



## APG RICHT DEVOPS-READY DATA CENTER IN “WIJ AUTOMATISEREN DE AUTOMATISERING”

Pensioenuitvoerder APG wil een moderne, wendbare organisatie blijven. Om dat te realiseren, wordt een compleet nieuwe IT-infrastructuur neergezet waarin de organisatie een grote hoeveelheid applicaties via de DevOps-methode kan ontwikkelen en beheren. Manager Architecture & Innovations Henk Dado vertelt over de voordelen. En de uitdagingen.

APG is een van de grootste uitvoerders van collectieve pensioenen ter wereld en regelt voor haar opdrachtgevers het pensioen van 4,5 miljoen mensen. Wil je wendbaar zijn in een turbulente markt waarin ook wetgeving snel kan veranderen, dan zul je je manier van werken daarop moeten aanpassen. Een hele klus in een organisatie met veel belanghebbenden, waar honderden applicaties voor intern en extern gebruik draaien. “Een aantal jaren geleden hebben we geconstateerd dat we om die snelheid te bereiken agile moesten gaan werken,” zegt Henk Dado, Manager architectuur en innovatie bij APG. “Als je iedere twee weken een release wilt doen of een sprint wilt afronden, moet je infrastructuur dat echter wel kunnen volgen. Dat betekent dat je de IT-infrastructuur verregaand moet automatiseren, zodat de DevOps-teams op den duur zo veel mogelijk zelfstandig kunnen doen.”

### HOE HEBBEN JULLIE DAT AANGEPAKT?

“Er zijn verschillende belangrijke onderdelen. In de eerste plaats zijn we software defined infrastructure gaan toepassen. Zo kunnen we onze hele infrastructuur-inrichting met software regelen. We stelden vast dat we specifieke softwareontwikkelaars nodig hadden om de software te maken waarmee het DevOps-team uiteindelijk de infrastructuur kon inrichten.

Maar daarmee ben je er nog niet. Ook aan de ontwikkelkant wilden we automatisering doorvoeren met continuous deployment in gedachten. Dat betekent dat we flink hebben

geïnvesteed in de manier waarop de ontwikkeling loopt en in de standaardisatie daarvan. Als je wilt automatiseren, ontkom je niet aan standaardiseren.”

### **HOE STAAT HET ER NU VOOR?**

“We hebben de eerste paar applicaties live op dit DevOps-Ready Data Center. De infrastructuur-componenten zijn softwarematig gecreëerd en worden softwarematig beheerd. Het onderscheidende element is overigens niet zo zeer de software voor een infrastructuurcomponent, maar wel de automatische inrichting daarin van zaken als authenticatie, autorisatie, logging, security, continuity, monitoring enzovoort. Basiscomponenten als een IIS-server, een managed Linux-server of een Tomcat-server kun je al bij wijze van spreken met een klik maken. Maar we zijn ook nog steeds infrastructuursoftware-bouwblokken, aan het maken. Zo kunnen we bijvoorbeeld wel een Oracle-schema in een database creëren, maar een Oracle-database of een SQL-database nog niet, maar dat gaat wel komen. We bekijken per applicatie die we in de infrastructuur neerzetten wat die nodig heeft. Als de applicatie daadwerkelijk gaat landen, zorgen we dat de bouwblokken die nodig zijn om de benodigde infrastructuur te creëren klaar zijn. Als je nieuwe applicaties laat landen, zie je echter dat ze soms weer specifieke eigenschappen hebben die niet helemaal passen bij de infrastructuur die we hebben neergezet. Maatwerkapplicaties kun je dan aanpassen. Of je past de infrastructuur-software aan. Bij maatwerk kun je zelf besluiten aan welke eisen een infrastructuurcomponent moet voldoen. Dan zeggen we bijvoorbeeld: we gebruiken deze versie van Java van die-en-die leverancier. Bij gekochte pakketten bepaalt de leverancier dat. Het hele proces is niet iets wat je even snel doet, maar ondoenlijk is het zeker niet.”

### **JULLIE HEBBEN NOG HEEL VEEL APPLICATIES DIE IN DE INFRASTRUCTUUR TERECHT MOETEN KOMEN. KAN DAT STRAKS SNELLER GAAN?**

“Zeker. Als je meer bouwblokken hebt gemaakt, kun je die blijven gebruiken. Het werk zit erin dat je steeds moet kijken hoe de applicatie in elkaar zit en hoe je met de verschillen omgaat. Als het aan de applicatiekant veel geld kost, kijken we aan de infrastructuurkant hoe we het oplossen, maar in principe houden we de standaard van de infrastructuur aan.”

