

DATA-DRIVEN

SUSTAINABILITY

# DUURZAME IT: HOE ONTWERP IK EEN DUURZAME IT STRATEGIE?

ORANJE BOEKJE DEEL 2

DOOR:

LAILA FETTAH

RONALD MEIJER

JAN SCHRAVESANDE

EDWIN VAN DER BURG



**ZIENSWIJZE OP DUURZAME IT :**

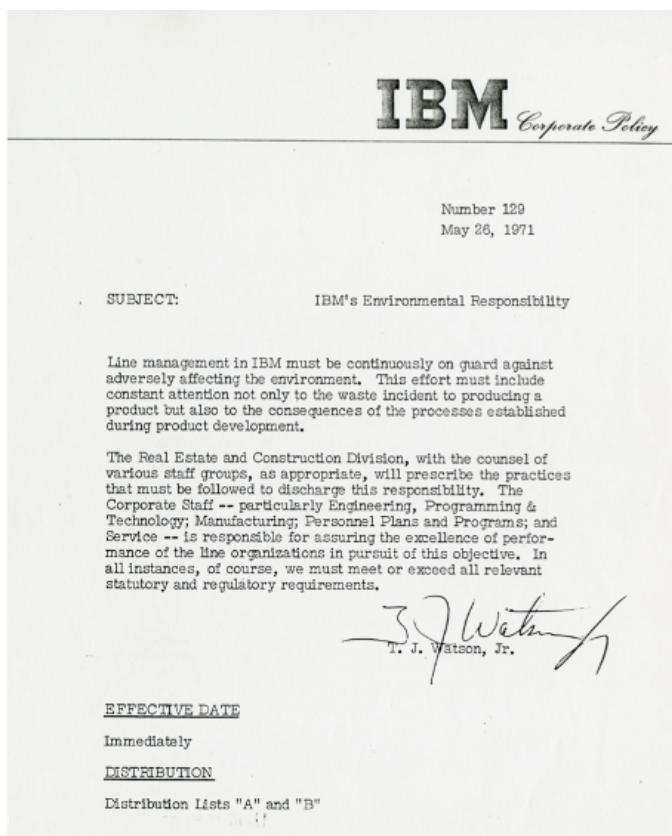
**DE REIS  
VAN  
ONBEWUST  
ONBEKWAAM  
NAAR  
ONBEWUST  
BEKWAAM**

# VOORWOORD

26 Mei 1971 was het de baas van IBM, Watson jr. die in een brief aan zijn managers aangaf dat IBM een grote verantwoordelijkheid heeft ten opzichte van het milieu en dat alle processen en regels hieromtrent tot in het kleinste details moeten worden gevuld.

Met dit boekje geven we opnieuw aan hoe belangrijk en actueel dit onderwerp vandaag de dag is. In dit boekje gaat het over cultuur en het nemen van verantwoordelijkheden. Ik durf te zeggen dat duurzaamheid in het DNA van IBM zit verweven en dat de schrijvers van dit boekje, Laila, Edwin, Ronald en Jan dit doen vanuit een diep verantwoordelijkheidsgevoel op hun domein: IT en IT Architectuur. Er is nog zo veel te winnen maar we zien dat bedrijven de stappen niet durven of kunnen zetten. Toch zal de wal het schip gaan keren en mocht het zover zijn dat ook uw bedrijf serieus is met de substantiële verduurzaming van de IT dan is dit een handig overzicht. Responsible Computing is mede door IBM tot stand gekomen en gedoneerd aan de Object Management Group (OMG).

Ik schreef in het voorwoord van het eerste boekje uit de oranje reeks "*Develop a Blueprint for a data-driven Enterprise Architectuur*" dat architecten geen auteurs zijn, maar dat boek werd met zoveel enthousiasme ontvangen dat ik dat niet meer durf te zeggen. Ik hoop dat dit boekje met evenveel plezier gelezen wordt en dat er af en toe een glimlach op het gezicht verschijnt om



de kwinkslagen. Nog meer hoop ik dat u af en toe de wenkbauwen fronst en dat de urgentie van dit onderwerp u tot actie zal bewegen.

In het hoofdstuk Responsible Impact wordt een “prijsvraag” uitgeschreven. Diegene die met het beste idee komt ontvangt gratis een sustainability workshop van ons met als uitkomst een MVP (Minimum Viable Product) voor uw duurzame ideeën!

Een geweldig idee. Ik sta daar van harte achter en ondersteun dit initiatief persoonlijk.

Johan Heij  
IBM Nederland

# INHOUDSOPGAVE

Zienswijze op duurzame IT :	3
Voorwoord	4
Inhoudsopgave	6
<b>Introductie van het raamwerk</b>	<b>10</b>
<b>1. Inleiding</b>	<b>10</b>
<b>1.1. Van onbewust onbekwaam naar onbewust bekwaam</b>	<b>15</b>
<b>1.2. Promoveren</b>	<b>16</b>
<b>1.3. Getallen</b>	<b>17</b>
<b>2. Responsible computing is duurzaam!</b>	<b>18</b>
<b>2.1. Responsible datacenter</b>	<b>19</b>
<b>2.2. Responsible infrastructure</b>	<b>19</b>
<b>2.3. Responsible code</b>	<b>20</b>
<b>2.4. Responsible data usage</b>	<b>20</b>
<b>2.5. Responsible systems</b>	<b>21</b>
<b>2.6. Responsible impact</b>	<b>21</b>
<b>2.7. Responsible Computing Principles</b>	<b>21</b>
<b>2.8. Aansprakelijk (accountability) geldt voor ieder RC domein</b>	<b>23</b>
<b>People</b>	<b>25</b>
<b>3. De IT Architect: Ontwerper van duurzaamheid</b>	<b>27</b>
<b>3.1. Non-functional duurzaamheid requirements</b>	<b>28</b>
<b>3.2. Transparantie</b>	<b>29</b>
<b>3.3. Energieverbruik</b>	<b>31</b>
<b>3.4. Herbruikbaarheid</b>	<b>32</b>
<b>3.5. Inclusiviteit</b>	<b>33</b>
<b>3.6. Echtheid</b>	<b>33</b>
<b>3.7. Prioriteren om te kunnen balanceren.</b>	<b>34</b>
<b>3.8. Architetcuurdomeinen en het raamwerk</b>	<b>35</b>
<b>4. De IT Specialist: duurzaamheid in uitvoering.</b>	<b>37</b>
<b>4.1. Het Manifesto van de Duurzame Specialist</b>	<b>38</b>
<b>4.2. Verbreden of verdiepen?</b>	<b>40</b>
<b>4.3. IT Specialist: De Programmeur</b>	<b>41</b>
<b>4.4. IT Specialist: De Data Scientist</b>	<b>42</b>
<b>4.5. IT specialist: infrastructuur specialist</b>	<b>44</b>
<b>5. De product owner: De bewaker van duurzaamheid</b>	<b>47</b>
<b>5.1. Brain Freeze: Complexiteit in de context van de economische situatie</b>	<b>47</b>
<b>5.2. Efficient Inzet van teams</b>	<b>51</b>

6. De chef	53
7. De Medewerker	55
<b>Responsible datacenter</b>	<b>57</b>
8. De systemen koel houden	58
8.1. Lucht gebaseerde koeling	58
8.2. Nadelen van lucht gekoelde oplossingen	59
8.3. Vloeistof gebaseerde koeling	60
8.4. Nadelen van vloeistof gekoelde oplossingen	61
8.5. Het kiezen van de juiste oplossing	61
9. Power Supply	62
9.1. Zonder stroom geen datacenter	63
9.2. Datacenter gebouwen	63
10. Datacenters en hyperscalers	65
10.1. Hoe sustainable is Cloud?	66
11. ODC als overheid strategie	68
11.1. Technologie platform	69
11.2. Data platform	69
11.3. Services platform	69
12. Code of conduct responsible datacenters	71
12.1. Responsible datacenter & responsible infrastructure	71
13. Responsible Datacenter Principles	72
<b>Responsible infrastructure</b>	<b>74</b>
14. De eerste aspecten	75
14.1. Omdenken	75
14.2. Schaarste versus overvloed	76
14.3. Energie zuinige systemen	80
15. Werklast Gedreven DataCenters	83
15.1. Systemen	83
15.2. Werklasten	86
15.3. De werkplek	90
16. Nog meer aspecten	91
16.1. Hoog beschikbaar als standaard	91
16.2. Opslag	92
16.3. Innovatie: 2NM (nano meter) technologie	92
16.4. Technology push	93
16.5. Duurzame processen	93
16.6. Systemen afschrijven	95

<b>17.</b> Een dagje baas van de ODC's	96
<b>17.1.</b> werklast gedreven inrichting	96
<b>17.2.</b> Nu met de duurzaamheid bril op:	97
<b>18.</b> Responsible Infrastructure Principles	99
<b>Responsible code</b>	<b>101</b>
<b>19.</b> Green Coding	102
<b>19.1.</b> Programmeer operating modellen	102
<b>19.2.</b> Software ontwikkeling	104
<b>20.</b> Architectuur	106
<b>21.</b> Design	108
<b>21.1.</b> Ontwerppatronen	108
<b>21.2.</b> Opsparen	109
<b>21.3.</b> Hergebruik	110
<b>21.4.</b> Netjes Organiseren	110
<b>21.5.</b> Opruimen	111
<b>21.6.</b> Anti-Patterns	112
<b>22.</b> Coderen	114
<b>22.1.</b> Programeertalen	114
<b>22.2.</b> Low Code en No code platforms	114
<b>22.3.</b> Definieren van Variabelen	116
<b>22.4.</b> De software ontwikkelstraat	116
<b>23.</b> Productie	119
<b>23.1.</b> Meten is weten	119
<b>24.</b> Kunstmatige Intelligentie Modellen	121
<b>24.1.</b> Model Training Services	121
<b>24.2.</b> Real-Time Geïntegreerde Analytics	122
<b>25.</b> Responsible Code Principles	123
<b>Responsible Data usage</b>	<b>126</b>
<b>26.</b> Inleiding	126
<b>26.1.</b> Verzamelwoede	128
<b>27.</b> Een duurzame data strategie als startpunt	130
<b>28.</b> De gereedschapskist	134
<b>28.1.</b> Verantwoorde technische tools	134
<b>28.2.</b> Aandachtspunten voor technische tools	135
<b>29.</b> Verantwoord Inzetten van Data ten behoeve van AI	139
<b>29.1.</b> Transparantie in Data Transformaties	140

<b>29.2.</b> Duurzame Trainingsdata	140
<b>29.3.</b> Explainability	140
<b>29.4.</b> Fairness	141
<b>30.</b> Operationalisatie	142
<b>31.</b> Responsible Data Usage Principles	143
<b>Responsible systems</b>	<b>145</b>
<b>32.</b> Raamwerk Responsible systems	147
<b>32.1.</b> De complexiteit van artificial intelligence	149
<b>33.</b> Aanpak voor het implementeren van betrouwbare AI systemen	
151	
<b>33.1.</b> idee	152
<b>33.2.</b> Bouw	155
<b>33.3.</b> Realisatie	157
<b>33.4.</b> Operatie	159
<b>34.</b> Cultuur en Skills	160
<b>35.</b> Responsible Systems Principles	162
<b>Responsible impact</b>	<b>164</b>
<b>36.</b> Een voorbeeld uit ons dagelijks werk	166
<b>37.</b> Een milieu management systeem (EMS): ISO 14000 normen	168
<b>38.</b> Bitcoin Mining	170
<b>39.</b> Sustainability Garage	171
<b>Conclusies</b>	<b>172</b>
<b>40.</b> De volgende stap	172
<b>41.</b> Saarinen & IBM als bron van inspiratie	174
Afkortingen	176
Bijlage B: Database Management Systemen	181
Over de schrijvers	184

# INTRODUCTIE VAN HET RAAMWERK

## 1. INLEIDING

**E**en vraag over duurzaamheid: Waarom willen bedrijven en organisaties verduurzamen? Is dit omdat ze hiermee kunnen bezuinigen op energiekosten of is dit omdat men hiermee het imago van het bedrijf of de organisatie kan oppoetsen? Of.... is dit omdat we **echt willen** verduurzamen omdat we inzien dat de huidige manier van invulling van onze bedrijfsvoeringen op de lange termijn niet houdbaar is? Onze huidige bedrijfsvoering is vooral gericht op efficiency en (economische) groei: meer, sneller, beter, mooier, groter, en ga zo maar door. Is dit op de lange termijn wel verantwoord en duurzaam?

Het zit veelal in de argumentatie die wordt gebruikt als we een duurzaamheid strategie uitgelegd krijgen door een bedrijf of organisatie. "Wij willen bezuinigen op energiegebruik" is een prima statement, maar de vraag is natuurlijk 'WAAROM?!'. Als het alleen gaat om kostenverlaging en omwille van de winstmarges en niet bezuinigen omdat het bedrijf dit **echt wil** dan is dit niet de juiste motivatie voor dit statement. Verduurzamen moet als een DNA patroon in organisaties en bedrijven zitten of komen. Daar is een behoorlijke mutatie van het huidige DNA (lees cultuur) voor nodig. Wellicht ietwat zware kost voor een inleiding maar het zet hopelijk onze gedachten tav verduurzaming op scherp.

Het goede is dat je vraag en aanbod op het gebied van duurzaamheid ziet groeien. Overal om je heen zie je tekenen van duurzaamheid ontstaan. In ons dagelijks leven wordt de roep om duurzaam gedrag en duurzame producten steeds groter en zien wij een bewuster gedrag bij zowel de consument als de producent ontstaan. Dit zal in de komende periode alleen maar toenemen waardoor met name bedrijven, hun bedrijfsvoering en producten

en diensten steeds meer een duurzaam karakter gaan krijgen. Je kunt je als bedrijf niet meer veroorloven om anti-duurzaam te zijn, en je kunt als consument het aanbod ook niet meer negeren.



*Duurzaamheid wordt steeds meer waarneembaar*

Een belangrijk onderdeel van de bedrijfsvoering is natuurlijk technologie. Technologie kan een grote impact hebben op de duurzaamheid van je organisatie. Duurzame technologie is een kreet die je overal om je heen hoort en tech bedrijven staan vooraan te dringen om aan te geven dat hun technologie bijdraagt aan duurzame oplossingen.

In dit boekje gaan wij in op de vraag of duurzame technologie echt bestaat en als dat zo is hoe kun je duurzaamheid onderdeel maken van je strategie?

Technologie is een wezenlijk onderdeel van ons bestaan. Wij gaan niet uitleggen wat de laatste twintig jaar ons heeft gebracht aan

technologische vernieuwingen maar we doen wel even een uitstapje naar de vorige eeuw. Midden in de twintigste eeuw zo een beetje. Technologie wordt enorm snel ontwikkeld en vooral ten dienste van de mens. Huishoudelijk apparatuur komt in grote getallen op de markt, stereo en televisie wordt toegankelijk voor iedereen en op deze manier ontwikkelen wij razendsnel een consumentenmarkt.

Een reclamebeeld uit de jaren zeventig waarbij een pijp rokende vader in zijn luie fauteuil zit, met het gezin gezellig om hem heen, waarbij hij met een afstandsbediening de televisie bedient kunnen de meeste van ons zich nog wel voor de geest halen. Voor de nieuwkomers onder ons (generatie Y&Z) is dit plaatje minder bekend maar er zijn legio voorbeelden die hen wel aanspreekt (CD speler, walkman, PC, smartphone, etc.).

Dit ideale plaatje zegt wel heel veel over hoe technologie een duidelijke stempel zet op onze beleveniswereld en hoe wij "het gemak dient de mens" graag adopteren. Het gemak gaat vaak ten koste 'van heel veel': Ons sociaal gedrag (geen goed gesprek in de trein maar staren naar Whatsapp), goedkope productie (gaat gepaard met afval en CO<sub>2</sub> uitstoot), wegwerpproducten (we repareren niet meer), plunderen van schaarse bronnen, en ga zo maar door.

Sinds de introductie van computer technologie in ons dagelijks bestaan is de automatiseringsdrang enorm toegenomen en is er eigenlijk geen grens aan deze drang te bekennen. Aan de andere kant waar we IT kunnen inzetten als verduurzamende versneller zijn we terughoudend als we kijken naar de 150.000.000 blauwe enveloppen die per jaar worden verstuurd door de Belastingdienst. Sinds 2014 heeft men plannen om het aantal blauwe enveloppen te verminderen door mensen te informeren middels de Berichten Box. Dit is een langzaam proces en het lijkt er bijna op dat men wacht bij de Belastingdienst tot de laatste digitale adem heeft uitgeblazen.

Dit is een voorbeeld uit de Nederlandse Overheid. Dat is het domein waar wij als architecten / schrijvers inzitten. Dat betekent niet dat de voorbeelden die wij geven niet gelden voor andere bedrijfstakken.

Met de milieuvraagstukken die de wereld beheersen en de energie crisis die wij in 2022 over ons heen hebben gekregen worden wij

wel gedwongen om ons te bezinnen op ons gedrag, zowel consument als producent, zowel burger als overheid.

Wij zijn er van overtuigd dat duurzame IT maar één ding kan betekenen: IT waar het moet en niet omdat het kan! Wat bedoelen we hiermee?

Neem een eenvoudig voorbeeld als digitale nieuwsbrieven, blogs, vlogs en andere elogs. Het is zo eenvoudig om je hierop te abonneren maar je moet wel heel erg savvy zijn om dit soort abonnementen op te zeggen. Twintig keer klikken, niet weten waar je op moet klikken en dan met een zucht denken "ach laat ook maar".

Dit is een voorbeeld van IT omdat het kan. Een simpele manier om een boodschap, aanbod of wat dan ook direct aan de man te brengen en de verantwoordelijkheid ook direct bij de ontvanger te leggen. Er bestaat nog niet zoiets als een digitale sticker op de mailbox met: NEE, geen ongedresseerd reclamedrukwerk, NEE, geen huis aan huis bladen. Het feit dat je de blogs, vlogs of andere elogs toch niet leest verandert hier niks aan.



En nu hoor ik je denken "het is maar een berichtje". Dat is precies het probleem. Op deze manier denken zit verankerd in de cultuur. Mijn opa zag een dubbeltje (dat is een muntje van 10 cent) op straat. Hij pakte het op, stopte het zorgvuldig weg in zijn 'knip' (zo noemde je dat) en zei: "zo dat is het begin van een miljoen".

Wij zien een overheid IT bedrijven als ware het commerciële bedrijven. Door IT in te zetten als onderscheidend vermogen zien wij sommige overheidsinstanties concurreren in plaats van maximaal te samenwerken. Het credo "de burger staat centraal" wordt vaak gebruikt als excus om de noodzaak van een bepaald IT project te onderbouwen terwijl er in veel gevallen het alleen om

technologische vernieuwing gaat. Daarnaast is het ook een excus om een nieuw technisch snufje te introduceren. Denk bijvoorbeeld aan een mobiele app die gebouwd moet worden om uitkeringsaanvraag te doen. Een voorbeeld van 'IT omdat het kan en niet omdat het moet'.

De burger centraal zou betekenen dat het aanbod wordt ontwikkeld op basis van de wens cq eis van de burger. Dat zou een mooie gedachte zijn ware het niet dat de wet bepaalt hoe een burger zich moet gedragen en zich moet conformeren aan de wet!

Een gedachte waarbij de overheid zich volledig inzet op een maximale informatiepositie van de burger en een IT dienstverlening baseert op deze informatiepositie levert best wel een wenkend perspectief (en daardoor duurzaam).

Hierbij zijn duurzaamheid aspecten als transparantie, voorkomen van fouten, beperken fysieke interactie met de overheid van belang en zal de privacy van burgers verbeteren.

Uiteraard kunnen wij organisaties en bedrijven niet veranderen door een boekje te schrijven maar we kunnen wel proberen om op basis van een aantal duurzaamheid principes te helpen met eenvoudige ingrepen om kleine stappen te maken die er toe bijdragen om onze IT middelen op verantwoorde wijze in te zetten. Hiervoor hebben wij het raamwerk van 'Responsible Computing' (RC) gebruikt en hanteren wij de domeinen van RC om een verdieping te maken op het gebied van verduurzaming.

De zes domeinen van Responsible Computing zijn:

1. Responsible Datacenters
2. Responsible Infrastructure
3. Responsible Code
4. Responsible Data usage
5. Responsible Systems
6. Responsible Impact

Schrijven over dit onderwerp in het licht van IT oplossingen is best lastig. Wij hebben ons best gedaan om in ieder geval het denkproces in gang te zetten. Er stromen nog heel veel bits door de cat 6 kabel voordat ieder bedrijf duurzaamheid als leidend principe heeft geadopteerd!

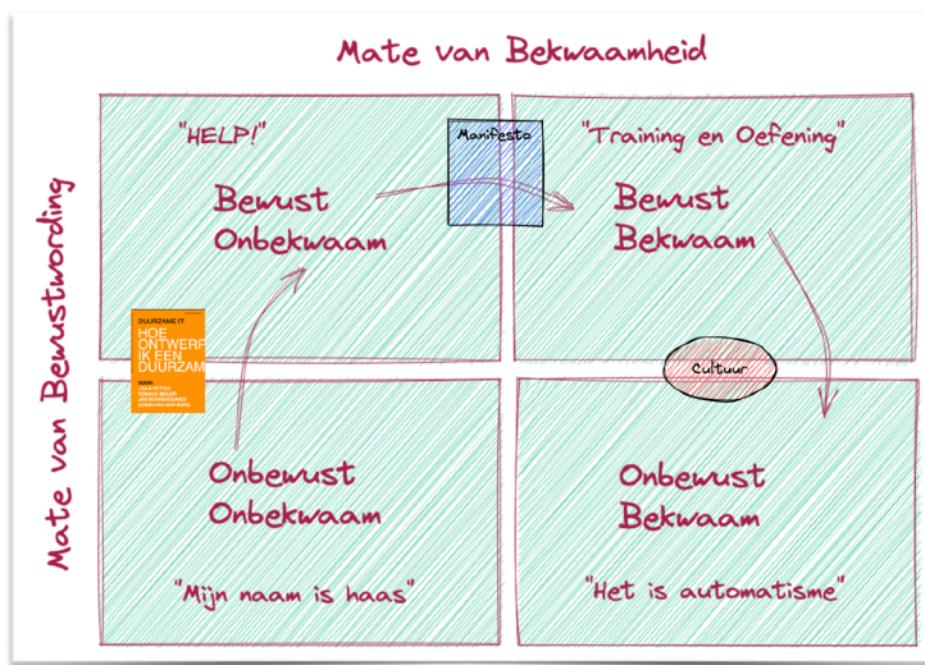
Ook voor dit boekje geldt dat wij geen experts zijn op het gebied van verduurzaming vraagstukken (maar wel op het gebied van IT!).

Door ons te verdiepen in de materie omdat klanten dit van ons vragen en soms eisen hebben wij wel een mening over dit onderwerp ontwikkeld. Wij hopen dat deze zienswijze gaat helpen bij mogelijke aanpakken en bij de gehele digitale transformatie waarbij "**sustainable by design**" wordt geadopteerd als hoofdprincipe.

## 1.1. VAN ONBEWUST ONBEKWAAM NAAR ONBEWUST BEKWAAAM

We spraken over het DNA (de cultuur) van een organisatie. Wat we hiermee bedoelen zijn gedragspatronen, identiteit. Als je over straat loopt en je eet een Mars dan doe je de verpakking terug in je zak om het later in de afvalbak te gooien. Daar denk je niet over na. Dat zit erin. Je vraagt je niet af waarom. Een mooi voorbeeld is dat van de Japanners, als zij een bioscoop of stadion verlaten nemen ze hun rommel mee en ziet het er weer keurig uit. Deze ingeslepen patronen kosten geen energie. Het is een vanzelfsprekendheid. Je bent dan onbewust bekwaam.

Als we het hebben over onbewust onbekwaam dan gaat het over dingen die we (nog) niet weten of onvoldoende realiseren. Met dit boekje willen we je bewust maken van de impact van IT op duurzaamheid en de impact van duurzaamheid op IT. Dat doet een beetje zeer, want het is soms heerlijk om van niets te weten. Maar daar willen we het niet bij laten want we willen je ook handvatten



(principes) geven om iets te doen om IT juist op een duurzame manier in te zetten. En we hopen natuurlijk dat het een automatisme wordt. Iets waar je niet meer over nadenkt, maar wat volkomen logisch is. Dan wordt het onderdeel van je DNA.

Een paar voorbeelden:

**Onbewust onbekwaam:** Te hard rijden doen we allemaal want de gaspedaal kan nog verder naar beneden. En tijd is geld! maar wat is het effect op het milieu? Wij weten zeker dat je daar niet over nadenkt op het moment dat je even gas bij geeft om op tijd op je afspraak te komen (de hoeveelheid brandstof die je extra gebruikt is evenredig aan de extra hoeveelheid CO<sub>2</sub> uitstoot).

De cloud is ook zo'n verleider om ongeremd IT resources te gebruiken. Het is goedkoop het is voorhanden dus gebruiken.....Duurzaamheid is hun probleem toch?

**Bewust onbekwaam:** De rekening van de cloud provider ligt op de mat. Hoe kan het nou dat het zo duur is? Moeten wij al bedrijf geen richtlijnen maken hoe om te gaan met IT resources? Het bedrijf begrijpt dat het noodzakelijk wordt om bewustwording te introduceren om bekwamer gebruik van resources te stimuleren.

**Bewust bekwaam:** Ontwerpers en ontwikkelaars zijn gehouden aan duurzaamheidsprincipes waardoor er geen overmatig gebruik wordt gemaakt van IT resources maar er bewust wordt omgegaan met middelen vanuit een schaarsheid principe. De kwaliteitsborging wordt geïmplementeerd in processen om de duurzaamheid van IT resources te garanderen.

**Onbewust bekwaam.** Iedereen in het bedrijf houdt zich onbewust aan duurzaamheid principes omdat het normaal is. Men is intrinsiek gemotiveerd om zich aan deze principes te houden. Het zit in het DNA van het bedrijf!

## 1.2. PROMOVEREN

Als voorbeeld hebben we in het bekwaamheid model de voorwaarden voor promotie door het model gezet.

- Van onbewust onbekwaam naar bewust onbekwaam: Lees dit boekje.
- Van bewust onbekwaam naar bewust bekwaam: implementeer een gedragsmanifest en principes. Voorbeelden hiervan beschrijven we verderop in dit boekje.

- Van bewust bekwaam naar onbewust bekwaam: onderga een cultuur verandering. Op het moment dat je elkaar binnen een organisatie kan aanspreken op het maken van duurzame keuzes ontstaat een zelfcorrigerend vermogen. Want we weten vaak wel wat goede keuzes zijn, maar we zijn gemakzuchtig en vinden het lastig en te veel energie kosten om ons gedrag te veranderen.

## 1.3. GETALLEN

Wat zeggen getallen? Door dit boekje heen gebruiken we regelmatig getallen. Als het gaat om duurzaamheid dan worden er allerlei artikelen gepubliceerd en in die artikelen vind je getallen terug vanuit een bepaalde context. Het bedrijf “duurzaam\_tot\_op\_het\_bot” zal het dik aanzetten. De “vervuiler tegen wil en dank” zal het juist rooskleurig voorstellen. De getallen die we geven zijn vooral bedoeld als indicatie. In jouw organisatie, op jouw gebied zal het beter zijn omdat je binnen een voor jouw bekende context werkt. Jij bent niet in staat die context weg te nemen, maar anderen in de organisatie wellicht wel.

Een voorbeeld. Vanuit de bedrijfsvoering wordt gesteld dat de omgevingen allemaal 7 x 24 uur hoog beschikbaar moeten worden uitgevoerd. Als infrastructuur architect bouw je een oplossing die hieraan voldoet door alles dubbel uit te voeren. Dat is je opdracht. Een bestuurder in de organisatie kan de business echter uitdagen op de realiteit van deze eis. Hoe erg is het als de Internet site een half uur niet beschikbaar is?

Het gaat ons erom dat we een gevoel krijgen van de omvang van de problematiek, maar ook dat we een gevoel te krijgen van mogelijke verbeteringen. Laat je niet verleiden tot cijfermatige details, maar pak de essentie op en ga daarmee aan de slag.

## 2. RESPONSIBLE COMPUTING IS DUURZAAM!

In deze editie van het oranje boekje concentreren wij ons op duurzame IT. Niet omdat het een trend is, of het beter zou “verkopen” (ook al is dit boekje gratis), maar omdat het nodig is.

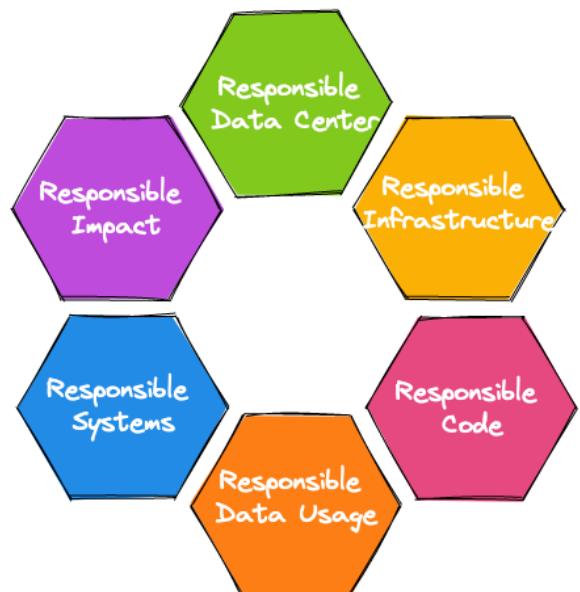
We hebben de kaders van Responsible Computing als uitgangspunt genomen. Dit raamwerk helpt ons de juiste focus te leggen op de onderwerpen mensen, processen en technologie in context van IT.

Hoewel een organisatie in staat zou moeten zijn om de Sustainable Development Goals (SDG's) van de Verenigde Naties te ondersteunen, vinden we dit een te groot en complex onderwerp en daarom beperken wij ons tot de reikwijdte van onze IT expertise. Anders wordt dit al snel de oceaan proberen te koken en we willen ons beperken tot maximaal 180 - 190 pagina's (is ook duurzaam).

Responsible Computing (RC) is een initiatief van een aantal bedrijven waaronder IBM. Dit raamwerk is gedoneerd aan de Object Management Group (OMG) en meer hierover te vinden op hun website<sup>1</sup>.

Responsible computing bevat 6 domeinen:

1. Responsible Datacenters
2. Responsible Infrastructure
3. Responsible Code
4. Responsible Data usage
5. Responsible Systems
6. Responsible Impact



*Responsible Computing Domeinen*

<sup>1</sup> <https://responsiblecomputing.net/>

Al deze domeinen zijn belangrijk, want veel organisaties zijn IT-intensieve organisaties. We bespreken hoe deze organisaties op elk van die domeinen hun verantwoordelijkheid kunnen nemen en schaarse middelen op een verantwoorde manier kunnen inzetten.

De eerste drie domeinen zijn technisch. De focus zal liggen op een optimaal gebruik van schaarse middelen. Dat is dus duurzaam! De andere drie domeinen bespreken de mogelijkheden om data-gedreven architectuur zodanig toe te passen dat het een positieve bijdrage levert aan de samenleving. Wil je meer weten over data-gedreven over het opzetten van een data-gedreven organisatie dan raden we je aan om het eerste oranje boekje te lezen: "How to: Develop a Blueprint for a Data-Driven Enterprise Architectuur".

In dit hoofdstuk geven we een kort overzicht van de zes domeinen en later gaan we hier in meer detail op in.

## **2.1. RESPONSIBLE DATACENTER**

Als we praten over het datacenter dan hebben we het over het fysieke gebouw en de faciliteiten die je nodig hebt om een datacenter te huisvesten. Denk aan stroom, koeling, noodstroom voorzienig, bekabeling enzovoort.

Datacenter providers meten hun duurzaamheid aan de hand van Power Usage Effectiveness (PUE) en Water Usage Effectiveness (WUE). PUE is de verhouding tussen de gebruikte stroom door de computer apparatuur en het hele datacenter. WUE is de hoeveelheid water gedeeld door de hoeveelheid gebruikte stroom van de computer apparatuur. Het meten van de efficiëntie van een datacenter is heel goed te doen en de meeste grote datacenters doen dit. Wat meestal niet gebeurt is de berekening van de kosten van de bouw van een DC. Transportkosten van bouwmateriaal en de materialen zelf (veelal beton) zorgen voor veel CO<sub>2</sub> uitstoot.

