.



INHOUDSOPGAVE

INHOUDSOPGAVE	1
1 INLEIDING	3
1.1 Een visie op architectuur op basis van principes	3
1.2 Leeswijzer	3
1.3 En verder	3
2 VISIE OP ARCHITECTUUR.....	5
3 HET CONCEPT ‘ARCHITECTUURPRINCIPES’	10
3.1 Doel en definitie	10
3.2 Principes in vele smaken.....	13
3.3 Wat zijn goede principes?	15
3.4 Vorm.....	17
4 GENETIC MAP	22
4.1 Normaalvorm.....	22
4.2 Tools.....	22
4.3 Analyses.....	27
5 ORDENING VAN ARCHITECTUURPRINCIPES	30
6 ARCHITECTUURPRINCIPES VERSUS BUSINESS RULES	33
6.1 Inleiding business rules	33
6.2 Business rules: Trends.....	34
6.3 Business rules, principes en requirements	35
6.4 Business rules en architectuur.....	36
6.5 Business Rules en DYA.....	37
7 BIJLAGE: BRONNEN	38

1 INLEIDING

DYA is hét label van Sogeti als het gaat om architectuur dienstverlening. Met [DYA](#) heeft Sogeti zich de afgelopen 5 jaar op de kaart gezet als een van de spelers –zo niet dé speler- in architectuurland. Die naam willen we graag verder uitbouwen.

Onze klanten vragen om een verdere inhoudelijke verdieping van de architectuur aanpak. En wel een die zowel aansluit bij de proces-insteek van DYA, als bij andere dienstverlening van Sogeti, zoals bijvoorbeeld RLcM (Requierelements Lifecycle Management).

Met het project DYA | Architectuurprincipes vullen we deze vraag deels in.

1.1 Een visie op architectuur op basis van principes

Met de opkomst van Enterprise Architectuur zijn architectuurprincipes steeds belangrijker geworden. Ze voldoen aan de behoefte aan een inhoudelijk sturingsinstrument op een hanteerbaar abstractieniveau. Toch blijkt het lastig om principes praktisch in te zetten in de IT-omgeving. De afstand tussen de vraagkant en de aanbodkant blijkt groot. Het alternatief –blauwdrukken van de gewenste situatie- is echter ook niet realistisch gebleken: te complex, te duur, te snel verouderd. In de afgelopen jaren heeft Sogeti een visie op architectuur en een aanpak van architectuur ontwikkeld op basis van principes. Kenmerkend voor deze visie en aanpak zijn:

- Gebaseerd op best practises;
- Hergebruik van generieke architectuurprincipes;
- Passend in de Just-in-time benadering van DYA.

Vanuit een analyse van principes in de dagelijkse praktijk geven we een –in de praktijk getoetste- benadering op wat principes (zouden moeten) zijn, hoe je ze opstelt en hoe je ze in de praktijk gebruikt. Zo slaan we een brug tussen proces en inhoud, theorie en praktijk, tussen corporate IT en systeemontwikkeling in projecten, tussen business en IT. Kortom: architectuur die wérkt.

1.2 Leeswijzer

In deze white paper (deel 1) werken we de visie uit. In [hoofdstuk 2](#) beschrijven we de visie op hoofdlijnen. Vervolgens gaan we in [hoofdstuk 3](#) in op het concept van architectuurprincipe. In [hoofdstuk 4](#) bouwen we hierop voort en ontwikkelen wat tools en analyses. In [hoofdstuk 5](#) bespreken we de samenhang tussen principes onderling aan de hand van ordeningen uit de praktijk. We sluiten dit theoretische deel 1 af in [hoofdstuk 6](#) met de relatie tussen principes en business rules.

1.3 En verder

Whitepaper Deel 2

In een tweede paper (medio 2008) gaan we in op de toepassing in de praktijk. Dit doen we aan de hand van de processen Opstellen, Gebruiken in projecten (Ontwikkelen onder Architectuur), Gebruiken in de Strategische Dialoog, en Handhaven.

Sogeti Repository van Architectuur principes (SRAP).

Naast deze papers reiken we ook een aantal instrumenten aan. Als eerste een uitgebreide [Repository](#) van in de praktijk gebruikte architectuurprincipes. De repository kan door (vooralsnog alleen Sogeti) architecten gebruikt (en aangevuld!) worden als checklist en inspiratiebron, en maakt het mogelijk om bij het opstellen van architecturen voor klanten een jumpstart te maken. Een permanente 'under construction'-versie bevat op dit moment circa 280 architectuurprincipes, deels



uitgewerkt met de achterliggende rationale en de praktische consequenties voor de IT en organisatie.

Support

Een presentatie is beschikbaar op www.dya.info. In het boek DYA Topics is een hoofdstuk (4) opgenomen over hoe architectuurprincipes vertaald kunnen worden naar requirements in de context van projecten. Tevens hebben we op de diverse onderdelen [cursussen](#) ontwikkeld:

- De 1-daagse cursus "DYA | Architectuurprincipes" (t.b.v. A&BS leergang architectuur) is gereed.
- De 1-daagse cursus "DYA & RLcM" (t.b.v. de ESE leergang RLcM Foundation) is gereed.

2 VISIE OP ARCHITECTUUR

Architectuur is een vakgebied waarop veel invalshoeken mogelijk zijn. DYA belicht van origine de proceskant, Zachman benadrukt meer de modelmatige productkant, SOA definieert alles in termen van services, en de Infrastructurele Benadering van van der Sanden benadrukt de centrale rol van informatie. Elk van deze benaderingen had of heeft zijn waarde en zijn beperkingen. De principe gebaseerde benadering die we hier presenteren bouwt voort op het DYA fundament. Het is een stijl van architectuur bedrijven die voortkomt uit de observatie dat architectuur in veel organisaties - ondanks alle theorie - onvoldoende uit de verf komt: de business pruimt het niet, niet in de strategische dialoog, en niet in projecten. Het doel is dan ook om architectuur toegankelijker te maken.

Alvorens in te gaan op de technische kant van de zaak -definities, vorm en werkwijze- positioneren we architectuurprincipes eerst in een bredere visie op architectuur:

Definitie van architectuur

Hoewel dit geen wetenschappelijke publicatie is beginnen we toch maar om het begrip architectuur te definiëren.

In 2002 werd in het eerste DYA boek de volgende definitie opgevoerd:

"Architectuur is een consistent geheel van principes en modellen dat richting geeft aan ontwerp en realisatie van de processen, informatievoorziening, technische infrastructuur en organisatorische inrichting."

Dit is een prima definitie. Feitelijk verschilt hij niet zoveel van die van Archimate:

"Architectuur is een consistent geheel van principes, methoden en modellen voor het ontwerpen en realiseren van processen, informatievoorziening, technische infrastructuur en organisatorische inrichting."

En ook alleen fijnproevers zullen discussiëren over de verschillen met bijvoorbeeld die van IEEE 1471-2000:

"The fundamental organization of a system embodied in its components, their relationships to each other, and to the environment, and the principles guiding its design and evolution."

Dit zijn alle drie definities waar we ons redelijk in kunnen vinden. Toch roepen dit soort definities altijd weer vragen op: Is het volledig? Is het ook descriptief? Geldt dit voor elke vorm van architectuur? Etc. Een belangrijker bezwaar is dat het technocratische definities zijn. Dat hoeft (in ons vak) geen bezwaar te zijn, maar het klinkt toch een beetje vreemd, alsof we het hele vak in één volzin willen pakken. Zoiets als: "Broodbakken is het proces van graan malen, verrijken met al of niet kunstmatige, voedingstoffen, tot deeg omvormen m.b.v. rijsmiddel en water, mengen, kneden, bakken en zo vers mogelijk ter consumptie aanbieden." Krijgt u ook niet het idee dat dit overkill is, het resultaat van eindeloos polderen tussen vakgenoten? In ieder geval nodigt het niet uit tot een discussie met klanten over wat hún visie op de business is. En vervolgens, hoe wij daar vanuit ónze visie op het vakgebied aan kunnen bijdragen. Daarom geven we ook een alternatieve typering:

"Architectuur = Vorm geven aan visie"

Daar kunnen we mee uit de voeten: het heeft ambitie zonder dat het te veel ruimte geeft. En het is tenminste direct duidelijk dat architectuur een proces is (werkwoord) en geen verzameling documenten. Geheel in de geest van DYA dus.

Principes als architectuurvorm

Met de opkomst van enterprise architectuur heeft het fenomeen architectuurprincipes een steeds belangrijker plaats ingenomen binnen de architectuur.

Architectuurprincipes zijn een heel effectieve vorm gebleken om IT beleid c.q. architectuur in te zetten als instrument om verandering te faciliteren.

Modellen vs principes

Enterprise Architectuur als de totale verzameling van informatie-, proces-, applicatie-, en nog een dozijn andere architecturen, schiet hopeloos voorbij aan de primaire doelstelling van inzicht en overzicht. Er is in grotere organisaties niemand die al deze architectuurproducten kan doorgronden. Van de andere kant: Een

A4 met daarop de 10 architectuurgeboden is misschien genoeg om op strategisch niveau de juiste discussies te voeren, maar zwaar onvoldoende voor het achterliggende IT contingent. Wat is nu een goede insteek?

Ten opzichte van (grafische) modellen hebben principes enkele nadelen, maar toch vooral voordelen. Eén voordeel van modellen is dat modellen in de vorm van platen / ontwerpen op één pagina heel veel kan communiceren. Een plaatje zegt meer dan duizend woorden. Maar datzelfde plaatje roept vaak ook weer duizend vragen op. Bij die plaatjes horen dus ook praatjes. Principes zijn die praatjes. In die zin zijn modellen en principes complementair. Soms communiceren platen beter, soms tekst [DYA Topics]. Principes vertegenwoordigen beter de Demand kant van de zaak, Modellen beter de Solutions kant.

Net zoals grafische modellen gericht kunnen zijn op communicatie (praatplaten) of specificatie (zelden volstaat één plaat voor beide doelen), zo kunnen principes afhankelijk van de context gepresenteerd worden als one-liner of als een uitgebreide specificatie voor 'onder de motorkap'. Waar principes beter presteren dan grafische modellen, is in het *communiceren van een visie*. Een architectuurprincipe kan prima verwoord worden in een alinea tekst waarin de consequenties van bedrijfsdoelen worden aangestipt. Het is immers niet het resultaat, maar de redenering die draagvlak creëert. Ook in de alignment naar de uitvoering toe werken principes beter. Hoe dat werkt hebben we al beschreven in [DYA Topics].

We onderschrijven daarom het standpunt dat de 'taal der architectuurprincipes' ook 'gewoon' een modelleertaal is. Een stelsel van principes is ook een model, net zo goed als een Archimate plaat [Proper].

Ontwerpen vs ontwikkelen

Een zeker zo belangrijke reden is dat architectuur in de vorm van principes veel sneller resultaten oplevert. Dat komt omdat de intentie niet meer die is van het opleveren van een blauwdruk van de gewenste situatie, maar het opleveren van kaders waarbinnen (het ontwerp van) de oplossing gerealiseerd moet worden. Het ontwerpparadigma is zagezegd vervangen door het ontwikkelparadigma. Beslissingen kunnen zo eerder genomen worden, zonder dat het overzicht ten onder gaat in een overvloed aan details. Het ontwerptraject komt later; effectief is er dus een 'separation of concerns' geïntroduceerd tussen vraag en aanbod.

Merk op dat we wél onderscheid maken tussen ontwikkelen en ontwerpen, maar niet dat we het ontwerpen geheel buiten de architectuur plaatsen. In de discussie over 'prescriptief vs descriptief' nemen we nadrukkelijk een én-én standpunt in [13].

Communiceerbaar

Een ander aspect van principes dat op prijs gesteld wordt, is dat principes dezelfde vorm behouden, om het even over welke soort architectuur of branche we het hebben. Bij modellen kent iedere architectuurdiscipline weer zijn eigen modellen, conventies en symbolen, die de communicatie behoorlijk in de weg kunnen zitten. Bovendien kunnen we een heel eind komen met de universeel beschikbare Office tools.

Professionalisering

Al vaak is geconstateerd dat architectuur als vakdiscipline nog erg onvolwassen is. Er is een wildgroei aan methoden en aanpakken, maar de praktijk is weerbarstig. In elke situatie lijkt een afwijking op de standaard gerechtvaardigd. Van Rees schreef er een bekend artikel over: De Methode Werkt Niet [12].

Het is hier niet de plaats om diep in te gaan op de tekortkomingen van een al te academische aanpak, of de veralgemeniseringen van aanpakken die ergens succes geclaimd hebben. Zij hebben allemaal hun waarde. Wil het vak vooruitgang maken, in de zin dat architecturen reproduceerbaar worden, is het zaak dat we een praktische theorie ontwikkelen.