### **DEVOPS HEB JE NIET VAN DE ENE OP DE ANDERE DAG STAAN. HOE WERKT DAT?**

“Het begint met agile werken. Daar zijn we nu al een jaar of zes mee bezig, waarvan de laatste drie jaar echt intensief. Daarbij begeleiden we de teams met agile coaches. Inmiddels is agile werken tot op het niveau van de raad van bestuur omarmd. Het is uiteraard belangrijk om behalve naar de agile

teams ook naar de managementkant te kijken. Ook daar zorgen we voor trainingen en begeleiding en veranderen rollen behoorlijk. Sommige bedrijfsonderdelen zijn al verder dan andere, maar het gaat absoluut steeds beter. Ten slotte hoort bij agile ook continu verbeteren.

Verder hebben we geleerd dat er nogal een verschil is tussen agile doen en agile zijn. Het gaat niet om het doen van de rituelen, want wil je het echt zijn, dan moet het in je genen gaan zitten.

In een traditionele wereld is het soms de projectleider die de meeste pijn heeft en het team zegt bij wijze van spreken: we gaan om 5 uur naar huis. Hier gaat het team het commitment aan om te leveren. De scrum master zorgt dat moeilijkheden worden weggenomen. Dat is best een cultuurverandering, benoemen wat er mis is gegaan en wat het team daar zelf van kan leren. En er dan ook echt iets mee doen.”

### **WAT IS DE VOLGENDE STAP?**

“Aan de ‘Ops’-kant is het nog veel ingewikkelder. Los van het feit dat de infrastructuur nog niet helemaal klaar is, moet je mensen genoeg middelen geven om hun werk te kunnen doen. Waar je van tevoren niet zo over nadenkt, is dat er allerlei organisatievraagstukken zijn die gevolgen hebben voor de automatisering. Als iemand een rol krijgt, moet hij de bijbehorende rechten automatisch krijgen. Je moet bijvoorbeeld niet hebben dat allerlei mensen die over security gaan een heleboel handelingen moeten verrichten voordat je verder kunt. Dat betekent dat je moet standaardiseren. Dat doen we role based. Het is daarbij een behoorlijke uitdaging om alle infrastructuur-componenten op dezelfde lijn in te richten. Voor de teams is het ook even wennen. Als er ‘s nachts een probleem was, werden voorheen eerst de datacenter-mensen uit bed gebeld. Nu is het DevOps-team als geheel verantwoordelijk: ‘you build it, you run it’. Het goede nieuws is dat er zo wel extra druk ontstaat om te voorkomen dat zich problemen voordoen.”

### **BIJ DEVOPS LIGGEN MINDER ZAKEN VAN TEVOREN VAST. IS DAT NIET LASTIG?**

“Aan een kant moet je zaken juist strakker regelen. Omdat de tijdlijnen korter zijn, moet je nu eenmaal duidelijke afspraken maken. Als er veel is geregeld omdat het automatisch gebeurt of omdat er goede afspraken zijn, hoeft het team niet alles van voren af te bedenken en kun je meer snelheid maken. Daarbij moet je een balans zoeken. We hebben nu een vaste set van tooling terwijl sommige ontwikkelaars misschien liever hun eigen favoriete tools gebruiken. Ik denk ook niet dat je alles aan een DevOps-team moet overlaten. Zaken als security patches en lifecycle management kunnen via expertise aan de zijkant worden gestimuleerd. In een team van zeven of acht mensen kan niet

iedereen alle aspecten kennen. Wat security patches betreft, baseren ons zo veel mogelijk op wat in de cloud gebruikelijk is. Daar worden de patches ook automatisch uitgevoerd.”