## **2.2. RESPONSIBLE INFRASTRUCTURE**

IT infrastructuur bestaat uit computers, opslag media en netwerkapparatuur. Het zijn de applicaties die de eisen stellen aan deze infrastructuur en uiteindelijk bepalen hoe groot, hoe zwaar en hoe snel de infrastructuur moet zijn.

In 1969 stuurden we de Apollo 11 inclusief bemanning naar de maan en zorgden we voor een veilige terugreis. Dit alles met behulp van een computer die zorgde voor de berekening van de juiste koers en het bijstellen van de koers. Deze computer bestond uit een

processor met een kloksnelheid van 12 microseconden en 72 Kb geheugen. Dit systeem is vergelijkbaar met de kracht van een Commodore 64!

Een gemiddelde smartphone vandaag heeft meer computer power dan heel NASA in 1969 bij elkaar kon brengen! En toch hebben we nog steeds te weinig computer power!

## 2.3. RESPONSIBLE CODE

Om een applicaties efficiënt te kunnen draaien is efficiënte infrastructuur vereist die weer draait in een efficiënt datacenter. Maar de code die geschreven wordt om de applicatie zijn werk te laten doen heeft een nog grotere impact. Met het efficiënt bouwen van een datacenter kunnen we 10-20% besparen, met efficiëntere infrastructuur kunnen we 60-90% besparen maar met de juiste code kunnen we soms wel tot een factor 2 besparen. Of moeten we het omkeren? Als je verkeerd programmeert dan kan het zo maar 2x zoveel computer power kosten. Rechtsom of linksom, responsible code is een belangrijk domein dat iets moet aangeven of code nu duurzaam is of niet.

Binnen dat domein zijn AI en Big Data nog eens grote stroomverbruikers.

Veelgebruikte ontwikkeltalen binnen AI zoals Python en R zijn typische voorbeelden van zeer "dure" talen (voor zover het CPU-gebruik wordt beschouwd). Voldoende dus om hier de tanden eens in te zetten. Hoe zorgen we nu voor duurzame code? Duurzame code, ook wel "green coding" genoemd deed enkele van ons denken aan de oude mainframe terminals?!



Green Screen

## 2.4. RESPONSIBLE DATA USAGE

In ons data-gedreven project zorgen we ervoor dat de data veilig is. Wij respecteren de gegevens van de klant, burger of ander bedrijf. We beschrijven duidelijk hoe we het gebruiken, en we gebruiken het alleen voor het doel waarvoor de toestemming is gegeven.

We weten precies waar welke data is en wat het is. We hebben ook een goede governance ingericht voor onze data en data gebruik. We weten ook waar de gegevens vandaan komen, hoe de gegevens zijn samengesteld en wie wanneer veranderingen heeft aangebracht aan de data.

## **2.5. RESPONSIBLE SYSTEMS**

We gebruiken data in onze systemen om beslissingen te nemen. We zijn transparant over de regels en algoritme die we gebruiken om tot die beslissingen te komen. Wanneer we gebruik maken van kunstmatige intelligentie zijn we transparant over de wijze waarop modellen tot stand komen en welke algoritme en gegevens daarvoor gebruikt zijn. We implementeren ModelOps om bias en drift te voorkomen en we archiveren en bijbehorende gegevens om aan te kunnen tonen dat wij deze modellen rechtmatig en ethisch hebben gebruikt.

## **2.6. RESPONSIBLE IMPACT**

Stel je voor: Wij bouwen systemen met de beste bedoelingen voor de samenleving. We maken de wereld een beetje beter met onze systemen. We kunnen bijvoorbeeld een systeem bouwen dat een lijst met beschikbare woningen vergelijkt met een lijst met mensen die er naar op zoek zijn naar een woning. Of we bouwen een systeem dat agressie op straat detecteert en we zorgen ervoor dat de politie op tijd aanwezig is om escalatie te voorkomen.

Hoe zal het systeem dat jij ontwerpt en bouwt het verschil maken voor de samenleving? Welke van de United Nations SDG's (UNSDG) wordt door jou project aangepakt?

## **2.7. RESPONSIBLE COMPUTING PRINCIPLES**

Naast deze zes domeinen kent het raamwerk ook zes principes. Deze principes zijn eerder waarden dan architectuur principes zoals we die kennen binnen de IT professie. Omdat wij in dit boekje een heel aantal architectuur principes definiëren zullen we deze zes principes van Responsible Computing voortaan waarden noemen. Deze waarden kun je grofweg indelen in waarden die terugkomen als kwaliteitseisen (randvoorwaarden) voor het systeem en waarden die vooral gaan over de cultuur. Denk bij dit laatste vooral ook even terug aan het raamwerk van onbewust onbekwaam naar onbewust bekwaam. Openheid, authenticiteit en

accountability moeten culturele waarden zijn van een duurzame organisatie.

De volgende waarden zijn gedefinieerd:

1. **Duurzaamheid.** Duurzaamheid in het kader van de UNSDG is heel breed. Bij duurzaamheid gaan wij daarom vooral uit van de milieu aspecten, met de nadruk op de energiegebruik en de daar aangekoppelde CO<sub>2</sub> uitstoot.
2. **Inclusiviteit.** Binnen de Nederlandse overheid beter bekend onder de naam: "De menselijke maat". Het gaat erom dat systemen door iedereen gebruikt kunnen worden en voor iedereen zijn. Inclusiviteit is ook aandacht voor de minderheidsgroeperingen, het individu.
3. **Circulariteit.** Binnen de IT gaat het dan over herbruikbaarheid en modulariteit. Denk bijvoorbeeld aan het hergebruik van kostbare metalen die in de elektronica van de computer zitten zoals goud. Deze materialen zijn bekend als 3TG (Tantalum, Tin, Tungsten en Gold) ofwel Tantaal, Tin, Wolfraam en Goud. Tantaal is sterk en buigzaam metaal met een hoge smeltemperatuur en werd ook gebruikt voor gloeilampen. Nu wordt het gebruikt voor o.a. condensatoren, een belangrijke component in de elektronica-industrie.
4. **Openheid.** Openheid, authenticiteit en het nemen van verantwoordelijkheid gaan verder dan randvoorwaarden die we aan onze systemen stellen. Het gaat hier over het gedrag van een organisatie die vervolgens weer bestaat uit personen. Openheid, transparantie laat zien waar je staat ten aanzien van het verantwoord computergebruik. Het is een kwetsbare opstelling die niet alleen maar laat zien wat er goed gaat, maar ook waar de problemen zijn. Het vraagt een leerbare openstelling om je te laten adviseren door anderen
5. **Authentiek.** Is het echt? Meen je het? Is Responsible Computing meer dan alleen een verplichting die voortkomt uit een rationeel besef dat we iets moeten doen aan de opwarming van de aarde en het opraken van schaarse resources?
6. **Accountable.** Accountable laat zich slecht vertalen in het Nederlands omdat het hier gaat over meer dan verantwoordelijkheid nemen, n.l. over verantwoording afleggen. Het is veel meer eigenaar worden van het probleem en hier *aansprakelijk* voor zijn. In de beschrijving van de verschillende rollen binnen de IT staat accountability centraal. Ben jij bereid aansprakelijk te zijn voor het domein waar jij over gaat? Sta je op

als beslissingen ten koste gaan van de duurzaamheidsprincipes en winstmarges omhoog moeten ten koste van alles en iedereen?

## 2.8. AANSPRAKELIJK (ACCOUNTABILITY) GELDT VOOR IEDER RC DOMEIN

Wanneer we de waarden loslaten op de domeinen dan zien we dat niet alle waarden even relevant zijn voor alle domeinen. Hieronder een overzicht van de relevantie van de waarden gerelateerd aan de domeinen. Ongeacht het domein is “accountability”, aansprakelijk zijn, altijd relevant. In de hoofdstukken over de diverse rollen zoals de IT architect, de programmeur en de product owner benadrukken we de relevantie van accountability in die rollen.

Domeinen Principes	Data Center	Infra-structuur	Code	Data Gebruik	Systems	Impact
Duurzaamheid ihkv energie	V	V	V	V	V	V
Inclusiviteit			V	V	V	V
Circulariteit	V	V	V			V
Openheid			V		V	V
Authentiek				V	V	V
Accountable	V	V	V	V	V	V

Duurzaamheid, inclusiviteit en circulariteit zijn heel goed meetbaar te maken en kunnen worden gebruikt om RC domeinen van verschillende bedrijven te vergelijken .

Openheid, authenticiteit en accountability zijn minder makkelijk meetbaar te maken omdat eigenlijk hiermee eigenlijk het duurzaamheidsDNA van het bedrijf wordt bedoeld.

Als we het hebben over DNA dan hebben we het over mensen. Dat klopt want de DNA wordt door mensen gecreëerd middels principes en ethische normen! Met andere woorden het is mensenwerk!

Wij hanteren in (business en IT) architectuur zogenoamde principes. Deze sets aan principes vormen de kaders voor de enterprise architectuur. Mochten er tegenstellingen bij het hanteren van deze

principes ontstaan dan gelden **de 'superprincipes'**. Deze overstijgen alle andere principes.' Net zoals '**Secure by Design**' zou je ook de superprincipe '**Sustainable by Design**' moeten implementeren.

# PEOPLE

In dit hoofdstuk bespreken we de rol van de IT Architect, de programmeur en de product owner of product manager en de strategische rol van de Chief Sustainability Officer (CSO). De IT architect hebben we gesplitst in de drie bekende rollen: business architect, applicatie architect en infrastructuur architect. De IT specialist hebben we ook gesplitst in drie rollen: de programmeur, de data scientist en de infrastructuur specialist.

Het onderstaande plaatje is slechts een model dat weergeeft waar de focus ligt voor de verschillende rollen. Het is bedoeld om de discussie te starten betreffende de zwaartepunten van de aansprakelijkheid van de verschillende rollen in onderstaand diagram. Dit kan helpen bij het opstellen van de profielen van medewerkers en hen bewust te maken van het belang van duurzaamheid principes.

Domeinen Rollen	Data Center	Infra-structuur	Code	Data Gebruik	Systems	Impact
Architect: Business						
Architect: Applicatie						
Architect: Infrastructuur						
IT Specialist: Programmeur						
IT Specialist: Data Scientist						
IT Specialist: Infrastructuur						
Product Owner						
CSO						

Deze verschillende rollen bepalen in grote mate de implementatie en realisatie van de duurzaamheid aspecten mbt Openheid,

Authenticiteit en Accountability. Zij moeten dus het DNA al hebben om deze vervolgens als een olievlek in de organisatie te verspreiden. Jammer dat Openheid niet met een 'A' begint dan hadden we er 3A van kunnen maken. De 3A's van het DNA, mooie titel.

# 3. DE IT ARCHITECT: ONTWERPER VAN DUURZAAMHEID

**D**it boek is geschreven door IT-architecten. Diegene die ons verdacht vinden zullen denken dat we voor eigen parochie spreken en dan bespreken we deze rol ook nog eens als eerste in dit hoofdstuk! Je zou ook kunnen denken dat wij als architecten aansprakelijkheid willen accepteren in het ontwerpen van duurzame IT-systeem.

Het is maar hoe je ernaar kijkt, maar wij vinden dat architecten een sleutelpositie hebben als het gaat om het adresseren en ontwerpen van duurzaamheid als kwaliteitsaspect in IT-systeem.

Voor diegene die geen IT-architect zijn en voor die architecten die moeite hebben om hun rol uit te leggen, wat is de rol van een IT-architect? De architect is verantwoordelijk voor een IT-systeem dat voldoet aan de specificaties van de opdrachtgever met in acht neming van “alle” eisen van andere stakeholders.

Wat bedoelen we hiermee? Een opdrachtgever wil een systeem laten bouwen dat moet voldoen aan een aantal functionele specificaties. Verder zal de opdrachtgever ook nog een aantal non-functional requirements hebben. Denk hierbij aan beschikbaarheid van het systeem of de snelheid. Daarnaast zijn er nog andere belanghebbende, bijvoorbeeld de beheer organisatie, security en een data privacy officer. De architect is de persoon die het geheel overziet en de balans zoekt tussen de eisen van de verschillende stakeholders. Die eisen kunnen tegenstrijdig zijn. Bijvoorbeeld het systeem moet eenvoudig te bedienen zijn maar ook veilig. De architect schrijft de mogelijke keuzes op en neemt een beslissing met een motivatie (de zgn. architectuur beslissing) en laat deze bekraftigen door een architectuurboard.

Een tegenstrijdigheid zit ook vaak in lange termijn doelstelling naar aanleiding van de bedrijfsstrategie en korte termijn doelstellingen van een project dat snel in de markt gezet moet worden. De architect heeft veel oog voor die lange termijn doelstellingen en

wordt daarom wel ervaren als een “pain in the ass” als er snel even iets gemaakt moet worden. Helaas zijn er heel wat situaties bekend van bedrijven die hadden gewild dat ze beter hadden geluisterd naar die lange termijndoelstelling.

Het middel dat gebruikt wordt om bedrijfsdoelstellingen te borgen zijn de zgn. architectuur principes. Deze principes zijn een afgeleide van de bedrijfsstrategie en zorgen ervoor dat het IT-systeem dat gerealiseerd wordt ook op langere termijn past in de doelstellingen van het bedrijf.

Waarschijnlijk voel je het nu wel aankomen. Duurzaamheid is onderdeel van de bedrijfsstrategie of zou dat moeten zijn, maar wij observeren dat er op project niveau niet of nauwelijks eisen worden gesteld mbt duurzaamheid aan het op te leveren product. Dus architect, maak je borst maar nat, de duurzaamheidsprincipes resulteren in nog meer eisen waar het systeem aan moet voldoen!

Als we bovenstaande taken koppelen aan de zes waarden die genoemd worden in het Responsible Computing raamwerk dan vraagt dat nog al wat van de instelling, de mindset, de betrokkenheid van de architect. De architect moet rekening houden met inclusiviteit, circulariteit, openheid en daarnaast authentiek zijn en aansprakelijkheid accepteren. Zijn of haar rol als trusted advisor is hiermee belangrijker dan ooit geworden.

Verder vereist het ontwerpen van een duurzaam systeem een brede kennis over de werking van software, hardware, netwerken en datacenters om ervoor te zorgen dat de totaaloplossing duurzaam is. Daarnaast moet het systeem voldoen aan wet- en regelgeving zodat er een systeem gemaakt wordt dat ook ethisch verantwoord is. Voor iedere rol, maar zeker voor die van de architect een ware uitdaging!

Het doel van dit hoofdstuk is om duurzaamheid concreet te maken voor de architect.

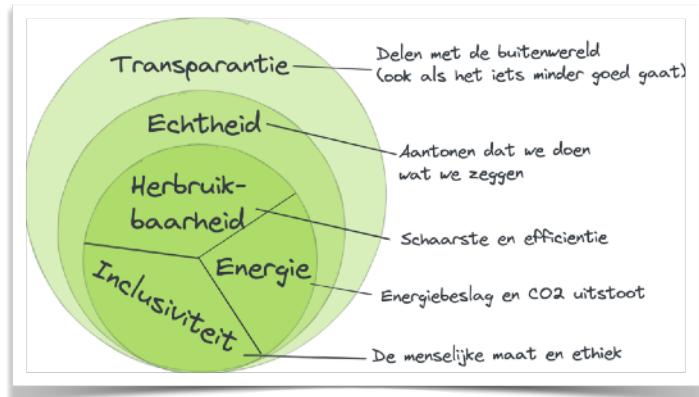
### **3.1. NON-FUNCTIONAL DUURZAAMHEID REQUIREMENTS**

Er zijn diverse standaarden als het gaat om het in kaart brengen van de eisen van een IT systeem. ISO9126 is één van de bekendere, maar deze zien wij niet vaak toegepast, waarschijnlijk om dat deze wat minder toegankelijk is.

IEEE 830-1993 geeft ook een lijst met non-functional requirements. Binnen jou bedrijf wordt er mogelijk een andere standaard gedefinieerd. Zo zien we bijvoorbeeld binnen de Nederlandse overheid het gebruik van NORA (Nederlandse Overheid Referentie Architectuur) met een eigen begrippenkader en definitie van kwaliteitseisen.

Binnen deze standaarden zijn diverse categorieën die impliciet duurzaamheidsprincipes ondersteunen. ISO9126 kent bijvoorbeeld usability als kwaliteitsaspect. Usability zou ook inclusiviteit moeten bevatten, of binnen de overheid beter bekend als **de menselijke maat**.

We hebben ervoor gekozen om de kwalitatieve duurzaamheidsaspecten los van een raamwerk te benoemen. Onze suggestie is om de aspecten die we noemen mee te nemen en een plek te geven binnen het raamwerk dat jouw organisatie gebruikt. We zien dat deze kwaliteitseisen een afgeleide zijn van de (principes) waarden die in onze optiek een duurzaam bedrijf zou moeten hebben. Verder hebben we niet de illusie compleet te zijn met deze lijst, maar wel bijna.



Vijf kwaliteitsaspecten in het kader van duurzaamheid

- Transparantie
- Energie
- Herbruikbaarheid
- Inclusiviteit
- Echtheid

## 3.2. TRANSPARANTIE

Bij transparantie gaat het erom dat wij van te voren bepalen *welke* informatie we *naar buiten moeten brengen* zodat onze stakeholders een eerlijk beeld krijgen van de duurzaamheid van ons gerealiseerde systeem. We kennen de uitdrukking: "Wie eerlijk is heeft niets te verbergen". Bij transparantie gaat het erom dat je

laat zien wat je stakeholders nodig hebben om te boordelen dat je hebt waargemaakt wat je beloofd hebt - of niet. Transparantie is een kwetsbare en leerbare opstelling. Waarom is het niet gelukt? Wat kunnen we ervan leren? Wat moet de volgende keer anders?

Het gaat daarbij niet alleen om het resultaat, maar ook de manier waarop. Stel je bouwt een energiezuinig stadion en je haalt deze doelstellingen, ben je dan transparant? Wat is de prijs die betaald is met de bouw?

Wanneer we een systeem ontwerpen moeten we transparantie meenemen en niet achteraf toevoegen. Een transparant systeem levert informatie aan de buitenwereld. Dit kan monitoring informatie zijn over het energiegebruik. Dit kunnen de datasets en algoritmen zijn die zijn toegepast in het systeem. Transparantie geldt ook voor de gebieden waar je nog niet bent maar waar je in de toekomst zou willen staan. Transparantie geldt ook voor de gebieden waar je niet transparant wilt of kunt zijn. **Wees transparant over waar je niet transparant over kan zijn!!??**

Transparantie zou kunnen leiden tot de volgende set aan requirements:

- Duidelijkheid waarvoor deze applicatie bedoeld is en liefst zonder bijbedoelingen. Als je op Google zoekt naar schoenen en je krijgt vervolgens allemaal ads over schoenen dan is zoeken op het Internet niet het (enige) doel van Google. De opzet van GDPR is om de gegevens van gebruikers te beschermen en alleen daarvoor te gebruiken waarvoor ze bedoeld zijn. Je geeft "consent" op het gebruik. We zien helaas veel toepassingen waar je consent moet geven, anders krijg je geen informatie of sites waar je persoonsgegevens achter moet laten om een document op te halen zodat ze je later kunnen spammen met diverse reclames.
- Meten is weten! Onderbrengen van monitoring data die gerelateerd is aan duurzaamheid. Er is vaak veel systeeminformatie op te halen over het gebruik van hardware en software.
- Een ander goed voorbeeld is het verzamelen van doorlooptijden van je toolchain. Er is een relatie te leggen tussen de duur van je compilatie en het testen en de efficiëntie van je applicatie.

- Ethisch gebruik van AI. Zoals beschreven in ons vorige oranje boekje kun je AI gebruiken om automatisch beslissingen te nemen die consequenties kunnen hebben op het leven van mensen. Je zult je moeten verantwoorden over het hoe en waarom van de keuzes. Hierover meer in het hoofdstuk over Responsible Systems.

### 3.3. ENERGIEVERBRUIK

Het mooie van IT is dat het op elektriciteit draait en dus in de operatie geen fossiele brandstoffen nodig heeft.

Toch is elektriciteit grotendeels evenredig met CO<sub>2</sub> uitstoot. Het grootste gedeelte van de energiecentrales in Nederland draait namelijk op gas. Nu kun je je datacenter op groen stroom draaien, maar uiteindelijk is van de opgewekte energie ruim 10% groen en de rest is grijs. Dus die 10% is schaars en moeten we uiteindelijk ook met z'n allen delen. Dus rechtsom of linksom, energie is een kostbaar iets waar je zuinig mee om moet gaan. De volgende requirements kunnen we hieraan toekennen:

Methaan + Zuurstof ==> Water + Kooldioxide  
(Aardgas)



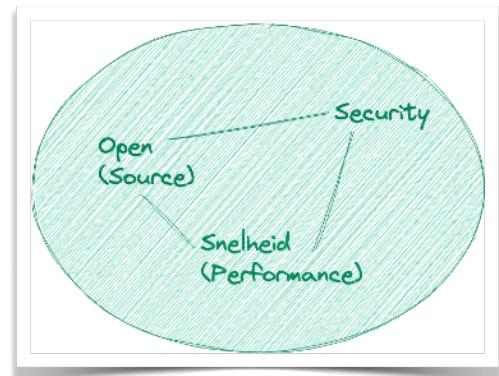
- Efficiëntie van de hardware. Of het nu compute, storage of netwerk betreft het is goed om te kijken naar de efficiëntie van de hardware die gekocht wordt. Het is echter zaak om je niet te beperken tot de specifieke hardware maar ook naar keuze van bv processor technologie of storage technologie. Hierover meer in het hoofdstuk Responsible Infrastructure.
- Productie van hardware. Ook de productie van hardware kost energie. Efficiëntie kan verhoogd worden door hergebruik. Zie de volgende paragraaf. Hoe en onder welke omstandigheden de productie van hardware plaats vindt is ook een duurzaamheidsaspect. We hebben het dan over de “supply-chain”. Daar gaan we in dit boekje verder niet op in, het is een complex onderwerp en uiterst relevant voor de CSO. We zien dat veel grote IT leveranciers zeer serieus zijn over de duurzaamheid van de hele keten. Je moet eenvoudigweg geen zaken doen met leveranciers die niet transparant (kunnen) zijn over hun keten.

- Efficiëntie van het datacenter. In een eigen datacenter is het mogelijk om het verbruik van het totale DC af te zetten tegen het gebruik van de hardware. De factor die daartussen zit noemen we PUE waarde. Hoe lager de PUE, hoe efficiënter het DC. Naast PUE kennen we ook de WUE. Dat gaat over het gebruik van water. Hierover meer in het hoofdstuk over Responsible Datacenters.
- Efficiëntie van code. De ene programmeertaal is de andere niet, er is een significant verschil in energieverbruik tussen de verschillende programmeertalen., Zie het hoofdstuk over Responsible Code waarin programmeertalen en patterns besproken worden.

Tot slot willen we hier nog schaalbaarheid noemen. Als het gaat over schaalbaarheid als non-functional requirement dan bedoelen we daarmee dat bij een fluctuerend aantal gebruikers het systeem dit aan kan door horizontaal of verticaal mee te schalen. Als het gaat om energieverbruik bedoelen we juist afschalen en uitzetten van systemen die minder of niet meer gebruikt worden.

### 3.4. HERBRUIKBAARHEID

De cyclische economie, wellicht beter bekend als het cradle 2 cradle (C2C) concept<sup>2</sup>. Het concept gaat ervoor om een product wat gebruikt wordt om te zetten in een nieuw product. Stel je eens voor dat een computersysteem dermate modulair is opgebouwd dat de kast, de voeding bekabeling hergebruikt kan worden. Dat het moederboard ook de volgende generaties CPU's aankan en dat de oude datacenter CPU's zeer geschikt zijn om te gebruiken in de automotive (die momenteel te maken hebben met een enorm chip tekort). Het lastige voor ons als architecten is dat wij die computer niet ontwerpen, want daar begint het. De computer moet ontworpen worden met herbruikbaarheid als basis.




---

<sup>2</sup> Cradle to Cradle: Remaking the Way We Make Things, Michael Braungart en William McDonough

Toch kunnen we bij de selectie van hardware minimaal kijken naar herbruikbaarheid daarvan en er zijn wel degelijk leveranciers die niet de hele computer vervangen maar uitsluitend de processoren.

- Kies infrastructuur die hergebruikt kan worden.
- Koop alleen van leveranciers die geleverde apparatuur weer innemen en verantwoord verwerken.
- Het opzetten van modulaire code. Code schrijven en produceren kost ook energie. Code die hergebruikt wordt hoeft niet meer gebouwd te worden. De besparing zou je bijvoorbeeld kunnen inzetten om de code te optimaliseren.

### 3.5. INCLUSIVITEIT

In het begin van het hoofdstuk hebben we het al even genoemd. We moeten rekening houden met de gebruikers van de systemen die we als architect ontwerpen. We houden wellicht nog rekening met mensen die een lichamelijke beperking hebben zoals mensen met een gehoorbeschadiging of slechtzienden, maar te vaak gaan we ervan uit dat de gebruiker comfortabel is met de computer.

Binnen de overheid hebben we te maken met een tegenstrijdigheid: Mensen die recht hebben en gebruik maken van diverse toeslagen kunnen dit aanvragen via Internet sites. Maar de mensen die gebruik maken van die toeslagen zijn veelal mensen die juist moeite hebben om om te gaan met deze moderne middelen. De overheid spreekt van *de menselijke maat*.

Inclusiviteit speelt ook een belangrijke rol bij de implementatie van geautomatiseerde processen waarin beslissingen genomen worden op basis van bedrijfsregels of AI modellen. Hoe gaan we om met de uitzonderlijke situatie die net niet binnen de regel valt? Hebben we voldoende data om ons model te trainen? Om iedereen gelijk te behandelen stelt de overheid wetten op. Die wetten zijn vaak heel goed voor het gemiddelde, maar niemand is gemiddeld, ook niet de mensen die we in het voorbeeld noemde die recht hebben op toeslagen.

### 3.6. ECHTHEID

Wanneer we een business case maken om een project te verantwoorden dan zien we over het algemeen twee dingen

gebeuren. Ten eerste is de business case rooskleuriger dan de werkelijkheid. We denken dat het nieuwe product dat we op de markt zetten 100.000 keer verkocht gaat worden. Meestal is het minder, soms meer. We denken dat de kosten om het product te maken 1 Mio is. Meestal is het meer, eigenlijk nooit minder. Het tweede dat we zien is dat een business case achteraf niet of nauwelijks gebruikt wordt om te controleren of onze berekening klopte. Gedane zaken nemen geen keer. Maar kunnen we er dan wel van leren?

Als het gaat om echtheid als non-functional requirements dan gaat het erom dat we aantonen dat we waarmaken wat we beloofd hebben. Een definitie voor echtheid in deze context is het op een *holistische manier aantonen door te meten*. Echtheid (authenticiteit) is een begrip dat juist in de context van duurzaamheid gebruikt wordt omdat het een cultuuraspect weergeeft. We moeten daarbij van te voren nadenken over de meetbaarheid van duurzaamheid om deze ook achteraf aan te tonen. Bijvoorbeeld

- We hebben gebruik gemaakt van een AI Model en we moeten aan kunnen tonen dat we de juiste data en algoritme gebruikt hebben.
- Iedere keer dat we in productie gaan met een nieuwe release gebruikt de applicatie 5% minder stroom. Dat betekend bijvoorbeeld dat bij Agile Development in de laatste sprint een analyse en optimalisatie wordt gedaan op stroomverbruik.
- Bij de Lifecycle management van onze storage moet het energieverbruik van de nieuwe storage 15% lager zijn.

Het kan zijn dat je ondanks het stellen van meetbare criteria het niet lukt om deze te halen. Met echtheid gaat het er vooral om dat je eerlijk rapporteert over deze doelstellingen en het niet mooier maakt dan het echt is. Echtheid en transparantie hebben veel met elkaar te maken. Waar transparantie aangeeft wat er gedeeld moet worden geeft echtheid de juistheid daarvan weer.

### **3.7. PRIORITEREN OM TE KUNNEN BALANCEREN.**

Als architect moet je keuzes maken. De non-functional requirements matrix is een hulpmiddel waarbij de requirements worden geprioriteerd. Je doet dit samen met al je stakeholders. Je geeft aan met een factor tussen 0 en 10 hoe belangrijk een requirement is. De prioritering helpt met het maken van keuzes. Interessant om te zien hoe hoog duurzaamheid scoort!

Het is niet altijd zo dat de non-functional eisen tegenover elkaar staan. Juist in het kader van duurzaamheid zien we drie non-functional eisen die elkaar juist versterken:

Open(source) systemen zijn modulair opgezet en het gevolg is dat de modules efficiënt zijn. Ze doen wat ze moeten doen. Dus de minimale code zorgt voor maximale performance en verhoogde duurzaamheid. Minimale systemen die alleen doen wat ze moeten doen zijn minder kwetsbaar. Een voorbeeld hiervan is container technologie. Docker is een bekende container technologie omgeving die zowel voorziet in applicatie “build” alsmede het draaien van de applicaties (“run”). Door de “build” en de “run” omgevingen uit elkaar te trekken krijg je twee kleinere omgevingen met een specifieke taak die efficiënter en veiliger zijn.

Een voorbeeld hiervan is Red Hat OpenShift. Red Hat heeft ervoor gekozen om niet compleet Docker maar CRI-O als container technologie te gebruiken. Met Docker bouw je en run je applicaties. CRI-O kan alleen maar runnen. De runtime is dus efficiënter, maar ook minder kwetsbaar want build commando's bestaan niet waardoor niet plotseling een nieuwe build gedaan kan worden

### **3.8. ARCHITETCURDOMEINEN EN HET RAAMWERK**

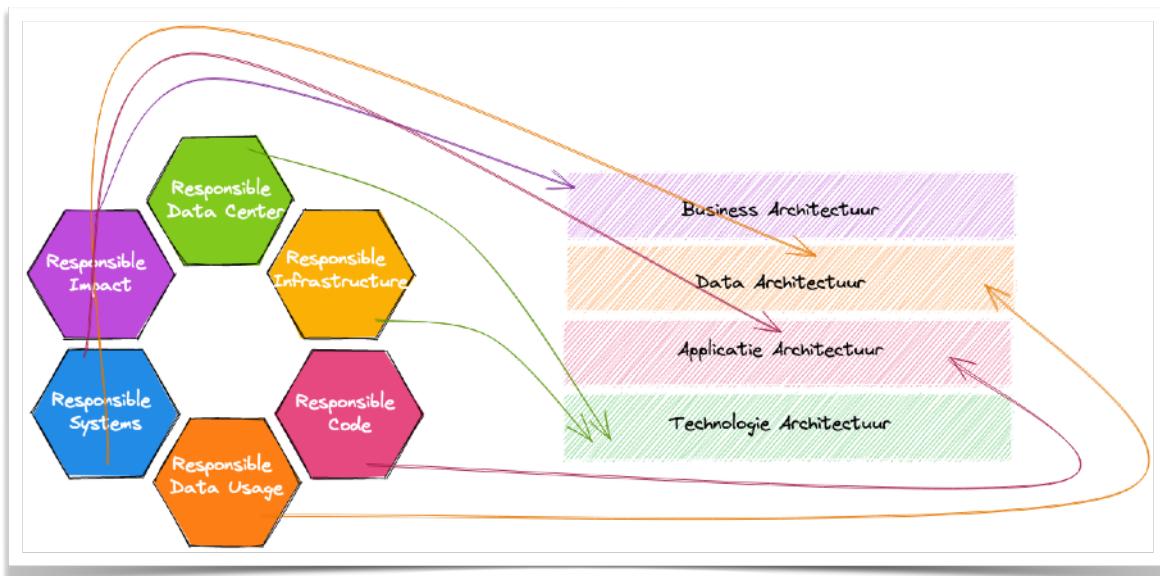
In het Responsible Computing raamwerk zijn zes domeinen gedefinieerd. Vanuit TOGAF en architectuur rollen onderscheiden we een viertal hoofddomeinen:

1. Business Architectuur
2. Data Architectuur
3. Applicatie Architectuur
4. Technologie Architectuur

De zes Responsible Computing domeinen zijn goed te relateren aan de TOGAF-architectuur domeinen. Dit kan enerzijds helpen om je duurzaamheidsverantwoordelijkheden binnen je rol af te

bakenen. Anderzijds kan het helpen om de zes Responsible Computing domeinen beter te positioneren. De Enterprise Architect is verantwoordelijk voor de consistentie tussen domeinen.