In onze visie zijn architectuurprincipes de basis waarop de architectuur, en vervolgens het hele bouwwerk van ontwerp tot beheer is gebaseerd. Zolang daar geen goede visie op is, onderbouwd met een theorie en methodisch kader, kan architectuur niet echt als wetenschap of als volwassen beschouwd worden. Een structurele inbedding van omgaan met architectuurprincipes is derhalve een voorwaarde voor de verdere professionalisering van architectuur. We verwachten dan ook dat de toenemende aandacht voor het onderwerp op universiteiten en in het bedrijfsleven de komende jaren zal blijven. Idealiter moet de 'body of knowledge' waar de architectencommunity uit put in een vorm gegoten worden zodat die beter onderwijsbaar wordt. Pas als we de techniek – de basics – onder de knie hebben, kunnen we ons met het echte werk bezig houden: het ondersteunen van management- en ontwerpbeslissingen.

80-20

Naar mate meer organisaties architectuurprincipes gaan formuleren, groeit ook het besef dat veel principes helemaal niet organisatiespecifiek zijn, maar vaak voortkomen uit best practises. Voor IT professionals vaak common sense, maar toch minder 'common' dan je zou willen. In het verleden hebben we ons gehaast om te vertellen dat architectuur natuurlijk afgeleid hoort te zijn uit de bedrijfsdoelstellingen, maar dat is feitelijk maar voor een deel waar. Bepaalde principes -bijvoorbeeld: "Gegevens worden bij de eerste invoer gecontroleerd"- kunnen natuurlijk best via een doel-middel-hiërarchie teruggetraceerd worden naar de doelstelling van efficiency, maar dat is toch een ietwat kunstmatige constructie achteraf.

Op basis van een jarenlange studie van architectuurprincipes die door organisaties worden geformuleerd, kan de volgende conclusie getrokken worden. De 'body of knowledge' van het architectuurvak omvat een verzameling van principes die betiteld kan worden met 'best practises'. Deze verzameling bevat – afhankelijk van hoe ruim je het begrip 'principe' wenst te hanteren – enkele honderden principes, waar heel veel organisaties uit putten bij het formuleren van hun versie er van. Slechts een beperkt deel van de architectuurprincipes zijn specifiek voor de branche, organisatie of business unit in kwestie. We stellen dan ook:

De set aan architectuurprincipes die een organisatie hanteert bestaat voor 80% uit best practises die niet specifiek voor die organisatie zijn.

De overige 20% zijn branche- of organisatiespecifieke keuzes die de organisatie onderscheiden van concurrenten of die de organisatie haar eigen look & feel geven. De couleur locale zogezegd: bestaande situatie, historie, jargon, cultuur, business rules en visie. Dit zijn de zaken die maken dat de ene bank/retailer/provider/... niet de andere is. We denken niet dat er een haarscherpe grens is tussen de 80% categorie en de 20% categorie, maar er is verschil.

Deze observatie heeft twee belangrijke gevolgen:

1. Die 80% kan in belangrijke mate vóóraf uitgewerkt worden. Sterker nog; die kan je al in de (nog nauwelijks bestaande) architectuuropleidingen stoppen.

2. Het liefst gaat de dialoog die de architect voert met de business over die 20%. Daar kan immers wél direct een link gelegd worden met de doelstellingen! Een architectuur c.q. het werken onder architectuur wordt 'verkocht' op die 20%. Die 80% horen onderwerp van gesprek te zijn binnen de IT organisatie. Het is de vakinhoudelijke kant van de zaak, en die moet gewoon goed geregeld zijn. Maar het is meer het werk onder de motorkap.

We denken dat een architect 80% van de tijd die hij besteedt aan het opstellen van principes, aan die 20% moet besteden. En daarbij is het proces –de dialoog– belangrijker dan de uiteindelijke set van principes. In de dialoog wordt het draagvlak gecreëerd dat je nodig hebt om dóór te zetten als het puntje bij het paaltje komt: in de projecten. Alle escalatieprocedures ten spijt, het uitvechten van compliance issues zijn dan vaak achterhoedegevechten.

Als we zeggen dat architectuur 'vorm geven aan visie' is, dan bedoelen we dus:

1. Zorgen dat we onze tijd zo kunnen verdelen dat we maximaal in dialoog zijn met onze klanten over wat hij wil bereiken, over samenhang en strategische planning.
2. Dat we het instrumentarium (visie op architectuur, een 'repository van principes', plus de tools om er mee om te gaan) beschikbaar hebben om een architectuur neer te zetten die IT best practises en business visie optimaal combineert;
3. Dat we een werkend proces ingericht hebben, van visie naar ontwerp en realisatie.

Dit laatste is vanouds het terrein waar DYA succesvol is. Met deze white paper richten we ons in eerste instantie op het tweede punt, zodat we de handen vrij krijgen voor het eerste punt. De set van principes die daar wordt opgeleverd zijn tenslotte wel de hefboom waarmee we de rest van de keten mee in beweging krijgen.

Voor DYA betekent het dus dat we de visie verder invullen en de procesbenadering aanvullen met een begrippenkader, een praktische manier van werken, en ondersteunende opleiding en tools. Doordat we aansluiten bij DYA, het werken met PSA's, en de Requirements Lifecycle Management methode van Sogeti (RLCM), is de aansluiting met ontwerp en realisatie in projecten gegarandeerd.

DYA en architectuurprincipes

DYA gaat met name in op het werken onder architectuur. Inhoudelijk zegt DYA (in de boeken) weinig over hoe je een architectuur opstelt. Het doet weinig uitspraken over welke modellen gemaakt worden of waar architectuurprincipes aan moeten voldoen. In DYA wordt het richtinggevende karakter van architectuur benadrukt (de prescriptieve benadering), waarbij zowel richtlijnen als modellen gebruikt kunnen worden.

Het DYA architectuurraamwerk onderscheidt algemene principes, beleidslijnen en modellen. Beleidslijnen zijn dan principes die betrekking hebben op een specifiek domein (business, informatie, techniek) of organisatieonderdeel.

Businessdoelen									
	Business-architectuur				Informatie-architectuur		Technische architectuur		
	Omgeving	Prod/dienst	Proces	Organisatie	Gegevens	Applicatie	Middleware	Platform	Netwerk
Algemene principes									
Beleidslijnen									
Modellen									

Architectuurprincipes in het DYA raamwerk

In het vervolg zullen we dit onderscheid niet langer maken en een algemene benadering kiezen, waar in concrete situaties verschillende ordeningen gekozen kunnen worden.

3 HET CONCEPT 'ARCHITECTUURPRINCIPES'

Inleiding

Hoe saai kan je beginnen? Beginnen met definities staat vrijwel zeker garant voor 50% afhakers op de eerste bladzijde. Alleen academici en een enkele 'die hard' komen daar doorheen. Toch is de verleiding groot om te proberen alle verwarring die er is rondom het concept architectuurprincipes weg te nemen door te komen met een definitie. Er zijn tientallen definities en termen in omloop, en je ontkomt er niet aan om er iets over te zeggen.

In deze white paper wordt het begrip architectuurprincipe van alle kanten belicht. Uiteindelijk komen we tot een pragmatische insteek. Architectuurprincipes is (hier) een verzamelterm voor al die smaken. Heeft u een nauwere definitie, dan is wellicht niet alles van toepassing. Heeft u een nog bredere definitie, dan zult u zaken missen. Mijn ervaring is dat je –als je eenmaal in discussie bent– vanzelf naar een begrip toegroeit. De beste leidraad is: wat wil *ik* er mee? Dat is geen theoretische discussie, maar een uiterst praktische. Lees dus met een open geest: er zijn andere interpretaties mogelijk.

Dat laat natuurlijk onverlet dat er onderweg wel een poging gedaan wordt om enige balans te vinden tussen 'rigor' en 'relevance'. M.a.w. we maken ook uitstapjes naar de theorie en actuele inzichten van de wetenschap. De hoofdstukken 3-6 gaan over het conceptuele begrippenkader. Dit legt de basis voor de volgende paper waar we het eindelijk gaan hebben over opstellen en gebruiken van principes.

Leeswijzer

In dit hoofdstuk willen we dieper ingaan op het concept van architectuurprincipes. We geven 'n definitie en een voorbeeld, om vervolgens uitspraken over architectuurprincipes te bespreken. Daarmee krijgen we al wat meer grip op de zaak. Daarna gaan we wat dieper in op de eisen die je kan stellen aan principes of aan een set van principes. We sluiten af met een beschrijving van de vorm waarin we principes moeten gieten om er ook daadwerkelijk mee uit de voeten te kunnen.

Daarmee hebben we dan de basis gelegd om de *samenhang* van principes te bespreken in hoofdstuk 5. Daarvoor is het handig om eens beter te kijken wat voor soort principes er zoal zijn, en welke onderverdelingen gehanteerd worden. Net zoals met definities gaan we daar soepel en pragmatisch mee om.

Om het beeld nog scherper te krijgen zetten we architectuurprincipes in hoofdstuk 6 af tegen een andere vorm van regels: business rules. Daarmee komen we – nog meer dan in voorgaande hoofdstukken in het gebied van meningen en discussie. Het gebied is nog niet uitgekristalliseerd, maar we nemen in ieder geval positie in.

3.1 Doel en definitie

Architectuurprincipes zijn een vorm waarin je architectuurproducten kunt gieten. Daarmee is niet gezegd dat het andere vormen uitsluit. Daarover later meer. Als je architectuurprincipes wilt gebruiken, is het wel zaak dat je goed begrijpt wat het zijn, en dat je een werkwijze hanteert die er bij past. Feitelijk is het andersom: we gebruiken principes graag als vorm omdat het effectief werkt.

Doel

Het doel van werken met architectuurprincipes is daarom onderdeel van het doel van werken onder architectuur. Nu is daar al heel veel over gezegd, dus we volstaan hier met een enkele kernpunten. We zullen zien dat aard van principes ons gaat helpen.

1. Ondersteuning van de Strategische Dialoog. We zien architectuur allereerst als een proces. Een proces van sturing, waarin beslissingen worden genomen over de inrichting van de informatievoorziening (IV) van een organisatie.. Architectuur geeft hierbij overzicht en inzicht in de samenhang Het kan gebruikt worden bij portfolioplanning en impactanalyses.
 - Op *strategisch* niveau: Hoe krijgen we de business strategie haalbaar en maakbaar in de IT? Welke capabilities moeten we ontwikkelen om ook op langere termijn de continuïteit te waarborgen? Wat willen we zelf doen en wat niet? Op welke middelen willen we inzetten?
 - Op *tactisch* niveau: Hoe realiseren we onze strategie in projecten? Hoe hangt het allemaal aan elkaar? Welk product / leverancier kiezen we en op basis waarvan?
 - Op *operationeel* niveau: Op welke processen, informatie en systemen heeft deze verandering impact? Hoe krijgen we de functionele eisen van de business en de huidige technische mogelijkheden bij elkaar? Wat moeten we regelen op de ontkoppelpunten?
2. Kaders stellen voor het Ontwikkelen onder Architectuur in projecten: op basis van de business strategie en IT best practises wordt beleid gedefinieerd dat kaderstellend is voor projecten. Op basis van deze kaders en business requirements kunnen Project Start Architecturen (PSA) gemaakt worden. Op basis van PSA's kunnen afspraken gemaakt worden over ontwerp en oplossingsrichting. Naarmate de architectuur beter is ingevuld zal het ondersteunen van projecten steeds makkelijker worden. En vice versa; naar mate er meer projecten onder architectuur worden uitgevoerd zal de architectuur steeds vollediger worden en beter passen. Projecten kunnen zo versneld starten en oplossingen realiseren die in lijn liggen met de bredere doelstellingen van de organisatie.
3. Leveren van Architectuurdiensten. Architectuur is de basis voor tal van diensten die aan de organisatie geleverd worden. Samenhang is daarbij de essentie. Architectuur is een instrument om in verschillende gremia de samenhang te realiseren én te handhaven tussen processen, systemen en informatie, tussen het IV beleid binnen de organisatieonderdelen - bovenliggend, nevenliggend en onderliggend, en tussen de organisatie en externe partijen.

Definitie

Een definitie geven van architectuurprincipes is een hachelijke zaak. Er zijn minstens zoveel visies op principes als op modellen. Op internet en in literatuur zijn duizenden bronnen te vinden. In organisaties waar principes gebruikt worden krijgt niet zozeer de definitie aandacht als wel het ordeningsprincipe: hoe delen we ze in in hokjes, en welke personen hangen we daar dan aan? En ook daar zien we veel variaties.

De definitie die wij hanteren is een ruime definitie:

Architectuur is vorm geven aan visie. Architectuurprincipes zijn beleidsuitspraken die in de lijn van die visie richting geven aan de inrichting van de informatievoorziening.
--

De term informatievoorziening moet (copafijth) breed opgevat worden. Wat opvalt aan deze definitie is dat ze voorbij gaat aan elk onderscheid dat je kan maken met betrekking tot principes. We nemen niet alles mee – bijvoorbeeld geen morele

principes – maar wel veel. Als organisaties er behoefte aan hebben is het ok. Verderop komen we tot een veel beperktere verzameling van wat *goede principes* zijn.