### **HEBBEN JULLIE SPECIFIEK NIEUWE MENSEN AANGENOMEN VOOR DE DEVOPS-TEAMS?**

“In principe doen we zo veel mogelijk met de mensen die er al zijn. We zien dat functies breder worden. Je verwacht van mensen dat ze niet alleen kunnen ontwikkelen of testen. Iedereen heeft een specialisme, maar je moet in de breedte ook veel kunnen. Van sommige rollen, zoals informatieanalisten en projectmanagers, neemt het volume af. In een meer traditionele waterval-methode specificeer je alles tot in detail en daarna gaat de ontwikkelaar aan de slag, waarna het wordt getest. Nu gaat het veel sneller en wordt het automatisch getest, een absolute voorwaarde om op deze manier te werken. Het is wel lastig om mensen te vinden die dit totaal van de automatisering van de automatisering kunnen overzien. Je moet heel veel infrastructuurkennis, software development skills én kennis van architectuur hebben. Dat is trouwens niet alleen een kwestie van opleiding, maar vooral ook van ervaring en mentaliteit. Mensen moeten het echt leuk vinden. Met dit traject zullen we nog wel een paar jaar bezig zijn en ook daarna zal het werk niet ophouden. Bij het onboarden van nieuwe applicaties kom je steeds nieuwe uitdagingen tegen en moet je nieuwe vraagstukken oplossen.”

### **HOE ZORG JE DAT DEZE INFRASTRUCTUUR TOEKOMSTVAST BLIJFT?**

“Ook dat is een uitdaging. Kun je een applicatie maken, maar die ook weer opruimen als je dat wilt? Ik vergelijk applicaties wel eens met kerncentrales: het kost veel meer tijd dan je had gedacht om ze te bouwen en het afbreken kost ook veel meer dan gedacht. De kunst is om duurzaamheidsprincipes toe te passen. Daarom is architectuur ook zo

belangrijk, ook in een DevOps-wereld: iets duurzaam maken dat je ook weer kunt weggooien, liefst tegen niet te hoge kosten. Mensen vragen ons ook wel eens of we niet een nieuwe legacy creëren, maar dan aan de infrastructuurkant. Het is een uitdaging om te zorgen dat dat niet gebeurt. Dat is een van de redenen dat je mensen nodig hebt die het geheel overzien. Wanneer ben ik legacy aan het bouwen waar ik nooit meer van af kan. Het is relatief nieuw en af en toe maak je een keuze waarvan je denkt: hadden we dat wel moeten doen? Dat moet je niet te vaak hebben, maar het hoort wel bij agile.”

### **JULLIE HEBBEN EEN TEAM VAN BAUHAUS/ MATRIXMIND INGESCHAKELD OM TE ADVISEREN OVER VOLGENDE STAPPEN. KUN JE IETS ZEGGEN OVER HUN AANBEVELINGEN?**

“Alles wat we in het datacenter aan het doen zijn, creëren we nu nog vooral on premises. Ons oorspronkelijke idee was om zelf een kerndatacenter te hebben met uitwijkmogelijkheid naar de cloud als we zouden willen opschalen. Ik weet nog niet zeker of we dat op die manier kunnen doen. Wat wel kan, is om infrastructuur-componenten via onze bouwblokken te creëren op een IaaS. Dat zit in het concept maar aanbevolen is om ook meer naar hybrid cloud-oplossingen te kijken. Dynamisch op- en afschalen is nog wel lastig. Je moet daar bijvoorbeeld bij je load balancing rekening mee houden. Bovendien zijn veel van onze applicaties niet bepaald gisteren of eergisteren gebouwd en dus niet op horizontale schaalbaarheid gemaakt. Waar we nu over nadenken is, is om zaken die niet zo'n variabele load hebben on premises zetten en kijken of we zaken waarbij de load varieert, bijvoorbeeld aan de front-office-kant volledig in de cloud kunnen doen. Zo ontwikkelen we nu ook een datacentervisie waarin het vraagstuk wat we aan datacenter-capaciteit nodig hebben geadresseerd wordt. Het zal zeer waarschijnlijk wel kleiner worden.”

## **AGILE EN DEVOPS**

In een DevOps-omgeving is de grens tussen ontwikkel- en beheerteams verdwenen. Ze werken samen en zijn verantwoordelijk voor het uiteindelijke bedrijfsresultaat. De vraag hoe software wordt uitgerold en ondersteund is even belangrijk als de vraag hoe het wordt ontwikkeld. Een groot voordeel van DevOps is dat de time-to-market wordt verkort tegen lagere kosten. Dit kan door bij iedere stap zo veel mogelijk te automatiseren en te monitoren. Van integreren, testen en releasen tot deployment en infrastructuurbeheer. Een belangrijk onderdeel van DevOps is agile werken: een kortcyclisch ontwikkelproces met korte lijnen van idee tot ontwikkeling.



**MATRIXMIND**

VAN DEVENTERLAAN 30-40  
3528 AE UTRECHT

T +31 30 711 88 41  
E OFFICE@MATRIXMIND.NL