We zien de mapping als volgt:



# 4. DE IT SPECIALIST: DUURZAAMHEID IN UITVOERING.

**E**n collega van ons heeft eens gezegd dat iedere IT specialist assembler moet kunnen schrijven, alleen zo kan je doordringen tot het hart van de computer, deze echt begrijpen en alle resources optimaal gebruiken.

Daar waar de architect verantwoordelijk is op het toezien dat duurzaamheid eisen explicet worden gedefinieerd voor een systeem is de specialist verantwoordelijk om dat ook daadwerkelijk de gedefinieerde oplossing te realiseren. Maar het gaat verder, de specialist is niet alleen verantwoordelijk om deze specificaties te realiseren, hij of zij is ook verantwoordelijk voor de *manier waarop* de oplossing gerealiseerd wordt.

Het is vergelijkbaar met de woningbouw. De architect geeft aan welke materialen en bouwblokken er gebruikt worden om de woning optimaal te isoleren en deze zo energieuwig mogelijk te maken. Hij maakt detailtekeningen van bijvoorbeeld koudebruggen waarin staat hoe deze voorkomen of geminimaliseerd kunnen worden. Een koudebrug, ook wel thermische brug genoemd, ontstaat doordat er een onderbreking is in de isolatielaag van de woning. Maar als het op dit detail aankomt is de praktische ervaring van de aannemer onontbeerlijk. Kan wat de architect bedacht heeft wel gerealiseerd worden? De aannemer bouwt, hij heeft het gereedschap in handen.

Specialisten hebben, net als beheerders veel macht op hun terrein. In de film Spiderman zegt oom Ben tegen Peter Parker (Spiderman): “*With great power comes great responsibility*”.

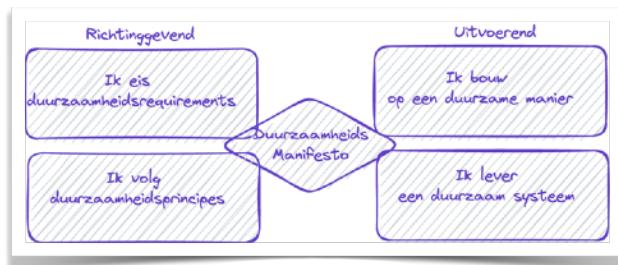
Het is soms een dunne lijn tussen het misbruiken of juist goed gebruiken van die machtspositie. Toch vinden wij dat wanneer een bedrijf onethische systemen bouwt de specialisten geen struisvogels mogen zijn! Ook wanneer een bedrijf schaarse resources verkwantelt MOET de specialist ook zijn of haar accountability (lees aansprakelijkheid) accepteren.

## 4.1. HET MANIFESTO VAN DE DUURZAME SPECIALIST

We zien dus dat de verantwoordelijkheden van de specialist verbreden.

Naast de bestaande verantwoordelijkheden is de specialist

verantwoordelijk om een duurzaam systeem te bouwen en het systeem duurzaam te bouwen. We hebben daarom een gedragsmanifesto opgesteld waar de duurzame specialist zich in zal herkennen:



Manifesto van de duurzame IT Specialist

1. **Ik eis duurzaamheidsrequirements.** Een specialist wil weten hoe hij zo duurzaam mogelijk kan bouwen. De oplossing moet ethisch verantwoord zijn en efficiënt met schaarse resources omgaan.
2. **Ik volg duurzaamheidsprincipes.** De architect definieert samen met de business duurzaamheidsprincipes die de doelstellingen van het bedrijf ondersteunen in de context van de organisatiecultuur. De specialist zal deze principes volgen, ook als druk vanuit de operatie toeneemt om wegen af te snijden en gevraagd wordt om principes te hanteren als richtlijnen.
3. **Ik bouw op een duurzame manier.** Duurzaamheid toepassen op de manier waarop gebouwd wordt. Vergelijk bijvoorbeeld de bouw van een datacenter. Deze zijn veelal van beton en esthetisch niet om over naar huis te schrijven. Er bestaan echter alternatieven voor beton en waarom ook niet aandacht geven aan de buitenkant van een pand om bijvoorbeeld horizonvervuiling te voorkomen? De manier waarop een infrastructuur of een applicatie wordt gebouwd wordt door specialisten voortaan duurzaam gedaan.
4. **Ik lever een duurzaam systeem.** Het actief managen van duurzaamheidsrequirements vanuit de business tot een succesvolle implementatie van het systeem. Specialisten weten ook wat de uiteindelijke bedoeling is van het systeem. "Mijn naam is haas, en ik weet nergens van" doen we niet aan. Zij zijn aansprakelijk.

In een duurzame wereld wordt de verantwoordelijkheid voor de specialist dus groter. Om duurzaamheidskeuzes te maken heb je

meer kennis nodig dan alleen programmeren, modellen bouwen of infrastructuur realiseren. Waar de specialist eerst de gebruiker was van een omgeving wordt hij nu mede-eigenaar. Laten we een voorbeeld geven om dit te verduidelijken.

Je hebt de mogelijkheid om een auto te leasen of zelf aan te schaffen. Als we generaliserend naar het verschil tussen eigenaar en gebruiker kijken zien we het volgende gedrag:

Eigenaar van de auto (privé)	Gebruiker van de auto (lease)
Rijdt efficiënt (de eco stand en meest economische route)	Staat altijd in de sportieve stand. Rijdt net iets harder dan toegestaan. Tijd is geld.
Zet in waar nodig	Zet in waar mogelijk
De fiets is een optie	Heeft geen fiets
Tanken kost geld	Tanken kost tijd maar levert wel spaarpunten op!
Onderhoud verlengt de levensduur	Onderhoud is verplicht door de leasemaatschappij
Kilometers verkorten de levensduur	Kilometers geven eerder zicht op een nieuwe auto
Vermijd iedere schade	Vermijd meer dan één schade per jaar
De auto moet efficient zijn en zuinig rijden	De auto moet mooi zijn
Weet hoe de auto werkt	Weet hoe je de auto bedient
Bij warm weer: ARKO (Alle Ramen Kunnen Open)	Als het heet is ga in in de auto zitten en doet de airco aan
Wast de auto regelmatig en houdt de banden op spanning	Wast de leaseauto bij inleveren

Hoewel dit natuurlijk wat overdreven is, zit er een duidelijke boodschap in; de eigenaar voelt zich veel meer betrokken bij de auto zelf, de eigenaar wil begrijpen wat er onder de motorkap gebeurt om de juiste beslissingen te nemen. De duurzame specialist begrijpt wat de impact is van de code op de processor of de impact van de keuze tussen een router en een switch.

De Responsible Computing principes gaan over authenticiteit en responsibility. Dat is precies waar dit ook over gaat. Van de

moderne specialist mag worden verwacht dat deze aansprakelijkheid (eigenaarschap) accepteert voor zijn code, infrastructuur en modellen en de duurzaamheidsprincipes ter harte neemt.

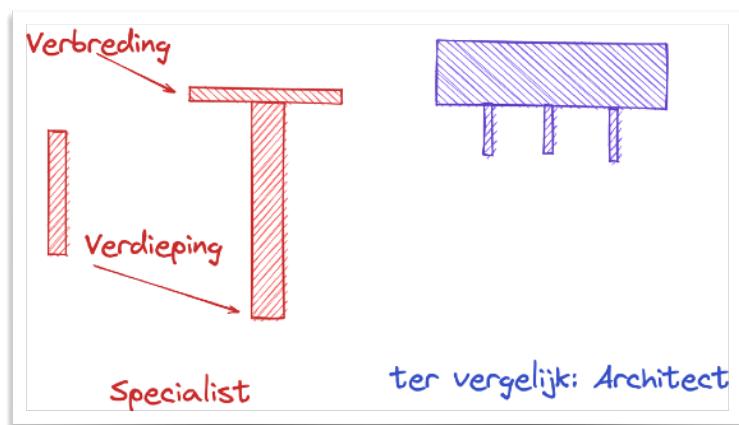
De huidige specialisten opleidingen zijn veelal ingericht op een manier waarop de leserijder zijn auto gebruikt. Opleidingen moeten aandacht besteden aan de gevolgen van de keuzes die specialisten maken bijvoorbeeld op het gebruik van de hardware.

## 4.2. VERBREDEN OF VERDIEPEN?

Het is langer een trend dat specialisten die net van school komen een brede kennis hebben van verschillende programmeertalen en computersystemen. Zij kunnen in principe in iedere taal ontwikkelen op elk systeem en deze brede ervaring zorgt ervoor dat zij ook makkelijk kunnen switchen van omgeving a naar b. Na verloop van tijd zien we dat zij specialist worden op één of enkele omgevingen. Dat heeft het voordeel dat zij binnen die omgeving resources optimaal kunnen inzetten. Het grote nadeel is dat ieder nieuw systeem op die ene omgeving landt. We noemen dat het golden hammer anti-pattern. Ieder systeem is voor die specialist een spijker.

Als het gaat om duurzaamheid moet de specialist zich verbreden en wordt dus iets minder specialistisch. Deze trend zien we al langer, denk bv aan SRE teams, hyper converged systems etc. Toch zit de waarde van de specialist in zijn specialisme. Het is goed om een huisarts te hebben, maar toch ook fijn als er specialisten in het ziekenhuis

werken die heel veel verstand hebben van bijvoorbeeld het hart. We spreken hier van een T-Shape profiel.



Het plaatje laat

T-Shape profiel van de Specialist

zien dat de duurzame specialist zich zowel in diepgang als in breedte moet ontwikkelen. Ter vergelijk de architect, die van nature die brede kennis heeft waar duurzaamheid aan is toegevoegd.

We gaan nu in op een aantal specifieke specialistenrollen. We kiezen de meest voorkomende. Uiteraard staat het vrij om specialisten toe te voegen die voor jou organisatie van belang zijn.

### **4.3. IT SPECIALIST: DE PROGRAMMEUR**

De programmeur is de IT specialist die zich toegelegd heeft op het schrijven van software. In het hoofdstuk over Responsible Code staat de nodige informatie over duurzaam coderen (green coding) die voor de programmeur zeer relevant is.

Laten we het gedragsmanifesto van de duurzame IT specialist eens projecteren op de programmeur:

1. Ik eis duurzaamheidsrequirements
2. Ik volg duurzaamheidsprincipes
3. Ik bouw op een duurzame manier
4. Ik lever een duurzaam systeem

#### ***Ik eis duurzaamheidsrequirements.***

De programmeur ontwikkelt de software op basis van requirements. Deze requirements worden veelal weergegeven in de vorm van use cases, business scenarios en non-functional requirements. Met de introductie van DevOps is het idee dat een programmeur een programmaatje maakt en het vervolgens over de schutting gooit om te worden opgepakt door Operations wel bijgesteld. Daarna werd Security belangrijk om in het ontwikkelproces mee te nemen en werd het DevSecOps.

*SusDevSecOps* is de practice die wij voorstellen om de programmeur niet alleen te laten ontwikkelen of ontwikkelen en beheren, maar ook om ervoor te zorgen dat ontwikkeling en beheer op een duurzame manier gebeurt. Van nu af aan accepteert de programmeur geen opdracht meer waarbij de duurzaamheidseisen niet duidelijk gesteld zijn.

#### ***Ik volg duurzaamheidsprincipes***

De architect heeft principes t.a.v. duurzaamheid opgesteld. Het idee van principes is dat ze de bedrijfsstrategie op duurzame manier ondersteunen. Urgentie, druk vanuit de business, die veelal veroorzaakt wordt door druk vanuit de markt zet deze principes

regelmatig onder druk. De specialist bezwijkt niet onder druk maar houdt vast aan de principes.

### ***Ik bouw op een duurzame manier***

In het kader van *SusDevSecOps* ontwikkeld de programmeur op een duurzame manier en denk dus na over zijn toolchain met tools en of dit een duurzame aanpak is. Onder tools verstaan we zowel programmeertalen, compilers, patterns en frameworks maar ook testscripts die code valideren.

### ***Ik lever een duurzaam systeem***

*SusDevSecOps* vereist ook dat de operatie op een duurzame manier gebeurt. De applicatie moet voldoen aan de daarvoor geldende ethische normen, energieverbruik en CO<sub>2</sub> uitstoot maar moet niet ingaan tegen de UNSDGs. Dit laatste is veel gevraagd, maar als je weet dat een gedeelte van de applicatie gebouwd wordt door AI die werkt vanuit China en minder krijgt dan het minimum loon dan zou je in beweging moeten komen.

### ***Moet je als programmeur breed inzetbaar zijn of juist specialist zijn?***

Als je de Java programmeur vraagt om een oplossing te bouwen dan weet je één ding zeker: het wordt een (efficiënte) oplossing in Java geschreven. Zou het niet mooi zijn als de programmeur in staat is om de juiste programmeertaal te selecteren voor deze vraagstelling en wellicht een nog efficiëntere oplossing te bouwen!

In het hoofdstuk over Responsible Code geven we een aantal operating modellen weer waarvoor gecodeerd kan worden. Een embedded programmeur is waarschijnlijk gebaard bij een diepgaande kennis van bv. Lua of C. Datzelfde geldt voor een programmeur in de gaming- of filmindustrie die gebruikt maakt 3D tools als Blenders. Als we kijken naar onze primaire doelgroep van algemene programmeren en AI dan hebben we waarschijnlijk het meest baat bij ontwikkelaars die breed inzetbaar zijn.

In dit hoofdstuk bespreken we ook verschillende codeerpatronen. Deze patronen zijn gebaseerd op fundamentele principes zoals wachtrijtheorie, sorteeralgoritmen, caching, etc.. Het is van belang dat de programmeur op de hoogte is van deze principes.

## **44. IT SPECIALIST: DE DATA SCIENTIST**

In de context van dit hoofdstuk beperken we de rol van de data scientist tot die van modellenbouwer. In de praktijk heeft de data scientist veelal ook de rol van de programmeur. Maar voor dit gedeelte focussen we op de rol van de data scientist die AI modellen bouwt. Als we kijken naar het gedragsmanifesto en we maken de vertaling naar de data scientist kunnen we daar een aantal dingen over zeggen.

### ***Ik eis duurzaamheidsrequirements***

Ten aanzien van de duurzaamheidsrequirements zal de data scientist vooral op zoek gaan naar de ethische en compliance gerelateerde requirements. In het hoofdstuk over Responsible Data Usage en Responsible Systems kun je daar het nodige over vinden.

### ***Ik volg duurzaamheidsprincipes***

Ten aanzien van de duurzaamheidsprincipes geldt hetzelfde als voor de andere specialisten. Laat je de kop niet gek maken onder druk of zogenaamde urgente business belangen. Afhankelijk van de toepassing kan een model enorme impact hebben in een beslissingsproces. Die impact moet voor de data scientist duidelijk zijn voordat hij of zij aan het bouwen van het model begint. Mag de data gebruikt worden voor het maken van het model is een andere vraag dan is de data beschikbaar. In veel gevallen mag het wel en is alleen vervelend genoeg niet beschikbaar. Echter er zijn ook situaties genoeg waarin het wel beschikbaar is maar niet voor die toepassing gebruikt mag worden. Bedrijven als Google likken nog steeds hun wonderen.

### ***Ik bouw op een duurzame manier***

De data is er, de privacy keurig op orde en nu het model bouwen! De GPU cycles vliegen je om de oren en als de knop ingedrukt wordt om het model te genereren dan gaan de lampen in het kantoor zachter branden, het datacenter wordt 5 graden warmer en de airco's maken overuren. Er zijn artikelen<sup>3</sup> waarin het gebruik van AI gekoppeld wordt aan een ecologische ramp. Toe maar, je zult data scientist zijn, nog even en je bent niet meer welkom op feestjes en je wordt op straat met de nek aangekeken!

In het artikel wordt ook verwezen naar codecarbon, een lichtgewicht pakketje dat je aan je code toevoegt om CO2 uitstoot

---

<sup>3</sup> [https://datanews.knack.be/ict/nieuws/hoe-vermijden-we-dat-ai-uitdraait-in-een-ecologische-ramp/article-opinion-1865833.html?cookie\\_check=1670936753](https://datanews.knack.be/ict/nieuws/hoe-vermijden-we-dat-ai-uitdraait-in-een-ecologische-ramp/article-opinion-1865833.html?cookie_check=1670936753)

te meten en hoe je ook verbeteringen door kan voeren. Helaas wordt er dan ook verwezen naar locaties waar je de code kan uitvoeren omdat daar providers zitten die groene stroom gebruiken. Dat voelt niet alleen als afkopen, maar dat is het ook.

### ***Ik lever een duurzaam systeem***

Tot slot wil je ook dat je model gebruikt wordt ten behoeve van het goede voor de mensheid. Dat kan lastig zijn omdat het model maar een heel klein stukje is uit een bedrijfsketen.

## **4.5. IT SPECIALIST: INFRASTRUCTUUR SPECIALIST**

Infraspecialisten in het kort, kunnen een grote stempel drukken op de duurzaamheid van de IT omgevingen. Zij bepalen in grote mate hoe de datacenter voorzieningen worden ingezet en welke infrastructuur voorzieningen worden gebruikt in het datacenter. Vaak zijn zij ook betrokken bij capaciteitsvraagstukken en staan zij aan de voet van beslissingen rond opslag. Tevens is de infraspecialist diegene die nieuwe technologie beproeft en deze introduceert in het IT landschap. Hoe deze infrastructuur componenten worden geconfigureerd om optimaal te presteren ligt in de handen van deze specialist.

Als er een eis vanuit de business komt voor een bepaalde beschikbaarheid van een systeem dan is de infraspecialist diegene die bedenkt hoe deze requirement wordt ingevuld. En wees eerlijk, de infraspecialist heeft zijn of haar zaakjes vaak prima op orde. Uiteindelijk moet er voorkomen worden dat bij iedere calamiteit hij of zij uit bed gebeld wordt om het probleem op te lossen. Een beter motivatie om problemen te voorkomen is er niet!

Laten we deze rol in de IT organisatie eens volgen aan de hand van het gedragsmanifesto.

### ***Ik eis duurzaamheidsrequirements***

Net als bij security bepaald de zwakste schakel de kracht van het beleid. De infraspecialist is in staat om het beleid voor duurzaamheid te vertalen naar zijn of haar werkomgeving. Dit kan betrekking hebben op technologie waarbij de specialist in veel gevallen een machtige positie heeft qua systeem rechten en daarom bij informatie kan die gevoelig is of het onvolledig of zelfs verkeerd documenteren van technische oplossingen zodat zijn of haar rol onmisbaar is en blijft (niet lachen dat gebeurt echt!). Dit

gaat natuurlijk over ethiek, maar iets eenvoudiger is natuurlijk het duurzaam adviseren over technologie en deze allen vervangen / verversen als het echt nodig is.

De Infraspecialist heeft requirements nodig om schakelketen voor duurzaamheid compleet te maken. Daarin moet de infraspecialist een actieve rol spelen.

Met actief bedoelen we: als de requirements ontbreken dan ga je ze ophalen. Bij implementatie van systemen wordt nog te weinig gekeken naar de duurzaamheid van de technologie keuze.

Gebruiken we voor niet-actieve data het tape medium als opslag of slaan we dit op harde schijven op zodat (gemak dient de mens) de informatie altijd direct beschikbaar is.

Overwegen we applicaties met identieke karakteristieken (werklasten) te consolideren op enterprise servers of blijven we doorgaan met massale uitbreiding van pizzadozen omdat we dat nou eenmaal gewend zijn?

### ***Ik volg duurzaamheidsprincipes***

In het voortbrengingsproces zitten alle handelingen die nodig zijn om een omgeving te bouwen en te testen.

In het voortbrengingsproces heeft de infraspecialist invloed op duurzame keuzes. Implementeren we nieuwe toepassingen alleen gebaseerd op technische specificaties of nemen we duurzaamheid requirements in acht? Laten we de testomgevingen actief doordraaien tot bij navraag blijkt dat deze niet meer nodig is of worden er vooraf afspraken gemaakt over de beschikbaarheid van de testomgevingen?

### ***Ik bouw op een duurzame manier***

Een infraspecialist begrijpt heel goed voor welke toepassing een omgeving gebouwd moet worden. Naast de functionele aspecten van de toepassing is de specialist goed geïnformeerd over de non-functional requirements. De specialist beveiligt de systemen tegen ongeoorloofd gebruik en zorgt dat data beveiligd is tegen tegen intern extern misbruik. Om ethisch gedrag in de organisatie te monitoren implementeert de specialist technologie om te voorkomen dat medewerkers ongeoorloofd websites bezoeken die door het bedrijf als niet gewenst of onethisch zijn bestempeld en is de specialist in staat om middels archief data informatie te achterhalen bij klachten over onethisch gedrag tussen medewerkers.

### ***Ik lever een duurzaam systeem***

Een systeem bestaat uit meer dan infrastructuur componenten. Het is een keten van business functionaliteit, applicatie, data en infrastructuur. De infraspecialist heeft als laatste schakel in de keten oog voor de duurzaamheid van de complete keten.

# 5. DE PRODUCT OWNER: DE BEWAKER VAN DUURZAAMHEID

We hebben de architect als de persoon die duurzame oplossingen ontwerpt, de programmeur die ze bouwt en de rol van de product owner of de project manager die verantwoordelijk is voor het bewaken ervan. Ten eerste willen we kijken naar de complexiteit en reikwijdte van het onderwerp duurzaamheid en hoe je hiermee om kan gaan als Product Owner. We zullen aantonen dat we leven in een tijd die zo onzeker is dat je er niet vanuit kunt gaan dat je van te voren een plan maakt en dat plan vervolgens stap voor stap uitvoert. Als Product Owner heb je ook te maken met culturele aspecten. We doen een eerste poging om projecten op een zo duurzaam mogelijke manier te ontwikkelen. Het is goed mogelijk om een groot gedeelte van het werk vanuit huis te doen zodat niet naar het werk gereden hoeft te worden. Anderzijds is er behoefte vanuit mensen om elkaar te ontmoeten. Dit sociale aspect leidt tot meer plezier in het werk en een betere kwaliteit. Op basis van social distance denken we na over de meest efficiënte vormen van vergaderen en samenwerken. Het agile manifesto stelt: ***"The most efficient and effective method of conveying information to and within a development team is face-to-face conversation."***. Hoewel dit een waar statement is heb je er niet zo veel aan omdat de praktijk anders is. Je moet kijken wat er wel en niet nodig is om informatie uit te wisselen.

## 5.1. BRAIN FREEZE: COMPLEXITEIT IN DE CONTEXT VAN DE ECONOMISCHE SITUATIE

Je kent het wel. Het is heerlijk warm, je hebt een lekker koude milkshake en na de 5<sup>e</sup> keer het koele ijs naar binnen te zuigen bevriezen je hersenen. Je kan even niets meer. Gelukkig is het na een paar seconde weg en je besluit, nu in een gematigder tempo je milkshake naar binnen te werken. Zo kan het ook wel zijn met het onderwerp duurzaamheid. Het is soms zoveel dat je hersenen bevriezen, je kunt niets meer. Waar moet je beginnen? Mocht je dat gevoel nog niet hebben dan heeft deze paragraaf tot doel om je op dat gevoel voor te bereiden.

Het is belangrijk om de complexiteit te begrijpen omdat je anders tegen de brain freeze aanloopt en het gevolg is moedeloosheid en

uitzichtloosheid. Daarom is het goed om de complexiteit onder ogen te zien en ook jou gebied af te bakenen tot een gebied dat voor jou wel behapbaar is. Een vriend zij eens, toen we het hadden over het geven aan goede doelen en of het geld dan wel goed terecht komt: Het is mijn verantwoordelijkheid om te geven, aan de organisatie om ervoor te zorgen dat het goed terecht komt. Dat is één van de doelen die we met dit boekje willen bereiken: ***neem verantwoordelijkheid op het gebied waar jij invloed op hebt.***

Hoe staat het ervoor met duurzaamheid als we dit plaatsen binnen het geheel van veranderingen waar we als mens, als land, als economie mee te maken hebben. Voor deze korte analyse gebruiken we het STEEPLED model. Het STEEPLED model is een model dat op acht domeinen aangeeft hoe het zit met de stabiliteit van een omgeving. Eén van deze domeinen is het milieu, daarnaast zijn er nog zeven andere. Acht min één is nog steeds zeven, dat had je natuurlijk ook al geconcludeerd.

S = Sociaal

T = Technologisch

E = Economisch

E = Environmental (milieu)

P = Politiek

L = Legislation (wetgeving)

E = Ethisch

D = Demografisch

Hoe stabieler deze factoren, hoe minder verandering. Dus als die andere zeven nu lekker stabiel zijn dan kunnen we ons echt richten op het oplossen van de milieuproblematiek. Maar wat zien wij in deze tijd? Bijna alle factoren zijn maximaal in beweging. Het is goed om daarvan doordringen te zijn als Product Owner. Deze wetenschap bepaald de duurzame inzet van je team en de keuze van ontwikkelmethoden. Maar eerst even een korte reflectie op deze tijd.

Op ***sociaal*** gebied is er nogal wat gebeurt. COVID-19 heeft een enorme impact gehad en nog steeds. Denk bijvoorbeeld maar aan het thuiswerken en de impact op de mobiliteit.

***Technologisch*** zijn de ontwikkelingen nog volop gaande. AI, 5G en quantum om er maar eens een paar te noemen, maar ook crypto currency, social media, elektrisch rijden, energietransitie.

**Economisch** staan we volgens deskundige aan de vooravond van een recessie. Of dit gaat gebeuren of niet het feit is dat we worden geconfronteerd met een ongekend hoge inflatie.

Qua **milieu** hoeven we niet veel te zeggen. Lees een paar publicaties en de conclusie is dat we eigenlijk te laat zijn. De klimaattop (COP27) in Egypte in 2022 is uitgelopen op een teleurstelling. De uitkomst is een fonds om arme landen te helpen bij het oplossen van de consequenties van de klimaatveranderingen. We hebben dus al toegegeven dat het een onomkeerbaar proces is. Zouden we niet juist moeten werken aan de preventie en daarin investeren?

**Politiek** is het op wereldniveau een zorgelijke situatie. De oorlog tussen Rusland en Oekraïne heeft wereldwijde impact. Maar ook lokaal rommelt het in veel landen. Dit staat niet los van alle bovenstaande ontwikkelingen.

Om de bovenstaande chaos te beheersen gaan we wetten maken. In Nederland zijn er voor gebouwen energie labels verplicht. Nu moeten vanaf 2023 bedrijfspanden ( $\geq 100\text{m}^2$ ) minimaal energielabel C zijn. Geen nieuwe auto's meer op civiele brandstof vanaf 2035. Geen gasaansluiting meer in nieuwe woning (let wel in 2020 was er nog het recht op een gasaansluiting!) De politiek in paniek. Zijn wetten nog een middel of worden ze een doel?

**Ethiek** is een hot topic. De druk om transparant te zijn gaat de komende jaren enorm toenemen. Hoe kom je tot welke beslissingen. Hoe toon je dat aan als je bijvoorbeeld gebruik maakt van AI? In het hoofdstuk over Responsible Systems komt dit uitgebreid aan de orde. Hier is een belangrijke rol weggelegd voor de Product Owner.

En tot slot **demografisch**. Wellicht is dit nog de meest stabiele factor in de westerse wereld. Wij mensen wonen, werken en leven in bepaalde omgevingen en dat veranderd niet massaal. De woningnood en COVID-19 leiden wellicht tot verschuivingen van (rand)stad naar platteland, maar de impact is (nog) beperkt. Toch kunnen de bovenstaande factoren impact hebben op demografische factoren. Denk maar eens aan de consequenties van het stijgen van de zeespiegel voor een land als Nederland wat voor een groot gedeelte beneden de zeespiegel ligt.

We zitten met elkaar in een “perfect storm”, alle factoren zijn in beweging en daarnaast beïnvloeden ze elkaar ook. De factoren kunnen elkaar versterken of juist compenseren, dat weten we niet. Het is onvoorspelbaar en onzeker hoe de komende jaren er uit zullen zien. Wat moet je nu als project manager doen? Twee dingen:

- 1) Zorg dat je **flexibel en veerkrachtig** bent. Dwight Eisenhower geeft hier een mooi voorbeeld van: *“In preparing for battle I have always found that plans are useless, but planning is indispensable”*. Hij zegt dat het gaat om het proces van plannen, de uitkomsten kunnen per dag verschillen, maar als je het proces onder controle hebt dan kun je meebewegen met de omstandigheden. Zoals Rinus Michels ooit zei: voetbal is oorlog, zo is de situatie nu ook best kwetsbaar en explosief. Als project manager / product owner moet je zorgen voor een agile of iteratieve aanpak. In het boek Agile Managen geeft de schrijver Mike Hoogveld een goed inzicht in de noodzaak van een flexibele en veerkrachtige aanpak en een manier om dit te doen.
- 2) Het tweede is: neem **non-regretable stappen**. Onafhankelijk van de omstandigheden zijn er stappen die altijd inherent goed zijn. Stappen die niet afhankelijk zijn van de context maar gekoppeld zijn aan universele waarden. Een voorbeeld van een non-regretable stap is het omlaag brengen van je energieverbruik. Een verkeerde stap is het overstappen van gas naar elektriciteit. Het eerste is altijd goed, het tweede is afhankelijk van de context. Hetzelfde geldt bv voor elektrisch rijden. Waar het eerst financieel interessant was (kosten per km) t.o.v. brandstof is door de 400% verhoging van elektriciteit dit business model binnen één jaar 180 graden gedraaid. Een non-regretable stap zou zijn het minderen van transport (kortere afstanden, minder frequent, zie ook ons idee hiervoor bij Responsible Impact)

Wat wij hier beschrijven zijn non-regretable stappen. Ze zijn altijd goed en onafhankelijk van de context.

Een druppel op de gloeiende plaat? Een mooie uitdrukking, maar niet erg bemoedigend. Maar als jij één druppel bent en we druppelen met zijn alle op die gloeiende plaat, dan koelt hij (letterlijk) af. 1,5 graad is het doel, dat is blijven staan tijdens het COP27 overleg.

## 5.2. EFFICIENT INZET VAN TEAMS

We hebben nu twee belangrijke principes gedefinieerd voor de Product Owner. Vervolgens willen we kijken hoe de teams zo efficient mogelijk kunnen worden ingezet. Met efficientie bedoelen we niet het maximaal uitbuiten van de medewerker, maar juist het optimaal laten floreren van de medewerker zodat deze tot optimale prestaties komt. Daarom een klein stukje onderzoek dat gedaan is naar de behoeftes van mensen.

Proxemics is een studie naar het menselijk gebruik van ruimte.

Hierin wordt onderscheid gemaakt in:

1. De intieme ruimte
2. De persoonlijke ruimte
3. De sociale ruimte
4. De publieke ruimte

Deze ruimtes zijn in de studie van proxemics fysieke bepaald, maar dit is ook verder te trekken naar een sociale context.

Persoonlijke Ruimte	Afstand	Relaties		Omvang
Intiem	Tot 50 cm	Partners	Core Family	2-5
Persoonlijk	0,5 - 1 meter	Vrienden	Extended Family	10
Sociaal	1 - 4 meter	Kennissen	Collega's	150
Publiek	Meer dan 4 meter	Onbekenden	Voorbijganger	> 250

Met de sociale context bedoelen we dat deze ruimtes ook allemaal een doel hebben. Als Product Owner heeft je team bepaalde behoeftes, hoe kom je daar nu het beste aan tegemoet? De intimiteit van een team is daarbij cruciaal.

*Stilstaan is achteruitgaan.* Traditioneel zien we een beweging naar groei om efficiëntie en invloed te vergroten. Dit betekent grotere organisaties, meer specialisme, complexere communicatiestructuren. Als je echter zoekt op “Stilstaan is ...” valt op dat ook de uitspraak stilstaan is vooruitgaan wordt omarmt. In het boek “Cradle to Cradle: Remaking the Way We Make Things”

beschrijven de auteurs<sup>4</sup> juist de beweging om in het kader van duurzaamheid meer focus te hebben voor de lokale context en de micro-economie. In Het boekje “Eckart’s notes” beschrijft Eckart Wintzen zijn organisatorisch concept voor BSO, een zeer wendbare organisatie die veelal geografisch bepaald werd. Denk eens na over het volgende.