Voorbeelden

Alvorens de verschillende smaken te onderzoeken is het goed om het begrip principes aan de hand van enkele praktijkvoorbeelden concreter te maken:

1. Elk gegeven heeft één eigenaar.
2. Business analisten zijn in staat om zelf analyses te definiëren en uit te voeren. Tooling en informatievoorziening ondersteunen dit proces.
3. Reuse before Outsource before Buy before Make.
4. Onze organisatie is zó ingericht dat wij onze klanten klantvriendelijk kunnen bedienen.
5. "Maximize Benefit to the Enterprise"
6. Ontwerp van processen, applicaties, services en infrastructuur gebeurt service geöriënteerd.

Je kan niet zo maar een waardeoordeel geven over deze principes. Het hangt heel erg van de context af of ze nuttig zijn voor de organisatie. Toch valt een aantal zaken op:

Ad 1: Dit principe gaat meer over governance dan over informatiearchitectuur. Het geeft uiting aan beleid, dat heel goed te vertalen is naar een requirement in de context van projecten, is helder, kan verantwoord en getoetst worden. Nu is 'elk' gegeven misschien wel veel, maar op het eerste gezicht is dit een prima principe om te hanteren.

Ad 2: Dit is typisch een voorbeeld uit een hele specifieke bedrijfscontext. Een onderdeel waar business analisten zich nog onvoldoende ondersteund voelen door IT in het uitvoeren van BI analyses. Je kan je zo maar voorstellen dat dit principe binnen enkele jaren overbodig wordt (gemeengoed). Het blijft wel waar, maar minder actueel.

Ad 3: Een dergelijk principe – in elke mogelijke variant – zien we veel. Vaak is het uiting van een strategische beslissing die telkens door de praktijk op de proef gesteld wordt. Het is een prima beleidsstatement om in het begin van een traject op de agenda te zetten. De handhaving ervan is een heel andere zaak.

Ad 4: Dit is een uiting van een customer intimacy strategie. Alvorens het praktisch te maken is er nogal wat uitleg nodig. Is die er niet bij gegeven, dan wordt de vertaling naar de realisatie in projecten een exercitie van goede wil van de projectleider. Als stand alone statement is het wel erg hoog over.

Ad 5: Dit is een TOGAF principe en klinkt wellicht als een open deur. Het neemt stelling in de eeuwige discussie tussen 'corporate' en 'business'. Hoewel in de toelichting fijntjes wordt opgemerkt: "No minority group will detract from the benefit of the whole. However, this principle will not preclude any minority group from getting its job done.", is de vertaling naar de praktijk uitermate onduidelijk. Ook hier is de uitwerking cruciaal.

Ad 6: Dit is principe die betrekking heeft op het ontwerpproces zelf. In het huidige tijdsgewricht, waarin SOA steeds belangrijker lijkt te worden, klinkt het als een harde strategische beleidlijn. Waarschijnlijk is het niet de bedoeling dat aan dit principe wordt vastgehouden tot de dood er op volgt, maar het triggert wel een fundamenteel discussie.

Belangrijker dan de principes zelf is natuurlijk de redenering die er aan ten grondslag ligt. Kennelijk was er in de betreffende context een probleem dat aanleiding was om het principe te formuleren.

We hebben nu een voorproefje gehad van het soort overwegingen dat kan spelen. Laten we eens kijken naar typering van principes, zowel uit de theorie als de praktijk.

3.2 Principes in vele smaken

Het begrip principes kent zoals gezegd vele interpretaties, synoniemen en etiketten. Met de onderstaande typering hoeft u het niet eens te zijn. In [3] en vele andere publicaties vindt u andere interpretaties. Het doel is niet om definities te geven, maar om de verschillende smaken voor het voetlicht te brengen. Hier vallen al deze smaken onder het begrip architectuurprincipe.

Architectuurprincipes

Deze term begint langzaam aan de standaard te worden. 'Architecture principles' is ook een term die TOGAF hanteert¹, en ook op Nederlandse universiteiten is de term gangbaar, zij het dat de definities verschillen. In het originele DYA-raamwerk wordt onderscheid gemaakt tussen algemene principes en beleidslijnen. Principes zijn hier bedoeld als algemene regels (*Waarom doen we dingen zo?*), terwijl beleidslijnen (policies) een vertaling van principes zijn. Ze adresseren meer de Hoe-vraag in de context van een specifiek domein als bijvoorbeeld informatiearchitectuur. Standaarden zijn een speciaal soort beleidslijnen: het zijn formeel geaccordeerde keuzes in lijn met de principes, en beantwoorden de Waarmee vraag. Dit is een aanzet tot een hiërarchie van principes. Meer hierover in hoofdstuk 4.

Business principes

Hiermee worden vaak de statements aangeduid die aangeven wat de missie of strategische doelen van de organisatie zijn. En ook wat de relatie met de buitenwereld is. Ook hier raken we in een vaag gebied tussen architectuurprincipes en business drivers en bedrijfsdoelstellingen (snelheid, kosten, innovatie, klantkennis, ...), strategische keuzes, waarover later meer.

Constructieprincipes

Deze term wordt vaak gebruikt voor richtlijnen die betrekking hebben op het ontwerpen zelf. Van Rees gebruikt de term heel nauwkeurig [12]. Hij typeert constructieprincipes als 'dit-moet-je-nooit-zo-doen'-principes en principieel anders dan informatie-architectuurprincipes. Deze hebben het karakter van 'dit-zou-ik-in-dit-geval-niet-zo-doen'-uitspraken, waarbij een andere keuze wel denkbaar is.

Architectuurprincipes betreffen keuzen die door de architect en de opdrachtgever zijn gemaakt.

Wij hebben het nu juist over keuzes die je als architect kan maken.

Constructieprincipes in de zin van van Rees lijken daarmee dus complementair.

Theoretisch kan dit onderscheid er misschien zijn, maar het is niet erg pragmatisch.

Ook van Rees geeft aan dat er redenen kunnen zijn waarom je willens en wetens afwijkt van constructieprincipes. Kennelijk geldt ook hier: zeg nooit nooit. Daarom hanteren wij liever een heel expliciete en positief geformuleerde vorm van beschrijven

¹ TOGAF onderscheidt drie nivo's van principes in een hiërarchische relatie:

1. *Enterprise principes, die de basis vormen voor besluitvorming enterprise breed, en richting geven aan de wijze waarop de enterprise zijn missie vervult.*
2. *IT principes die richting geven aan het ontwerp en gebruik van IT-resources.*
3. *Architectuurprincipes, die aangeven hoe het architectuurproces dient te verlopen en hoe architectuur moet worden geïmplementeerd. Architectuurprincipes worden opgevat als een subset van IT principes.*

van architectuur, zodat men inzicht krijgt in de achterliggende redenering en consequenties.

Inrichtingsprincipes

Waar actualiteit en bedrijfsdoelstellingen met name het WAT bepalen, zijn inrichtingsprincipes beslissend voor het HOE: definities, afbakening, en inrichtingskeuzen. Een algemeen geaccepteerde definitie van de term 'Inrichtingsprincipe' is niet voorhanden. Andere termen als architectuurbeginsel of inrichtingskeuze zijn min of meer synoniem.

Dietz onderscheidt in [3] architectuurprincipes, die richting geven aan de ontwikkeling van het functionele model en principes die richting geven aan het formuleren van ontwerpspecificaties (constructie). In de praktijk is dit onderscheid niet altijd even makkelijk te maken.

Ontwerpprincipes

Ook wel verander- of constructieprincipes genoemd. Dit zijn architectuurprincipes die iets zeggen over de wijze waarop ontworpen en gerealiseerd wordt. Typisch uitspraken over ontwikkelstraat, modelleerconventies en standaarden. Vaak zijn dit lastig te plaatsen principes omdat ze op een metaniveau lijken te staan. Het kan helpen om ze te zien als onderdeel van de procesarchitectuur, en dan in het bijzonder het ontwerpproces. De aandacht voor deze categorie is vaak buitenproportioneel ten opzichte van de principes die direct van een bedrijfsdoel zijn afgeleid. Dat heeft twee oorzaken: 1. Wat je ook ontwikkelt, je krijgt er mee te maken. 2. Enige beroepsdeformatie bij IT-ers zelf. Ontwerp- en constructieprincipes worden immers vooral door architecten, ontwerpers en software engineers gebruikt.

Patterns

In de bouwkunde heeft Christopher Alexander 'Patterns' geïntroduceerd [8][9]. Patterns zijn gangbare patronen. Het begrip is al snel overgenomen in de software engineering. De definitie van patterns uit [10] heeft veel verwantschap met die van principes:

*A **pattern** is the abstraction from a concrete form which keeps recurring in specific non-arbitrary contexts.*

Er is over software patterns al veel geschreven. Een verzameling patterns kan gezien worden als een taal. In [10] wordt de volgende definitie gegeven:

*A **pattern language** is a structured collection of patterns that build on each other to transform needs and constraints into an **architecture***

Hoe architectuurprincipes zoals wij die gebruiken verwant zijn aan patterns, hebben we niet onderzocht, maar dat concepten over elkaar heen liggen is duidelijk. Net zo kan er een relatie zijn tussen pattern languages en een repository van architectuurprincipes. Voer voor onderzoek?

Rolprincipes

Rolprincipes zijn een specifieke vorm van constructieprincipes. Ze worden ook wel werkprincipes genoemd [1] en definiëren wie wat doet. Het aanhangen van een set van Rolprincipes t.a.v. architecten geeft aanleiding tot een bepaalde stijl van architectuur/architecten. Zo is in [1] sprake van een aantal rolprincipes die hoort bij de 'Utrechtse School' van Cap Gemini. Waarschijnlijk zijn deze principes nog te algemeen om aanleiding te geven tot een eigen identiteit. Het vak architectuur is m.i. nog te onvolwassen om te kunnen zeggen wat mainstream is, wat stijl, en wat 'stijlloos vakmanschap' is.

Dit overzicht is aan te vullen met vele andere definities en varianten. Op dit niveau blijven het echter vage omschrijvingen, waar je moeilijk iets van kan vinden zonder enige praktijkervaring. Laten we eens gaan kijken naar de eisen die zoal gesteld worden aan principes. Een reflectie op academische en praktijkvisies.

3.3 Wat zijn goede principes?

Het goede nieuws is dat er al heel veel over geschreven is. Het slechte nieuws is dat het lastig is om zin en onzin van elkaar te scheiden. Op de keper beschouwd is het een stuitende hoeveelheid eisen, die soms ook nog strijdig zijn (volledig en compact). We gaan desalniettemin een poging wagen. Met Cruijff in gedachten (*'Je ziet het pas als je het door hebt'*) hopen we dat de lezer het na één keer lezen doorheeft. Daarna is gezond verstand hopelijk voldoende.

Allereerst is het zinvol om onderscheid te maken in:

1. Eisen aan de semantiek van een principe
2. Eisen aan de syntax van een principe
3. Eisen aan de set van principes die door een organisatie gehanteerd worden

Eisen aan de semantiek van een principe

Relevantie. Het principe maakt een keuze tussen realistische alternatieven.

Zijn deze principes in de hele branch/keten geldig? Onderscheid het ons van concurrenten? Als er geen realistische keuze is, is het principe overbodig.

Relevantie. Het principe stuurt daadwerkelijk op een manier die er toe doet.

Is er een kans dat er tegen gezondigd gaat worden? Is er in het verleden tegen gezondigd? Zo niet, is het principe overbodig.

Relevantie. Het principe doet er toe.

Is overtreden zó ernstig dat het een principe noodzakelijk maakt? Anders zijn er belangrijker zaken. Een architectuur moet overzichtelijk blijven en vooral niet irriteren met futiliteiten.

Richting gevend. Gericht op de toekomst.

Eenduidig geformuleerd.

Verifieerbaarheid. Het moet duidelijk zijn waarom het principe bestaat en wanneer het toepasbaar is.

Samenhang. Een principe is gericht op samenhang. Merk op dat dit iets anders is dan stabiliteit of duurzaamheid. Steeds meer wordt veranderbaarheid/flexibiliteit de norm. Desalniettemin worden principes nooit zomaar geaccepteerd of verworpen. Daar hoort een goed stuk communicatie aan vooraf te gaan.

Uitvoerbaar.

Doelgericht. Principes moeten in hun consequenties in lijn blijven met de bovenliggende bedrijfsdoelen, zoals: kostenbesparing, time to market, klantperceptie, complexiteitsreductie, ...

Handhaafbaar. Kan je naleving van het principe afdwingen? Als architectuur dat niet kan, moet iemand anders het doen. Heeft het principe anders wel zin?

Eisen aan de syntax van een principe

Gebiedende wijs.

Niet: het verdient de voorkeur ..., maar: Het gebeurt zus en zo.

Kort en bondig.

Positief formuleren (hoe wel).

In de toelichting kunnen de beperkingen toegelicht worden.

De negatieve formulering heeft het voordeel dat de ontwerper de maximale vrijheid behoudt. De constructieprincipes geven slechts de randvoorwaarden waarbinnen een oplossing moet worden gezocht, zonder dat -onbedoeld- andere alternatieven worden uitgesloten. De positieve formulering geeft daarentegen beter aan hoe het dan WEL moet. Beter alternatieven komen in de uitvoering vanzelf ter discussie. Negatief geformuleerde principes hebben ook de neiging té compact te zijn. Je moet al een hele vakman zijn om te begrijpen wat de relevantie is in een bepaalde context.

We kunnen gebruik maken van klare taal, of we kunnen gebruik maken van een formele of semi-formele taal.

Eisen aan de taal:

- Zowel geschikt voor specificatie als communicatie.
- Begrippenkader moet in staat zijn om de relatie te kunnen leggen tussen doelen van de organisatie en te maken ontwerpkeuzen
- De taal moet om kunnen gaan met de vaagheid van de menselijke taal (niet dwingen tot formele specificatie). In de praktijk van requirements engineering bestaan tools waarbij eenduidig taalgebruik wordt ondersteund. Wellicht ook geschikt voor architecten?
- De taal moet je redelijkerwijs in staat stellen om architecturen op te stellen binnen de eisen die de organisatie stelt in termen van tijd, geld, kwaliteit.
- De taal moet toepasbaar zijn binnen een methode die aansluit/aansluitbaar is op beleidsvorming/governance/ontwerp/verander processen in veel verschillende organisaties (geen te specifieke eisen doen aan dit soort processen).

Natuurlijke taal voldoet op deze punten prima. Waar natuurlijke taal *misschien* tekort schiet, is de automatiseerbaarheid. Het genereren van principes uit bedrijfsdoelen, van requirements uit principes, van modellen uit principes, e.d.. maar er staat niet voor niets 'misschien'. Het lijkt nu nog een brug te ver om dit te willen. Tenzij je onder de term principes ook 'business rules' vat. Dat doen we vooralsnog niet; meer over business rules in hoofdstuk 6.

Eisen aan de set van principes die door een organisatie gehanteerd worden

Beperkt in omvang.

Het juiste getal hangt natuurlijk af van de behoefte en de volwassenheid van de organisatie af. Onervaren business units hebben aan vijf one-liners al hun handen vol, terwijl de Nederlandse Overheid Referentie Architectuur (NORA 2.0) ruim 150 principes bevat, uitgewerkt in een lijvig (maar voortreffelijk) document.

Het zegt ook iets over je visie op architectuur: hoever ga je als architect, en hoeveel ruimte/leegte laat je over aan het vakmanschap van de ontwerper? Wij geven de voorkeur aan een situationele benadering. Er is niet één recept.

Toegankelijk.

Bij voorkeur op één plaats, in één document(setje) of op de architectuursite op het intranet. Waar iedereen er bij kan, zonder autorisatieproblemen of exotische tools.

Veel gehoorde eisen waar we het NIET mee eens zijn

Toetsbaar. SMART.

Dit is niet haalbaar, niet nodig en ook niet wenselijk. SMART wordt het in het gunstigste geval pas in de requirements fase van een project. En hoe smarter, hoe minder onderhandelingsruimte. Uiteindelijk gaat het om de redenering.

Een architectuur is geen wetboek om elke wetsovertreder op te kunnen aanpakken. De architect is de goede verstaander die de vertaling kan maken in een specifieke context.

De set van principes is volledig.

Volledig is een ontoetsbare eis. Dat betekent natuurlijk niet dat we wel ons best moeten doen. Gezond verstand dus. Het is ook onhaalbaar. Niet voor niets zien we vanuit DYA Just-in-time architecturen als een haalbare en realistische benadering [DYA]. Er zijn ISO standaarden (ISO 15704) waarin volledigheid wordt nagestreefd, maar die bieden een mechanisme zonder te claimen dat het ook kan.

De set van principes is inherent consistent.

Vrijwel alle architecturen bevatten interne tegenstellingen. Snel vs duurzaam; veilig vs gebruikersvriendelijk, etc. Daar is helemaal niets mis mee. Architectuur is bedoeld om besluitvorming te ondersteunen, niet om die onder het tapijt te moffelen. Dit soort spanningsvelden moet de architectuur juist naar voren halen! Architectuur is niet alleen het product van visie, maar ook van onderhandeling en compromis [van der Sanden].

De set van principes die een organisatie hanteert behoort stabiel te zijn.

Onzin. Architectuur hoort te leven! Elk project is een toets of de architectuur nog up-to-date is; principes zijn het begin van de discussie, niet het eind. De architectuur volgt als het goed is de dynamiek van de business, en probeert die enigszins te dempen. Architectuur is nu eenmaal een instrument om verandering te faciliteren, niet blokkeren. Een architectuur voor de eeuwigheid is een gevangenis.

In een set van principes is de onderlinge prioriteit expliciet aangegeven.

Het is duidelijk dat sommige principes belangrijker zijn dan andere. En ook dat principes onderling strijdig kunnen zijn. Principes hoeven dan ook niet gevolgd te worden tot de dood er op volgt. Ze moeten de juiste discussies faciliteren. Ze zijn als het ware een foto van de discussie halfweg (de bestuurlijke helft); hoe ze geoperationaliseerd worden wordt situationeel bepaald. We geloven er dus niet in dat je prioriteiten op voorhand voor alle situaties kunt vastleggen. Bovendien kap je zo elke dialoog af, en dat is nu juist precies een van de uitgangspunten van een Dynamische Architectuur. Het is aan de architect om elke keer weer te bepalen waar hij voor wil vechten, en wat hij als wisselgeld ziet.

Hierna zullen we een vorm bespreken die maximaal aan deze eisen tegemoet komt.

3.4 Vorm

Architectuurprincipes zijn niet nieuw. Wat wel nieuw is, is de wijze waarop we er mee omgaan. Te vaak worden principes als een one-liner gepresenteerd zonder dat er sprake is van enige toelichting of verantwoording. Daarmee is hun waarde beperkt tot een goed idee, maar weet niemand hoe dit te interpreteren. Laat staan dat er sprake is van zicht op de consequenties. Dit soort architecturen belanden als het goed is dan ook in de onderste la. Erger nog is de houding die er uit spreekt: het is de houding van de beleidsmaker die het beleid over de schutting gooit en onbereikbaar is voor dialoog. En vervolgens moppert dat niemand zich er aan houdt.

Kortom, architectuurprincipes hebben behoefte aan wat meer structuur en inhoud. Als wij het hebben over een architectuurprincipe dan duiden we die (net als voorheen) aan met een korte Statement, de one-liner. Maar achter deze krantenkop zit de inhoud waar het om gaat. Veel organisaties die hun enterprise of referentie architectuur voornamelijk in de vorm van principes hebben opgesteld, voegen aan krantenkop-statements nog twee rubrieken toe:

- De Rationale die verwijst naar bovenliggend doel of principe. In de rationale wordt duidelijk welke waarde (welk kwaliteitsaspect) in het geding is. Bijvoorbeeld: efficiency, betrouwbaarheid, veiligheid,...

- De Consequenties. Hierin wordt expliciet gemaakt wat de relevantie is: welke fundamentele of actuele ontwerpbeslissingen worden geraakt, en op welk object heeft het principe impact en hoe.

Voordat een set van principes wordt vastgesteld is er veel discussie nodig. Het is alleen zinvol om principes op te nemen waar iemand van het senior management zich als eigenaar hard voor wil maken in geval van discussie (meestal in de context van projecten). In onze visie hoort bij elk principe of groep van principes dan ook een Eigenaar. De eigenaar is dan degene waar naartoe geëscaleerd wordt, en degene die er samen met de project-opdrachtgever dan wel stuurgroep een knoop over doorhakt. Bij voorkeur dus iemand op minimaal gelijk niveau als de opdrachtgever. Alleen al het duidelijk hebben van de eigenaar helpt enorm. Het is niet meer de architect die omwille van vage IT argumenten iets vindt, maar het is een business discussie. En dat helpt om een goed compromis te vinden. Escaleren is immers voor niemand geen populair traject. Andersom is ook waar: zonder eigenaar is er nauwelijks redenen om de architectuur serieus te nemen. Er is immers geen sanctie. Met andere woorden: architectuur veronderstelt een of andere vorm van governance.

Daarmee komen we op de volgende structuur voor architectuurprincipes:

Statement:	One liner, waarin de essentie van het principe wordt weergegeven. Meestal wordt er een <i>norm</i> gesteld: "Dit doen wij zo".
Rationale:	Uitleg van het waarom. Deze verwijst naar bovenliggende bedrijfsdoelen of principes, en geeft aan welke <i>waarden</i> in het geding zijn. Meer geënt op de business is om hier 'the chain of pain' weer te geven (hoe noncompliance uiteindelijk de businessdoelstellingen ondermijnt). Of voor hen die geloven in de positieve benadering: 'the chain of gain' (hoe compliance de businessdoelstellingen ondersteunt). Evt. ook waarom alternatieve keuzes niet gemaakt zijn.
Consequenties:	Uitleg van de consequenties. Deze verwijst naar de gevolgen die het principe heeft voor ontwerp en inrichting. Voor concrete projecten worden met name de Consequenties vertaald naar projectspecifieke richtlijnen (in een Project Start Architectuur). Het noemen van tegenvoorbeelden (dit mag dus niet), of het verwijzen naar bestaande modellen (afbakening conform de functionele architectuur) of patterns is prima. Het aangeven van de stakeholders is dermate situationeel dat we dat uitstellen tot de fase van PSA. Het is ook lastig te onderhouden.
Eigenaar:	De senior manager die belanghebbende is bij het naleven van het principe. Een Eigenaar moet weten dat hij de eigenaar is! Hij moet begrijpen wat het betekent als hij er Ja tegen zegt, en bereid zijn er zich hard voor te maken.

Een organisatie die begint met het opstellen van architectuurprincipes moet het niet direct doodknuffelen en beginnen met bovenstaande subset. Grote en complexe organisaties kunnen onderstaande rubrieken overwegen. Het is een afweging tussen kwaliteit en werkbaarheid. Wij zien de bovenstaande vier rubrieken als Must have, de onderstaande niet.

Nr.:	Volgnummer. Handig voor referenties. Eventueel per rubriek. Gebruik verder betekenisloze nummers en hergebruik nooit nummer als een principe vervalt.
Architectuur:	Verwijzing naar het soort architectuur: business-, informatie-, technische-, ... Een dergelijke indeling bestaat meestal al. Een praktische indeling is ook naar eigenaar. Maar die moet je dan wel hebben vastgesteld. Default gebruiken we hier de termen uit het DYA raamwerk.
Datum update:	De architectuur leeft. Updates in de vorm van aanscherpingen, nuanceringen, verwijzingen naar toepassingen, etc. zijn voortdurend aan de orde. Beheer per principe heeft zo zijn voordelen (bijv. in een webomgeving).
Handhaving:	Verwijzing naar een proces en/of maat waarmee naleving gevalideerd dan wel gemeten kan worden.
Status:	In de beleidsvormende fase kan een principe nog de concept status hebben. Of later: vervallen
Relatie met:	Een verwijzing naar een gerelateerd principes/business rule, upward, downward, of sideward.
Opmerkingen:	Issues, open issues, afwijkingen, ...
Werkingsgebied:	Indien het principe alleen betrekking heeft op een specifiek domein of omgeving.
Toepassing:	Overzicht/voorbeelden van belangrijke toepassingen – of afwijkingen - in de organisatie. Eventueel een link naar de set van requirements (downward traceability).
Bron:	Indien er een hiërarchie bestaat van architecturen (corporate, BU, ...).
Prioriteit:	(af te raden) Een indicatie of het principe 1. Verplicht, 2. Sterk aanbevolen, of 3. Gesuggereerd is.

Toepassing

De reden waarom we hier zo uitgebreid op ingaan is dat we zo de aansluiting op het requirements management naadloos kunnen maken. In requirements management wordt vergelijkbare informatie van requirements bijgehouden. Belangrijk is bijvoorbeeld de traceerbaarheid. Stroomopwaarts om te traceren waar een requirement vandaan komt. Dat maakt het mogelijk om het belang ervan te achterhalen, en dus de impact van afwijken. En stroomafwaarts om de consequenties van een (nieuw of veranderd) principe te doorgronden.

Nog enkele opmerkingen uit de praktijk:

- Rationale en Consequentie geven samen de onderbouwing van de statement. Voor ondersteunen van beeld- en besluitvorming is het nodig dat actuele discussies hierin herkenbaar terugkomen.
- Het uitwerken van principes is geen doel op zich. Het moet een doel dienen. Uitleg vergroot de acceptatie, verlaagt het risico van ongewenste interpretaties, en kan hergebruik faciliteren, mits het behapbaar blijft. Open deuren worden niet toegelicht - totdat het tegendeel blijkt! Het DYA credo 'Just-enough Just-in-time' blijft van kracht.
- De *waarde* maakt duidelijk welk kwaliteitsattribuut er in het geding is, en het principe statement is in feite de *norm* die we stellen. De praktijk leert dat vrijwel elk principe te schrijven is in de volgende 'normaalvorm':

<Archifact²> voldoet met betrekking tot <Waarde> aan de norm
<Norm>

Dat is voor de communicatie geen handig format, maar voor de analyse wel. Door bij het opstellen van je principes rekenschap te geven van welke waarden je nastreeft in welke archifecten, maak je de vertaling naar een projectcontext veel makkelijker.

- De bovengenoemde vorm helpt enorm bij het maken van een PSA, en bij het verwerken van de feedback vanuit het requirements management proces. In het onderstaande kader is een voorbeeld uitgewerkt.
- Wat je uiteindelijk hebt is veel specifieker maar bekt natuurlijk niet zo lekker. Dat is dan ook de functie van de statement. De klassieke one-liner heeft het grote voordeel dat hij makkelijk hanteerbaar is in discussie. Pas de presentatie van principes aan de doelgroep c.q. stakeholders aan. Details zijn met name voor de specificatie, one-liners voor de communicatie. Voor communicatie en afstemming is een korte alinea in jip en janneke stijl, voorzien van een verhelderend plaatje, beter.

Een principe wordt uitgewerkt in circa 1 A4. Hebben we een principe één keer zo uitgewerkt, hebben we daar bij elk project weer profijt van. Het maken van een PSA kan zo heel snel uitgevoerd worden, als je eenmaal weet wat de bedoeling van het project is. Meer daarover in [\[PSA\]](#).

² De term 'archifact' verwijst in brede zin naar zaken waar we in architecturen iets over willen zeggen.

Voorbeelduitwerking van een architectuurprincipe:

AP-1	Informatie betrekken uit bronsystemen
Statement	Als een informatiesysteem gegevens uit een ander systeem nodig heeft worden deze gegevens betrokken uit het bronsysteem.
Rationale	<ul style="list-style-type: none"> Het bronsysteem is niet per sé het systeem van binnenkomst. Het bronsysteem heet zo omdat het gebruikt wordt als de bron van de waarheid. Het wordt geacht de gewenste kwaliteit te handhaven en het bevat de 'waarheid' in geval van verschillen. <u>Consistentie</u>. Consistentie over de enterprise heen. Wanneer dezelfde gegevens op verschillende plaatsen gemuteerd worden, resulteert dat in vervuiling. Altijd. <u>Kosten</u>. Geen redundantie, dus ook geen kosten die daar mee gepaard gaan. <u>Beheerbaarheid</u>. Als de organisatie dit consequent doet, is er de garantie dat de kwaliteit van de gegevens bekend is, namelijk: net zo goed als die van het bronsysteem. Kwaliteitsmaatregelen hoef je ook maar op één plaats te nemen. <u>Gebruikersvriendelijkheid, voorspelbaarheid</u>. Gegevens uit één bron hebben gemakkelijker één verschijningsvorm, kwaliteit en betekenis, dan gegevens die uit verschillende bronnen geput worden. Gebruikers hoeven er dan niet naar te raden. <u>Testbaarheid</u>. Elke vorm van integraal gegevensbeheer gaat hier van uit. Wil je aantoonbaar in control zijn (wettelijke normen, ISO audits, compliance, beveiligingseisen m.b.t. traceerbaarheid), dan is dit een must-have.
Consequenties	<ul style="list-style-type: none"> <u>Systeem</u>. Welke systemen voor welke (applicatie overschrijdende) gegevens als bron fungeren, moet op attribuut niveau worden vastgesteld op BU niveau of enterprise niveau (projectoverstijgend). <u>Systeem</u>. Om praktische redenen kan een systeem als tussenstation gebruikt worden. Dit is vanuit beheersoptiek ongewenst; er moeten maatregelen getroffen worden zodat de te gebruiken gegevens niet gewijzigd worden (toevoegen mag wel). <u>Gegevens</u>. De bron bevat de waarheid. Er zijn te allen tijde afspraken over welke gegevens leidend zijn. <u>Eigenaar</u>. Het bronsysteem heeft een houder. Afnemers zijn verplicht om evidente fouten aan de houder van het bronsysteem terug te melden. De houder is verplicht om deze meldingen te behandelen. <u>Eigenaar</u>. Voor alle applicatie overschrijdende gegevens wordt een eigenaar benoemd. De eigenaar is degene die beslist over gebruik van het gegeven (de CRUD-matrix). De houder van het bronsysteem is niet per definitie de eigenaar. <u>Eigenaar</u>. Ook als de bron buiten de enterprise ligt, zoals bij artikelnummers, BSN, DigiD, e.a. het geval is, wordt binnen de enterprise een bron en een eigenaar benoemd. <u>Gegevens</u>. Nooit ongekoppelde gegevens waar er centrale bronnen zijn. Ongekoppelde systemen leiden tot eilanden. <u>Informatiehuishouding</u>. Indien er enterprise breed gegevensbeheer is, dient er expliciet beleid te zijn t.a.v. bronsystemen m.b.t. beheer, autorisatie, authenticatie, kwaliteitsmanagement, rapportage, logging, ... Indien toch afgeweken wordt van het principe dienen de volgende zaken geregeld blijven/worden: <ul style="list-style-type: none"> <u>Gegeven</u>. Van elke gegeven (instantie) moet de bron traceerbaar zijn. <u>Proces</u>. Tenminste moet procedureel gewaarborgd worden dat er geen inconsistenties ontstaan. Dit betekent beperkingen in toekomstige aanpassingen. <u>Informatiehuishouding</u>. Het overtredende systeem is geen nieuw bronsysteem, maar voldoet zoveel mogelijk aan de beheernormen die gelden voor bronsystemen. <p>Merk op dat deze aanvullende eisen vaak het effect hebben dat het overtreden minder aantrekkelijk wordt.</p>
Eigenaar	Dit kan de verantwoordelijk informatiemanager zijn, de baas van de informatiemangers, de CIO,...

4 GENETIC MAP

4.1 Normaalvorm

Tot dusver hebben we principes als begrip geïntroduceerd, inclusief doel, definitie en vorm. Als doel hebben we voornamelijk gekeken naar de toepassingen Opstellen beleid, en Kaders voor projecten. De gekozen vorm ondersteunt de architect echter ook bij het uitvoeren van analyses. Daarvoor grijpen we terug naar de eerder geïntroduceerde normaalvorm. De normaalvorm³ ziet er als volgt uit:

<Archifact> voldoet met betrekking tot <Waarde> aan de norm
<Norm>

We hebben al geconstateerd dat de normaalvorm een analyse ding is. In de literatuur worden meer systeemtheoretische raamwerken gedefinieerd, maar deze lijken te abstract om praktisch inzetbaar te zijn.

De normaalvorm is geschikt voor verschillende analyses:

1. Bij het opstellen van *architectuurprincipes*: welke waarden zijn voor de business in het geding? Dit heet ook wel een Quint-analyse. Hiervoor gebruiken we een **checklist** die afgeleid is van ISO 9126 en IEEE 1061. Het helpt ons in het geval van een gegeven business doelstelling om de achterliggende rationale op te sporen.
2. Bij het opstellen van *architectuurprincipes*: Consequentieanalyse. Alvorens een principe voor te stellen is het zaak dat je de consequenties goed doordenkt. Niets is zo vervelend als wanneer je geloofwaardigheid onderuit gehaald wordt doordat je architectuur ondoordacht blijkt te zijn. Hiervoor gebruiken we een **checklist** die geïnspireerd is door Archimate.
3. Bij het opstellen van *architectuur*: Kernwaardenanalyse. Door de kernwaarden van de organisatie te vergelijken met een **repository van principes** (de 80% die we zien als standaardbagage van de architect) kunnen we snel een basisset van principes op het spoor komen die voor de organisatie van belang kunnen zijn.
4. Bij het opstellen van *architectuur*: Architectuur impact analyse. Door te kijken naar welke principes gehanteerd worden, welke waarden ze adresseren, en op welke archifecten ze de meeste impact hebben, kan inzicht krijgen in waar de 'hot spots' zitten. Hiervoor gebruiken we matrices die we een **Genetic Map** noemen.

Deze analyses zullen we hieronder toelichten. Merk op dat we ons zowel op het niveau van individuele principes begeven, als op het niveau van de hele set van principes – op architectuurniveau dus. Merk tevens op dat we hier het terrein van theorie en concepten deels verlaten, en het terrein van tools en praktische toepassing betreden.

Voordat we aan de analyses toekomen moeten we eerst de te gebruiken tools toelichten.

4.2 Tools

Checklist waarden

³ Deze normaalvorm is één op één is terug te voeren op kwaliteitsmanagement [36]. Je kan architectuur ook zien als kwaliteitsmanagement van de informatievoorziening.

Kwaliteitsattributen of waarden worden bepaald door de stakeholder. A la 'Beauty is in the eye of the beholder' is het geen objectieve aangelegenheid. Of een systeem snel genoeg is, of 0,01% uitval acceptabel is, het is afhankelijk van de norm die gesteld wordt.

Checklist waarden

<u>Betrouwbaarheid</u> R1. Volwassenheid R2. Fouttolerantie R3. Handhaafbaarheid R4. Fraudebestendigheid R5. Herstelbaarheid R6. Beschikbaarheid (continuïteit) R7. Consistentie R8. Voorspelbaarheid (Risicobeheersbaarheid)	<u>Functionaliteit</u> F1. Passendheid F2. Performance, productiviteit F3. Accuratesse F4. Interoperabiliteit F5. Diversiteit F6. Compliantie F7. Veiligheid (Toegankelijkheid, Vertrouwelijkheid, Integriteit, Onweerlegbaarheid) F8. Traceerbaarheid F9. Capaciteit	<u>Bruikbaarheid</u> U1. Begrijpbaarheid U2. Leerbaarheid U3. Uitvoerbaarheid U4. Explicietheid (Juistheid, volledigheid, nonredundant, eenduidigheid) U5. Aanpasbaarheid, flexibiliteit (maatwerk, modulariteit) U6. Aantrekkelijkheid (onderscheidend, herkenbaarheid) U7. Behulpzaamheid U8. Gebruikersvriendelijkheid (look & feel, relevantie, control, navigatie, personaliseerbaarheid, lokale ondersteuning (tijden, eenheden))
<u>Onderhoudbaarheid</u> M1. Analyseerbaarheid M2. Veranderbaarheid M3. Schaalbaarheid M4. Stabiliteit M5. Testbaarheid M6. Beheer(s)baarheid M7. Herbruikbaarheid	<u>Portabiliteit</u> P1. Adaptievermogen, uitbreidbaarheid P2. Installeerbaarheid P3. Compatibiliteit, conformance P4. Vervangbaarheid, afbouwbaarheid	
<u>Efficiency</u> E1. Tijdsbeslag E2. Resourcebeslag E3. Kosten (initieel, operationeel, transitie) E4. Kennisbeslag	<u>Samengesteld</u> C1. Standaardisatie (beschikbaarheid, consistentie, interoperabiliteit, diversiteit, begrijpbaarheid, leerbaarheid, eenduidigheid, stabiliteit, testbaarheid, beheer(s)baarheid, herbruikbaarheid, adaptievermogen) C2. Openheid (veiligheid, aanpasbaarheid, analyseerbaarheid, testbaarheid, adaptievermogen) C3. Modulair (herstelbaarheid, interoperabiliteit, veiligheid, begrijpbaarheid, uitvoerbaarheid, aanpasbaarheid, tijdsbeslag, resourcebeslag, kosten, veranderbaarheid, schaalbaarheid, stabiliteit, testbaarheid, beheer(s)baarheid, herbruikbaarheid, adaptievermogen, vervangbaarheid) C4. Operational Excellence (prijs, proces) (continuïteit, performance, kosten, schaalbaarheid) C5. Product Leadership (product, innovatie) (diversiteit, veiligheid, aanpasbaarheid, gebruikersvriendelijkheid, tijdsbeslag, kennisbeslag, adaptievermogen, vervangbaarheid) C6. Customer Intimacy (service) (beschikbaarheid, consistentie, passendheid, interoperabiliteit, diversiteit, aantrekkelijkheid) C7. Complexiteitsreductie (consistentie, compliantie, begrijpbaarheid, uitvoerbaarheid, explicietheid, tijdsbeslag, kennisbeslag, beheer(s)baarheid) C8. Time-to-market (tijdsbeslag, kosten, resourcebeslag, kennisbeslag, veranderbaarheid, stabiliteit, testbaarheid, herbruikbaarheid, adaptievermogen) C9. Innovatief/bewezen (diversiteit, kennisbeslag, schaalbaarheid, stabiliteit, vervangbaarheid) C10. (De-)centraal (beschikbaarheid, continuïteit, consistentie, schaal, accuratesse, interoperabiliteit, veiligheid, aanpasbaarheid, kosten, beheer(s)baarheid, herbruikbaarheid)	
<u>Business Effectiviteit</u> B1. Doelgerichtheid B2. Doeltreffendheid B3. Ondernemerschap (aliertheid, omgevings-sensitiviteit, durf) B4. Winstgevendheid B5. Wendbaarheid		

ISO-9126 bevat een lijst van kwaliteitskenmerken die gesteld kunnen worden aan informatiesystemen. Deze standaard is door een samenwerkingsverband van diverse bedrijven onder de naam Quint uitgebreid [5][6]. Voor architectuur zijn meer zaken van belang dan alleen systemen. Daarom hebben we de lijst uitgebreid tot iets dat werkbaar is gebleken in de praktijk. Bepaalde waarden blijken opgebouwd te zijn uit onderliggende waarden. Omdat ze vaak voorkomen hebben we een beperkte set van deze samengestelde waarden toegevoegd. We zien het als een 'under construction' lijst, hoewel de groei er al wel zowat uit lijkt. Het is geen wetenschappelijk verantwoorde tool; het wérkt en daar gaat het ons om. Zie het kader.

Het gebruik van een standaardlijst van waarden is nuttig omdat:

- het helpt om precies te zijn;
- het helpt om redundantie en strijdige eisen op te sporen;
- het helpt om principes en consequenties te ordenen;
- het leereffect het uitvoeren van analyses steeds gemakkelijker maakt.

Dezelfde argumenten zijn ook van toepassing op een standaardlijst van archifacten:

Checklist archifacten

Waar de Waarden aangeven wat een organisatie belangrijk vindt (bijvoorbeeld 'wendbaarheid') geven de archifacten aan waar een architectuurprincipe betrekking op heeft: wendbaar betekent dat onze processen ..., wendbaar betekent dat ons productportfolio.... Wendbaar betekent dat onze applicaties

Met andere woorden: de archifacten zijn de zaken waar de architectuur iets over zegt. Voor de kenners onder u: Het is handig om bij archifact te denken aan de Archimate-concepten (service, actor, applicatie, ..). Voor de niet-kenners is er ook een lijst in meer gangbare termen. Ook deze lijst is permanent 'under construction'.

NB: De kracht van de Archimate concepten is dat ze abstract zijn. Dat is ook hun zwakte. In communicatie met niet-architecten gebruiken we bij voorkeur business termen en voorbeelden.

Repository van principes

In [hoofdstuk 3](#) hebben we de observatie gedaan dat 80% van de architectuurprincipes die een organisatie hanteert bestaat uit best practises die niet specifiek voor die organisatie zijn. Helaas is een overzicht van die principes nog niet aangetroffen. Binnen Sogeti hebben we een aanvang gemaakt met een dergelijke [repository](#). Als input daarvoor hebben we uitvoerig bronnenonderzoek gedaan, waarbij we ons gericht hebben op principes die daadwerkelijk door organisaties vastgesteld zijn. Op dit moment bevat de repository circa 280 principes, deels uitgewerkt in de boven beschreven vorm.

Archifact checklist – Archimate termen

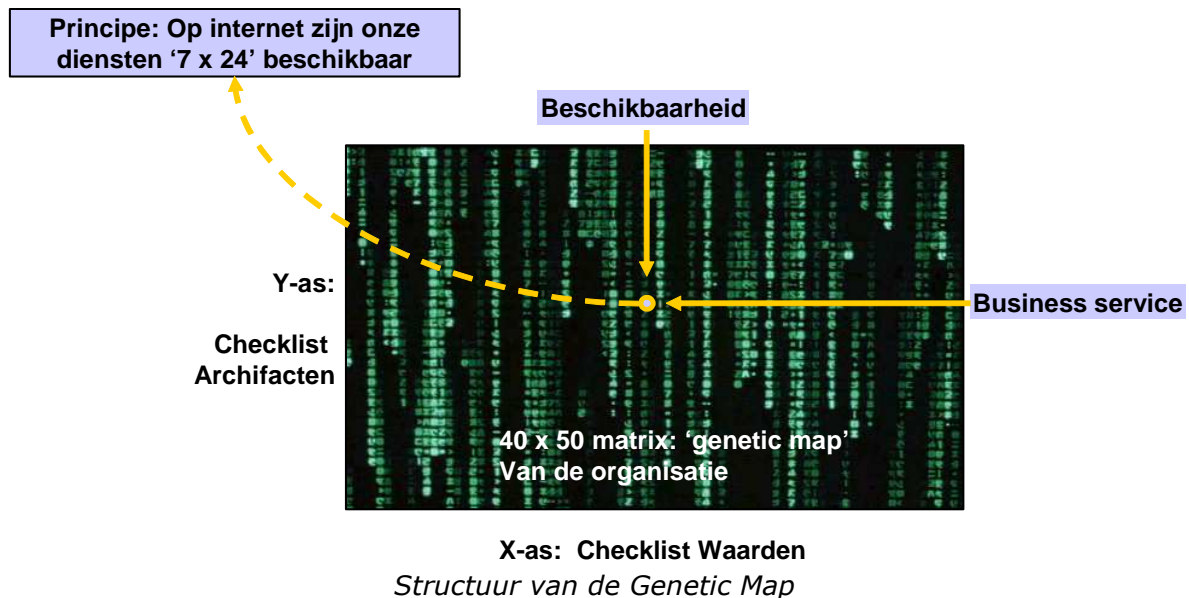
Behaviour	Information	Structure	Composed
Meaning	Data Object	Artifact	Generiek
Value	Application Collaboration	Node	Presentatielaag
Business Product	Application Component	Device	Proceslaag
Business Object	Application Interface	Network	Functielaag
Representation	Application Function	System software (applicatie, OS, ...)	Gegevenslaag
Business Actor	Application Interaction	Communication Path	Software
Business Role	Application Service	Infrastructure Interface	
Business Collaboration		Infrastructure Service	
Business Interface			
Business Process			
Business Function			
Business Interaction			
Business Service			
Business Event			

Archifact checklist – gewone taal

Omgeving	Organisatie	Goederen	Techniek & Infra
Wet- en regelgeving	Organisatie(onderdeel), Rol	Goederen (grondstof, halffabrikaat, product)	Netwerk, telefonie
Keten	Bedrijfsfunctie	Goederenstroom	Systeem, OS, hardware, middleware
Stakeholder	Business event	Goederen voorraad	Applicatie, code
Klant, Markt	Medewerker	Informatie	Component
Ketenpartner, Leverancier, Rol	Taken, verantwoordelijkheden, bevoegdheden	Gegevens, standaarden	(Gebruikers)Interface
Trends	Competentie	Informatiehuishouding	Projecten
Domein, sourcing	Eigenaar	Object	Project, programma,
Producten/doelen	Processen	Entiteit, attribuut	Migratie, plateaus
Bedrijfsdoel	Kanaal	Document, formulier	Risico
Formule, merken, componenten	Functie	Financien	Change event
Portfolio, Product, Dienst	Proces(keten, stap, ontkoppelpunt)	Geld, waarde, ROI, prijs	
	Service	Geldstroom	
	Activiteit		
	Locatie, tijd		

Genetic Map

Heb je de principes van een organisatie in beeld, dan kan je deze visualiseren in een matrix door horizontaal de Waarden uit de waardenchecklist te zetten, en verticaal de Archifacten uit de checklist. Door de cellen van de matrix te markeren waar er een principe een norm definieert voor een waarde m.b.t. het bijbehorende archifact, krijg je een soort van genenmap van de organisatie: de '**Genetic Map**':



Net als een DNA fingerprint zal de Genetic Map voor iedere organisatie uniek zijn (stelling). Het zegt iets over wat de organisatie belangrijk vindt (vision), en waar ze toe in staat is (ability to execute).

Nu is het maken van deze vorm van de Genetic Map 1. veel werk (40x50 matrix), en 2. met name geschikt voor een niet alledaagse analyse van de organisatie an sich (bijvoorbeeld in een benchmarkvergelijking met andere organisaties). Daarom zal in de praktijk het maken van eenvoudiger versies van de Genetic Map de voorkeur hebben.

4.3 Analyses

We werken drie analyses verder uit:

1 Quint analyse

Voor een Quint analyse kunnen we volstaan met een matrix waar we horizontaal de Waarden uitzetten tegen verticaal de principes. De cellen markeren we als het principe een bepaalde waarde raakt. In het uitgewerkte voorbeeld 'Informatie betrekken uit bronsystemen' zien we in de de Rationale o.a. de waarden consistentie en beheerbaarheid terug.

Hoe generieker de bedrijfsdoelstelling, hoe hoger in de doel-middel hiërarchie, en hoe meer waarden uiteindelijk geraakt worden. Merk op dat bij elke verzamelterm een hele reeks waarden geraakt *kunnen* worden, maar dat dat in een gegeven klantsituatie niet hoeft. Bovendien is het wel of niet raken aan een bepaalde waarde geen kwestie van wit of zwart, maar een met grijswaarden. Voorbeeld: Raakt het principe "Alle informatie wordt aan de bron gedigitaliseerd" aan resp. traceerbaarheid, veiligheid, of gebruikersvriendelijkheid? Er is wel een relatie, maar die is niet per sé de

basis voor het principe. **In de rationale bij een principe moet duidelijk worden welke waarden de kern zijn.**

Nog een voorbeeld:

We zijn met de business in gesprek over de architectuur waarvoor we principes willen opstellen. In het strategisch plan lezen we dat het voor de organisatie essentieel is dat ze sneller dan de concurrent deployment in andere landen moet kunnen doen, zo gauw als wetgeving dat toestaat. Wat betekent dat voor de architectuur? We stellen vast dat vereiste capabilities o.a. zijn: het kunnen kopiëren van operations naar andere landen, het aanpassen van producten aan lokale factoren en kennis hebben van internationale markten. Hierbij horen typisch waarden als: modulariteit (producten), standaardisatie (koppelvlakken, werkprocessen) en schaalbaarheid (processen, besturing, systemen). Het ligt nu voor de hand om te onderzoeken of de architectuur op deze punten aanvulling vereist. Dat vraagt nog steeds om een analyse van de bestaande situatie, maar nu zoeken we wel gericht!

2 Consequentieanalyse

Voor een Consequentieanalyse hoeven we ook geen Genetic Map te maken. We kunnen volstaan met een matrix waar we horizontaal de Archifecten uitzetten tegen verticaal de principes. In de voorbeelduitwerking 'Informatie betrekken uit bronsystemen' zien we bij de consequenties de archifecten systeem, gegeven en rol (eigenaar) terug. **In de consequenties moet duidelijk worden welke archifecten in het geding zijn.**

Voorbeelden:

Voor wie is het principe "Onze processen zijn papierloos" relevant?

Het archifact is 'Proces' of 'Alle interne administratieve processen'. De achterliggende waarde is 'efficiency' en de bijbehorende norm is: 'vrij van papieren dragers en opslag'.

De actoren die iets met het principe moeten zijn dus: procesontwerpers die papierloze processen moeten ontwerpen; IT architecten en ontwerpers die zorgen dat de onderliggende IT en IV er komt; product/dienst- en proces-eigenaren die opdracht (budget, prioriteit) moeten geven om te migreren naar papierloze processen; bovenliggend kader die de kaders moeten creëren om dit mogelijk te maken; en natuurlijk mensen in de uitvoering die moeten signaleren waar papier knelpunten veroorzaakt.

Is het (TOGAF) principe "Wij opereren binnen de wet" geen open deur?

Misschien wel. Het betekent dus wel dat wij over de hele linie compliance nastreven. Dat is nogal wat, want er zijn Veel wetten: wet op privacy, ARBO wetgeving, milieu, mededinging, etc. De achterliggende waarde is 'compliance' en de bijbehorende norm is: 'conform de wet'.

Archifecten zijn dus: Processen, beleid (wetten moeten vertaald worden); Actoren: medewerkers (worden geacht de wet te kennen), procesontwerpers, kwaliteitsmedewerkers (toezien op), managers (sturen op); Informatie (over wetten). Het nut van deze open deur kan zijn dat we ons tijdens veranderingen bewust zijn waar we al of niet tijdelijk afwijken.

Het analyseren van archifecten waar een principe impact op heeft brengt je op het spoor van actoren die er iets mee moeten bij verandertrajecten:

de ontwerpers van de geadresseerde en onderliggende archifecten; de mensen die er in de uitvoering mee van doen hebben, hun managers die er op moeten sturen, en de beleidsmakers die de kaders moeten zetten. Deze laatste groep zijn feitelijk de ontwerpers van de organisatie en business strategie, ergo: de top van het bedrijf.

3 Kernwaardenanalyse

Als er nog geen architectuur is kan je op basis van een kernwaardenanalyse snel met een startset komen, zonder direct heel veel tijd te besteden om het bedrijf van binnen en buiten te leren kennen.

Als er al wel een architectuur is kan je die ook prima gebruiken om de kernwaarden uit te destilleren. Niet ieder bedrijf heeft die tenslotte helder voor ogen, en als dat wel zo is wil dat nog niet zeggen dat de architectuur daar mee in lijn is. Door een Waarden-analyse te doen op de bestaande principes zie je direct wat belangrijk gevonden wordt. Een simpele toets of de architectuur (nog) in lijn is met de strategie van de organisatie.

Analyse	Volwassenheid	Fouttolerantie	Handhaafbaarheid	Fraudebestendigheid	Herstelbaarheid	Beschikbaarheid	Afbouwbaarheid	Passendheid	Performance	Schaal	Accuratesse	Interoperabiliteit	Diversiteit
	Betrouwbaarheid						Functionaliteit						
	6	1	5	1	0	18	1	12	4	1	7	20	
De bewijskracht, die aan													
Nieuwe informatietechnologie is een													
Alle uitvoeringsprocessen zijn													
De kritieke succesfactoren, de													
Historie van managementrapportages													
Business Object wordt gepositioneerd													
Er wordt een interface tussen XXX en													
De wet- en regelgeving, normen en													
Alle informatie over marktpartijen en													
Voorwaarde voor de invoering van													
I&A richt een specifieke digitale													
Nieuw in te zetten													
De gezamenlijke voorziening voor de													
Deze specifieke implementaties													
Deze specifieke implementaties m													

Uitwerking van een Waarden-analyse. De rode cijfers geven aan dat de waarden Beschikbaarheid en Interoperabiliteit toonaangevend zijn bij de inrichting van bedrijf en informatievoorziening.

5 ORDENING VAN ARCHITECTUURPRINCIPES

Een van de éérste vragen die gesteld worden bij het opstellen van architectuurprincipes is: "Hoe houd ik het overzicht?" Geen gekke vraag als je bedenkt dat het bieden van overzicht een van de redenen is waarom we aan überhaupt architectuur doen. Ook geen gekke vraag als je bedenkt dat nogal wat bedrijven zich bedolven voelen door alle modellen en ontwerpen die links en rechts door de organisatie zwerven. Dat willen we ons met principes niet nog een keer laten overkomen.

Structuur dus. Om een heel lang verhaal kort te maken: **er is niet één beste manier van organiseren die voor alle organisaties werkt.** Daarvoor zijn de verschillen te groot. Voor een specifieke organisatie is het wel goed om één systematiek te handhaven, maar welke, dat is van minder belang. Vanuit het perspectief van de externe architect is het zaak om daarbij aan te sluiten. DYA biedt structuur en een werkwijze, maar past zich aan aan de klant.

Het belang van één ordeningsprincipe

Een belang hebben we al genoemd: het bieden van overzicht. Omdat architectuurprincipes een breed gebeid bestrijken, is voor een doorsnee niet-architect slechts een heel klein deel relevant. Het helpt dan als je kan verwijzen naar het corporate deel, de informatie architectuurprincipes of de de richtlijnen die vallen onder zijn of haar bevoegdheid.

Daarmee raken we al direct een teer punt. Je kan (wilt) het maar op één manier organiseren. Voor de rest moet je andere hulpmiddelen verzinnen. In de praktijk willen nog wel eens twee manieren gecombineerd worden, maar daarna houdt het toch snel op. NORA bijvoorbeeld gebruikt een combinatie van de drie architectuurdomeinen <Business – Informatie – Technisch> en een dimensie met de aspecten <Wie – Wat – Hoe> wat resulteert in 9 gebieden. Het advies luidt: **keep it simple.**

Om relaties te ontdekken of een relevante set te selecteren kan je natuurlijk attributen toevoegen, zodat je makkelijk kan sorteren. Daar kleven echter direct nadelen aan als: onderhoud (doen architecten niet spontaan), ambiguïteit (niet uniek toe te wijzen) of toegankelijkheid (tooling, expertkennis). Het advies luidt wederom: **keep it simple.**

Er is ook een goede reden waarom we *niet* al te veel waarde moeten hechten aan een taxonomie of ordeningsprincipe: het werkt gewoon niet. Er zijn teveel principes die in meerdere categorieën vallen, of waar de rationale van links komt, terwijl de consequenties rechts liggen. Kiezen we, dan zoeken we natuurlijk op de verkeerde plek. Slaan we het twee keer op, genereren we redundantie en dus beheerproblemen.

Desalniettemin vraagt de menselijke maat om ordening. Keep it simple.

In de praktijk gebruikte ordeningsprincipes

We geven 10 verschillende ordeningen. Door hierin te variëren en combineren kan vrijwel elke organisatie er een weg in vinden.

Doel-middel hiërarchie

Een voor de hand liggende manier is om een doel-middel-hiërarchie aan te brengen met aan de top het missie statement en de business drivers, om dan vervolgens via strategische doelen af te dalen naar een decompositie van principes. Ervaring bij meerdere grote organisaties is evenwel dat dit niet leidt tot een uniforme gelaagdheid, en ook geen nette boomstructuur, maar wel een bepaalde oorzakelijkheid. Door onderscheid te maken tussen Business driver, consequentie en architectuurprincipe (lastig) kan wel enige ordening aangebracht worden maar consequent uitmodellieren blijkt lastig. Dat hoeft ook geen probleem te zijn: de manier van denken is belangrijker dan een perfecte modellering.

Strategisch – Tactisch – Operationeel

Deze onderverdeling lijkt intuïtief heel duidelijk, maar valt toch tegen omdat een hard criterium ontbreekt. Aanverwante verdelingen als <Waarom – Wat – Hoe – Doen> of <Doel – Beleid – IT Beleid – Constructie> hebben hetzelfde probleem.

Beschouwingsniveau's

Het woord architectuur dringt de metafoor op met de fysieke wereld, waar we begrippen tegenkomen als Ruimtelijke ordening, Stadsplan, Wijkplan, Gebouwarctitectuur en Binnenhuisarchitectuur. Die kan je bijvoorbeeld afbeelden op: Ketenarchitectuur, Enterprise Architectuur, Domeinarchitectuur, Informatiesysteemarchitectuur en Gegevensarchitectuur. Het sterke van deze benadering is dat het duidelijk is het architecturen op verschillende niveaus gaat. Het nadeel is ook hier weer dat er teveel discussie is over waar een principe thuishoort.

Abstractieniveau

Een variatie op de bovenstaande is om principes een formele status te geven, los van de inhoud: doelen, uitgangspunten, regels, richtlijnen, standaarden, etc. Je lost het probleem niet echt op, maar je hebt een aantal vakjes gecreëerd.

Organisatie hiërarchie

In grote organisaties worden architectuurprincipes op meerdere niveaus vastgesteld. Bijvoorbeeld: op corporate niveau, op divisieniveau, en op business unit niveau. Met een beetje (veel) fantasie kan je NORA zien als het hoogste niveau van architectuur voor de overheid. Het idee is dat de op het hoogste niveau de referentie-architectuur gedefinieerd wordt, en dat principes gelden voor alle onderliggende niveaus, die daar hun eigen principes aan toevoegen. Welke in geval van conflict prevaleert is dan een interessante discussie. Het goede van deze indeling is dat het makkelijk te combineren valt met een andere indeling, terwijl impliciet de samenhang in beeld blijft.

Generiek vs Specifiek: 80-20

Een indeling die logisch volgt uit de constatering dat veel organisaties dezelfde principes hanteren, is om dat onderscheid verder door te voeren. Bedrijfsspecifieke principes vs generieke principes.

Voorbeelden van specifieke ontwerpprincipes kunnen zijn:

- Belastingdienst: vertragingen in de geldstroom zijn niet toegestaan. Dit principe komt voort uit een bedrijfsdoelstelling, en leidt tot de keuze om te werken met automatische incasso en voorbedrukte OLA's.
- Dagblad: informatie wordt medium onafhankelijk opgeslagen. Leidt tot digitalisering, multichannel-architectuur, en op voorzieningen die het tijdskritisch editie-proces kunnen ondersteunen.
- Energiebedrijf: Portfolio handel en Speculatieve handel houden gescheiden boekhoudingen. Leidt tot aparte staf, en auditeerbare loggingfaciliteiten in de dealingroom.

Het is evenwel niet altijd even duidelijk wat generiek is en wat specifiek.

Branchspecifieke principes creëren alweer een grijs gebied. Bovendien zijn de de

groepen zo groot of divers dat een aanvullende onderverdeling gewenst is. Het onderscheid volgt wel automatisch uit de wijze waarop de principes tot stand komen, zoals we eerder beschreven.

Indeling gekoppeld aan een of ander referentie framework (Zachman e.a.)

Als de architectuur een specifieke methode of raamwerk volgt is dit per definitie de indeling. Nadelen als overal.

Indelen naar eigenaar

Niet veel organisaties beleggen elk principe bij een eigenaar. Maar er zit wel enige logica in om ze naar eigenaar in te delen. Zo is het voor iedereen direct duidelijk wie waar over gaat. Dat voorkomt wellicht inhoudelijke discussies waar een politieke of business beslissing aan de orde is. Nadeel is dat de organisatie sneller verandert dan je lief is.

Indeling gekoppeld aan IEEE1471 views

Ordering van principes naar stakeholder (IEEE1471). Helaas is ook dit niet eenduidig.

Architectuurdomeinen

Grote organisaties kennen binnen de architectuurdiscipline specialismen: enterprise architecten, informatie architecten, netwerk architecten, etc. Voorbeeld: het UWV heeft de principes destijds ingedeeld in: <Dienst – Proces – Organisatie en management – Gegevens – Informatievoorziening – Applicatie – Technische Infrastructuur>.

Dit is min of meer conform het DYA architectuurraamwerk. Veel organisaties gebruiken een variant van deze indeling. De repository van principes is daarom ingedeeld.

6 ARCHITECTUURPRINCIPES VERSUS BUSINESS RULES

De relatie tussen architectuur en business rules is onderwerp van veel discussie. De oorzaak zit voor een deel in het feit dat de onderwerpen vanuit verschillende richtingen aangevlogen worden. Architecten hebben veelal een top-down benadering en denken vraaggericht. Software engineers hebben veelal een bottom up benadering en denken meer oplossingsgericht. Aangemoedigd door leveranciers van tooling en consultancy komen deze partijen elkaar tegen in complexe projecten, beurzen en andere publieke arena's. **Verwarring alom.**

In dit hoofdstuk introduceren we kort het begrip Business rules. De trends op dit gebied en de relatie met principes maken duidelijk dat architecten iets moeten met business rules. Wat precies is nog niet helemaal duidelijk, maar we geven wel een aanzet.

"Business Rules⁴" is een rijk vakgebied, getuige de meer dan 1 miljoen hits in Google. Er zijn verschillende gremia in Nederland die zich met business rules bezig houden. Twee toonaangevende groepen zijn:

- BRPN: Business Rules Platform Nederland (<http://www.brplatform.org/>)
- NAF werkgroep Business Rules (http://www.naf.nl/nl/werkgroepen/business_rules.html).

Op het LAC wordt al enkele jaren een track verzorgd over de relatie tussen architectuur en business rules.

6.1 Inleiding business rules

Business rules zijn bedrijfsregels. Ze beschrijven de spelregels die het bedrijf hanteert bij de uitvoering van processen. Die spelregels kunnen gelden voor software, maar ook voor medewerkers. Feitelijk zeggen de regels niets over *hoe* ze uitgevoerd worden. Een prima beschrijving van wat business rules is het Business Rules Manifest (www.businessrulesgroup.org/brmanifesto/BRManifestNederlands.pdf).

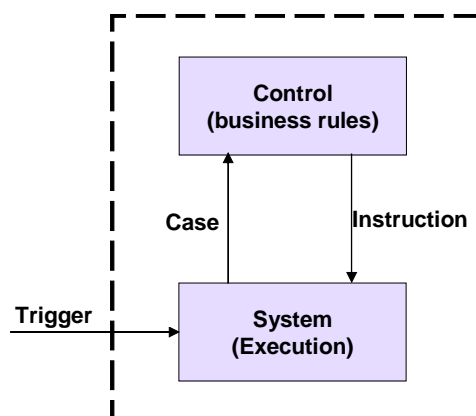
Voorbeelden van regels zijn:

- Bij een kredietaanvraag wordt altijd een BKR-toets gedaan.
- Electronische aangiftes zonder DigiD worden niet in behandeling genomen.
- Indien een order voldoet aan tenminste één van de volgende voorwaarden, neemt afdeling X de behandeling verder over: ...

Business rules gaan over het nemen van beslissingen. Beslissingen met betrekking tot de uitvoering van taken, of ten aanzien van routing. Een regel die zegt hoe een klant ingevoerd moet worden is ook een regel, maar niet een business rule in de zin waar wij het over hebben, en waarbij ter plekke een beslissing genomen wordt (hier zit wel een grijs gebied). Business rules hebben dus iets te maken met het begrip Control.

Hoewel het begrip soms wordt opgerekt, zijn business rules in de praktijk voornamelijk onderwerp van gesprek waar het automatisering betreft. Preciezer: waar het gaat over de vraag hoe de business rules gecodeerd worden in software. We hebben het dan over omgevingen waar complexe beslisbomen bestaan, zoals bijvoorbeeld bij het toekennen van rechten aan uitkeringsgerechtigden, het vaststellen van premies, of AI systemen.

⁴ Business Rules zijn bedrijfsregels. We gebruiken de gangbare Engelse term.



Figuur 1: Control vs Execution (stippellijn geeft Droste effect aan)

De basisgedachte is dat business rules niet hardcoded verstopt moeten worden in de software, maar liever apart gezet worden in een business rules engine (BRE) waar de business owner er in kan wijzigen zonder te hoeven programmeren. Dit heeft voordelen zoals:

- Beheerbaar kunnen houden van een complexe set van bedrijfsregels;
- Eenvoudiger doorrekenen van consequenties van wijzigingen (simulaties);
- Sneller wijzigingen kunnen doorvoeren (hogere productiviteit);
- Overzicht en inzicht door expliciete documentatie en visualisatie;
- Minder risico's, minder fouten; ;
- Traceerbaarheid van genomen beslissingen.

Nadelen zijn er ook:

- Het voegt een nieuw niveau van complexiteit toe aan de ontwikkelstraat;
- Vereist een volwassen software development omgeving;
- Alleen rendabel toepasbaar in zeer complexe en grootschalige businessomgevingen;

Of het ook goedkoper is, is maar de vraag. Het belangrijkste motief is meestal niet kostenbesparing maar flexibiliteit en/of kwaliteitsbeheersing.

6.2 Business rules: Trends

De aandacht voor business rules zal de komende tijd toenemen. De volgende trends zijn zichtbaar:

1. Meer en betere tooling voor business rules management (BRM). Business rules worden opgeslagen in een rules repository. De gebruiker krijgt grafische-, tabel- en (pseudo-natuurlijke) taal-interfaces ter ondersteuning van functies als: rules invoeren, structureren, analyseren, en testen. Meestal wordt code gegenereerd. Ook leveranciers uit de BPM- en architectuurtooling (ARIS) maken opgang naast gespecialiseerde BRM vendors (Fair Isaac, Ilog, Corticon, ...).
2. Ook de Big4 leveranciers ondersteunen de business rules approach. Rules worden uit de code gehaald. De komende jaren mag vanuit deze hoek een inhaalslag verwacht worden (overnames). Microsoft heeft voor de Vista omgeving al een Windows Workflow Foundation Rules Engine.

3. Standaardisatie. Er bestaat een hele stack aan standaarden. Wellicht nog een beetje veel, maar het schijnt dat een aantal standaarden boven komen drijven. Zie o.a. de presentaties van het BRPN .
4. Hoewel toolleveranciers business users als gebruikers van hun tools positioneren, zal dit beperkt blijven tot enkele expert users. In de meeste gevallen zal er ondersteuning vanuit de IT department nodig blijven. In het beste geval zal er een redactie-model ontstaan zoals dat ook in de web-wereld bekend is.

6.3 Business rules, principes en requirements

Het Business rules manifest zegt: "Regels zijn expliciete beperkingen op gedragingen en/of geven richting aan gedrag". Dat klinkt al heel erg hetzelfde als architectuurprincipes. Toch zijn het heel verschillende concepten, hoewel niet iedereen het daar over eens is. Stef Joosten stelt dat principes en business rules hetzelfde zijn, dat je rules kunt implementeren en dat architectuur daarmee overbodig wordt. Dat is wel erg kort door de bocht.

Architectuur gaat over samenhang en overzicht, over besturen, over communicatie, over algemene principes, over richten en inrichten. Business rules gaan over operationele beslissingen, over sturen, over specifieke processen, over verrichten. Architectuur heeft de vorm van principes en modellen; business rules hebben de vorm van volzinnen – al of niet in pseudo-code - en beslisstructuren.

Is er dan geen enkel verband behalve dat het gaat over directieve volzinnen? Jawel, beiden zijn bedoeld als instrument voor governance en communicatie. In beide gevallen is het proces lastiger dan de inhoud. En **beide zijn input voor requirements, waarbij architectuur meestal nonfunctional requirements dicteert, en business rules meestal functional requirements**. Als zodanig zijn beiden dan ook bedoeld om het ontwikkelproces sneller en beter te maken. Er valt dus wel iets te zeggen voor het standpunt dat het in kaart brengen van de business rules een verantwoordelijkheid is van een business architect. Maar dit is dan wel een heel andere invulling dan gebruikelijk. Het lijkt meer op het vlak van de business of informatie analist te liggen.

In [BR and Requirements management] worden business rules afgezet tegen requirements. **Business rules zijn criteria voor business beslissingen, ze dicteren business gedrag. Requirements specificeren wat een systeem moet kunnen, ze dicteren systeem gedrag.**

Het centrale uitgangspunt bij het vormgeven van architectuur in principes is de observatie dat architectuurprincipes voor een groot deel onafhankelijk zijn van de organisatie. 80% van de principes van een bedrijf komen uit een pool van ruim 300 architectuurprincipes of zijn een variatie daarop. Als dat zo is, wat maakt dan het onderscheid in informatievoorziening tussen twee bedrijven? Dat zijn niet alleen in die 20% bedrijfsspecifieke principes. Die zijn misschien wel het meest fundamenteel, maar dat kan natuurlijk niet alles zijn. Het verschil zit met name in de geaccumuleerde kennis (structural, human en relational capital), en die is neergeslagen in al of niet expliciete business rules.

Vanuit dit perspectief zijn architectuurprincipes en business rules complementair! Daarmee is niet gezegd dat je als architect niets met business rules te maken hebt. Maar wel dat je in de regel in de architectuur geen business rules tegen zult komen. En vice versa.

De relatie tussen business rules en beleidsregels gaat vaak via een hiërarchie van afleidingen. De grens is fuzzy. Om alleen de onderste laag business rules te noemen is wellicht te beperkt, want dan verlies je de samenhang. Als er op businessniveau iets verandert wil je de relatie kunnen leggen met alle rules die dan mee moeten veranderen. Neem bijvoorbeeld de 'life events' bij e-overheid. Daar probeert men een

overzicht te krijgen wat er allemaal moet gebeuren over overheidsinstellingen heen wanneer iemand geboren wordt, trouwt, een huis koopt, etc. En vice versa wil je ook opwaartse traceerbaarheid hebben.

6.4 Business rules en architectuur

Hoppenbrouwer en Coenen geven aan dat de Business Rules Approach (BRA) iets toevoegt aan het repertoire van de architect:

- Business rules helpen in projecten bij het specificeren van requirements;
- Business rules en principes spreken de taal van de business; het licht daarom voor de hand om 'het aftekenen door de business' op een zelfde manier in te steken.
- De architect moet aangeven wanneer het gebruik van een BRE aan de orde is (dit kan een principe zijn);
- De architect moet aangeven welke inrichtingskeuzes daarbij gemaakt worden;
- Business logic wordt een aparte laag in de applicatie architectuur (of service component in een SOA);
- De architect moet inzicht hebben in de 'fit' tussen competenties en ontwikkelstraat.

Er zijn nog enkele andere consequenties van BRA voor de architectuur:

- Architectuur beschrijft een systeem (in de systeemtheorie betekenis). Architectuur moet iets zeggen over het control-mechanisme. Met een dergelijk mechanisme kan je makkelijker een architectuur ontwerpen. In veel modellen ontbreekt eigenlijk een concept voor besturing. Hoe dit precies vorm te geven is niet zo maar te zeggen, maar het klinkt aannemelijk dat business rules daar in plek in dienen te hebben.
- Het invoeren van software tools in de architectuur-/ontwerp-/ontwikkelstraat blijkt steeds weer een heel lastig en omvangrijk project te zijn, omdat het diepgaand ingrijpt in de werkwijzen van mensen, in meerdere schakels van de keten. Dit geldt voor het invoeren van BRA nog meer dan voor de invoering van architectuurtooling of requirements management tooling. Het is maar de vraag of het mogelijk is om al deze tools in te voeren. Of: om ze níet allemaal in te voeren.
- De grens tussen architectureren, analyseren en ontwerpen was al vaag, maar wordt nog vager. Het onderscheid is a priori niet te maken. De architectuur-/ontwerp-/ontwikkelstraat verandert, zowel in competenties als in organisatie en tooling.
- Introduceren van business rules heeft impact op:
 - Businessarchitectuur. Zoals gezegd, als aan het repertoire van omgeving, organisatie, product en proces ook de business rules worden toegevoegd is er sprake van een nieuwe taak.
 - Procesarchitectuur. Hoewel de procesarchitectuur niet hoeft te veranderen (maar wel kán veranderen als je de control centraliseert), verandert wel het architectureren;
 - Informatiearchitectuur. Regels worden onderdeel van de informatie architectuur (i.c.m. proces en entiteit);
 - Applicatie architectuur. De BRE wordt een extra component. Rolverdeling tussen BRE, applicatie en procesflow ('stappen met beslissingen' vs 'beslissingen leiden tot stappen').

- Infrastructuur architectuur. De BRE moet praten met meerdere applicaties, en moet passen in de middleware of service architectuur;
- Software architectuur. Business rules ontvlechten uit de code. Vorm geven van de beslisboom. Implementeren van de BRE. Granulariteitsvraagstukken. Onderscheiden van rules en algoritmes.

Ergo: Invoeren van de BRA raakt vrijwel elke architect in de organisatie!

De vraag is echter in hoeverre het opstellen, vertalen, implementeren of beheren van business rules een verantwoordelijkheid van een architect is. Dat het relevant is voor de architectuur is duidelijk. Business rules toevoegen aan het repertoire van de architect impliceert wel een herdefinitie van de discipline, en doet de grens met analyse en ontwerp nog verder vervagen. Gezien de complexiteit van zowel architectuur als business rules is het maar de vraag of beide aandachtsgebieden te 'behappen' zijn door één persoon (de architect). Het devies luidt derhalve: Samenwerken in de architectuur-ontwerp keten.

Naarmate er meer ervaring komt met business rules, zal ook duidelijker worden hoe architecten hier mee om moeten gaan, en welke kansen en uitdagingen er zijn. In de context van architectuurprincipes is het goed om je steeds de vraag te stellen: is dit een principe of een business rule?

6.5 Business Rules en DYA

In de DYA boeken kwam de term business rules tot dusver niet voor. Naar mate business rules belangrijker worden, kunnen we ze op verschillende plaatsen in het DYA model tegenkomen:

Governance: Business rules zijn – net als architectuur – een asset, mits je ze ook als zodanig behandelt. Ergo, er dient ook enige sturing en bewaking op plaats te vinden. Omschrijving van het Wat en Hoe, op basis van best practises, is een mogelijke uitwerking.

Strategische dialoog: Indien BRA in een strategisch kader geplaatst wordt, heeft het impact op het projectenportfolio. Vanuit Architectuur Services kunnen kaders aangereikt worden vanuit de verschillende architecturen, en ook voor het managen van het proces.

Ontwikkelen onder architectuur: DYA adviseert het gebruik van architectuurprincipes, en een proces waarin deze via een PSA (Project Start Architectuur) worden vertaald naar projectspecifieke requirements (RLcM, Requirements Lifecycle Management). Het gebruik van business rules zou ten opzichte van dit proces een plaats moeten krijgen.

Architectuur Services: Indien de architectuurfunctie een rol neemt / toebedeeld krijgt in BRA, wordt het gebruik ervan onderdeel van de Architectuur Services.

Dynamische architectuur: Binnen DYA worden verschillende architecturen uitgewerkt in best practises (van business architectuur tot infrastructuurarchitectuur). Zoals gezegd heeft de BRA impact op deze architecturen. Best practises in deze kunnen in het DYA gedachtengoed opgenomen worden, indien de markt er om vraagt.

7 BIJLAGE: BRONNEN

- [1] «Architectuur, besturingsinstrument voor adaptieve organisaties » ; Rijsenbrij, Schekkerman, Hendrickx ; ISBN 90 5931 281 3, 2^e druk, 2004.
- [2] Potts, Colin and Glenn Bruns, Recording the reasons for design decisions, in Proceedings of the 10th international conference on software engineering, Singapore, Thailand, 11-1 April 1988, IEEE Computer Society Press, Washington DC, pp. 418-427, 1988
- [3] Jan Dietz, http://www.naf.nl/content/bestanden/xaf-1.1_fe.pdf.
- [4] ISO 15704 - Requirements for enterprise-reference architectures and methodologies* : 2000 ?
- [5] http://www.cibit.nl/site.nsf/page/ict_expertise_architectuur_kwaliteitseisen.
- [6] <http://www.se.l/index.html?/resources/publicaties/qnt-boek.shtml>,
- [7] http://www.the-art.nl/072_kwaliteit/1 On the syntax and semantics of architectural principles.pdf
- [8] Christopher Alexander: "The Timeless Way of Building", Oxford University Press, 1979.
- [9] Christopher Alexander: "A Pattern Language: Towns, Buildings, Construction", Oxford University Press, 1977.
- [10] Brad Appleton; "Patterns and Software: Essential Concepts and Terminology"; <http://www.sis.pitt.edu/~spring/patterns/patterns.html>
- [11] <http://www.opengroup.org/architecture/togaf7-doc/arch/p4/princ/princ.htm>
- [12] Jaap van Rees; diverse artikelen op <http://www.jaapvanrees.nl/>
- [13] http://www.dya.info/Images/Bijdrage_Serge_Bouwens_Archi_tekst_%20januari_2006_tcm13-32013.pdf