

MET DE KOMST VAN DE AGILE ONTWERPMETHODEN IS OOK DE ROL VAN DE ARCHITECT VERANDERD. RUTGER GOOSZEN, PAULUS MEIJS EN MARTIJN TEN NAPEL PLEITEN ERVOOR DAT DE ARCHITECT VEEL MEER EEN COACH WORDT. ZIJ MAKEN DUIDELIJK HOE HIJ OF ZIJ DIE NIEUWE ROL MOET OPPAKKEN EN INVULLEN.

door Rutger Gooszen, Paulus Meijs en Martijn ten Napel beeld Shutterstock



# ARCHITECTEN ZIJN DE NIEUWE COACHES

## Hoe de architect zijn nieuwe rol kan invullen

ALLES IS TEGENWOORDIG VERBONDEN MET INFORMATIE-TECHNOLOGIE (IT). Wat dat betekent, is bij gebruikers en managers inmiddels doorgedrongen. Het zijn de architecten van IT die soms nog moeite hebben met het loslaten van traditionele ontwerpmethoden. Als je onderkent dat onze maatschappij en daarmee onze organisaties toenemen in complexiteit (als een soort natuurwet van samenwerking van mensen), waarbij we voortdurend voortschrijdend inzicht toepassen om met die complexiteit om te gaan, is agile werken een andere benadering van dat vraagstuk. Het vraagt van een architect een andere manier van werken. Je zult coachend moeten acteren, naar zowel gebruikers als bouwers toe, en ze laten ervaren hoeveel leuker en beter hun werk wordt als ze samenwerken en verantwoordelijkheid pakken (of vaak: opeisen). Dat lukt niet altijd en overal in elke organisatie, maar als het lukt, in een organisatie waar de voedingsbodem goed is, dan bereik je een heel mooie transitie. In dit artikel wordt de nieuwe rol verkend van de architect bij complexe veranderingen met IT die volgens een agile aanpak worden uitgevoerd en wat dat vraagt van die architect.

### COMPLEXE PROJECTEN MET IT MISLUKKEN VAAK

Als we al langer weten dat veel complexe IT-projecten traditioneel mislukken, waarom blijven we dan elke keer weer hetzelfde doen? Complexe projecten zijn notoir moeilijk te beheersen volgens een ontwerpbenadering waarbij je alles in detail van tevoren wilt uitwerken. De vele stakeholders, belangen, afhankelijkheden, mogelijke opties en consequenties lijken soms een onontwarbaar kluwen. Het antwoord was lange tijd dat met architectuur, als hulpmiddel om samenhang en inzicht te geven, de complexiteit voorspelbaar en beheersbaar zou worden. Het effect is vaak dat bij complexe projecten ook de complete architectuur complex dreigt te worden, met lange doorlooptij-

den om deze compleet in kaart te brengen. Daardoor wordt architectuur eerder een belemmering dan een hulpmiddel.

### EEN ANDERE WIJZE VAN BESTUREN: AGILE ONTWIKKELING

De gepredikte oplossing tegen mislukking van complexe projecten met IT, is het volgen van de agile besturingsfilosofie: niet het hele systeem van tevoren in detail specificeren, maar de scope en kaders in grote lijnen duidelijk aangeven en daarna gewoon beginnen met de onderdelen die als meest waardevol zijn benoemd. Met veel betrokkenheid van de 'business' en in zelforganiserende teams. Maar een andere besturingsfilosofie is niet een nieuw IT-speeltje waar de rest van de organisatie geen last van heeft. Als de randvoorwaarden voor het toepassen van deze andere besturingswijze niet worden ingevuld, mislukt het project net zo hard als op de oude wijze. De 'Groundhog Day'-ervaring blijft dat elk project lijkt op het vorige.

### SOMMIGE ZAKEN BLIJVEN BIJ HET OUDE: RICHTING GEVEN IS NOODZAKELIJK

Randvoorwaarde één: het te bereiken doel en daaruit afgeleid de scope en kaders van de verandering, zijn bij de start goed afgebakend. Ook voor fundamentele beslissingen omtrent zogenaamde non-functionals, zoals performance, security en onderhoudbaarheid, is het belangrijk om van tevoren duidelijke criteria te stellen, keuzes te maken en richtlijnen op te stellen. Het highleveldesign is als kader nog steeds belangrijk, omdat teams decentraal aan het werk gaan maar wel een gezamenlijk beeld moeten delen. Architectuur, als bringer van structuur op hoofdlijnen en inzicht, is dus hetzelfde gebleven. Decompositie in het ontwerp (een bestemmingsplan of blauwdruk) op voorhand is en blijft nodig. De bewering dat je met agile geen gedefinieerde doelarchitectuur of ontwerp meer nodig hebt, alleen goede

#### AUTEUR



RUTGER GOOSZEN is managementadviseur in de strategie, governance- en architectuurpraktijk van Verdonck, Klooster & Associates. Hij heeft recentelijk ervaring opgedaan met het Scaled Agile Framework als aanpak bij complexe veranderprogramma's en is mede-initiatiefnemer van [www.onafhankelijke-architect.nl](http://www.onafhankelijke-architect.nl)

#### AUTEUR



PAULUS MEIJS is allround architect en analist met ervaring in diverse branches. Hij is gepassioneerd over zijn architectuurvak en de agile cultuur, alles ten dienste om business en IT zo waardevol en effectief mogelijk te laten samenwerken. Hij is dan ook specialist in agile architectuur met al zijn vertakkingen, waaronder naar ondersteuning van leadership voor de digitale transitie.

#### REACTIES EN BIJDAGEN

Voor reacties en nieuwe bijdragen van IT-experts: Tanja de Vrede 020-2356415 [t.d.vrede@agconnect.nl](mailto:t.d.vrede@agconnect.nl)

## Agile speelt in op hoe wij mensen als diersoort werken



businessvertegenwoordigers die kunnen prioriteren, is flauwekul. De detaillering van de doelarchitectuur is bij de start minder diepgaand, en de manier waarop je detaillering tot stand laat komen, is gaandeweg in plaats van 'up front'. Dat detailleren gebeurt in autonome teams, en om teams autonoom te kunnen laten werken, moet je afdoende ontkoppeling en prioritering op voorhand hebben bepaald. Daarnaast moet de opdrachtgever (en degenen die hem ter verantwoording roepen) ook achter deze manier van werken staan (van grof naar fijn), omdat anders verwarring ontstaat over de mate van detail die nodig is. Integratie en samenwerking blijven noodzakelijk. Daar komt de agile architect in beeld. Die werkt niet in een voorschrijvende en controlerende rol aan de oplossing, maar in een coachende en verbindende rol, werkend vanuit het overzicht dat hij als een van de weinigen kan geven.

### ARCHITECT: (MEDE)-ONTWERPER VAN DE BESTURING EN STRUCTUUR

Randvoorwaarde twee: agile methodiek vraagt om een fundamenteel andere besturing. De doelstelling verandert niet met agile, maar juist de manier van werken en dus besturen. Adopteren van welke methodiek dan ook is onzinnig als je niet ten diepste bereid bent om de besturingsfilosofie te veranderen. Veel organisaties starten met agile zonder

bereid te zijn de bestaande besturingsfilosofie (projectmatig werken en 'in control zijn') los te laten. Dat heeft een stuwmeer aan adviseurs agile werken en andere haarlemmerolie opgeleverd. Maar de opdrachtgevers moeten eerst bereid zijn hun werkwijze en besturing te reorganiseren. De architect, als adviseur van de opdrachtgever, moet daarom de opzet van de governance ontwerpen die past bij de agile werkwijze, doel en scope. De krampachtigheid bij sommige organisaties in de zoektocht om de besturingsfilosofie maar niet te hoeven veranderen, maakt dat ze zichzelf daarmee in de voet schieten. Eigenaarschap, verantwoordelijkheidsverdeling en betrokkenheid (tijd investeren) zijn daarbij de sleutelbegrippen. Om zelforganiserende teams met de juiste expertise te laten werken aan integreerbare componenten, speelt agile in op hoe wij mensen als diersoort werken: we zijn hiërarchische kuddedieren, waarbij taakverdeling op basis van onderhandeling (we noemen dat consensus) tot stand komt. Samenwerking en taakverdeling is wat ons als mens tot mens maakt. De taak van de architect is het bewaken van de consistentie tussen de decompositie in de blauwdruk en de consensus tussen de teams over wie wat doet en waar de grenzen van hun mandaat liggen. Niets meer, niets minder en even hard nodig als de andere taken.

### DE ARCHITECT ALS COACH:

- 1 is vanaf de start betrokken;
- 2 stelt kaders, maar is meer vraagbaak;
- 3 dwingt tot simplificatie;
- 4 is onderzoekend;
- 5 is facilitator in plaats van poortwachter;
- 6 durft in te grijpen.

### RANDVOORWAARDEN AGILE SUCCES

- 1 Scope, doel en ontwerpaders helder?
- 2 Besturing aangepast aan agile werken?
- 3 Organisatie stabiel en eigenaren benoemd?

### EERST (RE)ORGANISEREN EN DAN PAS AUTOMATISEREN

Randvoorwaarde drie: je kunt de IT pas goed realiseren als organisatorische knelpunten zijn opgelost en eigenaarschap en verantwoordelijkheden helder zijn. Hier ligt het echte pijnpunt van veel mislukte automatiseringsprojecten: de techniek moet de organisatorische impasse maar doorbreken, met alle drama's van dien. Als architect bewaak je de samenhang tussen mensen, processen, informatie en techniek en het is aan de architecten om af te dwingen dat het 'management' eerst de organisatorische problemen oplost voor we een IT-stap vooruit zetten. Dat gaat nooit vanzelf, dus ook daar zul je in je oplossing wendbaar moeten zijn en in deeloplossingen, vervangingstrajecten en evolutionaire progressie stap voor stap het organisatorisch probleem tot oplossing dwingen.

### DE ARCHITECT ALS COACH

Wat betekent dit voor de rol van de architect? Tijdens de start van het traject zal er een voortrekkersrol zijn om mede de belangrijkste componenten en systeem-eigenschappen in kaart te brengen. Deze bespreek je met de teams (van business tot en met beheer) en je toetst of zij de consequenties begrijpen en ermee instemmen. Met de opdrachtgever wordt een passende besturing afgesproken. Daarna laat de architect meer over aan de teams om het raamwerk in te vullen met te ontwikkelen functionaliteiten. De architect stuurt op samenhang en kwaliteit door eisen aan de te realiseren functies in de Definition of Ready en de Definition of Done op te laten nemen en opgeleverde producten hieraan te (laten) toetsen. Deze eisen zijn afgeleid uit de kaders die zijn afgesproken. De architect houdt de vinger aan de pols (bijvoorbeeld tijdens de sprintplanning-sessies en opleveringen) om te beoordelen of mogelijke consequenties van detailinvulling goed worden begrepen en dient als vraagbaak als teamleden oplossingen voorstellen die mogelijke

consequenties kunnen hebben op andere onderdelen van de blauwdruk.

Complexe projecten en complexe problemen vragen om simplificatie, en de agile architect zal voortdurend aan simplificatie moeten werken. Als je dat lineair doet, kom je nooit ergens op tijd, dus moet hij dat gaan paralleliseren en institutionaliseren in de teams: het stopt nooit, het is een manier van doen. Alles en iedereen tot simplificatie dwingen door decompositie, voortdurende terugkoppeling door leerervaring en de evaluatie van de leerervaring met als doel verdere simplificatie, zit in de genetica van alle agile werkmethode. Ergens in het agile geweld is die boodschap verloren gegaan.

### DE COMPETENTIES VAN DE AGILE ARCHITECT

De geschetste rol vraagt meer van de architect dan in zijn rol bij traditionele ontwerpgedreven veranderingen. Hij moet nog steeds verstand hebben van methoden en technieken voor het inzichtelijk maken van het vraagstuk, en de complexiteit communiceerbaar maken. Daarnaast moet hij veel meer onderzoekend en faciliterend zijn. Hij zal meer in interactie met de teams moeten werken; wel met zijn deskundigheid als basis voor zijn adviezen en antwoorden, maar minder als poortwachter. Hij zal ook overtuigend moeten kunnen zijn, als er issues onder het tapijt dreigen te worden geveegd die later grote consequenties kunnen hebben. Een allround adviseur en coach die op alle facetten van de verandering de spelers kan verbinden en kan aanzetten tot het (met afgesproken kwaliteit) bijdragen aan het gemeenschappelijke doel. Want als bij agile de programmanager en projectleider zijn verdwenen, wie bewaakt dan de kwaliteit en de samenhang van de oplossing die wordt gebouwd en hoe doe je dat? Dit 'hoe' is onderwerp voor een volgend artikel. 📖

COMPLEXE PROJECTEN EN COMPLEXE PROBLEMEN VRAGEN OM SIMPLICITATIE

### AUTEUR



MARTIJN TEN NAPEL is architect in het data-domein. Sinds 1998 is hij doorkneed met het gedachtegoed achter het agile manifest. In het datadomein komt de autonomie van professionals, de vraag naar sturing op samenhang en technologische complexiteit samen. Agile werkwijzen en denkwijzen helpen om deze uitdaging het meest effectief het hoofd te bieden. Martijn heeft met collega's de opgedane ervaring gebundeld in een raamwerk dat ze de naam 'connected architectuur' hebben gegeven.