Persoonlijke Ruimte	Behoefte	Rol	Ontmoetingen
Intiem	Delen persoonlijke gevoelens in de context van werk	Maatje	Dagelijkse ontmoeting
<i>Persoonlijk</i>	<i>Resultaat / Loyaliteit</i>	<i>Team / mentor / Coach</i>	<i>Weekelijks ontmoeten</i>
Sociaal	Kennisdeling / Belangrijkheid /	Afdeling / Kennisgroep	Maandelijks ontmoeten
Publiek	Trots / Bihoren	Bedrijf	Jaarlijks ontmoeten

Wintzen vind het belangrijk dat mensen onderdeel zijn en voelen van de afdeling, niet van het bedrijf. Voor bedrijven groter dan 150 mensen zou dit een goed model kunnen zijn. Geografisch groeperen, daar ligt je betrokkenheid. Daarmee is er geen noodzaak meer om grote fysieke bedrijfsontmoetingen te regelen, dat kan prima die een virtuele meeting.

Bedenk dus als Product Owner dat je team opereert in de persoonlijke ruimte. Dat je dus ruimte maakt voor die wekelijks ontmoeting. Dat mensen hun maatjes hebben binnen dat team of net erbuiten, dat je onderdeel bent van een afdeling of kennisgroep waarbij je maandelijkse betrokkenheid stimuleert. En ten slotte is ook ieder teammember onderdeel van het grote bedrijf, die staat voor duurzame normen en waarde.

In het hoofdstuk over Responsible Impact hebben we een voorbeeld gegeven over de inzet van IT om duurzaam te teamen.

---

<sup>4</sup> Boek van Michael Braungart en William McDonough

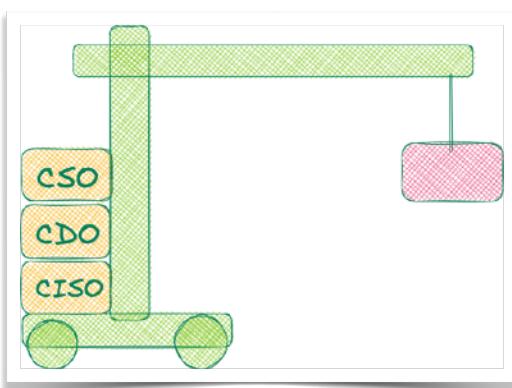
# 6. DE CHEF

**D**e chef officier duurzaamheid is de letterlijke vertaling van de “Chief Sustainability Officer”. Nou het klinkt wel even anders! Daar zitten wel twee salarisschalen tussen. Laten we het daarom maar hebben over de CSO. De CSO is de functionaris die zich bezig houdt met het opstellen en uitvoeren van het duurzaamheidsbeleid van het bedrijf. Een nieuwe rol die in het leven is geroepen om meer en beter focus te krijgen op de duurzaamheidsaspecten. Waarschijnlijk zal deze persoon zich het eerst bekommeren over de verplichte rapportages (ESG) die moeten worden opgeleverd. Zie daarvoor ook de sectie in het hoofdstuk Responsible Data Usage. Daarnaast is ook het duurzaamheidsimago van het bedrijf belangrijk.

Afhankelijk van de industrie zal de CSO zich in meer of mindere mate bezig houden met IT. In de transportsector zal dit minder zijn, in de financiële sector meer, eenvoudigweg omdat financiële instellingen bijna IT bedrijven zijn en IT nauwelijks nog een secondair proces te noemen is.

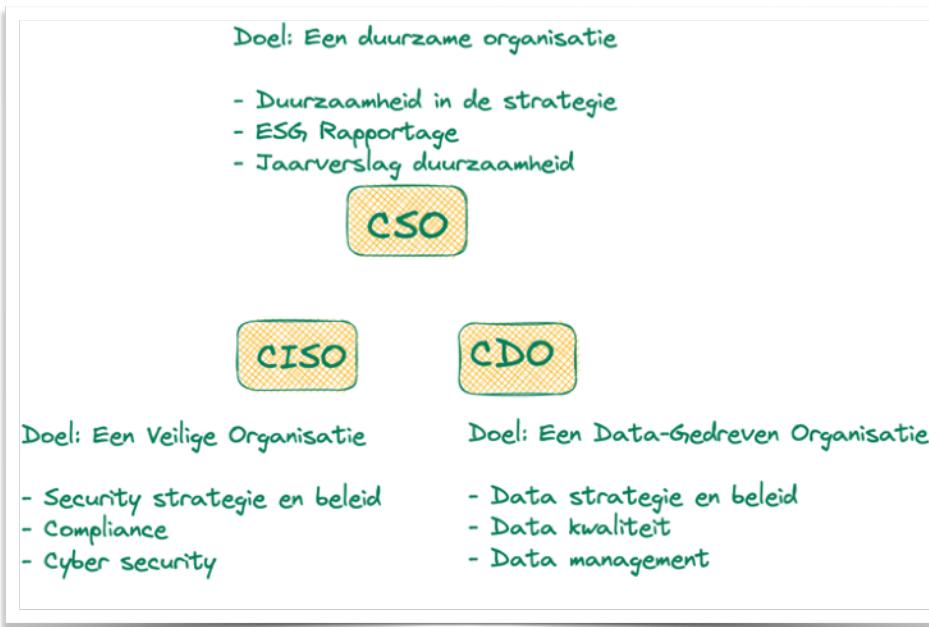
De CSO zal vanuit zijn of haar rol ook veel samenwerken met andere chiefs: de Chief Data Officer (CDO) als het gaat om data privacy en eigenaarschap en de Chief Information Security Officer (CISO) als het gaat om veiligheid.

Je hebt hele hoge en lange hijskranen die bouwmateriaal verplaatsen. Als dat materiaal aan het uiteinde van de hijskraan hangt dan is er een enorme hefboomwerking. Dat materiaal is te vergelijken met de kwartaaldoelstellingen van het bedrijf, de verwachtingen van aandeelhouders of beloften van ministers. De kraan kan alleen blijven staan door de contragewichten. De CSO, CDO en CISO zijn



CSO, CDO en CISO zijn belangrijk voor de balans

de contragewichten van de organisatie. Ze zorgen voor de perfecte balans.



*De triage voor balans*

## 7. DE MEDEWERKER

**E**ven iets totaal anders. We wisten niet waar we dit onderwerp moesten adresseren, maar je hebt soms van die momenten dan heb je iets wat je kwijt moet. We moeten dit even kwijt. We hebben het over het verantwoord gebruik van IT systemen. Veel van deze IT systemen zijn bedoeld om het leven van de medewerker te vergemakkelijken. Toch? We maken een systeem dat ervoor zorgt dat eenvoudige vragen door een chatbot afgehandeld worden. We bouwen case management systemen zodat werk toebedeeld wordt aan de juiste medewerker zodat hij of zij zich efficiënt kan inzetten. We willen het proces zo optimaal mogelijk laten verlopen dus we gaan meten: wat is de doorlooptijd, waar liggen bottlenecks? Voor je het weet is er een afrekencultuur op basis van prestaties. Is dit duurzaam? Waar liggen de grenzen? Wanneer voel je je gecontroleerd?

Een persoonlijk verhaal. Op een goede dag verdiepte ik mij in de mogelijkheden van het automatisch laten scannen van documenten. Ik zag mooie besparingsmogelijkheden en ging met een business case naar de directeur. Besparen op tien mensen a 125k per jaar is toch 1,25 mio per jaar! Zet daar een investering tegenover van 1 mio en een ROI binnen 1 jaar is een no brainer. De directeur keek me aan, zij niets en bracht me naar de afdeling waar het scannen van deze document plaatsvond. Je kent het misschien wel, in de buurt van de postkamer, diep in het binnenste van het gebouw waren mannen en vrouwen bezig met hun werk. Druk bezig met hun belangrijke taak. Toegewijd, serieus en met aandacht voor elkaar. Ze kwamen 's ochtends op de fiets met hun broodtrommel want eten in de kantine was te duur. Ze kosten zeker geen 125k per jaar. Dit was hun werk, dit was waarom ze iedere dag uit hun bed kwamen. Dit was hun bestaan.

Daar vertrok ik, in mijn leaseauto op naar een lunch in een restaurant. Het voorstel ging in de prullenbak, wel of geen business case was niet meer interessant. Het gaat om veel meer dan dat. Dankbaar voor deze directeur en een levensles wijzer.

Het is natuurlijk geen eenvoudig dilemma, maar er zijn grenzen aan het continue verhogen van productiviteit.

Een duurzame organisatie gaat op duurzame wijze om met haar medewerkers.

# **RESPONSIBLE DATACENTER**



Datacenters worden gebruikt om de fysieke infrastructuur onder de juiste condities te huizen. Er moet voldoende stroomaanvoer zijn en voldoende koeling om de infrastructuur optimaal te laten functioneren.

Naast de aanvoer van stroom en de koeling levert de datacenter ook een backup indien de stroom van het net uitvalt. Bij de opkomst van datacenters was koeling altijd een grote uitdaging omdat de computersystemen 'los' op de vloer stonden, veelal gegroepeerd in type systemen. Het mainframe, de gedistribueerde systemen zoals UNIX en Windows bij elkaar en de periferie apparatuur (printers, tape units) stonden ook veelal gegroepeerd in een bepaalde hoek. Door beperkingen in datacommunicatie werd vaak apparatuur decentraal op afdelingen geplaatst in zgn. SER's: Satelite Equipent Rooms. De centrale datacenter stond bekend onder de naam MER: Main Equipment Room.

De restricties in datacommunicatie hebben we door de introductie van moderne communicatietechnieken niet meer. Alle informatiesystemen staan nu centraal in het datacenter. Dat kan natuurlijk een datacenter in het bedrijf (on premise) zijn maar uiteraard ook een cloud datacenter.

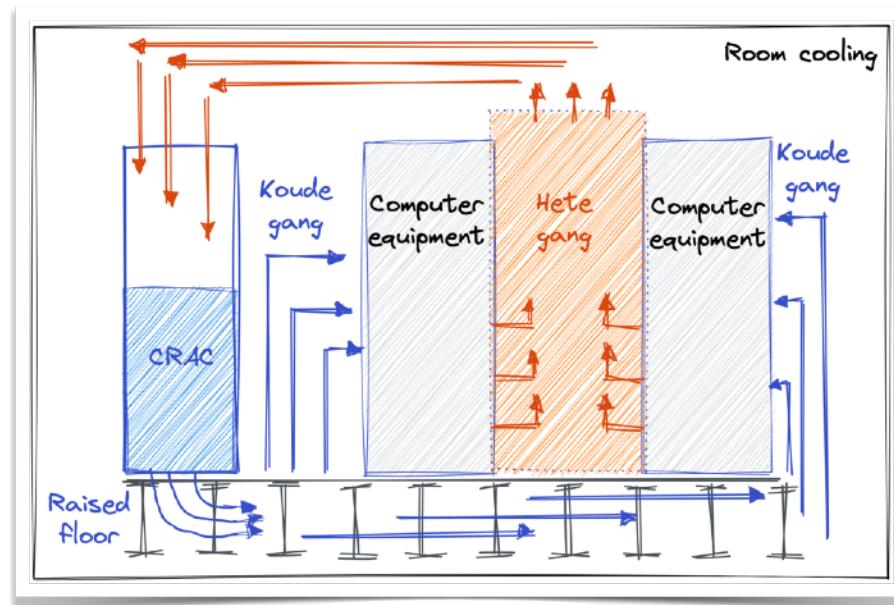
# 8. DE SYSTEMEN KOEL HOUDEN

**K**oeling van de systemen is altijd een uitdaging geweest in een datacenter. Traditioneel werden dat met grote airco units gedaan die in de afgesloten ruimte van het DC stonden opgesteld. Tegenwoordig worden de servers opgesteld in afgesloten 'gangen' die optimaal gekoeld kunnen worden. Deze techniek werkt veel efficiënter dan de losse units die traditioneel op vloer van het DC stonden. De techniek van koelen is ook erg geavanceerd geworden. De traditionele 'airco's' die gekoelde lucht uitblazen zijn uitgebreid met zeer uitgebreide watergekoelde systemen die op het precieze niveau van CPU, GPU of geheugen kaarten koeling kunnen aanbrengen.

## 8.1. LUCHT GEBASEERDE KOELING

De basis Principe van lucht koeling is vrij simpel. Koude lucht wordt de ruimte ingeblazen waarbij de warmte in de ruimte wordt 'afgebroken' door deze te vermengen met koude lucht. Er zijn wel verschillende manieren hoe lucht koeling systemen worden toegepast:

- Room cooling
- Row cooling
- Rack cooling

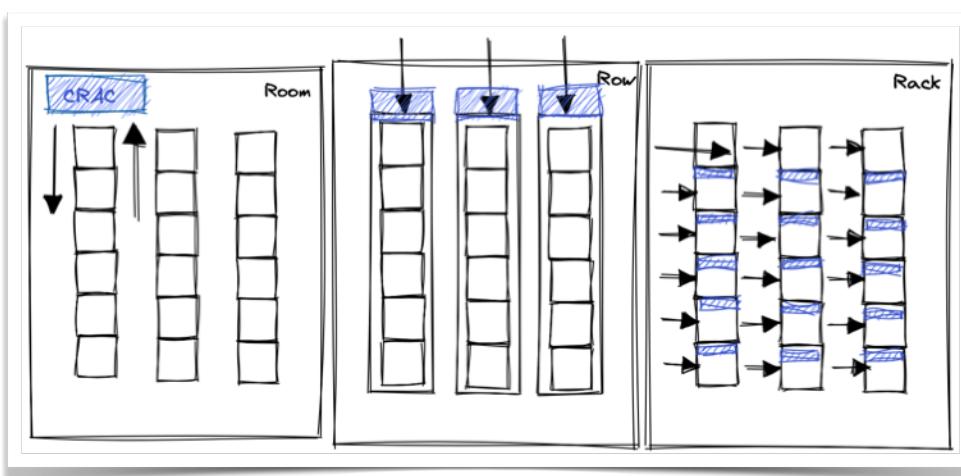


Room cooling Principe

**Room cooling:** Het koelen van de ruimte komt tot stand door koude lucht de ruimte in te blazen (CRAC - Computer Room Air Conditioning). Dit kan worden gedaan door de lucht te circuleren of via de verhoogde computervloer bij de systemen te brengen. Tegenwoordig wordt er wel veel gebruik gemaakt van warme en koude gangen zodat de lucht circulatie beter beheersbaar is en de koude lucht op de gewenste plaats gebracht kan worden. Dit scheelt zeker in de portemonnee!

**Row cooling:** Rij-gekoelde oplossingen zijn preciezer dan ruimte gebaseerde koeling systemen. Elke rij heeft 'eigen' units waarbij de lucht circulatie met grotere precisie in de rij kan worden gebracht. Deze techniek is efficiënter en daardoor energie zuiniger.

**Rack cooling:** Deze vorm van koeling is nog efficiënter waarbij de koeling tot op rack (kast) niveau wordt gebracht. De units worden op of in het rack geplaatst zodat de koeling capaciteit kan worden aangepast aan de situatie van een specifiek rack waar bijvoorbeeld CPU of GPU intensieve werklasten draaien.



Lucht gekoelde oplossingen

## 8.2. NADELEN VAN LUCHT GEKOELDE OPLOSSINGEN

Er zijn wel wat uitdagingen met luchtgekoelde oplossingen. De werklasten worden steeds meer CPU en GPU intensiever. Dit vraagt om veel koel capaciteit. Lucht is in veel gevallen niet meer

toereikend om de gevraagde koel capaciteit te leveren. De kosten (vanwege de energie) zijn enorm hoog en deze vertegenwoordigen een groot gedeelte van het budget om een datacenter te runnen. Naast deze financiële tegenvaller is het ook niet echt goed voor de gezondheid van je collega's die regelmatig op de computerzaal moeten werken. Het geluid die de units produceren kunnen een gehoorbeschadiging opleveren!

### 8.3. VLOEISTOF GEBASEERDE KOELING

Vloeistof gebaseerde koeling wordt steeds breder toegepast dan alleen voor mainframes en supercomputers. Vloeistoffen zijn wel 50 tot 1000 keer efficiënter dan lucht! Systemen worden door de 'zware' applicaties meer en meer eisend tav koeling. Vloeistof koeling belooft in ieder geval in veel gevallen een oplossing te bieden aan de uitdaging van computer concentratie en zware werklasten.

Er zijn eigenlijk drie belangrijke vloeistof gebaseerde koeltechnieken:

- direct-to-chip cooling / direct-to-plate cooling
- rear-door heat exchangers
- Immersion cooling

**direct-to-chip cooling:** Brengt het koelsysteem helemaal binnen het computer chassis. Koude vloeistof wordt direct geleid naar koude plaatjes die vlak naast de warmte gevoelige componenten zijn geplaatst (CPU, GPU, memory). Gekoelde vloeistof wordt direct naar deze plaatjes geleid die vervolgens de warmte absorberen van deze componenten. De warme vloeistof wordt vervolgens weer afgevoerd naar een koel mechanisme . Zodra de vloeistof is afgekoeld wordt deze weer naar de plaatjes geleid.

**Rear-door heat exchangers:** Een vergelijkbaar concept kun je op rack niveau toepassen. De achterdeur wordt vervangen door een warmte wisselaar en de warme lucht van de server ventilatoren wordt door de wisselaar geblazen waarbij de lucht wordt afgekoeld. De warmte wordt afgevoerd door een gesloten circuit naar buiten

**Immersion cooling:** Een veel nieuwere technologie waarbij de interne server componenten worden ondergedompeld in een niet-

geleidende vloeistof. De componenten en de vloeistof worden vervolgens ingepakt in afgesloten containers zodat deze niet gaan lekken. De hitte van de componenten wordt op deze manier naar de koelvloeistof getransporteerd. Dit proces vergt veel minder energie dan andere manieren van koelen.

## 8.4. NADELEN VAN VLOEISTOF GEKOELDE OPLOSSINGEN

Water en electronica gaan niet echt goed samen. De vrees voor risico op lekkage is voor veel IT professionals heet hangijzer, speciaal bij de methode direct-to chip koeling. Als deze oplossing begint te lekken dan is de schade enorm!

Daarnaast vergen dit soort oplossingen specifieke skills van de datacenteroperators en specifieke management frameworks om deze units te configureren en te beheren. Dit is geen sinecure en heeft een aardig prijskaartje.

## 8.5. HET KIEZEN VAN DE JUISTE OPLOSSING

In veel gevallen wordt de TCO (total cost of ownership) als meetlat gehanteerd. Het is best wel complex om deze berekening goed te maken en te vergelijken met elkaar. Hoe efficiënter en preciezer de koeling wordt ingezet hoe hoger het prijskaartje van de investering zal zijn. Aan de andere kant is het energieverbruik en waterverbruik van een op vloeistof gebaseerde koeling vele malen minder. Dit komt in een TCO vergelijking dan weer positief naar voren omdat bij vloeistof koeling meer computercapaciteit per m<sup>2</sup> kan worden geconcentreerd.

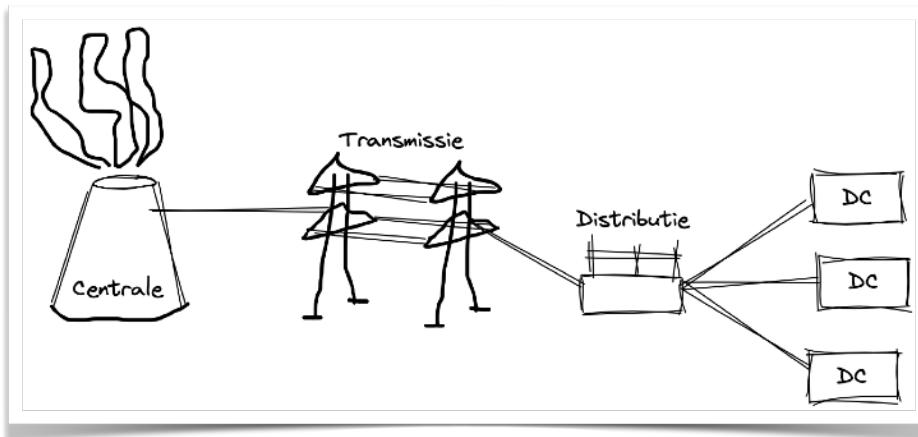
Wij hopen natuurlijk dat datacenters duurzaamheid hoog in het vaandel hebben. Zeker als je als duurzaam bedrijf een eigen datacenter hebt.

Bij het kiezen van een Cloud provider stel de vraag naar Power Usage Effectiveness (PUE) en Water Usage Effectiveness (WUE) als een van de selectiecriteria. Of je er invloed op hebt of niet, die vraag moet je stellen! Je moet duidelijk maken dat duurzaamheid voor jou belangrijk is!

# 9. POWER SUPPLY

Laten we maar meteen met de deur in huis vallen. Servers die op USB-C technologie draaien is voor ons een sub-optimale duurzame oplossing. Nog steeds heb je stroom en een adapter (transformator) nodig om deze technologie om te kunnen zetten in 5, 9, 12 of 15 volt.

Er zitten tegenwoordig overal adapters tussen. Bij je telefoon, laptop, maar ook in keuken apparatuur, hifi, en ga zo maar door. Deze stroom conversies zorgen voor veel energieverlies. Iedere conversie stap is weer een stukje energieverlies. Op onderstaand plaatje is de energieketen geschetst. Van hoogspanning tot spanning die we in ons huishouden gebruiken. Hier treedt ook energieverlies op. Daarna bij gebruik van veel apparatuur waar we de standaard 230 volt weer terugbrengen door adapters naar voor dit apparaat bruikbare voltages zoals gangbare 5 of 12 volt.



*Energieketen en conversiepunten*

Zonnecellen kunnen 12 en 5 volt leveren. Als je de opgewekte energie (spanning) direct de computer kan invoeren heb je geen conversie nodig (en dus geen energieverlies).

Een zonnecel levert ongeveer 1,1 volt. Dus 5 zonnecelletjes voor de componenten die op 5 volt draaien en 12 zonnecelletjes die 12 volt nodig hebben.

Het is maar een gedachte

## 9.1. ZONDER STROOM GEEN DATACENTER

Uiteraard hebben datacenters stroom nodig, veel stroom. De kritische systemen worden vaak uitgerust met dubbele voedingen (power supply units) en hangen daarnaast ook aan de noodstroom voorziening (UPS, uninterrupted power supply system) indien de stroom levering vanuit het grid hapert. Deze houden het ongeveer 5 minuten vol zodat de generatoren (meestal diesel) de noodstroomvoorziening kunnen overnemen.

De stroomvoorziening voor datacenters wordt meestal direct uit het grid afgenoem. Wereldwijd gebruiken de datacenters 1 - 1,3% van alle stroom maar in Nederland is dat maar liefst ongeveer 2,8%! Dit is omgerekend 3,2 miljard kilowattuur<sup>5</sup>.

Logisch dat de politiek zich is gaan bemoeien met de vestigingsstrategie van grote datacenters.

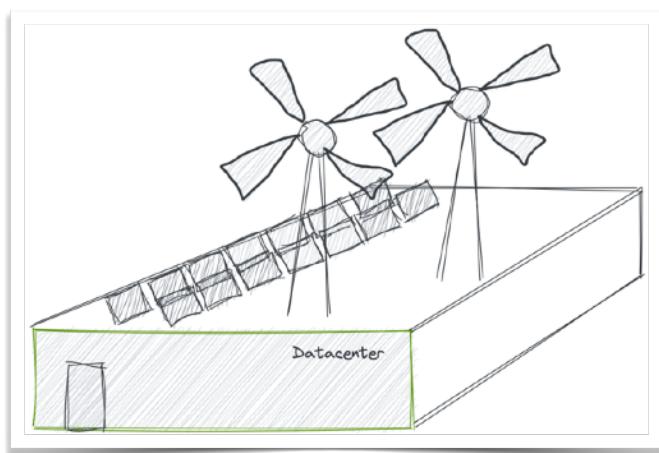
## 9.2. DATACENTER GEBOUWEN

'Beauty is in the eye of the beholder', maar er zullen weinig mensen zijn die worden geroerd door de bekoorlijke eigenschappen van het datacenter gebouw.

Grijs, beton, vierkant, allemaal elementen die een datacenter kenmerken.

Uiteraard valt hier vanuit een duurzaamheid gedachte nog veel te winnen. Waarom bouwen en waarom niet hergebruik van bestaande gebouwen? Platte daken die geduldig wachten om vol gelegd te worden met zonnepanelen voor het datacenter of voor de omwonenden.

Ook esthetisch gezien kunnen we verduurzamen. Als de daken niet met zonnecellen worden vol gelegd dan maar met groen. Naast een manier om de bio diversiteit te stimuleren is dit ook



<sup>5</sup> <https://www.dutchdatacenters.nl/thema-energie/>

nog eens een geweldige isolatie tegen de warmte.

# 10. DATACENTERS EN HYPERSCALERS

W ellicht zou je het terugdringen van het aantal M2 van het datacenter als principe willen implementeren. Hoe minder 'floorspace' hoe minder elektriciteit, koeling, UPS'en en andere basisvoorzieningen behoeven. De hoeveelheid servers worden voor veel datacenters een probleem. De keuze voor distributed systemen (pizzadozen) is ooit gemaakt omwille van horizontale schaalbaarheid. De keuze voor enterprise servers zou een goede oplossing zijn om de computer dichtheid te vergroten. Een datacenter dat geheel of gedeeltelijk direct gevoed kan worden vanuit renewable energiebronnen, zoals zon en wind levert een mooie gedachte op!

Grote datacenters zijn een gevolg van een strategie om centrale controle te hebben over de datacenter diensten en om maximale consolidatie van software (middleware en apps) en data te kunnen realiseren. Datacenters waar enorme hoeveelheden servers draaien (5000 of meer) en groot in omvang en oppervlakte noemen we hyperscale datacenters. Denk hierbij aan de datacenters die door bedrijven zoals Amazone, Google, Microsoft, IBM en Facebook worden gebruikt  
Centralisatie to the max is nu het principe.

Onlangs was er een brede maatschappelijke discussie rond de vestiging van een nieuwe META (Facebook) datacenter in Zeewolde. Van techniek naar politiek. Zelf hebben wij tot een recent verleden nog nooit discussies gehoord of tweede kamer debatten gezien in het licht van datacenter locaties. De begrippen MER (main Equipment Room) en SER (Satellite Equipment Room) zijn vervangen door datacenter of hyperscale datacenters. Maar een datacenter dat net zoveel energie verbruikt als 460.000 huishoudens!! Vandaag de dag kan het niet zo zijn dat een modern datacenter zoveel energie vanuit het netwerk opeist. Datacenters zouden meer autonoom moeten zijn op het gebied van energievoorziening. Meer over dit onderwerp in het volgende hoofdstukje over cloud.

Denk nog even terug aan de inleiding van dit boekje waarbij het voorbeeldje van 'gemak dient de mens' een illustratie geeft van hoe wij worden gevoed met technologie en hoe graag en gemakkelijk we deze adopteren en consumeren. Er zit een groeimodel in onze vraag naar technologie die oneindig lijkt. IT uit de muur leek 20 jaar geleden nog een sprookje. Kijk eens om je heen en tel het aantal IT systemen (telefoon, laptop, horloge, TV, radio, oven, wasmachine.....) binnen handbereik!

Houdt dit nog eens op?

Het is altijd gevaarlijk om steeds te praten over 'vroeger', maar 'vroeger' is onze referentie! Het plaatje van de vader in zijn luie stoel die de TV kan bedienen met een afstandbediening is inmiddels een plaatje geworden waarbij de TV een 'one stop shop' is geworden van streaming diensten, informatie en nieuwszenders en al wat je hebt gemist dat je alsnog kunt bekijken.

Geen platen of CD's meer in de kamer en geen kast vol met videotapes, DVD's of blue ray's. Soms heeft nieuwe technologie ook voordelen :-).

Het is een feit dat diensten voor film en muziek streaming datacenters nodig hebben. Evenals Cloud providers die een plek nodig hebben om hun diensten aan te bieden.

## 10.1. HOE SUSTAINABLE IS CLOUD?

Kortweg: Cloud is vele malen meer duurzaam dan on premise datacenters. In het rapport van Microsoft (Microsoft cloud carbon study 2018) staan veel interessante cijfers. Zo staat geschreven dat de Microsoft Cloud tussen de 22% en 93% meer energie efficiënt zijn dan de traditionele datacenters maar ook dat de CO2 efficiency tussen de 72% en 98% beter scoort.

Veel bedrijven gebruiken inmiddels technologie uit de cloud. In Amerika gebruiken de datacenters gezamenlijk zo'n 75 miljard kWh per jaar (Europa 100 miljard kWh in 2020 volgens ingewijden), dezelfde hoeveelheid van zo'n slordige 6 miljoen Amerikaanse huishoudens volgens een studie van Berkeley National Laboratory. Dit gebruiksgetal zou vele malen hoger zijn zonder de inzet van commerciële datacenters (cloud).

Uiteraard zijn naast Microsoft ook de andere grote cloud spelers actief op het gebied van sustainability.

AWS is naar eigen zeggen 88% meer efficiënt op het gebied CO2 footprint dan traditionele datacenters. Onder andere ook door gebruik te maken van specifieke chip technologie in de computers. De Graviton3 (vergelijkbaar met de chips in de telefoon, nl ARM)

gebruikt 60% minder energie dan andere general purpose chip technologieën.

AWS, Google, IBM, Oracle en Azure (om er een paar te noemen) zijn allemaal druk bezig met duurzaamheid.

Wel een kritische noot dat er door deze partijen veel wordt gewerkt met opgekochte emissie rechten maar over de lengte lijkt de cloud nog altijd duurzamer dan on premise datacenters.

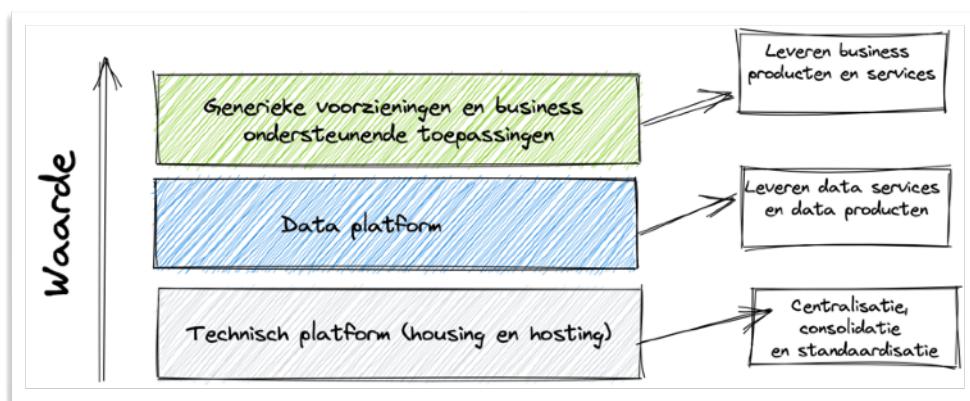
# 11. ODC ALS OVERHEID STRATEGIE

**A**ngezien wij architecten zijn in het overheidsdomein kijken we ook naar wat hier gebeurt op het gebied van datacenters. Onze overheid heeft een centralisatie van de datacenters gerealiseerd middels een Overheids DataCenter strategie. Dit heeft de overheid gerealiseerd door de ca vijfenzestig overheid datacenters in vier datacenters centraliseren. Hiermee is een serieuze besparing op energie behaald maar zou een vervolg traject van de ODC's een echt duurzame ODC kunnen opleveren?

Op dit moment worden de ODC's met name gebruikt voor housing en en hosting maar er zijn nog niet veel diensten die worden aangeboden vanuit ODC's die overheids-breed worden gebruikt. Hier zou wellicht nog een grote slag gemaakt kunnen worden. Denk aan een generieke voorziening voor werkplekken. Die bestaat al! Mooi, maar waarom wordt deze niet door iedereen gebruikt? Daarnaast hebben organisatie uitvoeringsorganisaties als SVB en UWV een 'eigen' DC strategie die niet aansluit op de ODC strategie.

Wij hebben een stukje opgenomen in het volgende hoofdstuk waarbij een wenkend perspectief wordt geschatst voor de ODC's.

Wellicht dat een eenvoudig model kan helpen bij deze discussie?



Leveringsmodel

## **11.1. TECHNOLOGIE PLATFORM**

De ODC's leveren op dit moment voornamelijk een technologie platform. Draai je toepassingen bij ons in het datacenter en je hoeft geen zorgen meer te maken over de voorzieningen voor je platform. Op zichzelf al een mooie stap, maar zou het niet een interessante gedachte zijn om een dienstenportfolio te ontwikkelen vanuit de ODC's die gericht zijn op de eigenschappen van de werklasten zodat platform en applicatie als een generieke voorziening kan worden geleverd. Zo zou je kunnen bedenken dat de mainframe werklasten van de SVB prima zouden kunnen worden ondergebracht in ODC Apeldoorn.

Deze werklasten kenmerken zich als applicaties met hoge transactie volumes ontworpen om op een mainframe te draaien. De technische kennis en kunde (I-vakmanschap) rond dit platform is aanwezig in ODC Apeldoorn, maar tevens de kennis en kunde om deze applicaties te moderniseren en door te ontwikkelen naar een moderne applicaties in functionele blokken in plaats van de traditionele 'stovepipe' applicaties.

## **11.2. DATA PLATFORM**

Hosten van applicaties is al een goede stap maar het wordt nog interessanter als je keten data, data dat tussen verschillende overheid organisaties wordt uitgewisseld kan hosten als dienst. Bij dit data platform wordt data uit verschillende bronnen en voor verschillende doeleinden bij elkaar gebracht en als dienst aangeboden aan de verschillende organisaties. Zo ontstaat er een één overheids data platform waarbij data dat door verschillende organisaties wordt gebruikt centraal kan worden opgeslagen en hierbij kunnen datadiensten ontwikkeld worden die ervoor zorgen dat deze data niet door het hele overheidslandschap hoeft te reizen.

Denk aan data virtualisatie technologie en storage virtualisatie aan huidige moderne technologie om de data footprint te reduceren. Naast centrale platformen zouden er ook domein specifieke datadiensten gehost kunnen worden. Deze diensten worden ontwikkeld door de domeinen (werkgebieden) en kunnen worden gebruikt door andere organisaties

## **11.3. SERVICES PLATFORM**

Bovenop het data platform kunnen uiteindelijk diensten worden ontwikkeld en worden aangeboden. In de overheid wel bekend als

generieke voorzieningen. Denk hierbij aan voorzieningen die door vrijwel alle departementen worden gebruikt maar die in veel gevallen op een eigen specifieke manier zijn ontwikkeld. Hierdoor wordt hergebruik vaak niet mogelijk en dat is toch jammer! Voorbeelden van generieke voorzieningen zijn: werkplek diensten, archivering diensten, ontwikkel omgevingen, etc.

# **12. CODE OF CONDUCT RESPONSIBLE DATACENTERS**

**E**r is zelfs een code of conduct opgesteld voor datacenters door de EU. Organisaties kunnen hier vrijwillig aan meedoen. Deze gedragscode is in het leven geroepen als antwoord op de enorme energieverbruik in datacenters. Deelnemers aan dit initiatief kunnen zelfs in aanmerking komen voor een jaarlijkse EU Data Centres Code of Conduct Awards.

Meer informatie is te vinden op de website:

<https://e3p.jrc.ec.europa.eu/communities/data-centres-code-conduct>

## **12.1. RESPONSIBLE DATACENTER & RESPONSIBLE INFRASTRUCTURE**

Uiteraard zijn deze onderwerpen nauw met elkaar vervlochten. In het volgende hoofdstuk kijken we naar infrastructuur maar af en toe zie je dat er verwezen wordt naar datacenter voorzieningen vanuit infra of andersom vanuit datacenter naar infra voorzieningen.

# 13. RESPONSIBLE DATACENTER PRINCIPES

In dit hoofdstuk worden een aantal voorbeeld principes beschreven die kunnen helpen bij verduurzaming van de datacenters. We gebruiken hiervoor een template uit TOGAF<sup>6</sup>. TOGAF is een bekende architectuur aanpak die door veel bedrijven wordt gebruikt.

Principle RDC1	<b>Maak gebruik van de meeste efficiënte koeling techniek</b>
Statement	Kies voor het datacenter een koeling techniek die recht doet aan de eisen van de infrastructuur.
Rationale	Door de juiste koeling techniek toe te passen voorkom je dat deze op hun tenen moeten draaien om voldoende te koelen. Bij servers waar CPU's en GPU's zwaar worden gebruikt is een andere techniek nodig dan servers waar alleen I/O op plaats vindt.
Implicaties	Bij de inrichting van het datacenter kunnen werklasten met een bepaalde eigenschap worden geconsolideerd zodat de meeste efficiënte koel techniek kan worden toegepast.

Principle RDC2	<b>Maak datacenters zelfvoorzienend in hun energie behoeft</b>
Statement	Datacenters dienen hun energie behoeft zoveel mogelijk te halen uit renewable energie die zijzelf opwekken.
Rationale	Door datacenters te maximaal voorzien van zonnecollectoren, windturbines en andere energiebronnen wordt de druk op de lokale energie centrales minder en worden de datacenters groener.

<sup>6</sup> The Open Group Architecture Framework: <https://www.opengroup.org/togaf>

Principle RDC2	<b>Maak datacenters zelfvoorzienend in hun energie behoefte</b>
Implicaties	De groene energievoorziening dient onderdeel te zijn van het ontwerp van datacenters.

# **RESPONSIBLE INFRASTRUCTURE**



In dit hoofdstuk beschrijven we infrastructuur vanuit een duurzaamheid perspectief. Onze infrastructuur is modulair opgebouwd en zo is het ook met dit hoofdstuk gesteld. Vanuit verschillende aspecten wordt de infrastructuur beschouwd. Als klap op de vuurpijl (niet zo duurzaam) gaan we een dagje de baas spelen van de overheidsdatacenters.

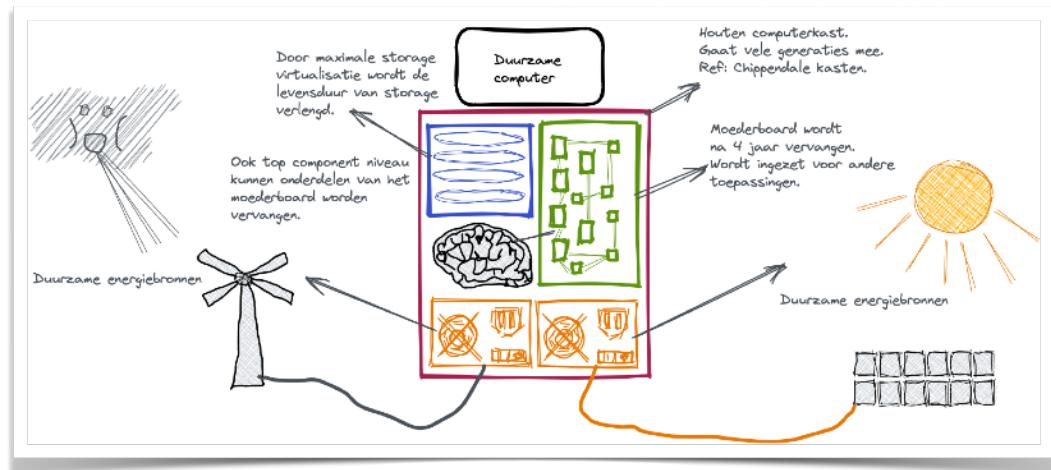
# 14. DE EERSTE ASPECTEN

## 14.1. OMDENKEN

Stel je voor je zou een computer bouwen puur vanuit een duurzame gedachte.....

Om te beginnen bouw je natuurlijk waar mogelijk op basis van duurzame onderdelen.

We beginnen met de kast van de computer. Waarom moeten deze vervangen worden als de technische of economische levensduur ten einde komt. De kast zelf is nog prima en kan nog jaren mee.



Duurzame computer

Natuurlijk, een kast gemaakt van hout!!

Thomas Chippendale (1718-1779) maakte kasten die vandaag de dag nog steeds pronken in musea of bij mensen thuis.

Uiteraard wordt de computer gevoed vanuit duurzame energiebronnen. De voedingen zijn modulair en kunnen bij gebreken als gevolg van leeftijd vervangen worden. Nog sterker als je geen hoge beschikbaarheid eisen hebt voor een dergelijke computer dan zijn de voedingen niet op basis van active-active geconfigureerd maar op basis van active - standby.  
Dit levert de keuze mogelijkheden:

- Zon en wind: je kunt kiezen welke voeding actief is
- Allen zon: Kies voor de rechter voeding
- Alleen wind: kies voor de linker voeding.
- Geen win en geen zon: schakel over op traditionele energievoorziening.

Het moederboard bevat de intelligentie van de computer. Vaak worden computers vervangen op basis van economische afschrijving en niet zo zeer omdat de technologie zoveel verbeterd is. Het tijdperk van de '*Wet van Moore*' (het aantal transistors in een geïntegreerde schakeling elk twee jaar verdubbelen) is geëindigd. Dus de vernieuwing vanwege significante verbetering is niet echt meer geldig.

Het moederboard zou je dus kunnen vervangen als er echt sprake is van technische verbetering of omdat je andere technologie nodig hebt. Bijvoorbeeld naast CPU capaciteit ook GPU capaciteit voor specifieke analytische toepassingen.

Dit geldt ook voor het intern geheugen. Als er meer capaciteit nodig is of als er sneller intern geheugen nodig is is de memory als modulair component te vervangen.

En in het kader van cradle-to-cradle denken zou je de verouderde componenten kunnen hergebruiken in toepassingen die geen hoge eisen stellen aan de technologie. Denk hierbij aan de inzet van processors in bijvoorbeeld auto's, huishoudelijk apparatuur of andere laagdrempelige technologie.

Het is natuurlijk geen serieus voorstel maar hopelijk zal het de lezer prikkelen om eens anders te kijken naar computersystemen ipv de traditionele manier.

Als we op deze manier van denken zitten laten we dan eens kijken door naar de IT wereld door een duurzaamheidsbril.....

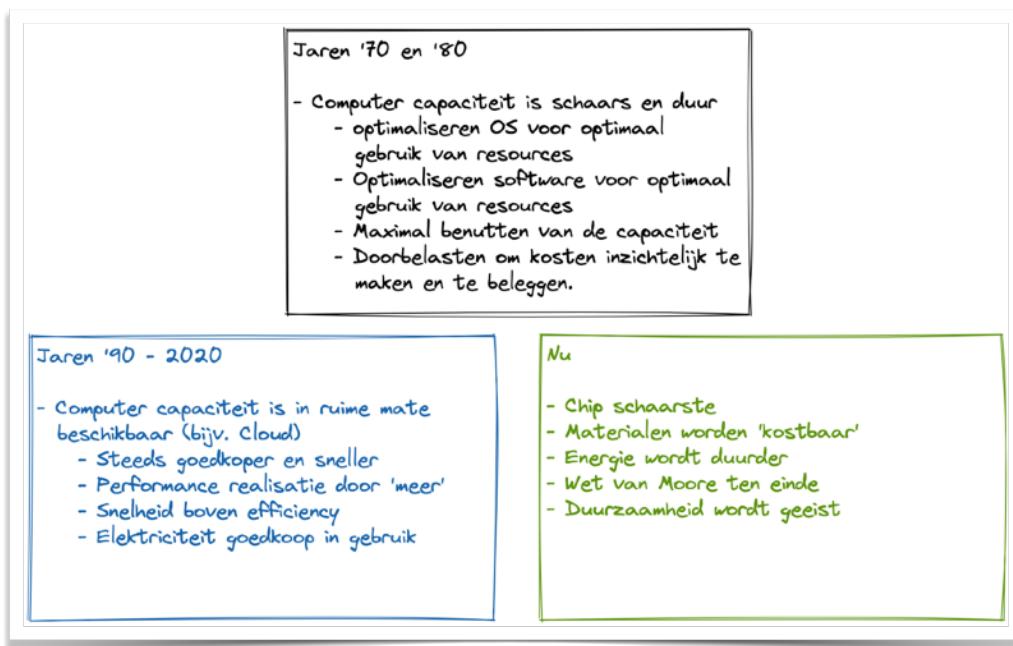
Wij zien een aantal onderwerpen die de moeite waard zijn om eens tegen het licht te houden in het kader van Data Centers en infrastructuur:

- 1. HOE DUURZAAM IS ONZE TECHNOLOGIE?**
- 2. HOE DUURZAAM ZIJN ONZE PROCESSEN?**
- 3. HOE DUURZAAM IS CLOUD?**
- 4. VERNIEUWEN OM TE VERNIEUWEN?**

## **14.2. SCHaarSTE VERSUS OVERVLOED**

Door de jaren heen hebben wij al IT'ers leren omgaan met schaarste en overvloed van IT resources. In de beginjaren van de

IT industrie ('70 en '80) waren capaciteit en middelen schaars en duur. Een RS2-32 kabeltje koste maar liefst Fl 350,- dus was de business case bij het bedrijf waar ik voor werkte al snel gemaakt. Een soldeer apparaat aanschaffen en deze was in een handomdraai terugverdient. Deze schaarste gold ook voor de ontwikkelaars. De programmatuur zo efficiënt mogelijk laten draaien zodat de impact op het middelen wordt beperkt.



### Schaarsheid IT door de jaren

Kun je dit nog voorstellen: Een gebruiker van het mainframe (in dit geval een Bull L66) zag direct bij het aanloggen hoeveel geld hij of zij had verbruikt op het mainframe. Omdat de centrale computer gedeeld moest worden door veel gebruikers was 'Timesharing' een prima oplossing. Hierbij had je veel een centrale computer die werd gedeeld door zgn terminal gebruikers. Een goede oplossing voor ontwikkelaars die veel code moesten editeren en vervolgens kleine testruns konden doen.

By the way dit was ook de periode dat nog lang niet iedereen een telefoon op het bureau had staan. Was toen ook nog een beetje een status symbool.

In de jaren '90 werd de computer schaars overwonnen door de introductie van personal computers. Opeens had iedereen een klein mainframpje ter beschikking. Er ontstond een enorme dynamiek in de ontwikkeling van nieuwe technologie. Ipv hard wired point-to-point verbindingen van terminals naar mainframe werd de local area netwerk topologie geïntroduceerd. Men kon resources delen in kleinere communities en de printer van zo'n community stond lekker makkelijk op de afdeling ipv in de centrale computerruimte. (Dit natuurlijk tot men er achter kwam dat een printer ook niet echt gezond was op de afdeling..)

Client server technologie zorgde voor een behoorlijke modernisatie van programmeren waarbij de logica op de PC draaide en deels op de centrale computer. Het ging schoorvoetend.....eerste de afdelingsmanagers een PC, later de IT'ers (lang niet iedereen). Na de massale introductie van PC's kwam het dilemma van email. Wie mag email gebruiken en wie heeft dit absoluut niet nodig voor het werk....

De technologie werd in razend tempo doorontwikkeld en met name kennis was schaars. De middelen waren door het grote aanbod redelijk betaalbaar maar door gebrek aan standaardisatie in de markt ook zeer complex in gebruik.

Dit had tot gevolg dat IT in roerig vaarwater kwam. Veel verstoringen en instabiele omgevingen. En dit terwijl bedrijven en organisaties steeds afhankelijker werden van de IT voorzieningen. Wellicht dat de outsourcing golf enigszins heeft geholpen bij het weer normaliseren van de inzet van IT middelen. Er was plotseling een partij die de IT diensten leverde op basis van een strak kosten model in plaats van een interne IT organisatie die de kosten vaak uit de hand lieten lopen. Voor iedere handeling wordt betaald en bovendien worden de prijzen marktconform en transparant gehouden wat bij een 'eigen' IT organisatie nog wel eens kon ontsporen. De broodjes van de overwerkers in het weekend werden bijvoorbeeld doorbelast in de firewalls als gemeenschappelijke voorziening.

Energie was totaal geen issue. Dit ging pas een rol spelen op het einde van het eerste decennium in de 21e eeuw: Green datacenters! Dit was een allereerste poging om datacenters met energieslurpende systemen bewust te maken van het energiedrag. Dit waren met name op faciliteiten (airco, ups, verbruik van koelwater) gerichte oplossingen. Het was even een hype maar dit verdween op de achtergrond bij de grote financiële crisis die rond 2008 begon. Alle investeringen op het gebied van IT

kwamen min of meer tot stilstand. Laat staan dat bedrijven oog hadden voor een milieubewuste IT beleid.

Wat is het vandaag de dag toch anders. Schaarste is weer de norm. Gebrek aan grondstof om computerchips te maken, energieprijzen zijn toenhoog en de consumenten eisen dat de bedrijven duurzaamheid omarmen. Dat betekent overigens niet dat daarmee de grote duurzaamheid transformatie een feit is!

***Bezuinigen op energie betekent niet automatisch verduurzamen.....***



*Besparen op energie is  
niet hetzelfde als  
verduurzamen*

We hebben het veel over energiebezuiniging. Daar is niets mis mee, maar is dit nu de juiste motivatie voor verduurzaming? Indien je energie bezuinigt vanuit een economisch aspect dan is de kans heel groot dat je weer in oude gewoontes vervalt als de energieprijzen weer normaliseren. Dan drukt de hoge energieprijs niet meer op de winst of omzet resultaat en zijn we weer terug bij 'business as usual'.

Gelukkig hebben we voor zeer zware toepassingen de Timesharing weer van de plank gehaald. Met de introductie van Quantum technologie kunnen we er rustig vanuit gaan dat dit fenomeen niet een PQC (Personal Quantum Computer) variant gaat krijgen. Gebruikers moeten dus op basis van gedeelde resources gebruik maken van deze technologie.

Wij denken dat verduurzaming moet voortkomen vanuit een intrinsieke motivatie. Dan heb je ook geen Chief Sustainability Officer nodig die wanhopig probeert uit te leggen aan de buitenwereld hoe het bedrijf bezig is met verduurzaming en aan alle kanten probeert te voorkomen dat de markt het bedrijf verdenkt van 'greenwashing'.

Het is dus voor een groot gedeelte cultuur!

Alleen vanuit een 'onbewust bekwaam' modus ontstaat deze intrinsieke motivatie voor verduurzaming.

### 14.3. ENERGIE ZUINIGE SYSTEMEN

Wist je dat er een groot verschil in energieverbruik is tussen de verschillende systeem hardware?

Vooral voor AI is behoorlijk wat rekenkracht nodig. Intelligent gebruik van CPU- en GPU-werklasten kan een aanzienlijke energie besparing opleveren. In tegenstelling tot wat je in de naam zou verwachten, biedt het OpenPower-consortium een technische oplossing die niet alleen open is, maar ook een andere processor architectuur gebruikt, dat is slechts 1/3 van de energie in vergelijking met traditionele processors in je datacenter.

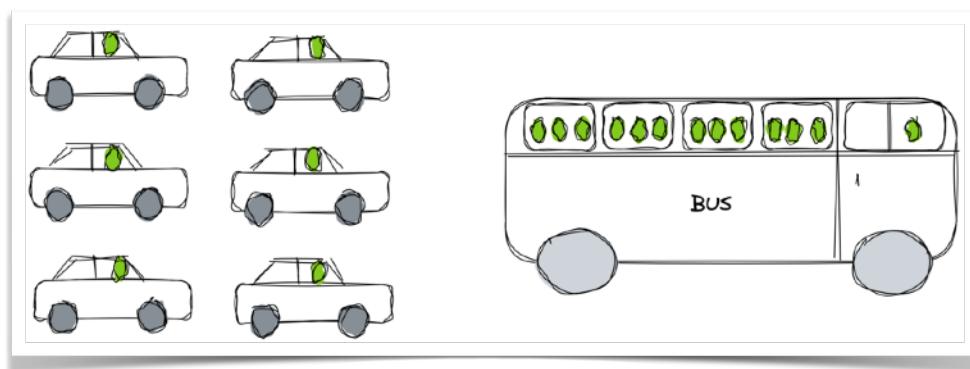
We zijn Intel dankbaar voor het gemak waarmee we onze infrastructuur modulair konden opschalen maar gewenste capaciteit om onze applicaties te draaien en onze ontwikkelaars met een druk op de knop konden voorzien van capaciteit. Herkent u dit ook: "Deze server is van onze afdeling. Hij is aangeschaft vanuit ons projectbudget dus we gaan hem niet delen met een andere afdeling".

Een klein voorbeeldje hoe dit in het verleden ging (maar wij weten vrijwel zeker dat dit nog steeds gebeurt) waardoor systemen sub-optimaal werden gebruikt. Denk alleen al eens aan de fysieke ruimte die deze modulaire oplossing gebruikt: x86 gebruikt voor ca 43 m<sup>2</sup> terwijl een enterprise server (bijvoorbeeld IBM LinuxOne)slechts iets meer dan 6 m<sup>2</sup> nodig heeft. En dan nog het energie gebruik: In deze voorbeeld case van Asia Pacific Insurance Company gebruikte x86 890 kWh terwijl de IBM oplossing niet meer dan 335 kWh nodig heeft om dezelfde prestaties te leveren. Dat is maar liefst 62% besparing op energie en 86% op floorspace. Dit soort cijfers zijn erg interessant voor bedrijven en organisaties die bewust bezig zijn met hun CO<sub>2</sub> footprint.

Kijk, je moet een dergelijk enterprise server niet overwegen als je deze ook niet volledig kan benutten. Het mooie is dat dit soort systemen het meest nuttig zijn als je ze voor de volle 100% kan benutten. Wanneer ontstaan het kantelpunt om te overwegen of een dergelijk systeem aantrekkelijk is voor jouw bedrijf of organisatie?

Vergelijk het eens met een bedrijf dat medewerkers voorziet in vervoer om van kantoor naar een andere werklocatie te rijden. Bij

15 medewerkers is het best nog wel interessant om een wagenpark aan te schaffen met kleine personenauto's en deze door de medewerkers te laten gebruiken pm van A naar B te rijden. Met drie of vier medewerkers per auto kom je op een klein wagenparkje van zo'n vier auto's. Dat kun je nog steeds uitbreiden naar 20 medewerkers maar ergens houdt het op. Dan is de economische voordeel van de aanschaf van kleine auto's niet rendabel meer als je kijkt naar aanschaf, afschrijving, onderhoud verzekering, brandstof, enz. Dan zou je als bedrijf kunnen overwegen om een bus te aan te schaffen en het personeel daarin te vervoeren! Aanschaf is wel hoger maar als je de prijs per stoel uitrekent dan valt de aanschafwaarde weg tegenover het gebruik. Als je de bus kan vullen dan maak je optimaal gebruik van het busconcept!



*Het busconcept*

Consolidatie van werklasten is best een aantrekkelijke gedachte. Zeker als de mate van beveiliging daarmee ook een stuk beter wordt, de ruimte die het systeem gebruikt en bovendien de vermindering van energiegebruik.....zorg overigens wel voor een goed balans tussen consolidatie en afhankelijkheid. Één is geen gaat voor sommige situaties wel degelijk op.

Maar moet je geen specifiek kennis in huis hebben om een dergelijk systeem te kunnen beheren of inrichten. O ja, vast en zeker maar dat is éénmalig bij de installatie van het systeem. Daarna is het gewoon een Linux server die op dezelfde wijze wordt beheerd als gedistribueerde systemen.

Er is kennelijk altijd een soort stammenstrijd gaande tussen gedistribueerde systemen en Enterprise systemen. Wij zien het zo: kies het juiste systeem wat bij je past. Niet distributed omdat het moet, maar hanteer een aantal principes waarop je de juiste keuze kan baseren!

# 15. WERKLAST GEDREVEN DATACENTERS

**W**ij geloven in werklast gedreven datacenters. Datacenters die zich specialiseren in technologie voor bepaalde werklasten om deze zo optimaal (lees efficiënt) mogelijk in te zetten. Een “fit for purpose” en niet een “one size fits all”. Overigens past die “one size fits all” kleding mij nooit, altijd te klein, maar dat terzijde. We willen hier een poging doen om werklasten te mappen op de infrastructuur die hiervoor het meest geschikt is. Het meest geschikt betekent in dit geval ook het meest efficient, het meest duurzaam.

## 15.1. SYSTEMEN

Grofweg zijn er vijf soorten servers te onderscheiden, waarbij de eerste twee weer een paar varianten kennen.

1. CISC gebaseerde servers
  1. Single Servers (ook wel bekend als “pizza dozen”)
  2. Blades
  3. High end
2. RISC gebaseerde servers
  1. Single Servers
  2. High end
3. GPU gebaseerde servers
4. Mainframes
5. Quantum computers.

### CISC en RISC gebaseerde servers

CISC en RISC zijn processor architecturen.

CISC staat voor Complex Instruction Set en RISC voor Reduced Instruction Set. CISC processoren zijn complexer en bevatten tot 3x zoveel transistoren. RISC processoren zijn efficiënter en zuiniger en worden daarom toegepast in telefoons, tablets, huishoudelijke apparaten en zien we ook veel terug in communicatie apparatuur in datacenters. Tot voor kort was CISC de meest populaire processor in datacenters, maar door de opkomst van de RISC processor zien we, vooral bij de hyperscalers een enorme toename van gebruik.

Doordat je bij deze leveranciers gebruik maakt van services is het als gebruiker niet relevant om te weten op wat voor infrastructuur deze service draait.... Bijvoorbeeld de eerder genoemde Graviton3 processor van AWS. Windows draait alleen op CISC en dat is de reden dat er nog veel CISC processoren worden toegepast. Voor het gebruik van een RISC processor ben je aangewezen op een Unix variant van een Operating System.

Dit verschil is fundamenteel belangrijk voor een duurzaamheidsdiscussie omdat RISC veel zuiniger is, maar veelal niet de huidige standaard. Om een duurzaam datacenter te bouwen moet je hier een strategische keuze maken.

### **GPU gebaseerde servers**

GPU staat voor Graphics Processing Unit. Ooit ontworpen om snel 3D processing te doen. Een GPU is in staat om heel veel berekeningen parallel uit te voeren, ideaal als je het volgende plaatje op je scherm moet tonen. Iedere pixel op je scherm wordt gelijktijdig berekend en gelijktijdig getoond. Deze servers zijn daarmee ook uitermate geschikt voor het trainen van een neuraal netwerk voor AI modellen. We noemen het GPU gebaseerd omdat er allerlei varianten zijn ontwikkeld, bijvoorbeeld de Tensor Processing Unit van Google.

### **Mainframes**

Mainframes zijn flinke servers die bekend staan om hun betrouwbaarheid, veiligheid en snelheid in verwerken van transacties. Dit komt omdat bij een mainframe server de componenten in hoge mate geïntegreerd zijn. In het hoofdstuk over architectuur zagen we dat hoge mate van integratie goed is voor energieverbruik maar minder goed voor modulariteit. Dat is precies de afweging die je hier ook moet maken. Bij veel grote bedrijven draaien de belangrijkste bedrijfsprocessen op deze servers. Het beeld kan zijn dat een mainframe groot en duur is maar mainframes zijn er in diverse smaken en prijsklasse. Een Linux mainframe is niet veel duurder dan een flinke high-end server. Dus afhankelijk van de omvang van je organisatie en je duurzaamheidsstrategie kan een mainframe een opties zijn.

### **Quantum computers**

Tot slot nog quantum computers. Een uitermate krachtige computer die zeer veel berekeningen paralel kan doen. Ontwikkelingen zijn nog volop gaande. Hoewel je deze niet in je datacenter zal

neerzetten is er al wel de mogelijkheid om een quantumcomputer via de cloud te gebruiken. Het is nog wel een niche product maar bedrijven die te maken hebben met chemie, medicijnen, medische wetenschap, encryptie technologie, etc. zijn hier volop mee bezig. We zullen de quantumcomputer nu nog niet meenemen in onze beschouwing.

## 15.2. WERKLASTEN

Wat we verder nog zien is dat er steeds meer combinaties van technologieën op de markt komen. Een integratie van CPU en GPU bijvoorbeeld. Maar voor het denkproces trekken we dit even los. Het kan goed zijn dat je voor jezelf een werklast definieert die optimaal gebruik kan maken van zo'n combinatie.

Terug naar de servers en werklasten. Wat zijn relevante karakteristieken voor een server die bepalen welke werklast hier het best op kan draaien?

Relevant voor servers zijn eigenlijk maar een paar eigenschappen.

1. Ten eerste is de verwerkingscapaciteit van de CPU van belang. Hoe snel kunnen instructies worden uitgevoerd?
2. Het tweede dat belangrijk is is de snelheid waarmee externe data opgehaald kan worden om door die CPU verwerkt te kunnen worden.
3. Het derde wat belangrijk is is de connectiviteit van de CPU met externe CPU's. High end servers zijn wat we noemen channel attached. Door deze te koppelen kunnen zeer krachtige systemen gebouwd worden. Supercomputers zijn een voorbeeld hiervan.

In onderstaande tabel zijn voorbeelden opgenomen van veelvoorkomende werklasten met de bijbehorende karakteristieken. Per bedrijfstak kan er meer of minder focus liggen op bepaalde werklasten. Vooral voor grote organisaties is het lonend om onderscheid te maken in de werklasten die er zijn.

Werklast	Omschrijving	Karakteristieken
Collaboratie omgeving	Omgeving waarin medewerkers en ecosystemen met elkaar samenwerken, communiceren en informatie uitwisselen.	Veel parallelle processen. Gemiddelde eisen aan performance. Netwerk snelheid belangrijk.
Transactieverwerkende systemen	Systemen die transacties verwerken zoals betalingen, beurskoersen, bestellingen, uitkeringen, prolongaties.	Serieel process met veel load en daardoor IO dat geprocessed moet worden.
Operationele systemen	Systemen die direct de operatie ondersteunen. Dit kan een IT systeem zijn dat verantwoordelijk is voor het mobiele telefoon netwerk, een omgeving die walsen aanstuurt voor het maken van staal, de lopende band bij een fabriek.	Systemen die dicht tegen de operatie aanzitten. Veelal realtime behoeften. Communicatie met periferie. Zeer hoge betrouwbaarheidseisen. Security vaak door ontkoppeling bedrijfsnetwerk.
Rapportages	Omgeving die standaard of ad-hoc rapporten maken. Data wordt gehaald uit een data warehouse.	Queries op veelal gestructureerde data. Processor intensief, maar een rapport maken mag even duren.
Informatievoorziening (Web sites / Self Service)	Internet sites die veel gelijktijdige gebruikers verzoeken af moeten handelen.	Veel gelijktijdig gebruik. Hoge mate van parallelisatie mogelijk. Grilige gebruik. Horizontale schaalbaarheid van belang.
Data analytics	Analyse die gedaan worden op grote hoeveelheden data.	Analyse op data is CPU/GPU intensief. Mogelijk ook ad-hoc, (near) real time. Kan zeer hoge eisen stellen aan systemen. Bijvoorbeeld uitvoeren op real time data.
Content beheer	Omgevingen voor het opslaan van grote hoeveelheden ongestructureerde data zoals documenten, video's en foto's.	Nadruk ligt op efficiënte opslag. Niet zo zeer afhankelijk van de CPU. Kan goed parallel werken en heeft baat bij horizontale schaalbaarheid.
Ontwikkel-platformen	Platformen voor ontwikkelaars van systemen.	Gaat vooral om de snelheid waarmee het platform ter beschikking wordt gesteld en grote behoefte aan snelle processoren en veel memory.
Innovatie-omgeving	Omgevingen om iets uit te proberen, te testen en prototypes bouwen.	Uitprobeeromgeving. Vergelijkbaar met het ontwikkelplatform met dien verstande dat CPU snelheid waarschijnlijk minder relevant is. Isolatie van productie kan een voordeel zijn.
Eindgebruikers-omgevingen (burgers en bedrijven)	De werkplek. In een aantal gevallen virtueel.	De werkplek. In sommige gevallen wordt er gebruik gemaakt van een virtuele werkplek, bijvoorbeeld op basis van VDI.

De tabel op de volgende bladzijden geeft de karakteristieken van de servers (tabel links) en de relatie tussen servers en werklasten (tabel rechts). Welke servers passen bij welke werklast.

Server	CPU Capaciteit	IO - Bandbreedte	CPU Koppelingen	Energie
CISC - Single	++ (16 cores, 3,7 Ghz) (AMD Ryzen)	--	--	--
CISC - Blades	++	-	-	-
CISC - High End	++	+/-	+	-
RISC - Single	++ (16 cores, > 4 Ghz) (Power 10)	+/-	--	+
RISC - High End	++  (Tot 480 cores)	+/-	++	++
GPU	-	-	++	+/-
Mainframe	+ (8 cores, >5 Ghz) (Telum)	++	++	+

<b>Server</b>	<b>Werklast</b>	<b>Voorbeeld Systeem</b>
CISC - Single	Ontwikkelplatformen Innovatieomgeving Afgeschermd server in security zone	Lenovo ThinkSystem SR630 7X02
CISC - Blades	Collaboratie omgeving Eindgebruikersomgevingen VDI	HPE Blade System c7000
CISC - High End	Operationele systemen Informatievoorziening (Web sites / Self Service) ERP systems Content beheer	Atos Bullion Servers
RISC - Single	Ontwikkelplatformen Firewall, Routers, Switches Appliances	ARM Server
RISC - High End	Operationele systemen Rapportages Data warehouses Content beheer	IBM Power E1080
GPU	Data analytics AI Model Calculation, Graphics	Nvidia
Mainframe	Transactieverwerkende systemen	IBM Z16

Uiteraard geeft deze tabel maar een indicatie. Een werklast die gebaat is bij verticale schaalbaarheid zal vooral kijken naar high-end RISC servers of een mainframe. Als daarnaast I/O belangrijk is dan past de mainframe beter. Is CPU snelheid belangrijk dan is het RISC of CISC systeem de betere keuze. Hoewel de CISC en RISC strijd in volle gang is, is deze eigenlijk al beslist in de datacenters van de hyperscalers en op de mobile eindgebruikersapparaten, waar in beide gevallen RISC processoren massaal omarmt worden.

## 15.3. DE WERKPLEK

We gaan in ons boekje nauwelijks in over het onderwerp de werkplek. Steeds vaker zien we een BYO (Bring Your Own) beleid. Neem mee wat jezelf prettig vindt. Er wordt dan een virtuele werkplek geboden. Eenvoudig uitgelegd installeer je een applicatie op je zelf meegebrachte device die toegang geeft tot een werkplek die draait op een server in een datacenter. Het voordeel is dat dit veilig is en je niet allerlei devices hoeft te ondersteunen en bij te werken. Wel één groot nadeel: je hebt een computer op je bureau staan die energie kost en je hebt een werkplek draaien die energie kost. Niet efficient dus.

Het tweede aspect is de keuze van het device zelf. Een nieuwe M1 (RISC) gebaseerde Apple is 3x zo zuinig dan een nieuwe Intel/AMD (CISC) gebaseerd systeem. Wat doe je met die wetenschap. Datzelfde geldt voor Chromebooks, als je toch in de cloud werkt... Tsja nu je dit weet moet je her toch wel even over nadenken als je een bedrijf hebt met wat medewerkers.

Een laptop gebruikt tussen 150 en 300 kWh per jaar. Een Macbook ongeveer 1/3, dat is 50 tot 100kWh, dat scheelt dus 100 - 200 kWh. Heb je 20 medewerkers dan hebben we het hier over het stroomverbruik van één huishouden. En als je nu denkt, maar dat hoef ik toch niet te betalen als bedrijf, ga dan terug naar bladzijde één van dit boekje en begin opnieuw met lezen.

# 16. NOG MEER ASPECTEN

## 16.1. HOOG BESCHIKBAAR ALS STANDAARD

Het gemak waarmee wordt omgesprongen met onze IT middelen is soms wel verontrustend. Vaak worden vanuit een beheersgemak patronen toegepast die helemaal niet nodig zijn. Een organisatie heeft een twin datacenter en een prachtige IT inrichting die er voor zorgt dat het bedrijf 7x24 operationeel beschikbaar is.

Bij het inrichten van de omgeving voor een nieuwe applicatie van de afdeling A heeft de IT afdeling begrepen dat het om een belangrijke applicatie gaat. Deze wordt dus (want IT'ers hebben het beste voor met het bedrijf) als hoog beschikbaar ingericht. In IT termen active-active. Met andere woorden als bij de ene datacenter problemen ontstaan dan draait de omgeving zonder enige verstoring door in het andere datacenter. Er is uitgegaan dat men absoluut geen data mag verliezen en dat de operatie ook niet verstoord mag worden. In IT noemen ze dit RPO = 0 (data verlies is 0) en de RTO = 0 (tijd om operationeel te worden is 0).

En ja, het gaat om een echt belangrijke business applicatie en niet om de verzamelde recepten van de hobby kookclub van de afdeling marketing,

In een zeldzame sessie tussen de business owner en het team en de IT dienstverlener blijken deze beschikbaarheid aannames helemaal niet te kloppen. Nog sterker de applicatie mag er wel een dagje uit liggen en de business weet precies van wie de data de laatste 24 uur is verwerkt. Dus kunnen we makkelijk vragen om het nog een keer op te sturen.....

Wij als IT'ers weten dat dit natuurlijk geen optimale oplossing is en dat er prima oplossingen zijn om deze data te bufferen om later te verwerken, maar toch!

Het laat zien dat we soms gemakshalve (met de beste bedoelingen) een enorm beslag leggen op IT middelen. De consequentie voor ons duurzaamheid beleid komt op deze manier aardig op de tocht te staan.

Maar wees eens eerlijk: hoe vaak gebruiken de IT organisaties bepaalde patroneren omdat dit lekker makkelijk is vanuit een beheersaspect?

Hier valt nog veel te winnen!

Als we naar de toekomst van IT kijken dan gloort er wel een wenkend perspectief. Applicaties zijn cloud native ontwikkeld en draaien in containers. De infrastructuur is slim genoeg om te bepalen waar een applicatie moet draaien, de beschikbaarheid is op applicatie niveau geregeld en de capaciteit wordt automatisch op- en afgeschaald.

Dit vraagstuk is dan vrijwel geheel opgelost!

## 16.2. OPSLAG

Gegevens vereisen ook veel opslagruimte. Overweeg het gebruik van tape indien mogelijk, want tape technologie is een zeer duurzaam medium. Op de een of andere manier is tape verdwenen uit het IT landschap. Misschien omdat men het ouderwets vindt of omdat het niet voldoet aan een waarschijnlijk niet-bestaaende eis (non-functional requirement) dat de data binnen een 'split second' beschikbaar moet zijn. Als het gaat om archiefdata bijvoorbeeld dan kan de kenniswerker ook wel even 17 seconden (data retrieval tijd voor tape) wachten tot de gegevens van het opgeslagen dossier beschikbaar zijn.

Ook opslag-compressietechnologie kan erg handig zijn om de benodigde hoeveelheid opslagruimte te verminderen.

Uiteraard zijn oplossingen als data virtualisatie ook belangrijke ontwikkelingen om de storage footprint terug te dringen. We

kopiëren ons een slag in de rondte als het gaat om data. Kopie van kopie wordt opgeslagen om te gebruiken in testomgevingen, machine learning, BI, backup en ga zo maar door.

Met technologie als data virtualisatie maar ook data fabric (zie uitleg in het boekje 'how to build an architecture blueprint for a data-driven organisation') wordt onnodig data 'heen en weer gesleep' en kopiëren teruggedrongen. Door veel meer data processing bij de bron te doen alleen relevante data uit bronnen te ontsluiten voorkom je dat complete datasets continue moeten worden gekopieerd om te gebruiken bij andere processen.

## 16.3. INNOVATIE: 2NM (NANO METER) TECHNOLOGIE

2nm zijn maar twee letters en een cijfer. Het lijkt niets en het is nog minder. 2nm staat voor twee nanometer. Dat is klein, héél erg piepklein.

Tijdens de Covid-19 pandemie heeft het research en development center van IBM niet stilgezeten. Zij hebben een chip ontwikkelt die 45% betere performance levert terwijl deze technologie maar liefst

75% minder energie verbruikt. Kortom de levensduur van je smartphone verviervoudigen!

Wat dacht je van één keer in de vier dagen je smartphone nog maar opladen.

De energie winst zit hem in de grootte (eigenlijk kleinte) van de chip. Alle transistors bij elkaar, zo'n 50 miljard transistors op een chip ter grootte van een nagel, zitten heel erg dicht op elkaar. Zo kan de verwerking van elektrische stroomjes supersnel worden verwerkt. Geen 'lange' afstanden tussen de verschillende transistoren. Afstand is natuurlijk een relatief begrip in deze context!

Wij gaan niet in detail uitleggen hoe deze technologie werkt, maar het spreekt voor zich dat dit soort ontwikkelingen enorm bijdragen aan het duurzaamheid dossier in IT.

## 16.4. TECHNOLOGY PUSH

Naast het verdwijnen van tape uit het IT landschap zien wij ook steeds vaker dat de IT'ers van vandaag af willen van batch processing. Real time verwerking van informatie is natuurlijk veel leuker, spannender en moderner dan het verwerken in batches. Maar is dit ook altijd nodig vanuit een bedrijfsvoering perspectief? Je kunt prachtige oplossingen bedenken waarbij ieder stukje informatie wordt opgepakt en automatisch op georchestreerde wijze door het IT landschap wordt geloosd. Dit in plaats van de informatie verzamelen en verwerken op het moment dat er weinig bedrijfsactiviteiten zijn, meestal 's nachts.

Zoals beschreven in het onderdeel schaarste versus overvloed was batch processing ten tijde van schaarste een geweldige oplossing. Overdag raadplegen van systemen en invoer nieuwe data door de gebruikers en 's nachts verwerken. In plaats van naast elkaar. Op deze manier hoef je niet alle systemen te schalen op piekbelasting maar meer op continue belasting. Veel minder energie.

## 16.5. DUURZAME PROCESSEN

Wie kent ze niet de architectuur principes 'Always on' en toegang tot systemen op basis van 'Anytime anywhere'? De impact van dit soort principes op een duurzaam IT beleid is groot. Wellicht dat het goed is om iets genuanceerder te kijken naar dit soort principes en starten met het onderscheid te maken tussen systemen die ECHT altijd aan moeten staan en systemen die helemaal niet altijd aan hoeven te staan.

Denk hierbij aan systemen waar ontwikkelaars mee werken. Ontwikkel omgevingen, test omgevingen, acceptatie. Omgevingen, etc., maar ook systemen die worden gebruikt voor machine learning en analytics doeleinden.

Deze systemen staan 24x7 ter beschikking van ontwikkelaars en projecten maar ze worden vaak niet gebruikt na kantooruren. Ze kunnen dus ook uitgezet worden en 's morgens weer opgestart zodat ze weer klaar staan voor gebruik wanneer de mensen overdag weer aan het werk gaan. Het zijn vaak kleine dingen die



*De tekst op deze foto is bedoeld voor veiligheidsredenen maar bij duurzaamheid krijgt het een andere betekenis.*

ertoe kunnen bijdragen dat gedrag en bewustzijn over duurzaamheid een stap in de richting kan betekenen in het 'bewust-bekwaam' worden.

Het meten van systeem activiteit is met huidige technologie zeer gemakkelijk. Daarnaast kun je scripts activeren die een power down doet van deze systemen bij inactiviteit.

Een timer zorgt dat 's morgens een script wordt geactiveerd om de systemen weer op te starten.

## **16.6. SYSTEMEN AFSCHRIJVEN**

Vaak wordt een standaard afschrijving gehanteerd van 4 jaar. Dan is de technologie zowel technisch als financieel afgeschreven. Is dat wel nodig? In ieder geval zou je kunnen overwegen om de technologie langer te gebruiken in bijvoorbeeld niet-kritische omgevingen.

In de engelse taal is duurzaamheid sustainability. In de muziek wordt de term sustain ook gebruikt. Het is het effect om te zorgen dat de noot nog doorklinkt nadat de snaar is bespeeld of de toets op de piano is bespeeld. Doorklinken dus eigenlijk of langer duren. Dat is precies ook wat wij voor ogen hebben als we kijken naar gebruik van infrastructuur. Langer gebruik maken van de technologie omdat deze nog prima functioneert. De vervanging werd in het verleden vaak gedaan omdat na deze periode kans op verstoringen gingen ontstaan. De harde schijf was niet meer vooruit te branden of het systeem werd onbetrouwbaar. Eigenlijk zijn dat klachten en zorgen die bijna niet meer voorkomen. Systemen zijn ongelooflijk betrouwbaar en gaan zelden of nooit stuk.

# 17. EEN DAGJE BAAS VAN DE ODC'S

**S**tel je voor dat wij een dagje de baas konden zijn over de ODC's.... Hoe zouden wij deze kans aangrijpen om een invulling te geven aan de ODC's? Uiteraard doen we dit primair vanuit een duurzaamheid gedachte en een veiligheid gedachte. De superprincipes sustainable by design en secure by design staan bij ons voorop!

## 17.1. WERKLAST GEDREVEN INRICHTING

Als we de vier locaties nu eens indelen en inrichten op basis van type werklasten (werklasten, toepassingen, applicaties) dan krijgen we een hele andere strategie. Namelijk, bepaalde type werklasten worden geconsolideerd en gecentraliseerd dan zou je de onderliggende infrastructuur ook kunnen inrichten conform deze type werklasten.

Je zou bijvoorbeeld onderscheid kunnen maken in informatiesystemen die worden gebruikt door kenniswerkers op het gebied van beleidsvorming, dossier verwerking bij uitvoeringsorganisaties en systemen ter ondersteuning van de WOO. Voor WOO (WOB) verzoeken wordt zeer geavanceerde toepassingen gebruikt die op basis van AI modellen de specialist voor WOO moet ondersteunen. Om deze systemen goed te gebruiken moet je ongeveer zijn afgestudeerd als astronaut!

Aan de andere kant heb je systemen die specifiek zijn gericht op werklasten met grote transactievolumes. Denk aan uitvoeringsorganisaties al UWV, SVB, Douane en Belastingdienst. Dit zijn systemen waar de meeste transacties (aangiftes, WW uitkering, AOW of Kinderbijslag) zonder menselijke interventie worden uitgevoerd. Alleen de afgekeurde transacties (aangifte met fouten, onvolledigheid, etc) worden 'uit het proces geworpen' en worden als dossiers verder behandeld door de kantoor / kennis medewerkers. De transacties die zonder menselijke interventie worden afgewerkt noemen we 'straight through processing'. De systemen die dit soort grote volumes verwerken zijn over het algemeen grotere enterprise systemen zoals een mainframe.

Vervolgens kun je denken aan een ODC die specifiek is ingericht voor applicatie ontwikkeling en testing. Hier heb je een mogelijkheid om systemen in te zetten bij calamiteiten in de

productie omgeving nieuwe versies van middleware of applicaties te testen en ketentesten uit te voeren.

Systemen voor analytische doeleinden mogen natuurlijk niet in dit rijtje ontbreken. Hiervoor zou je ook een ODC kunnen inrichten waarbij modellen worden getraind en getest op basis van data, data warehouses worden gebruikt voor analytische data en rapportage en waar partijen zoals CBS data kan gebruiken voor statistische informatie.

## **17.2. NU MET DE DUURZAAMHEID BRIL OP:**

Door de systemen in te richten op basis van type werklasten ontstaat er een grote mate van standaardisatie. De systeem kenmerken worden immers voor een groot deel bepaald door de non-functional requirements van de bovenliggende applicatielaag. Door deze standaardisatie ontstaat ook de mogelijkheid tot massale consolidatie. Deze type werklasten kunnen overheidsbreed worden aangeboden vanuit één locaties op éénduidige wijze. Uiteraard kun je nog onderscheid maken in type gebruikers waarbij een beleidsmedewerker een andere werkwijze volgt (proces) dan een kantoormedewerker in een uitvoeringsorganisatie. Het grote voordeel kan ook worden gevonden in het functioneel gebruik van dit soort toepassingen. Door processtandaardisatie wordt de informatie ook op éénduidige wijze in beheer genomen en opgeslagen. Door bijvoorbeeld metadata (beschrijving van de data) te standaardiseren voor specifieke gebruikersgroepen wordt het indien gewenst ook weer gemakkelijker om te voldoen aan WOO verzoeken omdat deze op metadata niveau kan worden teruggevonden.

Resultaat is minder diverse systemen (lees werklasten) en daardoor minder infrastructuur footprint zoals servers. Daarnaast wordt de lifecycle management (nieuwe versies van systemen) beter beheersbaar en kun je de specialistische kennis (I-vakmanschap) van deze systemen concentreren.

De ODC waar de ontwikkelomgevingen staan wordt een absoluut voorbeeld van duurzaamheid. Wij hebben jarenlang principes als on-demand, always-on en andere 'gemak dient de mens' omarmt met het oog op flexibiliteit. Maar flexibiliteit heeft een prijs. Stel je voor dat we technologie inzetten om te scannen (meten) of systemen worden gebruikt door ontwikkelaars in het datacenter.

Zodra er geen activiteit is geweest het laatste uur wordt het systeem gedeactiveerd en uitgezet. Zodra de omgeving weer gewenst is kan deze weer worden geactiveerd.

Het zijn natuurlijk maar losse ideeën, maar als hier serieus naar zou worden gekeken dan zou de overheid datacenters een geweldige voorbeeld functie kunnen zijn voor duurzaam IT.

<b>Werklast</b>	<b>Lokatie</b>
Collaboratie omgeving	ODC Haaglanden / Cloud
Transactie-verwerkende systemen	ODC Belastingdienst
Operationele systemen	ODC Amsterdam Eigen beheer: RWS, Politie, ...
Rapportages	ODC Belastingdienst
Informatie-voorziening (Web sites / Self Service)	ODC Noord
Data analytics	ODC Belastingdienst
Content beheer	ODC Haaglanden
Ontwikkel-platformen	ODC Noord / Cloud
Innovatie-omgeving	ODC Noord / Cloud
Eindgebruikers-omgevingen (burgers en bedrijven)	ODC Haaglanden

# 18. RESPONSIBLE INFRASTRUCTURE PRINCIPES

Hieronder staan een aantal voorbeeld principes voor infrastructuur.

Principle RI1	<b>Consolideer werklasten waar mogelijk</b>
Statement	Consolideer werklasten met dezelfde eigenschappen op enterprise servers en standaardiseer de onderliggende infrastructuur
Rationale	Door werklasten te identificeren met hetzelfde karakter creëren we patronen voor standaardisatie. Door de werklasten te consolideren op enterprise servers reduceren we het aantal fysieke servers die sub-optimaal worden benut.
Implicaties	Een keuze zal moeten worden gemaakt bij de investeringsronde voor server technologie. Er komt een ander investeringsplaatje kijken bij enterprise servers.

Principle RI2	<b>Always off is de standaard</b>
Statement	Zet systemen die niet worden gebruikt uit en bespaar onnodig energieverbruik.
Rationale	Systemen zijn altijd beschikbaar terwijl deze soms helemaal niet worden gebruikt. Door werklastconsolidatie op een slimme manier te doen worden systemen doelgebonden ingezet. Hierbij kunnen keuzes als continue gebruik of gebruik op aanvraag een rol spelen.
Implicaties	Systemen worden beschikbaar gemaakt op basis van vraag. Inactieve systemen worden uitgezet en zodra deze weer beschikbaar zijn voor gebruik worden de activiteiten gemeten. Zodra de systemen weer inactief zijn worden deze uitgezet.

Principle RI3	<b>Verleng de technische levensduur van de infrastructuur</b>
Statement	Door werklasten te consolideren op enterprise systemen wordt het aantal m2 en stroomverbruik sterk gereduceerd.
Rationale	Tijdens de ontwerpfase wordt er nagedacht over duurzaamheidsaspecten van componenten. Per component wordt er iets gezegd over de herbruikbaarheid en de energieconsumptie.
Implicaties	Het verdient extra aandacht om een applicatie onafhankelijk van de onderliggende technologie te bouwen.

# RESPONSIBLE CODE



Duurzaam programmeren is ook bekend onder de naam “green coding”. Toen we dit voor het eerst hoorde keken we elkaar aan met grote vraagtekens in nog grotere ogen. Gaan we hier het verschil mee maken? Wanneer we het hebben over duurzaam programmeren hebben we het over het voortbrengingsproces van de code, het schrijven en bouwen van de applicatie zelf en het draaien van de applicatie in productie. Het gaat niet over de toepassing van de applicatie, dat wordt beschreven in het hoofdstuk Responsible Impact.

# 19. GREEN CODING

**H**et gaat bij “green coding” vooral over resource gebruik, energieconsumptie. Als we het hebben over herbruikbaarheid van code doen we dat vooral met het oog op voorkomen van de energieconsumptie die nodig is om code te ontwikkelen.

Het programmeren heeft ook een ethische kant, deze wordt besproken in Responsible Systems. Programmeren heeft ook te maken met dataprivacy, deze wordt besproken in Responsible Data Usage. De focus hier is dus het gebruik van computers als schaarse resource en de energie die we gebruiken voor het ontwikkelen en testen van applicaties en de efficiëntie van applicaties in productie.

Nu valt het in de praktijk niet mee om het energieverbruik van software te meten. Er zijn wel initiatieven en ons advies is om op een pragmatische manier te meten en je in eerste instantie vooral te richten op verhoudingen tussen applicaties en niet per se op de absolute getallen.

Een tweede hulpmiddel om energieverbruik te meten is software performance. Software die sneller is gaat efficiënter om met resources in vergelijking met software die langzamer is. Deze twee kwaliteitseisen zijn dus niet tegengesteld, maar werken bijna altijd dezelfde kant op.

## 19.1. PROGRAMMEER OPERATING MODELLEN

Laten we beginnen om kaders aan te brengen. Grofweg zouden we programmeren in een aantal operating modellen kunnen splitsen:

1. **Embedded programmeren.** Dit is het programmeren van koelkasten, koffiezetterapparaten, auto's, etc. Ook gerelateerd aan de Internet of Things (IoT). Dit is een uitstekend voorbeeld voor andere operating modellen als het gaat om minimalistisch gebruik van resources.
2. **Gaming, Mining, Movies, VR and 3D modelering.** Deze modellen zijn enorm CPU en GPU intensief. Enerzijds zit hier een belangrijk educatief element in. Deze is zeker voor bedrijven interessant. Anderzijds betreft dit ook de ontspanningsindustrie. Dit is een op zichzelf staand business model. Je zou dit kunnen vergelijken met vliegen voor vakantie, marketing. We kunnen prima zonder dit businessmodel en de toepassing bied geen voordelen

voor het milieu. Dit zijn dus activiteiten waarmee we zonder kunnen. Het vraagt daarom een ander afrekenmodel. De druk op het milieu van deze activiteiten zou als een financiële compensatie terug moeten komen. Bijvoorbeeld het afkopen van CO<sub>2</sub> uitstoot bij vliegreizen.

3. ***Applications and apps development.*** We hebben het hier over het bouwen van applicaties die de bedrijfsvoering ondersteunen. Anders dan bij het vorige voorbeeld kun je applicaties bouwen die juist milleudoelstellingen ondersteunen. Denk bv maar aan het aanvragen van een vergunning. Wanneer je dit digitaal kan afhandelen bespaar je papier, gebouwen en reiskosten.

4. ***Software development.*** Hier gaat het om het ontwikkelen van software pakketten voor een grote markt. Bijvoorbeeld SAP, IBM WebSphere, SalesForce, etc. We zullen niet expliciet naar deze modellen kijken maar er is overlap met het derde model. Omdat het over pakketten gaat waarbij herbruikbaarheid belangrijk is zien we dat veel van deze pakketten veel focus hebben op performance. Jaar over jaar is de performance van WebSphere verbeterd om op deze manier de concurrentie voor te blijven. Deze verbetering zorgt voor een duurzamer product.

5. ***AI development.*** Steeds vaker wordt AI ingezet en het ontwikkelen van een AI modellen is echt anders dan de ontwikkeling van applicaties. In ons boek dat gaat over een blueprint voor een Data gedreven organisatie gaan we kort in op ModelOps. Het voortbrengingsproces van AI modellen. In het hoofdstuk Responsible Systems behandelen we ModelOps nu uitgebreider omdat dit essentieel is voor het ethische aspect voor het bouwen van modellen.

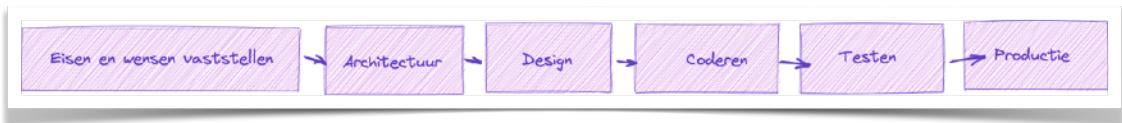
6. ***Quantum development.*** Quantum is een andere tak van sport. De ontwikkelingen gaan razendsnel, maar vooralsnog is deze ontwikkeling vooral interessant voor bepaalde domeinen.

	<b>Operating model</b>	<b>Impact</b>	<b>Mindset</b>	<b>Remarks</b>
1	Embedded programmeren	Zeer kleine CPU footprint  Veel devices	Energie efficiëntie is noodzakelijk part of the design	Wat kunnen we hiervan leren?
2	Gaming, Mining, Movies, VR and 3D modelering	Grote CPU/GPU footprint	Focus op resolutie en performance is noodzakelijk	

3	Applications and apps development	Medium CPU footprint	Focus on functionality	
4	Software development	Medium footprint	Inherit focus on performance, security and footprint	
5	AI development	Large CPU/GPU footprint	Focus on functionality	Maakt gebruik van relatief dure ontwikkelaars
6	Quantum development	Large power footprint Some devices	Focus on applicability	Grote koelinstallaties noodzakelijk.

We gaan in op de modellen 3 en 5. Dit zijn de meest voorkomende modellen die in het alledaagse leven gebruikt worden binnen bedrijven. Model 4 zal veel weg hebben van modellen 3 en 5. De andere modellen zijn vrij specifiek.

## 19.2. SOFTWARE ONTWIKKELING



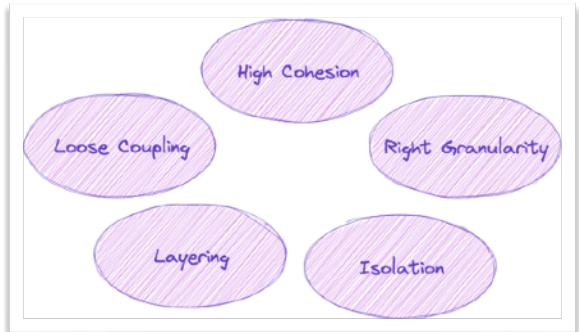
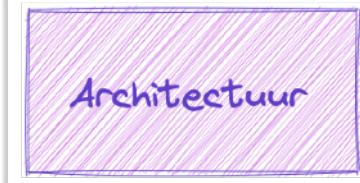
Software ontwikkeling ziet er grofweg uit zoals we dat in bovenstaand plaatje hebben weergegeven. Stel, we hebben het programma van eisen gekregen van de business en gaan bezig met de architectuur, het design, coderen, testen en in productie brengen van de applicatie. Dus alles wat zich in en om de DevSecOps pipeline afspeelt. De architect en de programmeur spelen hierbij een essentiële rol. Naast een hoofdstuk over de architect over de hoeder van duurzaamheid hebben we daarom een hoofdstuk toegevoegd over de programmeur: De uitvoerder van duurzaamheid. In dit hoofdstuk gaan we in op wat er gedaan moet worden door de architect en de programmeur in de verschillende fasen van het software ontwikkeltraject. We zullen achtereenvolgens ingaan op wat er in de architecturfase moet

worden gedaan, wat belangrijk is in de design fase, waar rekening mee moet worden gehouden met coderen en testen (de ontwikkelstraat) en wat er mogelijk is in productie.

# 20. ARCHITECTUUR

Wanneer de business haar eisen heeft vastgesteld voor de software die gebouwd gaat worden zal de applicatie architect een componenten model gaan maken. Dit proces bestaat uit een aantal fasen: De eerste stap is om de componenten te identificeren. Vaak wordt dit gedaan op basis van referentie architecturen. De volgende stap is om ieder component en de noodzakelijke verbindingen te specificeren. De derde stap is om te bepalen hoe componenten gerealiseerd zullen gaan worden. Wordt er gekocht of wordt er iets zelf gebouwd? Tijdens dit proces van component modelering worden er vijf principes gehanteerd:

1. Loosly coupling – zijn de componenten zoveel mogelijk autonoom en is de communicatie zo los mogelijk.
2. High cohesion – Is er logische samenhang van de functies binnen één component
3. Layering – Is er duidelijk sprake van generieke componenten (die makkelijk gekocht of hergebruikt kunnen worden) en specifieke componenten, vooral bedoeld voor de specifieke toepassing
4. Granulariteit – Hebben de componenten een juiste vorm van fijnmazigheid. Zijn ze allemaal even groot?
5. Isolatie – Zitten er technische afhankelijkheden in die we liever in een aparte component willen onderbrengen.



Als je deze vijf principes op je in laat werken is het duidelijk dat het modelleren vooral bedoeld is om een complex systeem in stukje op te hakken op een zodanige manier dat de componenten waaruit dit systeem wordt opgebouwd **hergebruikt** kunnen worden. Nu is herbruikbaarheid in het kader van duurzaamheid een goed principe. Tegelijkertijd zien we dat herbruikbare componenten een communicatie overhead introduceren die qua energieverbruik juist weer extra kosten met zich meebrengt. Vergelijk de "monoliet" met

de microservices architectuur. De REST API calls tussen de microservices zijn veel minder efficiënt dan de interne calls in de monoliet. De keuze van de communicatie tussen componenten is in het kader van duurzaamheid een relevant vraagstuk.

Hoewel 80% van de (externe) communicatie via REST API's wordt gecommuniceerd en de data in JSON (tekst) formaat wordt gecommuniceerd is dit zeker niet het meest efficiënte manier. Als er sprake is van een client-server situatie is gRPC een betere keuze<sup>7</sup>. Maar voor lokale communicatie kunnen ook interne (LPC) calls gebruikt worden, maar zorg altijd wel voor loosely coupled interface design ook als de implementatie resulteert in het samenvoegen van diverse componenten in één applicatie! Het zodanig opdelen van je applicatie in de juiste files en altijd primaire argument types gebruiken en nooit objecten doorgeven.

**Principe: tijdens het modelleren van componenten wordt duurzaamheid (energieconsumptie en herbruikbaarheid) meegenomen als eis.**

In de laatste stap van component modeling wordt de vraag gesteld of een component (of een groep van componenten) gekocht moet worden of zelf gebouwd. Deze stap is meestal gebaseerd op een TCO voor dat component, waarbij in de praktijk de TCO van zelfbouw te positief wordt ingeschat. Vandaar dat het principe “Buy before Build” is geïntroduceerd, maar dat terzijde. Waar het ons om gaat is dat in de keuze duurzaamheid moet worden meegenomen. Wat zijn de voordelen als het gaat om duurzaamheid in Buy vs Build?

**Principe: In de keuze Buy vs Build is moeten de energiekosten van het bouwen en runnen van een applicatie meegenomen worden.**

Voordat software gebouwd gaat worden wordt er per software component een design gemaakt. In het design wordt gebruik gemaakt van zgn. ontwerppatronen.

---

<sup>7</sup> <https://www.wallarm.com/what/the-concept-of-grpc>.

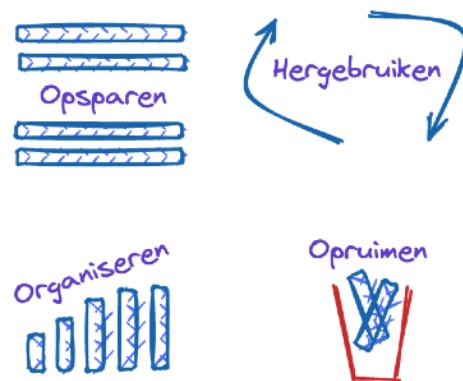
# 21. DESIGN

**E**en goed ontwerp is vol van patronen. Daarom ligt de aandacht van dit hoofdstuk volledig daarop. Met betrekking tot codepatronen is het zeer aan te bevelen om een serieus notie te nemen van het initiatief van Prof. Dr. Patricia Lago (Universiteit van Amsterdam) waar zij verschillende artikelen publiceerde over software en duurzaamheid (<http://patriciaalgo.nl>).



## 21.1. ONTWERPPATRONEN

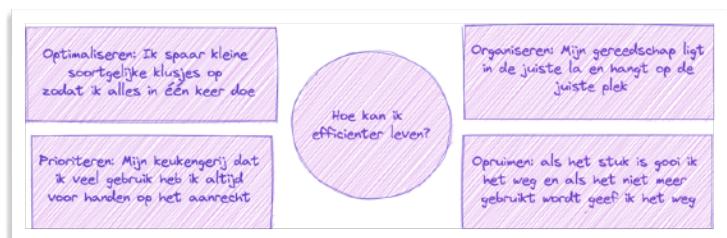
Software ontwikkelaars maken veel gebruik van patronen. Patronen zijn een oplossing voor een veel voorkomend probleem in een specifieke context. Het is niet de bedoeling om alle patronen te evalueren maar wel om bewust te worden van duurzame en minder duurzame patronen.



We bespreken vier categorieën van ontwerppatronen die de efficiëntie van software kunnen verhogen:

1. Opsparen
2. Hergebruiken
3. Organiseren
4. Opruimen

Overigens gelden deze categorieën niet alleen voor software ontwikkeling, maar je kunt ze ook prima in je dagelijks leven toepassen.



## 21.2. OPSPAREN

Als je de afwas doet ga je niet 1 kopje awassen. Je spaart op tot het de moeite is (bij studenten vaak als er geen servies meer is, een enkeling begint al als het aanrecht vol staat, maar dat is een uitzondering). Software kent diverse patronen die te maken hebben met opsparen:

1. Batch verwerking
2. Asynchrone communicatie
3. Queing



### Batch is Back

We dachten misschien dat dit iets is waar we juist vanaf willen. Maar in het kader van duurzaamheid is niets minder waar. Batch verwerking is enorm efficiënt wanneer we dit bijvoorbeeld vergelijken met Services Oriented Architectuur of een microservices architectuur waar iedere transactie de aanroep van één of een aantal services betekent. Bij batch verwerking wordt er met één call honderden of duizenden transacties in één keer verwerkt. Daarnaast is batch veel eenvoudiger in de uitvoering omdat het een serie van sequentiële voorspelbare stappen is. En je hoeft echt niet te wachten tot het nacht is voor het uitvoeren van je batch processen. Je kunt ook 1 keer per uur je batch proces laten lopen.

### Asynchrone communicatie

We maken als mens zelf ook heel veel gebruik van deze manier van communiceren. Denk aan de post, de email, aan SMS of Whatsapp berichten, Slack, allemaal asynchrone communicatiemiddelen. Dat is prettig je stuurt iets weg en op jouw voorkeurmoment lees je de reactie. Maar als je haast hebt pak je de telefoon, om te bellen bedoel ik, want dat andere kun je ook met je telefoon, maar dan kan je op dat moment niets anders meer. Een computersysteem werkt hetzelfde. Bij asynchrone communicatie wordt een bericht verstuurd en er komt vanzelf een trigger met een antwoord. Ga dus niet zitten wachten!

### Queing

Wanneer je alweer in de verkeerde rij voor de kassa staat dan ben je misschien minder gelukkig met dit patroon, maar vanuit de kassières gezien is het wel efficiënt. Overigens is het mechanisme

in Amerika: 1 rij en meer kassa's, niet één rij per kassa. Een voor de gold customer is er die aparte kassa!

## 21.3. HERGEBRUIK

### Caching

Normaal gesproken is opruimen een heel goed idee. Zeker als je het efficiënt doet. Toch, soms, als je iets heel veel gebruikt, denk bijvoorbeeld aan je telefoon dan ruim je deze niet op maar hou je die altijd bij je. Dat in essentie is caching, niet opruimen maar altijd voor handen op de plek waar je op dat moment bent.



### Delen

In het kader van de deeleconomie een goed idee. Een voorbeeld hiervan is een pattern dat heet connectie pooling, helaas een verkeerde naam want het zou moeten zijn connectie sharing. Het opzetten van een verbinding naar bv een database of een server kost vrij veel tijd. Als je de verbinding eenmaal hebt opgezet en de gegevens zijn uitgewisseld wordt de verbinding weer netjes afgesloten. Je kunt er ook voor kiezen om de verbinding niet af te sluiten maar weer vrij te geven zodat de volgende applicatie er gebruik van kan maken. Je kunt het zien als een vorm van caching, je ruimt dus niet netjes op.

## 21.4. NETJES ORGANISEREN

### Sorteren

Het behoeft nauwelijks aandacht, maar voor de volledigheid doen we het toch maar. Je kunt op heel veel manieren sorteren. Er zijn veel verschillende algoritme en de ene is nu eenmaal efficiënter dan de ander: Selection, Bubble, Merge, Heap, Quick zijn een aantal algoritmen en je kunt op het internet op diverse plaatsen informatie hierover vinden<sup>8</sup>. Het beste sorteeralgoritme is niet altijd quicksort, naar het hangt af van jouw context en situatie.



<sup>8</sup> Bijvoorbeeld: <https://www.toptal.com/developers/sorting-algorithms>.

## **Indexeren/sorteren**

Als je een bibliotheek binnenkomt staan de boeken gelukkig op een bepaalde manier georganiseerd. Hierdoor kun je een boek snel terugvinden. Ook bij de supermarkt liggen producten bij elkaar. Dit is efficiënt en werkt in de IT net zo. Door op de juiste manier te organiseren kost het minder energie om iets terug te vinden. In onze publicatie: How to: Develop a blueprint for a Data-Driven Enterprise Architecture” hebben we services opgenomen die hierbij helpen.

## **Hashing**

Hashing zorgt ervoor dat een grote hoeveelheid data gereduceerd wordt tot een veel kleiner bereik. Als je de inhoud van de data zelf niet nodig hebt kan je dit energie en ruimte besparen.

## **21.5. OPRUIMEN**

Garbage collection is een typische patroon in software ontwikkeling dat na verloop van tijd kijkt of niet gebruikt geheugen vrij gegeven kan worden omdat deze bezet wordt gehouden door objecten die niet meer door de applicatie aangeroepen zullen worden.



Dit kun je ook breder trekken door als programmeur ervoor te zorgen dat je bijvoorbeeld tijdelijke data in database weer weggooit, je logging opschoont en niet langer bewaart dan nodig.

Data cleansing is een onderwerp dat we in het hoofdstuk Responsible Data Usage bespreken.

Het is ook goed om te kijken naar patterns die juist minder duurzaam zijn. Toch zijn deze patterns er met een reden. Het gaat er hier niet om of de patterns wel of niet gebruikt moeten worden, maar veel meer een bewustwordingsproces.

## 21.6. ANTI-PATTERNS

Naast de bovenstaande goede patronen, zijn er ook patronen met een risico. Je moet deze niet gedachteloos implementeren maar goed nadenken over het energiegebruik.

### Client Server / Application Tiering

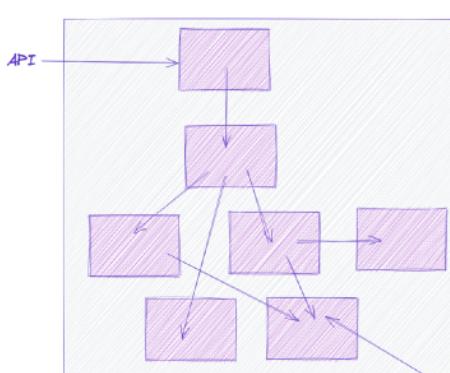
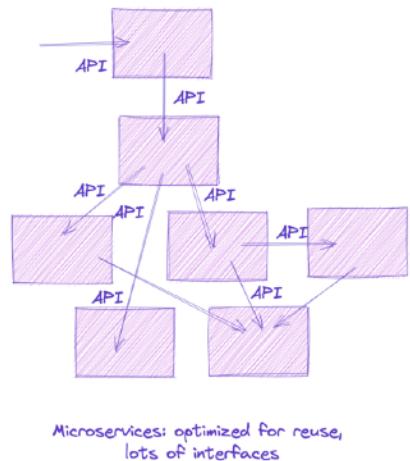
Als we het hebben over client server of application tiering dan hebben we het over een applicatie die verdeeld is over twee of meer servers. In het hoofdstuk Responsible Infrastructure gaan we in op de kosten van communicatie. Client server en multi-tier applicatie hebben deze overhead. De andere kant is weer dat de applicatie modulair is opgezet waardoor hergebruik weer hoger is. De kunst is de juiste balans te vinden en het meest efficiënte communicatie protocol.

### SOA en Microservices

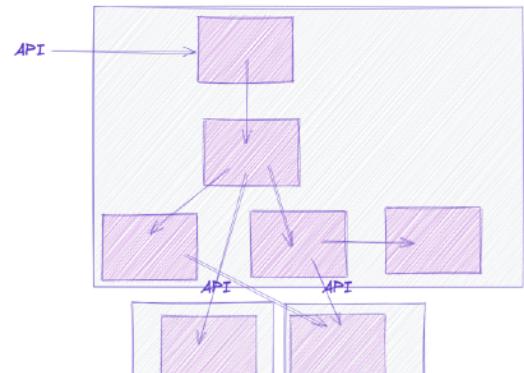
Het opbreken van een applicatie in componenten is een must.

De manier waarop die componenten vervolgens met elkaar communiceren maakt een groot verschil SOA services hebben een hogere granulariteit dan microservices.

Concepten als minservices of macroservices kunnen ook een goed compromis zijn. Het



Monolith: optimized for performance, reuse with overhead, one external interfaces



Macroservices: 80% optimized for performance, 80% reuse, 20% external interfaces

op de juiste wijze componentiseren van de applicatie is essentieel om de communicatiekosten te bewaken.

Het gebruik van patterns zegt iets over de senioriteit en vakbekwaamheid van de applicatieontwikkelaar. Bij goede ontwikkelaars zien we massaal gebruik van patterns. Vaak zal het beter om een pattern te gebruiken dat niet energie efficiënt is dan geen pattern te gebruiken omdat dit onder de streep nog meer kost.

Het tweede wat je ziet is dat energie patterns ook de performance van systemen ten goede komt. Hiervoor is één uitzondering. Als het pattern parallelisatie voorkomt dan is het wel energiezuiniger maar niet sneller.

Principe: In de keuze van patterns moet verantwoord worden waarom welk pattern is gekozen. Duurzaamheid is hierbij een van de criteria.

# 22. CODEREN

## 22.1. PROGRAMMEERTALEN

Er is een tijd geleden een onderzoek gedaan naar de efficiëntie van programmeertalen<sup>9</sup>. Interessant genoeg wordt er recentelijk veelvuldig naar deze studie gerefereerd. Helaas één van de weinige onderzoeken die gedaan is, maar het resultaat is onthutsend. Het verschil tussen C en Python of Perl in efficiëntie is bijna een factor 80!

Grofweg kunnen programmeertalen in tweeën worden gedeeld:

1. Gecompileerde talen
2. Geïnterpreteerde talen

Bij gecompileerde talen wordt de code door een compiler gehaald en vertaald in een machinetaal module zodat het operating systeem de applicatie direct uitvoert. De machinetaal module kan alleen maar werken op het platform waarvoor het gebouwd is. Doordat de applicatie en de processor via het operating systeem direct met elkaar communiceren is een gecompileerde taal veel sneller als een geïnterpreteerde taal. Daar komt bij dat bij een gecompileerde taal ook bekend is of bepaalde software al dan niet aangeroepen gaat worden. De footprint van een gecompileerde applicatie is daardoor ook kleiner.

Bij geïnterpreteerde talen is er een interpreter die gekoppeld is aan het platform en die op run time de code vertaald. Iedere keer dat de code wordt aangeroepen moet er opnieuw worden vertaald. Deze talen zijn dus minder efficiënt. Daartegenover staat dat dezelfde programma's op verschillende omgevingen kunnen worden uitgevoerd. De interpreter zorgt ervoor dat het operating system altijd de bedoeling van het programma begrijpt. De prijs is dat er altijd een interpreter nodig is en altijd een extra vertaalstag zit tussen de applicatie en de CPU.

## 22.2. LOW CODE EN NO CODE PLATFORMS

Te pas en te onpas worden ze gebruikt, low en no code platforms. Het idee van deze platformen is dat de gebruiker van deze platformen geen programmeer ervaring hoeft te hebben om toch

---

<sup>9</sup> <https://greenlab.di.uminho.pt/wp-content/uploads/2017/10/sleFinal.pdf>

zelf programma's te schrijven. In een tijd van schaarste op de markt van goede ontwikkelaars is dit een boodschap die we graag willen horen. De schaarste zorgt ook nog eens voor dat er flink in de buidel getast moet worden om die software ontwikkelaars te betalen. Kunnen we inderdaad iedereen applicaties laten bouwen met deze platformen? We zijn zeker niet tegen het gebruik van low en no code platformen maar een waarschuwing is wel op zijn plaats. Als het te goed is om waar te zijn dan is het waarschijnlijk niet waar. Er zitten duurzaamheidsrisico's aan het gebruik, of moeten we dan zeggen misbruik van deze platformen. De ontwikkelaars of ontwerpers komen zo ver van de technologie te staan dat zij niet kunnen zien wat de impact is van keuzes op de onderliggende efficiëntie van de generatie van de code. Hoe bepaald een low code platform de keuze van bijvoorbeeld een sorteeralgoritme? Niet voor niets vragen we aandacht voor het belang van kennis in het hoofdstuk over de programmeur.

Een tweede probleem bij low en no code platformen is de vendor lock in. Principe GC1 geeft aan dat de code moet kunnen draaien op het meest efficiënte platform. Het is dus zaak dat de gegenereerde code niet binnen een eigen omgeving draait maar bv wel gedeployeerd moet kunnen worden op bijvoorbeeld een Kubernetes platform.

Nu vegen we hier no en low code even op één hoop, maar dat is niet terecht<sup>10</sup>. Hoewel er gemeenschappelijke impact is op duurzaamheid is er ook een verschil. Bij low code platformen is er de mogelijkheid om ook stukken te programmeren. Het geeft je dus de mogelijkheid om daar in te grijpen waar de omgeving niet efficiënt is. Bij no code wordt je heel erg gedwongen gebruik te maken van wat er is. Als er een development mogelijkheid aanwezig is dan wordt er vaak voor een scriptie taal gekozen en dat zijn niet de talen die bovenin de lijst met meest efficiënte talen staan.

Alleen maar nadelen? Zeker niet! Het ontwikkelen op een low or no code platform gaat sneller. Je bespaart hier energie in de development omgeving. Er worden versnellingen tot wel 10x beloofd. Realiseer je dat de codeerfase ongeveer 30% is van een applicatie ontwikkeltraject. Het overige werk zit in het verzamelen

---

<sup>10</sup> <https://www.ibm.com/cloud/blog/low-code-vs-no-code>

van de eisen, het ontwerpen van de applicatie en ook het testen van een applicatie kost veel tijd.

## 22.3. DEFINIEREN VAN VARIABELEN

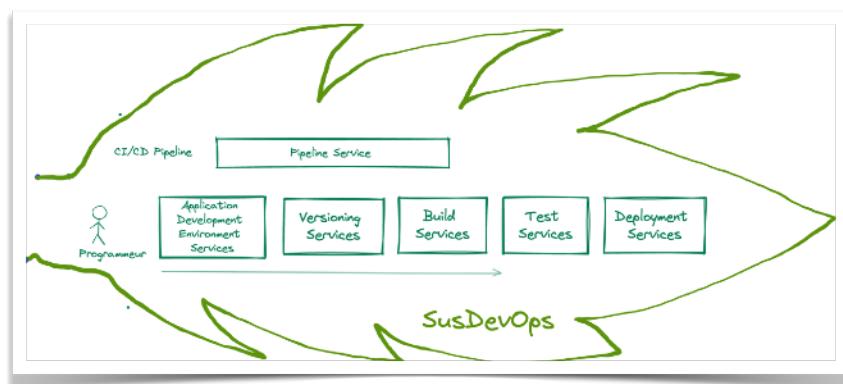
```
int iCounter = 1;           // integer 4 (or 8 bytes)
short iSustainableCounter = 1; // 2 bytes saves at least 50% of storage
```

Vele kleintjes maken één grote

In code definieer je variabelen als een integer, character string floating point, etc. Afhankelijk van deze definitie reserveert de computer een aantal bytes. Zorg ervoor dat je niet meer reserveert dan nodig is.

## 22.4. DE SOFTWARE ONTWIKKELSTRAAT

Een van de omgevingen die bijna nooit groot en snel genoeg is is de software ontwikkelstraat. Ontwikkelaars bouwen applicaties, compileren deze en testen. Passen weer aan, snel compileren en testen en dit tientallen malen per dag. En liefst snel want tijd is geld en wachten op een compile actie om te kunnen testen is ook irritant. Ontwikkelplatformen zijn energievreters. In ons vorige oranje boekje hebben we een basis keten van tools geschatst:



SusDevSecOps

Voor een gedetailleerde beschrijving van de componenten verwijzen we graag naar dat boekje. Als we kijken naar de duurzaamheidsaspecten van de toolchain kunnen we daar een aantal opmerkingen over maken.

Ten eerste de pipeline services. Deze service is de regisseur van de hele keten. Een duurzame regisseur is niet aanwezig als deze niet nodig is en zal de hele keten pas opzetten als hier vraag naar is. Serverless is een must voor een duurzame regisseur.

Vervolgens krijgen we de applicatie development environment. Dit is de omgeving waarbinnen de ontwikkelaar zijn applicatie bouwt en unit tests uitvoert. De ontwikkelaar heeft liever 32 cores dan 16 en liever 128 Byte RAM dan 64. Als je de specs van een moderne development machine gebruikt dan is dat niet mis. Als er vervolgens een applicatie gebouwd wordt veranderd de laptop of desktop in een verwarming, in de winter wel lekker, maar in de zomer wat minder. Deze kosten zijn vaak niet zichtbaar omdat de developer het liefst dit lokaal op zijn of haar computer doet.

Als de ontwikkelaar tevreden in wordt de code in een repository gezet. Dat kost weinig CPU, enig bandbreedte maar alles bij elkaar wel wat storage. Misschien goed om ook eens wat op te ruimen?

Dan de build service. Alle code van alle developers wordt bij elkaar geharkt en vervolgens een build gedaan van het hele systeem. Ook dit tikt aan. Kun je dit wellicht gedurende de nacht doen als de ontwikkelaars liggen te slapen. We kunnen dan schaarse resources gebruiken die we even niet nodig hebben.

Uiteraard wordt de code vervolgens getest. Een regressietest, maar wordt er ook gekeken naar performance? Maar ook andere tests zijn mogelijk. Die overigens ook wel voor de build service gedaan worden. Worden er code inspecties uitgevoerd om te kijken of er efficient geprogrammeerd is?

En vervolgens hebben we de deployment service. Deze zet de applicatie neer in de meest optimale omgeving!

Hoewel we niet in kunnen gaan op ieder detail en elke mogelijkheid om de pipeline te optimaliseren is de bedoeling duidelijk. En vooral in die pipeline geldt: vele kleintjes maken één grote!

Eén ding willen we nog wel noemen. Het is zeer goed om de hoeveelheid energiegebruik in de pipeline te meten. Wellicht zegt het getal niet direct iets in verhouding tot een andere omgeving, het getal kan wel helpen als relatief vergelijk tussen de versies van de applicatie die gebouwd wordt. Wat als bv de bouwkosten van release 3 naar release 4 verdubbelen? We moeten dat kunnen uitleggen!

Het is wellicht ook mogelijk om relaties te leggen tussen energiegebruik in de bouw en in productie. Wellicht een AI modelletje voor maken?

# 23. PRODUCTIE

**H**et ontwikkelen van applicaties kost energie. Meestal is dit een activiteit die in een periode van een paar maanden tot een paar jaar plaats vindt. En onder applicatieontwikkeling verstaan we ook applicatieonderhoud. Vervolgens is het de bedoeling dat de applicatie gebruikt gaat worden, meestal voor jaren, het liefst voor zoveel mogelijk gebruikers. Dan gaat de energiemeter pas echt lopen.

## 23.1. METEN IS WETEN

Het beste manier om goede vergelijkingen te maken is het meten van het energieverbruik van een applicatie. Voor applicaties is dat echter nog niet zo eenvoudig. IBM System Z heeft een ingebouwd meetstelsel en geeft rapportage van het gebruik. Het resource gebruik van een IBM Systeem Z wordt dan ook als kostbaar gezien. In het kader van duurzaamheid moeten we ook andere processoren zoals Power, Intel, AMD, ARM, RISC-V als kostbaar beschouwen.

Er komt echter steeds meer aandacht voor het meten van applicaties. Denk maar aan je smartphone die aangeeft hoeveel data een applicatie gebruikt of hoeveel interne geheugen? Waarom? Precies, je moet er voor betalen, dus is het iets waardevols. Hoeveel mensen zijn in 2022 niet bezig geweest met het meten van hun energiegebruik! Heb je een MacBook dan kun je kijken in de energy tab van je Activity monitor hoeveel een applicatie gebruikt en hoeveel het de afgelopen 12 uur heeft gebruikt.

Omdat het gebruik van AI libraries relatief duur is heeft huggingface een tool gemaakt dat CO<sub>2</sub> emissie meet!<sup>11</sup> En zij zijn niet de enige. Een ander tools is Turbonomic<sup>12</sup>, dit tool meet in de operatie wat de CO<sub>2</sub> uitstoot is, rapporteert dit en geeft ook de mogelijkheid om je applicatie te verhuizen naar een duurzamer platform. Dus zeker de moeite waard om hier afwegingen te maken. Wij verwachten dat het komende jaar de druk vanuit de

---

<sup>11</sup> <https://huggingface.co/blog/carbon-emissions-on-the-hub>

<sup>12</sup> <https://www.turbonomic.com/sustainability-calculator/>

CSO zal toenemen om energieverbruik van applicaties te gaan meten.

Een organisatie moet ervoor zorgen dat er een raamwerk gemaakt wordt waarlangs het applicaties gebruik gemeten wordt. Op deze manier kan er gekeken worden waar de grootgebruikers zitten en door doelstellingen te zetten zou er jaar over jaar bijvoorbeeld 5% bespaard kunnen worden op de energierekening. En als je naar de cloud gaat moet je daar natuurlijk ook meten! Een typisch voorbeeld van een grootgebruiker kan bijvoorbeeld Active Directory zijn. Nu valt het niet mee om Active Directory op een andere omgeving te draaien en daarmee worden gelijk restricties duidelijk die je moet voorkomen bij keuze van je applicaties en software.

Snellere programmeertalen betekenen niet vanzelfsprekend minder energie verbruik. Mooi rapport en benchmark, wat zou er volgen in de toekomst? Handleiding/ python scripts om ontwikkelaars te ondersteunen in de keuzes die ze maken?

# 24. KUNSTMATIGE INTELLIGENTIE MODELLEN

**H**et bouwen van AI modellen is een specifieke tak van sport. Voor het bouwen van modellen is een ongelofelijke hoeveelheid energie nodig in vergelijking met de code developer waar we het hier over hadden.

Daarnaast hebben we bij de bouw van modellen te maken met de ethische aspecten van AI. Die gelden overigens ook voor de code developer, maar bedrijfsregels zijn nu eenmaal veel eenvoudiger herleidbaar dan een model dat getraind wordt met een bepaalde set van data.

Vervolgens zijn data scientist grootgebruikers van Python en R. Op het gebied van duurzaamheid de meest beroerde talen.

De duurzaamheidsaspecten die we in dit hoofdstuk beschreven hebben over het ontwikkelen van software gelden ook voor AI modellen.

In het hoofdstuk Responsible Data Usage beschrijven we hoe we omgaan met het respecteren van privacy. In Responsible Systems beschrijven we de ethische aspecten van AI systemen. Daarbij komt ook ModelOps aan de orde, een variatie op SusDevSecOps. We beperken ons daarom tot een paar onderwerpen die in deze hoofdstukken niet aan de orde komen maar in het kader van duurzaamheid wel relevant kunnen zijn.

## 24.1. MODEL TRAINING SERVICES

De model training service is een dienst die we ook in ons eerste oranje boekje beschrijven. Het is een implementatie van een batch proces waarbij het maken van een model als een batch job gescheduled worden om zo resources als GPU's optimaal te benutten. In het kader van duurzaamheid een aan te bevelen service.

## **24.2. REAL-TIME GEÏNTEGREERDE ANALYTICS**

Misschien een beetje vreemde in het kader van duurzaamheid. Het idee van real time geïntegreerde analytics is dat tijdens transactieverwerking direct een analyse wordt uitgevoerd doordat GPU en CPU processing op 1 chip zitten. Het bekendste voorbeeld is de M1 Chip van Apple. Doordat GPU, CPU en memory op 1 chip zitten met een operating system dat daarvoor geoptimaliseerd is, is deze tot 3x sneller of zuiniger dan een traditionele Windows laptop. Windows laptops kunnen alleen draaien op X86 processortechnologie. Zoals je in het hoofdstuk over Responsible Infrastructure hebt kunnen lezen is deze technologie flink op achterstand komen liggen en zien we langzamerhand deze combinatie verdwijnen uit datacenters. Hoewel geïntegreerde analyse enorme winsten oplevert heb je wel te maken met een verplichte combinatie van CPU, GPU, Memory, Operating System en AI Libraries.

# 25. RESPONSIBLE CODE PRINCIPES

H

ieronder geven we een aantal voorbeelden van principes.

Sommige hiervan zijn boven eerder aangegeven en hier verder uitgewerkt op basis van het TOGAF principe template.

Principle GC1	Code moet op het meest efficiënte platform kunnen draaien
Statement	Code die zelf ontwikkeld wordt of aangekocht is soeverein en werkt op ieder platform, bij voorkeur in containers.
Rationale	Applicatie zijn efficiënter op bijvoorbeeld een RISC platform dan op een traditioneel CICS platform. De applicatie moet platform onafhankelijk zijn om eenvoudig te kunnen verplaatsen naar het meest efficiënte platform.
Implicaties	Het verdient extra aandacht om een applicatie onafhankelijk van de onderliggende technologie te

Principle GC2	OTAP omgevingen zijn beschikbaar als code
Statement	Of het nu om een ontwikkel, test, acceptatie of productie omgeving gaat, de omgevingen worden gedeployed vanuit code zodat deze na gebruik weer weggegooid kan
Rationale	Omgevingen gebruiken schaarse middelen zoals CPU en memory. Omgevingen die als code beschikbaar zijn kunnen worden gebouwd, gebruikt en weer weggegooid. Het denken moet anders. Omgevingen zijn tijdelijk en worden alleen gebouwd als het nodig is.
Implicaties	Dit betekent dat omgevingen ontwikkeld worden zoals code ontwikkeld wordt. “Infra as code” is een kreet die gebruikt wordt. Dit vraagt specifieke kennis van infrastructuur beheerders. Tools als Ansible en kunnen

Principle GC3	<b>Voor iedere component wordt de duurzaamheidsfactor beschreven</b>
Statement	Tijdens het modelleren van componenten wordt duurzaamheid (energieconsumptie en herbruikbaarheid) meegenomen als eis.
Rationale	Tijdens de ontwerp fase wordt er nagedacht over duurzaamheidsaspecten van componenten. Per component wordt er iets gezegd over de herbruikbaarheid en de energieconsumptie.
Implicaties	Het verdient extra aandacht om een applicatie onafhankelijk van de onderliggende technologie te bouwen.

Een voorbeeld van de extra informatie van een componentbeschrijving zou er als volgt uit kunnen zien:

Component:	Printing	
Vraag	Antwoord	
Hoe vaak wordt de module	Generieke module die 100.000 keer per dag wordt aangeroepen.	
Hoe noodzakelijk is de data?	Gemiddeld	
	Functie	Patterns
Welke functie zijn de meest kostbare, welke patronen kunnen bijdrage aan de efficiëntie?	Sorteren gegevens	Quick Sort

Principle GC4	<b>We meten energiegebruik</b>
Statement	Tijdens de bouw en de run van de applicatie wordt het CPU gebruik gemeten.
Rationale	Dit hoeven geen absolute getallen te zijn, maar mogen relatieve getallen zijn om een bewustwording te creëren en een cultuur die realiseert dat het bouwen en runnen van software energie kost.
Implicaties	Er moet extra code gemaakt worden om het energieverbruik te meten. Een voorstel kan zijn om jaar over jaar 5% te besparen.

Principle GC5	<b>Energie in afweging Build vs Buy vs Reuse meenemen</b>
Statement	Bij de keuze van Build vs Buy moet gekeken worden naar de TCO.
Rationale	Onderdeel van de TCO is energieverbruik. Applicaties die gekocht worden zijn over het algemeen energie efficient. Anderzijds kan de generieciteit van een Buy applicatie juist een negatieve impact hebben.
Implicaties	Leveranciers moet gevraagd worden naar de energie footprint. Leveranciers moeten beleid overhandigen hoe zij hun applicaties op een verantwoorde manier ontwikkelen.

# RESPONSIBLE DATA USAGE



Verantwoord data gebruik gaat over het duurzaam en ethisch verantwoord omgaan met data. Duurzaam betekent niet dat de data lang mee moet gaan maar dat we data inzetten voor duurzaam gebruik op een duurzame manier. Het gaat erom dat we een verantwoorde manier ontwikkelen voor het gebruik van data en de opslag van data. Dat kan alleen als we bewust zijn van wat we verzamelen, produceren en (mogen) gebruiken.

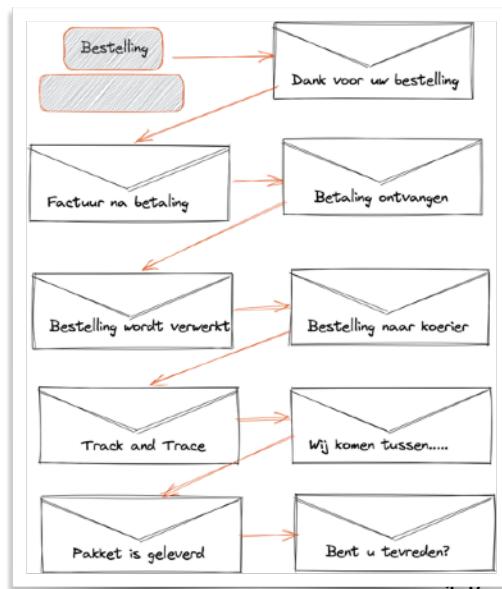
## 26. INLEIDING

**E**n voorbeeldje uit de dagelijkse praktijk:

Daar zit je dan, om half 9 uur achter je bureau met je eerste kop cappuccino met havermelk, om eerst maar eens je mailbox op te schonen. Voor velen van ons misschien een semi-automatische handeling? Of misschien een bewuste actie tijdens je eerste koffie-pauze op kantoor om door je mailbox te lopen, om zo even je hoofd te leggen?

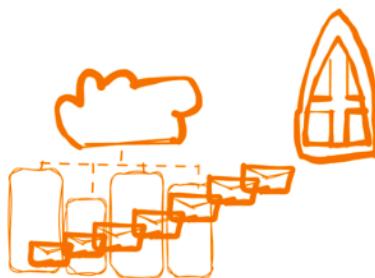
En daar kunnen zo maar vijftig berichten in zitten en hoeveel bewaar je er of lees je serieus? Neem bijvoorbeeld een bestelling van een cadeautje voor Sinterklaas. Binnen korte tijd dwarrelen er een flinke hoeveelheid e-mailtje binnen:

Helaas ontvangen we met zijn allen wel vaker een aantal mails per bestelling. En vaak genoeg, gelukkig, niet altijd 9



stuks. Maar al zijn het er een paar, voor al die bestellingen, ook buiten Sinterklaas, die we met zijn allen doen. Maar bedenk ook eens wat er allemaal op de achtergrond gebeurt en wat voor data daar gegenereerd wordt en opgeslagen. Het hele logistieke proces erachter om mensen op de vloer in te roosteren (data), de workflow

die ervoor nodig is om het bestelde product in een cadeautje te krijgen (data); de communicatie tussen koeriersbedrijf en leverancier (data), al die busjes (elektrisch of niet) die gepland worden (data), het inpakwerk (ook data), en de infrastructuur waar systemen op draaien (on-premises en/of in de cloud)... Al deze activiteiten gebruiken IT systemen die op hun beurt weer stroom kosten en waarin data verwerkt moet worden op een verantwoorde manier.



Simple actie, veel reactie

En vervolgens kan al die data nog voor andere doeleinden gebruikt worden (mits daar toestemming voor gegeven is). Bijvoorbeeld in de vorm van training datasets voor 'cross- en up sell' aanbiedingen, wat verantwoord dient te gebeuren, voor mail nummer tien, kerst komt er immers aan!

*In 2020 werden 335 miljoen online orders geplaatst in Nederland. Als je het transactiepatroon zou volgen dan hebben deze transacties 1.6 miljard mailtjes tot gevolg. Niet omdat het moet, maar omdat het kan.*

*Het is maar een eenvoudig voorbeeld maar deze mailberichten worden bij de verzendende partij opgeslagen (de webshop) alsmede de ontvangende partij (de koper) dus 2x1.6 miljard!*

*En dit is slechts 1 bestelling. Alleen al PostNL bezorgt gemiddeld zo een 1,1 miljoen pakketjes per dag. En rond de feestdagen (piekperiode) is dat 70% tot 100% meer<sup>13</sup>.*

*Stel je nou eens voor dat we dat met 10% zouden kunnen verminderen met zijn allen...?*

---

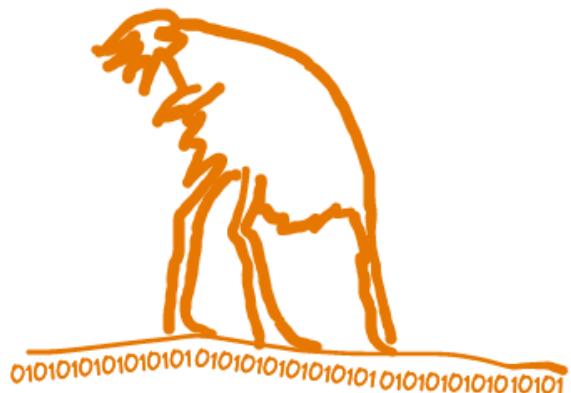
<sup>13</sup> <https://www.transport-online.nl/site/148079/postnl-ziet-geen-afname-pakketbezorging-ondanks-hoge-inflatie/>

Sinterklaas is inmiddels het land weer uit, maar hoe zit dat met de data in de systemen van al deze bestellingen? Wordt de data ook verwijderd of wordt deze data nog ingezet in allerlei cross- en up-sell acties?

## 26.1. VERZAMELWOEDE

We durven het statement wel te maken dat de meeste organisaties meer data verzamelen dan nodig is voor hun operatie. En ook, dat de meeste organisaties nog niet zover zijn in het verwijderen en vernietigen van data wanneer dat zou moeten.

Al met al zou je kunnen zeggen dat door simpelweg een minimalisme principe te hanteren, je niet alle data meer hoeft te verzamelen. Je bent dan per definitie duurzamer bezig bent en met minder data wordt ook het data beheer eenvoudiger. Maar helaas, zo simpel zit de wereld niet in elkaar. Want stel nou dat er tussen die enorme berg aan data iets magisch verstopt zit? Het is dus verleidelijk om zoveel mogelijk data te behouden. Hoe maak je de keuze tussen weggooien of bewaren?



Kop in ‘de data’ steken

Een aantal jaren geleden ontstond de term “dark data”. Gartner definieerde die als volgt:

*“the information assets organisations collect, process and store during regular business activities, but generally fail to use for other purposes.”*

In onze woorden: een enorme berg data die het licht nog nooit gezien heeft en ook zeer waarschijnlijk niet tot verlichting zal leiden, ondanks alle mooie beloftes en pogingen ten spijt, en energie die ervoor verstoot word om deze data beschikbaar te houden!

Maar wat als er dan toch iets tussen zit? Bijvoorbeeld dat de dienstverlening rond rijbewijs aanvragen sterk verbetert door onderscheid te maken tussen verlengingen en nieuwe aanvragen. Een ander voorbeeld is dat er sneller oorzaken gevonden kunnen worden voor klapbanden die ontstaan op de A-38756. Door de klapband meldingen van de ANWB te koppelen aan wegwerkzaamheden van Rijkswaterstaat kan snel de oorzaak worden gevonden en het gat in de weg worden gedicht.

Dit zijn hypothetische voorbeelden van data analyses. Maar betekent dit dat we massaal zoveel mogelijk data moeten verzamelen?

*Heb het lef om data op te ruimen!*

*1 jaar ≠ gebruikt = DELETE!*

Verantwoord data gebruik is sneller gezegd dan gedaan. Voor wat betreft dit domein gaan we eerst in op de aspecten van een data strategie en wat we hier moeten doen om Responsible Data Usage te kunnen implementeren. Vervolgens kijken we welke middelen we tot onze beschikking hebben om deze duurzame strategie uit te voeren. We noemen dit onze gereedschapskist. De strategie en de gereedschapskist zijn relevant voor alle data. In het hoofdstuk erna bespreken we een aantal aspecten van data die vooral met AI te maken hebben. Het hoofdstuk operationarisatie is een evaluatie van data projecten op het moment dat data analyses niet langer een speeltje zijn, maar dat de business er serieus gebruik van wil maken.

# 27. EEN DUURZAME DATA STRATEGIE ALS STARTPUNT

In dit hoofdstuk zullen niet alle aspecten volgen van een goede datastrategie, daar is genoeg over geschreven. Er zal vooral aandacht gegeven worden aan de aspecten waar in onze ervaring meer aandacht voor mag zijn om **verantwoord** met data om te gaan. Dit is een belangrijk onderdeel in de evolutie van bewust onbekwaam naar bewust bekwaam in het kader van Responsible Computing.

**Stap 1:** Het überhaupt hebben van een data strategie. En niet een paar abstracte slides over hoe belangrijk data is en hoe die tot inzichten kan leiden of dat er geïnvesteerd wordt in het aannemen van data scientists etc. Nu zetten we het hier wel heel zwart wit neer, maar in de kern moet die er gewoon zijn voor elke organisatie en by design breed gedragen maar ook ontwikkeld worden met de CDO aan het roer. In deze data strategie is Responsible Data Usage een belangrijk speerpunt. Een maturity model kan helpen bij het ontwikkelen van een data strategie<sup>14</sup>.

**Stap 2:** De data strategie moet heel concreet en breed gedragen zijn om data als middel in de bedrijfsvoering in te kunnen zetten. Draagvlak kun je bereiken door een breed gremium te betrekken in de ontwikkeling van de data strategie. Juist door Responsible Data Usage te benadrukken win je het vertrouwen van de betrokkenen en voorkom je dat dit een feestje van de data strategen wordt. Door specialisten te betrekken voorkom je dat het te abstract blijft, en onduidelijk is wat voor een impact data gebruik kan hebben op een proces of op iemands leven.

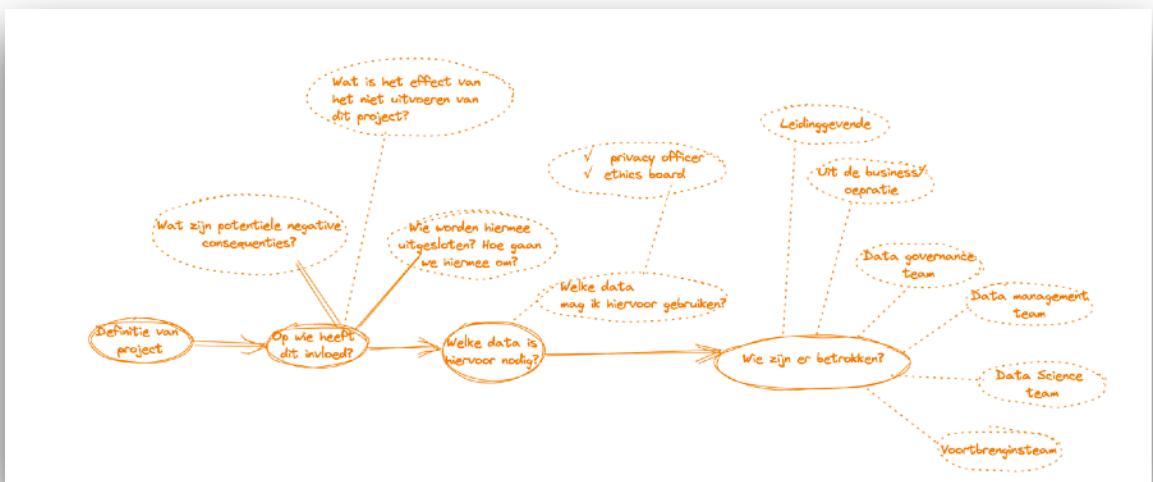
**Stap 3:** Zorg voor een goed investeringsplan. Een strategie werkt alleen als daar ook resources (middelen en mensen) en eigenaren aan gekoppeld worden. In het begin heeft de focus teveel gelegen op tooling voor het oplossen van data vraagstukken, en daarmee te weinig op het inrichten van data governance en data management.

---

<sup>14</sup> [https://ictworks.org/wp-content/uploads/2019/11/Responsible\\_Data\\_Maturity\\_Model\\_10-16-19.pdf](https://ictworks.org/wp-content/uploads/2019/11/Responsible_Data_Maturity_Model_10-16-19.pdf)

Responsible Data Usage vraagt om eigenaarschap, kwaliteit en herkomst.

**Stap 4:** De data strategie moet organisatiebreed begrepen en gedragen worden en wat het betekent om data in te zetten om beslissingen te nemen in bedrijfsprocessen. Men is zich er van bewust dat de data volgens specifieke ethische normen wordt gebruikt. Dit is wederom een stap van bewust onbekwaam naar bewust bekwaam in Responsible Computing.



Realiseren van een use case binnen de data strategie

## Privacy aspect

In relatie tot de AVG en daarmee de bescherming van persoonsgegevens wordt er in de overheid veel gesproken over doelbinding.

De definitie die de **NORA hanteert**:

*“data wordt verzameld en uiteindelijk verwerkt voor een welbepaald en uitdrukkelijk omschreven en gerechtvaardigd doel”*

Data gebruik binnen een proces, dient verantwoord, ethisch en transparant te gebeuren en op basis van toestemming van de data eigenaar. Hopelijk volgen er in de komende jaren dashboards of portalen waar je als klant in een oogopslag kunt zien wat een partij met jouw gegevens doet.

## Milieu aspect

Data heeft ook enorme impact op ons milieu. De juiste ontwerp keuzes binnen de data strategie zijn een grondslag voor het efficiënt opslaan, verwerken, terugvinden en vernietigen van data. Er nog veel te winnen op dit gebied. Denk aan data opslag op tape, verminderen van data transport door virtualisatie, goed classificeren en metadateren van data waardoor het zoek en vindt proces sneller gaat en natuurlijk een goed life cycle management hanteren.

Impact van data (transport, opslag en verwerking) op het Milieu is onderbelicht terwijl de footprint onderschat wordt.

## Checklist

Hieronder vind je een handige checklijst met vragen die als inspiratie kunnen dienen voor een duurzame datastrategie:

### Mijn data checklist:

1. Over welke data beschik ik?
2. Welke data hiervan bevat sensitieve gegevens?
3. Voor welk doeleinde is de data verzameld?
4. Waar mag ik data voor gebruiken? Wie is de eigenaar van de data?
5. Wie mag er toegang hebben tot mijn data, en onder welke voorwaarden?
6. Hoe verzorg ik (self-service) toegang tot data?
7. Waar dient data opgeslagen te worden?
8. Hoe kan ik data benaderen?
9. Welke datakwaliteit is acceptabel?
10. Wat voor policies ...
11. Hoe monitor ik datagebruik en toegang?
12. Hoe wordt trainingsdata geselecteerd?
13. Hoe zorg ik ervoor dat er geen bias ontstaat in het ontwerpen van diensten?
14. Zijn analyses die wij draaien verenigbaar met onze waarden als organisatie?
15. Kan ik uitleggen hoe een beslissing tot stand is gekomen, vanuit de hele pipeline bekijken?
16. Aan welke wet-en regelgeving heb ik mij te houden?
17. Hebben we een stempel van goedkeuring van de privacy officer of een Ethics board (als je die überhaupt al hebt)?
18. Hoe kan ik duurzamer omgaan met de technologie in huis?

# 28. DE GEREEDSCHAPSKIST

**A**ls we het hebben over onze gereedschapskist dan refereren we naar alle middelen die we tot onze beschikking hebben om data te gebruiken. Denk hierbij aan:

- Technische tools zoals een rapportagetool, business intelligent tools maar ook opslag van data, bijvoorbeeld een SSD schijf of een tape.
- Processen en procedures zoals Data Management en Data Governance. Data governance (de richtlijnen en procedures) en data management (de processen en de tools oftewel de uitvoering van data governance).

Een brede definitie waarbij we onze gereedschapskist op een verantwoorde en duurzame manier willen inzetten. We willen enerzijds ethisch verantwoord met de data omgaan en anderzijds ook efficiënt als het gaat om energiegebruik en de CO<sub>2</sub> footprint.

## 28.1. VERANTWOORDE TECHNISCHE TOOLS

*A fool with a tool is an even more dangerous fool!*

### Focussen data specialisten op nieuwste van de nieuwste tools?

Voor sommige van ons misschien wel herkenbaar; de fascinatie en focus van kinderen op nieuwe speeltjes in de speeltuin of thuis. Het nieuwste van het nieuwste is interessant om te gebruiken en dat levert dus ook onderling nog weleens strijd op. Want al neem je de zak van Sinterklaas vol nieuw speelgoed mee, dat andere speeltje waar die ander mee bezig is of het over heeft, wow.. Dat wil ik hebben, dat moet het antwoord zijn. Als specialist is het een gezonde eigenschap om geïnteresseerd te zijn in het nieuwste van het nieuwste. Dat geldt ook voor data specialisten. Data specialisten houden van data en nog meer van de tools die uit data waarde kunnen halen. Of het nu gaat om een rapportagetool, een BI tool of een AI tool. Nieuw is interessant. Maar niet alles wat

nieuw is is bruikbaar en verantwoord.

### **Hebben data specialisten een specifieke voorkeur voor een tool?**

Data mensen hebben een gereedschapskist vol met tools. De één gebruikt bij voorkeur de schroevendraaier en schroef en de ander slaat er liever op los met een hamer en spijker. In beide gevallen zit het plankje vast meer de vraag is voor hoelang? P.S. Er zijn er ook die een schroef met een hamer erin slaan, maar die categorie laten we even buiten beschouwing.

Er zijn er genoeg tools ontwikkeld de afgelopen jaren, een aantal zijn beschreven in het eerste oranje boekje over data-gedreven organisaties (van data naar informatie tot inzichten).

En zeker op data gebied is er een ruime keus vorhanden waarmee je kunt vermaken en ieder heeft zo zijn of haar voorkeur.

Strijd om tools hebben altijd al bestaan. De kunst is om de juiste tool voor het juiste probleem in te zetten. Dit betekent dat de tools niet leidend moeten zijn maar de resultaten die we willen boeken.

Als je data en tools beschouwd als Lego dan krijg je een andere gereedschapskist dan wanneer je hamers en schroeven draaiers gebruikt.

Schroevendraaiers en hamers zijn ontworpen voor een specifiek doel (hameren en schroeven) door een Lego aanpak te hanteren kun je de tools bouwen die je nodig hebt voor jouw specifieke data omgeving.

Samenvattend; een gereedschapskist om verantwoord om te gaan met data is zoveel meer dan de selectie van hamers en schroevendraaiers, maar bovenal waarvoor en hoe je die gebruikt.

## **28.2. AANDACHTSPUNten VOOR TECHNISCHE TOOLS**

### **Collectie**

Hier staat bewust het zelfstandig naamwoord collectie en niet het werkwoord collecteren of in beter Nederlands inzamelen. Het refereert hiermee aan de eerdere paragraaf verzamelwoede.

Een data collectie is de totale verzameling data van een bedrijf of organisatie. Hieronder valt ook: welke data heb ik/heb ik toegang

toe, welke data heb ik nodig en mag ik gebruiken, waar komt de data vandaan, voor welk doeleinde was deze data geproduceerd, wat betekenen de velden, hoe staat het met de datakwaliteit?

Dit kan aangevuld worden met de data principe:

hou ik me aan het zo min mogelijk data te betrekken, door oprecht mijn collectie slechts uit te breiden met wat ik nodig heb?

## **Security**

Daarnaast is voor het gebruik collectie een belangrijke taak weggelegd voor security oplossingen om data beschikbaar te maken voor een derden. Dit kan zijn voor een andere afdeling, een andere organisatie, of aan een klant. op basis van diens wensen en eisen wat leidt tot toestemming voor datagebruik.

Hiervoor zijn acces management, encryptie en pseudonimisatie technieken om dit te realiseren.

Soms kan de ene afdeling wel bij bepaalde data, maar kan dat voor een andere afdeling niet gelden of zelfs voor een bepaalde persoon binnen een afdeling. Of misschien mag er gebruik gemaakt worden van data mits niet te herleiden is om wie het gaat in een analyse?

Een andere oplossing is homomorphic encryptie. Bij homomorphic encryptie vindt de bewerking van de data ook versleuteld plaats. Hierdoor hoeft de versleutelde brondaten niet ontsleuteld te worden.

## **Duurzaam opslaan van data**

En wat betreft duurzaamheid mogen diverse technische oplossingen rondom data opslag niet vergeten worden; compressie,

- deduplicatie, het elimineren van teveel data kopieën om data opslag capaciteit in de hand te houden
- auto-scaling, het dynamisch aanpassen van de omgeving op basis van de gewenste vraag (CPU, memory, network)
- storage tiering, Een oplossing waarbij data over verschillende type storage devices wordt opgeslagen
- over-committing, het toekennen van meer virtuele capaciteit dan daadwerkelijk fysiek beschikbaar
- storage virtualisatie, verschillende opslag media virtueel bijeenbrengen als een enkel opslag medium

## **Duurzaam beschikbaar stellen van data**

Ook bij het beschikbaar stellen van data kunnen we gebruik maken van duurzame technologie.

- Data virtualisatie, het virtueel samenbrengen van data uit verschillende bronnen
- Remote data processing, verwerk data bij de bron en verstuur alleen de resultaten
- optimalisatie van queries, determineer de meest efficiënte manier om een query uit te voeren op relationele database en andere databases (bijvoorbeeld NoSQL en graph databases)
- locatie van data inclusief analyse omgeving, wat het laatste betreft, zouden we veel kunnen leren van edge implementaties en data als het ware streamend analyseren

### **Data virtualisatie uitgelicht**

Het gebruiken van data hoeft technisch en fysiek gezien niet te betekenen dat je het naar je toe koopt of naar de afdeling die ermee aan de slag gaat.

Wat een duurzaam en krachtig tool is om toegang tot data te verschaffen, is een relatief nieuwe, maar al wel veel gebruikte: Data Virtualisatie.

Dit maakt het mogelijk om data samen te brengen door het te ontsluiten, transformeren en integreren vanuit meerdere bronnen zonder het fysiek bij elkaar te brengen. En daarbij kun je gebruik maken van de capaciteit van de onderliggende technische bronnen.

### **Data Catalogus zorgt voor duurzame toegang tot data**

De kern van een catalogus is toegang te bieden tot data. Data die onder controle is van de organisatie. We weten wat voor data het is en hoe we het mogen gebruiken. Data die vertrouwd kan worden en conformert aan wet- en regelgeving door middel van voorgedefinieerde regels en policies. Het biedt duidelijkheid welke personen of systemen wel of geen toegang mogen hebben tot bepaalde data. De catalogus kan ook data maskeren afhankelijk van de gebruiker die de data opvraagt of de use case die de data gebruikt. Er is nog veel meer over te zeggen, maar daarvoor verwijzen we graag naar andere bronnen<sup>15</sup>. Sorry, wij van WC-eend.

Dat betekent dat de catalogus zowel gestructureerde als ongestructureerde data kan beschrijven, maar ook machine learning modellen. Metadata, maakt het mogelijk om data te vinden, te profileren en te verrijken, door een samenspel van

---

<sup>15</sup> <https://www.ibm.com/topics/data-catalog>

technische-, proces- en business metadata. Hiermee speelt het een belangrijke rol in compliancy en data kwaliteit, maar ook in het managen van de levenscyclus van data. De data catalogus geeft ook aan welke data niet meer relevant is of gebruikt mag worden. Dit toont aan dat een data catalogus relevant is.

Belangrijk blijft hierbij te realiseren dat technologie zoals een data catalogus niet genoeg is. Naast een data strategie zijn er degelijke interne processen nodig en is er een cultuur(verandering) nodig binnen een organisatie om data gebruik op een verantwoorde, ethische manier in te zetten.

Ongeacht hoe groot je team ook is, al heb je een leger aan data engineers en stewards, aan alle data verzoeken voldoen en dat verantwoord laten plaats vinden is een onmogelijke taak en daarbij ook nog eens inefficiënt, daar heb je echt tools voor nodig.

Daarom is self-service belangrijk. Om dit technisch mogelijk te maken is de data catalogus essentieel.

### **Monitoren en beheren van de Data Collectie**

Een belangrijke taak is ook het monitoren van de data collectie en het gebruik. Dashboards die inzicht bieden in wat er bekeken en gebruikt wordt en door welke organisatieonderdelen en door wie. Dit geldt niet alleen voor de data, maar ook voor de modellen en de systemen die gebruik maken van de modellen.

# 29. VERANTWOORD INZETTEN VAN DATA TEN BEHOEVE VAN AI

**A**lles wat we tot nu toe hebben beschreven in het gedeelte over Responsible Data Usage gaat over willekeurig gebruik van data. Het maakt hierbij niet uit of de data gebruikt wordt voor het maken van een rapport of dat data gebruikt wordt voor het profileren van potentiële klanten voor marketing doeleinden. Ethisch verantwoord omgaan met data geldt voor alles en helemaal voor PII (Personal Identifiable Informatie). In dit hoofdstuk een paar toevoegingen die vooral belangrijk zijn als we data inzetten ten behoeve van AI.

Stel dat we van het meest gunstige geval uit gaan: een volwassen organisatie die data governance en data management heeft ingericht. Data scientists hebben toegang tot data die zij voor hun analyse mogen gebruiken en zij kunnen erop vertrouwen dat de data kwaliteit in orde is. Hoe goed de data kwaliteit ook is het is altijd mogelijk dat velden in een database niet goed gevuld zijn of incompleet zijn. Het is van belang dat dit duidelijk is voor de data scientist zodat hij hier rekening mee kan houden met de invloed op de analyse.

Er ligt een belangrijke verantwoordelijkheid bij de data scientist om de data kwaliteit goed in de gaten te houden. Hierover hebben we ook het een en ander geschreven in het hoofdstuk over de duurzame specialist en het gedragsmanifest.

---

*Ter illustratie, als er een veld tussen zit dat om ethische redenen of vanwege doelbinding mogelijk niet gebruikt mag worden dan moet er aan de bel getrokken worden. Of dat nou bij de data steward is, een data privacy officer of een ethics board, het is belangrijk dat er een proces voor is om dit te melden. Het is ook noodzakelijk om een plan te hebben als het onverhoopt toch mis gaat en blijkt dat er data oneigenlijk gebruikt is.*

*Als je informatie waar je geen toegang toe hebt toch kan afleiden door middel van combinaties van variabelen dan doe je er goed aan om dat niet te gebruiken. Neem bijvoorbeeld geslacht, als je*

*dat kan afleiden voor een deel van de bevolking omdat je een variabele hebt met zwangerschapsverlof, maar je geen onderscheid mag maken tussen mannen en vrouwen, dan gebruik je dit dus niet.*

---

## 29.1. TRANSPARANTIE IN DATA TRANSFORMATIES

Data transformaties en manipulaties om een model te ontwikkelen moeten vastgelegd worden in het kader van transparantie. Dit kun je doen door gebruik te maken van een data catalogus of een AI Factsheet. Een AI Factsheet is als het ware een recept dat laat zien hoe tot een model gekomen is; de databronnen die gebruikt zijn, hoe de data getransformeerd is, de trainingsset, inclusief de locaties, de velden, het model zelf etc.

## 29.2. DUURZAME TRAININGSDATA

Om tot de juiste trainingsdata te komen kan een karwei op zich zijn, zelfs als het data management op orde is. In het kader van duurzaamheid voegen we daar een taak aan toe:  
*daag de organisatie uit om zo spaarzaam mogelijk tot een dataset te komen.*

Gebruik een kleinere set, want dat scheelt in het gebruik van resources, en alle beetjes helpen. Ook bij een data scientist ligt zowel in aanpak als het gebruik van resources een verantwoordelijkheid, met als mooi voordeel soms ook performance winst, maar ook uitlegbaarheid van een model. Zeker wat betreft resource intensieve algoritmen.

Nu was het mooi geweest als we hier concrete handreikingen kunnen geven om tot een optimale dataset te komen, maar het is helaas niet zo eenvoudig.

## 29.3. EXPLAINABILITY

Als je dan tot een goede trainingsset bent gekomen en het modelleren kan beginnen, hoe zorg je ervoor dat je de resultaten kan uitleggen? Immers, verantwoord omgaan met data, betekent ook dat resultaten uitlegbaar zijn. Er zijn tal van algoritmes die

uitlegbaar zijn. Maar er zijn ook net zoveel black-box<sup>16</sup> algoritmen, zoals AI modellen.

Afhankelijk van de case en het toepassingsgebied kan het wel of niet acceptabel zijn.

Leuk om over na te denken: explainability van black-box algoritmen.  
Er worden toolkits ontwikkeld om achteraf te bepalen  
hoe een black-box algoritme tot resultaat is gekomen.  
Vindt jij dit acceptabel of wil je eigenlijk vooraf weten hoe het  
algoritme werkt?  
Wie in de organisatie zou hierover moeten beslissen? Ben jij dat?

Naast Explainability, wat zorgt voor vertrouwen zijn er andere indicatoren die van belang zijn in het verantwoord gebruiken van data om naar inzichten en daarmee beslissingen te gaan.

## 29.4. FAIRNESS

Bij Fairness gaat het erom zorg te dragen voor gelijke behandeling en het voorkomen van bias. Dit kan al dan niet bewust in de data verstopt zitten of in de algoritmen die gebruikt worden.

Het gaat hier niet alleen om de technisch aanpak voor het voorkomen van bias maar raakt ook de organisatie en het cultuuraspect. Een speciaal team, gespecialiseerd in diversiteit, kan bijvoorbeeld helpen in het ontwikkeltraject en kan helpen om te zorgen dat belanghebbenden worden betrokken in de ontwerp fase.

Ook hier zijn wat techniek betreft diverse algoritmen beschikbaar om te controleren of de trainingsdata en het model fair zijn.

Een andere tak van sport, is gericht op het ‘hack proof’ maken van algoritmen, waardoor de uitkomst niet veranderd kan worden door invloeden van buitenaf. Dit is van belang voor zowel de trainingsdata als de modellen zelf, zodat voorkomen kan worden dat er door manipulatie andere uitkomsten kunnen ontstaan.

---

<sup>16</sup> De gebruiker kan niet zien welke algoritme is gebruikt. Voorbeelden van black box algoritme zijn Google, Instagram en Netflix.

# 30. OPERATIONALISATIE

## Voor het echie

Dat zeiden we vroeger als we gingen knikkeren. Je had een nep potje, dan kreeg ieder zijn knikkers weer terug en een potje voor het echie, dan was je je knikkers kwijt als je verloor of je kreeg die van de ander als je won.

Dat was heel spannend want dan stond er wat op het spel. Zo is het ook met het serieus gebruik van data. We hebben het in één keer over echte persoonsgegevens. Jan van Vliet uit Vlaardingen is niet langer een verzonnene virtuele persoon maar een persoon van vlees en bloed die echt bestaat en als je hem knijpt zegt ie “au”. Dan wordt het spannend!

Organisaties die al langere tijd onderweg zijn met analyse werk, maar wellicht veel projecten hebben die maar niet verder komen dan de Proof of Concept fase met gesimuleerde data zullen meer tijd en energie moeten steken in data management. Data management is een vakgebied dat zich bezig houdt met het duurzaam gebruiken van data. Als je weet over welke data je het hebt kun je gaan spelen voor het echie. Het is randvoorwaardelijk om de experimentele fase waar de analisten en data scientists mee bezig zijn te brengen naar een productie waardig systeem. Dat kan praktisch gezien door van begin tot eind ook data management ondersteund door data governance voor het project mee te nemen.

Dan focus je niet alleen op de mooie modellen, maar wordt je gedwongen om eerst na te denken over wat het effect is van het (ethische) gebruik van de data en de non-functionals mee te nemen. Hierbij spelen eigenaarschap en verantwoordelijkheid een belangrijke rol. Met als mooi bijeffect, een samenwerkingen met diverse lagen en rollen binnen een organisatie, die hopelijk wel leidt tot een hoger succesratio van de promotie van PoC naar succesvolle productiefase.

# 31. RESPONSIBLE DATA USAGE PRINCIPLES

Principle RDU1	Data wordt online benaderd en niet gekopieerd
Statement	We halen data op bij de bron en gaan niet alles kopiëren.
Rationale	Het kopiëren van data heeft een aantal nadelen: <ul style="list-style-type: none"> <li>— Data actualiteit. Een kopie loopt achter op de realiteit.</li> <li>— Transport en opslag. Een extra kopie is extra opslag en transport.</li> <li>— Risico op ondoorgrondelijke ketens. Wat als er een kopie van een andere kopie van de kopie gemaakt wordt?</li> </ul>
Implicaties	De consequentie is dat we de data moet virtualiseren en een verbinding naar de data realiseren die voldoet aan eisen als security en latency. De Data Mesh architectuur ondersteunt dit principe.

Principle RDU2	We conformateren ons aan de data strategie
Statement	Er is een data strategie, deze is breed gedragen en we houden ons hieraan.
Rationale	We zorgen ervoor dat er een data strategie is die de duurzame bedrijfsbelangen behartigt en tegelijkertijd ook uitvoerbaar is. Individuele teams geven prioriteit aan algemeen belang boven individuele successen.
Implicaties	Voldoen aan een lange termijn strategie betekent meestal dat er extra investeringen gedaan moeten worden: <ul style="list-style-type: none"> <li>— Inrichten van een Data Management team</li> <li>— Inrichten van een Data Governance board</li> </ul>

Principle RDU3	Data minimalisatie
Statement	We houden er een principe op na om minimaal data op te slaan en vast te houden. Bij twijfel weggooien en niet bewaren.
Rationale	We herleiden de data die we willen bewaren vanuit onze data strategie. Als er dus geen strategische reden is om data te bewaren gooien we het weg.

Principle RDU3	Data minimalisatie
Implicaties	Misschien gooien we teveel weg en missen we een kans om omzet te verhogen. Jammer dan, duurzaamheid gaat voor.

Principle RDU4	Alle data elementen hebben een eigenaar staan in de corporate catalogus
Statement	We zorgen ervoor dat de data waar we mee werken vastgelegd is in de corporate catalogus met alle informatie en eigenaarschap dat daarbij hoort.
Rationale	We hebben controle over de data waar we mee werken. We laten ons geen dingen overkomen maar sturen.
Implicaties	We hebben administratie en een tool nodig om ervoor te zorgen dat dit goed wordt uitgevoerd.

# RESPONSIBLE SYSTEMS



Responsible Systems zijn systemen die op een verantwoorde manier met (persoonlijke) informatie omgaan en daarover ook verantwoording afleggen.

Bijvoorbeeld je vraagt een hypotheek aan bij een bank en je hypotheekaanvraag wordt niet geaccepteerd. Je wilt dan graag weten waarom die keuze gemaakt is. Dit kan bijvoorbeeld zijn doordat er een bedrijfsregel in het systeem zit die een zelfstandige die korter dan drie jaar een eigen bedrijf heeft (nog) geen hypotheek kan krijgen.

Deze rekenregels, ook wel bedrijfsregels (business rules) genoemd worden opgeslagen en gebruikt in systemen. Soms zijn deze regels wettelijke eisen of statistische gegevens (denk aan sterftetabellen) om de hoogte van een pensioenuitkering te berekenen. Sommige systemen bevatten tientallen regels die weer uit honderden parameters bestaan. Het kan flink complex worden, maar je kunt altijd de resultaten van de berekeningen achterhalen. