

# Dimensies in architectuur

Danny Greefhorst en Henk Koning

*In de afgelopen jaren zijn er een aantal architectuurraamwerken gedefinieerd, zoals Zachman, IFW en 4+1, met ieder zijn eigen dimensies en terminologie. Een beginnend IT-architect vraagt zich af wat hij heeft aan al deze raamwerken. Een ervaren IT-architect heeft waarschijnlijk zijn eigen gewoonten ontwikkeld voor het beschrijven van architecturen, maar vraagt zich af of hij geen voordeel zou hebben van onderdelen van recent ontwikkelde raamwerken. Wij kunnen die vragen ook niet geheel beantwoorden maar willen wel een kleine bijdrage leveren. Het doel van dit artikel is om te komen tot een lijst van raamwerkafhankelijke dimensies die een rol spelen bij architectuur. Deze lijst vereenvoudigt communicatie over architectuurbeschrijvingen, maar kan ook gebruikt worden als een checklist bij een architectuurontwerp. Ter illustratie zal eerst worden ingegaan op bestaande architectuurraamwerken.*

IT-architectuur kan worden beschreven vanuit verschillende gezichtspunten. Dit wordt onder andere tot uitdrukking gebracht in de IEEE/ANSI P1471 standaard [1] voor beschrijving van architectuur. IEEE 1471 spreekt over ‘views’ die volgens een voorgedefinieerd ‘viewpoint’ gemaakt zijn. Alleen schrijft IEEE 1471 niet voor welke viewpoints er zijn en welke gebruikt moeten worden. En de architecten zijn het er onderling ook niet over eens welke viewpoints gehanteerd moeten worden! In het verleden zijn er wel initiatieven geweest om standaard gezichtspunten te definiëren in de vorm van architectuurraamwerken. Denk bijvoorbeeld aan het Zachman framework voor informatiesystemen [1], de RM-ODP standaard voor gedistribueerde systemen [2] en het 4+1 model voor software architectuur [3]. Als je op zoek bent naar een architectuurraamwerk voor je project, dan loop je tegen het probleem aan dat al deze raamwerken een andere afbakening kennen. De viewpoints zijn, soms behoorlijk, verschillend! Kennelijk zijn deze raamwerken in heel verschillende omstandigheden ontstaan en passen ze niet bij alle situaties. Het gevolg is dat veel architectuur-ontwerpprojecten op zoek gaat naar een goede indeling. Een ander gevolg is dat bestaande architectuurrapporten slecht toegankelijk zijn voor lezers die niet vertrouwd zijn met de gevolgde indeling. Ook is de informatie uit gelijksoortige rapporten moeilijk onderling te relateren. Problemen genoeg dus. De vraag die wij ons stellen is ‘hoe kunnen we wat meer houvast krijgen op al deze verschillende denkrichtingen?’

Wij hadden zelf namelijk behoefte om meer houvast te krijgen op de beschikbare raamwerken en hebben ze daarom eens op een rijtje gezet, wat geleid heeft tot dit artikel. Wij hebben de inleidende informatie van diverse raamwerken bekeken en de gebruikte dimensies geïnventariseerd en daaruit een basisset afgeleid. Voor alle duidelijkheid: wij spreken dus niet vanuit diepgaande kennis van elk raamwerk afzonderlijk. Voor een vergelijkende studie leek ons dat ook niet nodig. Wat ons, behalve de behoefte aan houvast, verder nog drijft is een drang om de ‘blijvende waarden’ te extraheren uit deze vastgelegde deskundigheid van anderen. Anders gezegd, om onze eigen mentale gereedschapskist uit te breiden met concepten die nuttig kunnen zijn voor het begrijpen van en communiceren over architecturen. Merk op dat het niet onze bedoeling is een kwaliteitsoordeel over de architectuurraamwerken te geven. We willen de raamwerken alleen gebruiken om onze ideeën te illustreren.

Een groot probleem bij het schrijven van een artikel zoals dit, is dat sommige woorden zoveel betekenissen hebben. Woorden als “type”, “informatie”, “perspectief” en “gezichtspunt” heb je nodig als je beschouwend ergens over wil praten (meta-niveau). Als je ook nog wilt uitleggen hoe je te werk bent gegaan, praat je al gauw op meta-meta-niveau. Anders gezegd, over het perspectief waarmee je gekeken hebt naar perspectieven op perspectieven. Anders gezegd, welke benadering je hebt gekozen voor het kijken naar gezichtspunten op invalshoeken. In de verschillende raamwerken worden ook veel dezelfde woorden gebruikt, met steeds een eigen betekenis. Kortom, ga er maar aan staan. In dit artikel hebben wij in elk geval geprobeerd zoveel mogelijk alledaags Nederlands te spreken (voor zover wij dat nog kunnen). In de opgestelde lijst van dimensies proberen wij zoveel mogelijk algemeen gangbare termen te gebruiken.

In de volgende paragraaf geven wij korte beschrijvingen van enkele bekende raamwerken. Voor wie ze nog niet kent is dit informatief en het geeft een beeld van wat wij bedoelen met “dimensies”. We presenteren een samengevatte tabel met de dimensies van raamwerken waarover wij informatie konden vinden. In de

daaropvolgende paragraaf vergelijken we de raamwerken onderling. Uit de overeenkomsten en verschillen leiden we een lijst van dimensies af. De laatste paragraaf laat zien hoe deze lijst van raamwerk-onafhankelijke dimensies gebruikt kan worden om een bestaande architectuurbeschrijving te positioneren. We sluiten dit artikel af met een korte conclusie, een wens voor de toekomst en een oproep voor reacties.

## Architectuurraamwerken

Architectuurraamwerken zijn verzamelingen van dimensies, waarbij een dimensie een mogelijke classificatie van architectuurmodellen is. Architectuurraamwerken zijn onder te verdelen in twee categorieën: enterpriseraamwerken en applicatieraamwerken. Enterpriseraamwerken doen architectuuruitspraken op niveau van een divisie, heel bedrijf of zelfs een afdeling. Ze omvatten vaak een grote verzameling documenten. Applicatieraamwerken beschrijven de architectuur van een specifieke applicatie of van een groep overeenkomstige applicaties, en bestaan in de regel uit één document, of een kleine verzameling documenten. Voorbeelden van enterpriseraamwerken zijn het Zachman Framework for Information Systems Architecture (ISA) [4, 5], het Information Framework (IFW) [6], The Open Group Architecture Framework (TOGAF) [7], Integrated Architecture Framework (IAF) [8] en Methodology for Architecture Description (MAD) [9]. Bekende applicatieraamwerken zijn het 4+1 model [3], het raamwerk van Siemens [9, 10], en het 2+2 model van de VU [12]. Hierna volgt een korte beschrijving van de Zachman, IFW, 4+1 en Siemens raamwerken.

### *Zachman*

De basis voor enterprise-raamwerken is gelegd door John Zachman in zijn artikel uit 1987 [4] waarin hij een raamwerk voor de architectuur van informatiesystemen beschrijft. Zijn idee was dat architectuur voor informatiesystemen zou kunnen worden geïnspireerd door architectuur in meer volwassen engineering disciplines. Hij zag dat de architectuurmodellen in deze engineering disciplines veel overeenkomsten vertonen en zouden kunnen worden samengevoegd tot een algemeen model. Zachman onderkende daarin twee dimensies: de betrokken partijen en de soort informatie.

Betrokken partijen zijn: de planner, de eigenaar, de ontwerper, de bouwer en de onderaannemer van een informatiesysteem. Later krijgen deze perspectieven logische namen en worden het de contextuele, conceptuele, logische, fysieke en uit-context perspectieven. Het uit-context geeft aan dat op dit niveau vaak onderdelen (op bestelling) gefabriceerd worden buiten het zicht op het grotere geheel waarin ze gebruikt zullen gaan worden.

De soort-informatie-dimensie is ontstaan door de observatie van Zachman dat steeds de elementaire vragen 'wat, hoe, waar, wie, wanneer en waarom' beantwoord worden. Voor informatiesystemen wordt dat vertaald naar 'gegevens, functies, lokaties, mensen, tijd en motivatie'. De observatie was verder dat beide dimensies onafhankelijk van elkaar zouden kunnen variëren, zodat er uiteindelijk  $5 \times 6 = 30$  verschillende soorten architectuurmodellen zouden kunnen worden gemaakt om te komen tot een informatiesysteem. In het raamwerk worden deze modellen weergegeven in een tabel met kolommen voor de soort-informatie en rijen voor de betrokken partijen.

### *IFW*

Veel enterpriseraamwerken zijn een uitbreiding of aanpassing op het raamwerk van Zachman. Een voorbeeld hiervan is IFW, wat staat voor "Information FrameWork". IFW heeft zich begin 90-er jaren ontwikkeld binnen de tak van IBM die zich richt op banken en verzekeringsmaatschappijen. Het is geïnspireerd door Zachman, maar heeft een eigen koers gekozen. IFW heeft twee hoofddimensies (zie de kolommen en rijen in Figuur 1).

Types of Information Levels of Constraint		Organisation View			Business View				Technical View		
		Strategy	Structure	Skills	Data	Function	Workflow	Solution	Interface	Network	Platform
Deconstruction Level	Domain Concept										
	Domain Classification										
Composition Level	Generic Template										
	Design Context										
Implementation Level	Operational Bound										

**Figuur 1 IFW – Information FrameWork**

Wat opvalt aan IFW in vergelijking met Zachman is dat de kolommen nu namen hebben die sterk IT-gericht zijn. Veel sterker dan de algemene woorden ‘wat, hoe, waar, wie, wanneer en waarom’. De kolommen zijn verdeeld in drie groepen: organisatie, business en techniek. De groep organisatie heeft de kolommen strategie, structuur en bekwaamheden, de min of meer vaste bestanddelen van de organisatie. De groep business heeft de kolommen gegevens, functies, werkstroom en oplossingen. Hier zie je de huidige activiteiten van de organisatie. Oplossingen zijn de producten en diensten die de organisatie naar buiten levert. De groep techniek heeft de kolommen interfaces, netwerken en platforms. Onder ‘interfaces’ valt de structuur van de applicaties. De applicaties zelf zitten in de onderste rij van de kolommen van de business groep. De verschillende kolommen vormen samen de dimensie “type informatie”.

De naamgeving van de rijen is wat cryptisch. We lopen ze even langs van boven naar beneden. Domeinconcepten zijn de basisconcepten die gemeenschappelijk zijn voor alle bedrijven in een bepaalde sector, de financiële sector bijvoorbeeld. Domeinclassificaties geeft aan welke plaats de organisatie inneemt binnen de domeinconcepten. Generieke sjablonen zijn de basis voor meerdere ontwerpen binnen de organisatie. De ontwerpcontext verankert een systeem binnen de organisatie, maar laat nog keuzen open. Operationeelgebonden geeft de feitelijke objecten aan. In IFW beperkt een rij de keuzevrijheid van de rij eronder. De waarden van de verschillende rijen vormen samen de dimensie “mate van keuzevrijheid” (‘levels of constraint’).

Behalve de twee hoofdimensies onderkent IFW nog vier dimensies, die we kort even noemen. De derde dimensie wordt gevormd door de inhoud van cellen in de tabel. Immers elke cel kan veelvoudige informatie bevatten. De vierde dimensie is de transformatie in de tijd. De vijfde is het eigenaarschap van de verschillende componenten in de architectuurbeschrijving. De zesde en laatste dimensie is de methodologieketen. Met methodologieketen worden de relaties aangeduid die onderkend worden tussen componenten in cellen in de tabel en de processen om deze cellen goed te vullen.

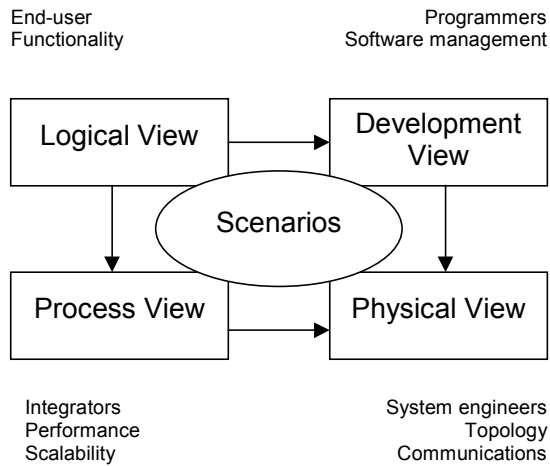
IFW is evenals andere raamwerken niet formeel en eenduidig beschreven en laat keuzevrijheid over aan de architect.

### *Evernden Eight*

Een van de belangrijkste auteurs van IFW was Roger Evernden. Hij past het werken met dimensies tegenwoordig toe in een methode voor “informatie-architectuur”, de Evernden Eight [13]. Informatie-architectuur is daarbij gericht op het gegevensaspect, de structuur en toegang tot gegevens, wat o.a. de basis is voor het opzetten van websites. De in Evernden Eight geïdentificeerde dimensies zijn “type informatie”, “begripsniveau”, “type representatie”, “niveau van transitie”, “type kennis”, “niveau’s van verantwoordelijkheid”, “type proces” en “meta niveau’s”. We zullen hier niet verder ingaan op deze dimensies, aangezien zij zijn meegenomen in de dimensies die verderop in het artikel uitgebreid uiteen worden gezet.

## 4+1

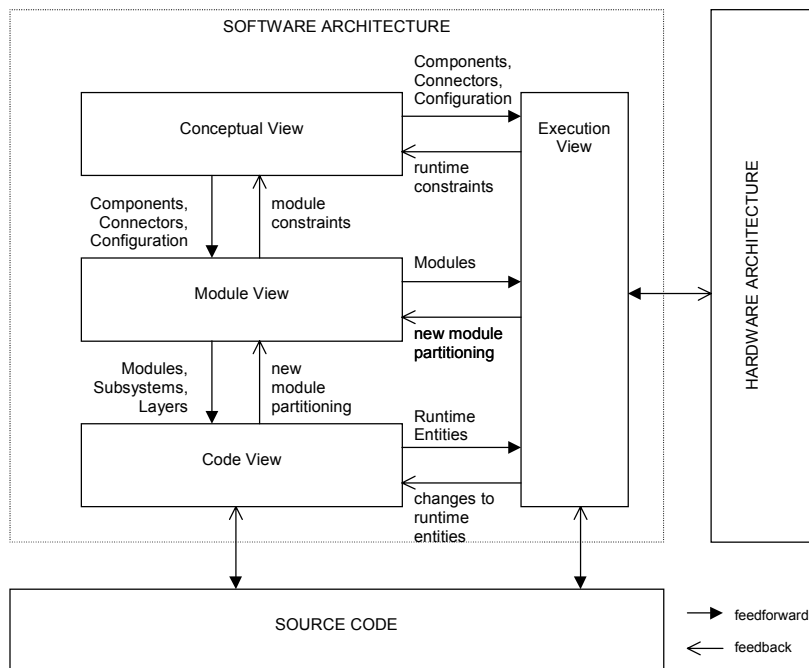
Een bekend applicatieraamwerk is het 4+1 model [3] voor software (zie Figuur 2).



**Figuur 2 Het 4+1 model**

In tegenstelling tot de enterprise-raamwerken heeft dit raamwerk maar één hoofddimensie, die niet benoemd wordt. De gezichtspunten zijn evenals het Zachman framework onderverdeeld naar betrokkenen en hun belangen. Er zijn vier gezichtspunten, namelijk een logisch-, ontwikkel-, proces- en fysiek gezichtspunt. Deze gezichtspunten hebben een herkenbare relatie naar respectievelijk gebruikers (klassen), ontwikkelaars (packages en bestanden), integrators (processen, berichten) en systeem engineers (nodes en netwerken). Het vijfde gezichtspunt bevat scenarios die beschrijven hoe de eenheden in de andere gezichtspunten met elkaar samenwerken. Het 4+1 model is initieel gepubliceerd door Philippe Kruchten en vormt de basis voor de producten van de firma Rational.

## Siemens



**Figuur 3 Siemens raamwerk**

Het applicatieraamwerk van Siemens is erg vergelijkbaar met het 4+1 model, maar hanteert vier gezichtspunten die lichtelijk afwijken van die in het 4+1 model. Zo onderkennen zij een conceptueel model dat vergelijkbaar is met het logische gezichtspunt, splitsen zij het ontwikkelgezichtspunt op in een module en een code gezichtspunt en voegen zij de proces en fysieke gezichtspunten samen tot een executie gezichtspunt.

## Overzicht

Wij willen de informatie over bestaande raamwerken graag afsluiten met een samengevat overzicht van raamwerken en anderssoortige architectuurindelingen waarover wij informatie hebben gevonden (zie Tabel 1). Bij elk raamwerk vermelden wij de bron, de dimensies en de waarden in de dimensies. De dimensies zijn weergegeven als rijen bij een raamwerk. Een uitsplitsing van waarden in een dimensie wordt tussen haakjes weergegeven. De tabel geeft u een indruk welke dimensies er allemaal een rol spelen, en misschien wekt het uw interesse voor een bepaald raamwerk.

Raamwerk	Bron	Dimensies
2+2 model	[12]	Context, Technical Infrastructure, Conceptual, Development
4+1 model	[3]	Logical, Process, Development, Physical, Scenarios
ARIS	[18]	Organizational, Data, Control, Function, Product/Service
Boar	[15]	Infrastructure, Data, Applications, Organization
		Inventory, Principles, Models, Standards
DYA	[14]	Business (Product, Proces, Organisatie), Informatie (Gegevens, Applicatie), Technisch (Middleware, Platform, Netwerk)
		Algemene Principes, Beleidslijnen, Modellen
Herzum/Sims	[16]	Functional, Application, Technical, Project Management
IAF	[8]	Business, Information, Information Systems, Technology Infrastructure
		Contextual, Conceptual, Logical, Physical, Transformational
		Business and ICT System, Security, Governance
ISA	[4, 5]	Data, Function, Network, People, Time, Motivation
		Contextual, Conceptual, Logical, Physical, Implementation, Out-of-Context
IFW	[6]	Organization (Strategy, Structure, Skills), Business (Data, Function, Workflow, Solution), Technical (Interface, Network, Platform)
		Deconstruction (Domain Concept, Domain Classification), Composition (Generic Template, Design Context), Implementation (Operational Bound)
MAD	[9]	Inter-organizational, Organizational, Process, Information, Application, Distribution, Configuration
Maier/Rechtin	[17]	Data, Behavior, Form, Purpose, Performance, Managerial
MArch	[21]	Product, Proces, Organisatie, Informatievoorziening, Infrastructuur
		Context, Concept, Logisch
		Nu, minder dan 2 jaar, 2 tot 5 jaar
RM-ODP	[2]	Enterprise, Informational, Computational, Engineering, Technology
Siemens	[10, 11]	Conceptual, Module, Execution, Code

Tapscot	[19]	Business, Work, Information, Application, Technology
TOGAF	[7]	Business (People, Process, Function, Business Information, Usability, Performance), Engineering (Security, Software Engineering, Data, System Engineering, Communications Engineering), Operations (Security, Software, Data, Computing/Hardware, Communications), Acquirers (Building Blocks Cost, Standards)

**Tabel 1 Bestaande architectuurraamwerken**

## Dimensies

Wij willen nu proberen om het scala aan indelingen te ordenen en samen te vatten in één lijst met raamwerk-onafhankelijke dimensies. Dat doen we door eerst de bekeken architectuurraamwerken onderling te vergelijken. Daarna stellen we een nieuwe lijst voor van raamwerkonafhankelijke dimensies. De betekenissen van de dimensies zullen we kort omschrijven en enkele manieren aangeven om de lijst van dimensies te gebruiken.

### *Kijk en vergelijk*

In deze paragraaf gebruiken wij het woord ‘aspect’ in de algemene betekenis van ‘zijde, kant van waaruit iets beschouwd wordt’, dat valt hier en daar samen met een waarde die een dimensie kan hebben in onze tabellen.

Kijken we nog even terug naar Tabel 1 dan vallen een aantal zaken op:

- Er worden veel verschillende termen voor de aspecten gebruikt en gelijke termen hebben vaak een andere betekenis. Zo vallen er onder de term informatie-architectuur in IAF, MAD en DYA niet dezelfde aspecten.
- De aspecten zijn vaak informeel gedefinieerd waardoor de afbakening tussen aspecten onduidelijk is.
- De aspecten zijn vaak van een net iets andere grootte, waardoor een aspect in het ene raamwerk wordt opgesplitst in meerdere aspecten in het andere raamwerk. Zo wordt in tegenstelling tot andere raamwerken in IFW geen expliciet onderscheid gemaakt tussen middleware en platformen.
- In IAF wordt er een duidelijk onderscheid gemaakt tussen bedrijfsgegevens en IT-gegevens, waarbij de laatste de fysieke representatie van de eerste is. Het onderscheid tussen bedrijfsaspecten en IT-aspecten is algemener en is ons inziens ook op andere aspecten toepasbaar, zoals bijvoorbeeld ‘functies’. De bedrijfsaspecten beschrijven dan de situatie zoals deze in de business bestaat onafhankelijk van hun IT-implementatie. Een IT-aspect is een aspect dat gerealiseerd wordt door een applicatie.
- Een aantal raamwerken definiëren aspecten rond kwaliteitseigenschappen zoals performance of security. Ons inziens zijn deze kwaliteitseigenschappen orthogonaal aan andere aspecten en verdienen zij een eigen dimensie.

Het verschil in terminologie en mate van uitsplitsing blijkt als je probeert om enkele raamwerken naast elkaar te zetten (zie Tabel 2). In deze vergelijking hebben wij IFW als uitgangspunt genomen. Wat in IFW “Strategy” heet, is een deel van wat bij Boar onder “Organization” valt. Wat in IFW “Data” heet, valt in IAF uiteen in “Information” en “Information Systems”. Enzovoorts.

IFW	Boar	DYA	IAF	MAD	Zachman
Strategy	Organization	Organisatie	Business	Organisational	Motivation
Structure	Organization	Organisatie	Business	Organisational	Network
Skills	Organization	Organisatie	Business	Organisational	

Data	Data	Gegevens	Information Information Systems	Information	Data
Function	Application	Proces	Information Information Systems	Process	Function
Workflow	Application	Proces	Business	Process	Time
Solution		Produkt	Business	Inter- organisational	
Interface		Applicatie	Information Systems	Application	Network Function
Network	Infrastructure	Netwerk	Technology Infrastructure	Configuration	Network
Platform	Infrastructure	Platform Middleware	Technology Infrastructure	Configuration Distribution	Network

**Tabel 2 Vergelijking van zes enterprise-raamwerken**

Voor de vergelijking van de applicatieraamwerken nemen wij vast een voorschot op de waarden in onze nieuwe lijst dimensies (zie kolom "Aspect" in Tabel 3). De IFW begrippen vormen een goede basis, maar moeten worden aangevuld met extra begrippen. Het geheel is meer technisch van aard en nog meer operationeel gebonden. Op dit niveau is er een herkenbare link te leggen naar de UML-concepten, die wij ter informatie nog toevoegen.

Aspect	4+1 model	Siemens	2+2 model	UML
Data				Class diagram
Function	Logical	Conceptual	Context, Conceptual	Use-cases Class diagram
Workflow	Scenarios			Sequence diagram
Ontwikkeling	Development	Module, Code	Development	Packages Class diagram
Deployment	Physical	Code		Components
Runtime	Process	Execution		Classes
Platform	Physical	Execution	Technical Infrastructure	Deployment diagram
Network	Physical	Execution	Technical Infrastructure	Deployment diagram

**Tabel 3 Vergelijking van drie applicatieraamwerken en UML**

### *Voorgestelde dimensies*

Dan willen we nu komen tot een voorstel voor een lijst van raamwerk-onafhankelijke dimensies (zie Tabel 4). Wij maken daarvoor gebruik van de geschetste vergelijking in terminologie. Daarnaast maken we gebruik van onze intuïtie en ervaring in architectuur en we hebben nog wat andere bronnen nageslagen over algemeen bruikbare beschrijvingsaspecten, zoals de al genoemde Evernden Eight [13]. In de

hiernavolgende paragraaf worden de betekenissen uitgelegd. We zeggen er direct bij dat ook deze lijst van dimensies informeel is en veel ruimte voor eigen interpretatie over laat.

<b>Dimensie</b>	<b>Mogelijke Waarden</b>
Type informatie	organisatie (strategie, structuur, bekwaamheden), business (bedrijfsgegevens, bedrijfsfunctie, bedrijfsworkflow, producten, middelen), applicatie (IT gegevens, IT functie, IT workflow, ontwikkeling, deployment, runtime), infrastructuur (netwerk, platform, middleware)
Abstractieniveau	contextueel, conceptueel, logisch, fysiek, implementatie
Mate van keuzevrijheid	domeinconcept, domeinclassificatie, generiek sjabloon, ontwerpcontext, operationeel gebonden
Detailniveau	alle informatie, deel van de informatie, verschil met andere/eerdere informatie
Scope	sector, organisatie, domein, systeem-familie, systeem, component
Kwaliteit	functionaliteit, betrouwbaarheid, bruikbaarheid, efficiëntie, onderhoudbaarheid, portabiliteit
Vorm	beleid, principe, richtlijn, model, standaard
Planning	huidig, korte termijn, middellange termijn, lange termijn, streef
Meta-niveau	instantie, model, metamodel, metametamodel
Welgedefinieerdheid	niet gedefinieerd, informeel gedefinieerd, formeel gedefinieerd
Representatie	tekst, plaatjes, simulatie
Houdbaarheid	in maanden of jaren
Relaties	relaties naar andere gezichtspunten
Doelgroep	doelgroep van architectuurbeschrijving
Betrokkenen	bij architectuurbeschrijving betrokken partijen
Overeenstemming	betrokkenen met wie architectuurbeschrijving is afgestemd

**Tabel 4 Voorgestelde dimensies**

Hier een korte uitleg van de verschillende voorgestelde dimensies:

*Type Informatie* – Dit is de belangrijkste dimensie, waarvan de waarden het mogelijke onderwerp van een architectuurmodel beschrijven. Het is in een architectuurbeschrijving altijd zinvol om een duidelijk onderscheid te maken tussen deze waarden. De “organisatie” groep beschrijft de blijvende waarden, de assets. De “business” groep beschrijft de huidige activiteiten en geleverde diensten en producten. De “applicatie” groep beschrijft de softwarematige IT-ondersteuning die gerelateerd is aan de bedrijfsactiviteiten. De “infrastructuur” groep beschrijft de algemene software en hardware.

*Abstractieniveau* – Deze dimensie is overeenkomstig het Zachman-raamwerk, alwaar het de rijen in het model vertegenwoordigd. Het kan gebruikt worden als tweede dimensie, na ‘type informatie’, maar het kan ook de hoofdindeling zijn. Het is geen verplichting om deze dimensie te hanteren. Een alternatief wordt gevormd door de volgende drie dimensies.

*Mate van keuzevrijheid* – Dit is de dimensie zoals die wordt gehanteerd in het IFW-raamwerk. Zie verder de eerdere beschrijving van IFW.

*Detailniveau* – Deze dimensie biedt de mogelijkheid om een grote hoeveelheid informatie in te delen in lagen, zonder dat de lagen specifieke betekenis hebben. Het is mogelijk dat een volgende laag slechts een deel van de informatie of alleen de verschillen ten opzichte van andere kennis of informatie bevat. Hierbij gaat het ondermeer om een beschrijving van de benodigde voorkennis.



*Scope* - Deze dimensie beschrijft hoe groot het domein is dat de architectuurbeschrijving afdekt. In het algemeen kun je architectuurbeschrijvingen onderscheiden die een gehele bedrijfssector beschrijven, een organisatie, een domein binnen een organisatie (bijvoorbeeld kredieten), een familie van systemen, een specifiek systeem, of een component (onderdeel) van een systeem. Als de scope van een architectuurbeschrijving een gehele organisatie is wordt ook wel gesproken over een enterprise-architectuur of corporate-architectuur. Als een architectuurbeschrijving alleen een applicatie beschrijft dan wordt ook de term applicatie-architectuur gehanteerd, alhoewel deze term ook andere betekenissen kan hebben.

*Kwaliteit* - We hebben reeds eerder opgemerkt dat kwaliteitseigenschappen orthogonaal staan op de andere dimensies. Daarom is er een aparte dimensie voor gedefinieerd, waarbij de waarden bestaan uit de te hanteren kwaliteitseigenschappen. Weergegeven zijn de hoofdeigenschappen van het Extended ISO model [20], maar het is natuurlijk ook mogelijk een ander kwaliteitsmodel te hanteren. Consequentie van het onderkennen van deze dimensie is dat het mogelijk wordt om te spreken over bijvoorbeeld een performance-architectuur of een security-architectuur.

*Vorm* - In deze dimensie wordt bepaald in welke vorm een architectuurbeschrijving precies wordt gegoten; in de vorm van beleid, principes, richtlijnen, modellen, of standaarden. De scheidslijn tussen deze waarden is niet altijd even duidelijk te definiëren. Verder zal het in de praktijk zo zijn dat een specifieke architectuurbeschrijving meerdere vormen tegelijk hanteert, waarbij standaarden en richtlijnen worden vergezeld van modellen.

*Planning* - De planningsdimensie heeft aan welke situatie de architectuurbeschrijving weergeeft: de huidige situatie, een toekomstige situatie of misschien zelfs een streefbeeld waarvan duidelijk is dat deze nooit gehaald zal worden. Deze dimensie komt overeen met het bewustzijn dat het niet voldoende is een lange termijn architectuur te definiëren; minstens net zo belangrijk is het definiëren van het migratiepad om te komen tot deze architectuur. Hierbij kunnen relevante tussensituaties worden onderscheiden, met daaraanvast een tijdslijn

*Houdbaarheid* - Gerelateerd aan de vorige dimensie geldt voor vrijwel alle architectuurbeschrijvingen een bepaalde houdbaarheid, alhoewel daar vaak erg onduidelijk over wordt gedaan. Feit is dat de wereld verandert; er komen nieuwe technologieën, organisaties fuseren en nieuwe producten vragen om nieuwe processen. Het zou dus goed zijn om deze houdbaarheid expliciet te maken; bijvoorbeeld door expliciet aan te geven wanneer volgende releases van de architectuurbeschrijving zullen worden vrijgegeven. Van een architectuurbeschrijving met planningstermijn drie jaar, kan over een jaar al een bijgestelde versie verschijnen.

*Meta-niveau* - Veel architectuurbeschrijvingen hebben niet zozeer betrekking op een bedrijfsspecifieke situatie, maar zijn veel meer een algemene indeling of classificatie. In dat geval is het eigenlijk een meta-architectuur; een architectuur die andere architecturen beschrijft. Deze metastap kan ook meerdere malen worden toegepast, maar gaat zich dan in zeer vage gebieden begeven die meer het terein zijn van tooleveranciers en bedenkers van architectuurbeschrijvingstalen. De waarde "metamodel" in deze dimensie lijkt op de waarde "domein classificatie" in de dimensie "mate van keuzevrijheid". Een verschil is dat metamodellen meestal domeinonafhankelijk zijn.

*Welgedefinieerdheid* - Architectuurbeschrijvingen zijn niet altijd even duidelijk; ze gebruiken allerlei termen waarvan geen definitie wordt gegeven. Beter zou zijn elke architectuurbeschrijving minimaal gepaard te laten gaan van een definitielijst, tezamen met een meta-model die de relaties tussen verschillende termen beschrijft. In specifieke gevallen kan het zelfs noodzakelijk zijn een formele taal te hanteren in een architectuurbeschrijving, om alle mogelijke dubbelzinnigheid te vermijden.

*Representatie* - Deze dimensie beschrijft dat architectuurmodellen op meerdere manieren kunnen worden gerepresenteerd. In het bijzonder is het mogelijk om modellen tekstueel, middels plaatjes of in de vorm van een simulatie (animatie, prototype) weer te geven. Hierbij kan de keuze worden gemaakt uit allerlei standaarden. In geval van visuele modellen ligt bijvoorbeeld het gebruik van UML voor de hand. Vaak gaan tekstuele en visuele representaties hand in hand; use-cases zijn bijvoorbeeld voor een deel visueel, maar voor een veel belangrijker deel tekstueel.

*Relaties* - Zoals eerder beschreven kan architectuurinformatie niet los van andere informatie worden beschouwd. Vrijwel altijd is het van belang deze relaties expliciet te documenteren. De algemene implicatie

is dat moet worden bepaald welke relaties van een architectuurmodel met andere informatie moeten worden beschreven. Deze relaties vormen dus een aparte dimensie in architectuur.

*Doelgroep* – Een architectuurbeschrijving is altijd geschreven voor bepaalde betrokkenen, aangezien deze betrokkenen een bepaalde informatiebehoefte hebben. Dit is ook terug te zien in de abstractieniveaus van Zachman die voor een groot deel worden vormgegeven door de doelgroep.

*Betrokkenen* – Naast de doelgroep heeft een architectuurbeschrijving in het algemeen betrokkenen, die allerlei rollen kunnen hebben. Zo kan de eigenaar, klant of kwaliteitsbewaker invloed uitoefenen op de architectuurbeschrijving, bijvoorbeeld door bepaalde richtlijnen te hanteren.

*Overeenstemming* – Het is relevant om te weten in hoeverre een architectuurbeschrijving is afgestemd met betrokkenen [22]. Als er geen overeenstemming over een architectuurbeschrijving is dan heeft het een stuk minder waarde.

## **Gebruik**

De lijst raamwerk-onafhankelijke dimensies kan op verschillende manieren gebruikt worden. Als checklist kan de lijst gebruikt worden door bij een (concept van een) bestaande architectuurbeschrijving de lijst langs te lopen en te controleren of aan alle voor het probleemdomein relevante aspecten is gedacht.

Een andere toepassing is om een bestaande architectuurbeschrijving te positioneren. Door aan te geven waar de beschrijving zich begeeft in de beschreven dimensies wordt duidelijk wat het doel en de strekking van de architectuurbeschrijving is. Op deze manier ondersteunt de lijst het communiceren over architectuurbeschrijvingen. Dit gebruik komt terug in het hiernavolgende praktijkvoorbeeld.

Tot slot zou de lijst met dimensies gebruikt kunnen worden om de indeling van een nieuwe architectuurbeschrijving vast te stellen of zelfs van een familie van beschrijvingen (raamwerk). Welke dimensie daarbij gebruikt wordt als hoofdindeling is geheel afhankelijk van het probleemdomein en het doel van de beschrijving. Het is dan wel jammer dat er weer een nieuwe indeling gemaakt wordt, maar het voordeel is dat er geput wordt uit een (vanaf nu...) bekende lijst dimensies!

## **Praktijkvoorbeeld**

Eén van de auteurs heeft voor een organisatie in de financiële sector een standaard software-architectuur gedefinieerd. We zullen aan de hand van de beschreven dimensies toelichten wat de strekking van het document is.

Het document bevat de *typen informatie* “IT functie”, “IT workflow” en “ontwikkeling” en beschrijft daarbij de *relatie* met “middleware”. Opdrachtgever voor het document is een programma waarbinnen nieuwe CBD-systemen worden ontwikkeld. Tot de *doelgroep* behoren technisch architecten en ontwikkelaars die betrokken zijn bij het ontwikkelen van deze systemen. Een andere belangrijke *betrokkene* is de centrale architectuurafdeling die inmiddels verantwoordelijk is voor het beheer van het document. Er is *overeenstemming* over document bereikt met deze architectuurafdeling, lopende projecten en algemene stafafdelingen die betrokken zijn bij het beheer van de ontwikkelstraat. In het document worden modellen voorgesteld die IT functies op logisch en fysiek *abstractieniveau* beschrijven. Deze modellen zijn *meta-modellen*; ze beschrijven een standaard onderverdeling van IT functies in de vorm van componenten. Het fysieke model is bedoeld als generiek sjabloon voor uiteindelijke ontwerpen van applicaties. Het *detailniveau* is zodanig dat wordt uitgegaan van voorkennis op het gebied van object-oriëntatie, component-gebaseerd ontwikkelen, de gehanteerde ontwikkelmethode en de gebruikte technologie. Het fysieke model is onderverdeeld in twee lagen van detail, waarbij de eerste de relatie legt tussen het logische model en het gedetailleerde fysieke model. De tweede beschrijft de soorten programmeertaalklassen die typisch in dergelijke systemen voorkomen. De *scope* van de architectuurbeschrijving zijn twee systeem-families; systemen die de gebruiker in de front-office ondersteunen en mid-office componenten. De architectuurbeschrijving richt zich met name op het borgen van de *kwaliteitseigenschappen* “efficiëntie” en “onderhoudbaarheid”. Naast genoemde modellen beschrijft het document ook ontwerpprincipes, richtlijnen en standaarden (standaard interface-definities). De architectuurbeschrijving is qua *planning* een korte tot middellange termijn architectuur, die iedere zes maanden (*houdbaarheid*) wordt bijgewerkt aan de hand van nieuwe inzichten. Termen in de architectuurbeschrijving zijn *informeel gedefinieerd*; het document

begint met een metamodel waarin de gehanteerde termen en hun relaties in natuurlijke taal zijn beschreven. De modellen en standaarden zijn visueel beschreven met UML (*representatie*); alle andere informatie is tekstueel.

In Tabel 5 is de plaats van de architectuurbeschrijving in de verschillende dimensies samengevat door de van toepassing zijnde waarden vet aan te geven. Naar ons idee is het zo mogelijk duidelijk te maken wat de “standaard software architectuur” beschrijft.

Dimensie	Mogelijke waarden	Toelichting
Type informatie	organisatie (strategie, structuur, bekwaamheden), business (bedrijfsgegevens, bedrijfsfunctie, bedrijfsworkflow, produkten, middelen), applicatie (IT gegevens, <b>IT functie, IT workflow, ontwikkeling</b> , deployment, runtime), infrastructuur (netwerk, platform, middleware)	<ul style="list-style-type: none"> <li>• componenten</li> <li>• scenario's</li> <li>• packagestructuur</li> </ul>
Abstractieniveau	contextueel, conceptueel, <b>logisch, fysiek</b> , implementatie	
Mate van keuzevrijheid	domeinconcept, domeinclassificatie, <b>generiek sjabloon</b> , ontwerpcontext, operationeel gebonden	
Detailniveau	alle informatie, deel van de informatie, <b>verschil met andere/eerdere informatie</b>	<ul style="list-style-type: none"> <li>• voorkennis van CBD</li> <li>• fysiek model in twee detailniveau's</li> </ul>
Scope	sector, organisatie, domein, <b>stelsel-familie</b> , systeem, component	<ul style="list-style-type: none"> <li>• front-office applicaties</li> <li>• mid-office componenten</li> </ul>
Kwaliteit	functionaliteit, betrouwbaarheid, bruikbaarheid, <b>efficiëntie, onderhoudbaarheid</b> , portabiliteit	
Vorm	beleid, <b>principe, richtlijn, model, standaard</b>	
Planning	huidig, <b>korte termijn, middellange termijn, lange termijn</b> , streef	
Meta-niveau	instantie, model, <b>metamodel, metametamodel</b>	<ul style="list-style-type: none"> <li>• logisch en fysiek metamodel</li> <li>• metametamodel voor terminologie</li> </ul>
Welgedefinieerdheid	niet gedefinieerd, <b>informeel gedefinieerd</b> , formeel gedefinieerd	
Representatie	<b>tekst, plaatjes</b> , simulatie	<ul style="list-style-type: none"> <li>• UML</li> </ul>
Houdbaarheid	in <b>maanden</b> of jaren	<ul style="list-style-type: none"> <li>• elke 6 maanden nieuwe release</li> </ul>
Relaties	relaties naar andere gezichtspunten	<ul style="list-style-type: none"> <li>• relatie met middleware</li> </ul>
Doelgroep	doelgroep van architectuurbeschrijving	<ul style="list-style-type: none"> <li>• technisch architecten</li> <li>• ontwikkelaars</li> </ul>
Betrokkenen	bij architectuurbeschrijving betrokken partijen	opdrachtgever: <ul style="list-style-type: none"> <li>• programma beheerder</li> <li>• <u>architectuurafdeling</u></li> </ul>
Overeenstemming	betrokkenen met wie architectuurbeschrijving is	<ul style="list-style-type: none"> <li>• <u>architectuurafdeling</u></li> </ul>

	afgestemd	<ul style="list-style-type: none"> <li>• projecten</li> <li>• stafafdelingen die ontwikkelstraat beheren</li> </ul>
--	-----------	---

**Tabel 5 Positionering “standaard software-architectuur”**

## Conclusies en toekomstig werk

Bij het beschrijven van informatie over architectuur zijn veel verschillende dimensies betrokken. De dimensies die worden gebruikt in bestaande architectuur-raamwerken zijn in dit artikel samengevoegd tot een lijst van raamwerkonafhankelijke dimensies. Het belangrijkste doel van deze lijst is het vereenvoudigen van communicatie over architectuurbeschrijvingen, maar daarnaast heeft deze lijst ook andere handige toepassingen. In het algemeen denken wij dat communicatie ook het belangrijkste doel van architectuurraamwerken is. Welk raamwerk je daarbij precies gebruikt is niet zo relevant.

Bij gelegenheid zouden wij graag de voorgestelde lijst beter funderen dan wij nu gedaan hebben en de beschrijvingen preciezer maken. Graag maken we daarbij gebruik van aanvullingen en correcties die ons worden toegezonden. Anders gezegd: reacties welkom!

Tot slot kunnen wij opmerken dat wij dit een leuke en nuttige exercitie vonden; we hebben er veel van geleerd. Wij danken de reviewers voor hun goede commentaren.

*Danny Greefhorst is werkzaam als IT Architect bij IBM Global Services en bereikbaar onder [greefhorst@nl.ibm.com](mailto:greefhorst@nl.ibm.com). Henk Koning is werkzaam als onderzoeker bij SERC en bereikbaar onder [koning@serc.nl](mailto:koning@serc.nl).*

## Referenties

- [1] IEEE Std 1471-2000: “IEEE Recommended Practice for Architectural Description of Software-Intensive Systems”, 2000.
- [2] ISO/IEC CD 10746-1, “Basic Reference Model of Open Distributed Processing”, 1994.
- [3] P. Kruchten: “The 4+1 View Model of Architecture”, IEEE Software, 1995.
- [4] J. A. Zachman: “A Framework for Information Systems Architecture”, IBM Systems Journal 26, No. 3, 1987.
- [5] J.F. Sowa and J.A. Zachman: “Extending and formalizing the framework for information systems architecture”, IBM Systems Journal 31, No. 3, 1992.
- [6] R. Evernden: “The Information FrameWork”, IBM Systems Journal, 1996.
- [7] The Open Group: “The Open Group Architectural Framework”, Version 7
- [8] H. Goedvolk, D. Rijsenbrij: “White Paper Integrated Architecture Framework”, version 1.0, 1999.
- [9] J.L. Meinema: “Corporate architecture: a conceptual approach”, University of Twente, 1999.
- [10] C. Hofmeister, R.L. Nord, D. Soni: “An Industrial Perspective of Software Architecture”, In Proceedings of the Eleventh International Conference on Data Engineering, IEEE Computer Society, Taipei, Taiwan, March 1995.
- [11] C. Hofmeister, R. Nord, D. Soni: “Applied Software Architecture”, Addison Wesley, 2000.
- [12] N. Lassing, D. Rijsenbrij, H. van Vliet: “Zicht op aanpasbaarheid – 2+2 viewmodel”, Informatie, September 2001.
- [13] R. Evernden: “Evernden Eight”, 4<sup>th</sup> resource, 2002.
- [14] R. Wagter et al.: “DYA: snelheid en samenhang in business en ICT-architectuur”, Tutein Nolthenius, 2001.
- [15] Bernard H. Boar: “Constructing Blueprints for Enterprise IT Architecture”, Wiley, 1998.

- [16] P. Herzum, O. Sims: “Business Component Factory”, John Wiley & Sons, 2000.
- [17] M.W. Maier, E. Rechtin: “The art of systems architecting”, CRC Press, 2002.
- [18] A.W. Scheer : “Architecture of Integrated Information Systems”, Springer, 1992.
- [19] D. Tapscot: “Paradigm Shift”, McGraw Hill, 1993.
- [20] B. van Zeist et.al.: “Kwaliteit van software producten, praktijkervaring met een kwaliteitsmodel”, Kluwer Bedrijfsinformatie, 1996.
- [21] L. Hermans: “Uitbuiten synergie ICT- en businessstrategie”, Informatie, September 2002.
- [22] D. Verhoef: “Eisen aan een architectuur”, Landelijk Architectuur Congres, 2000.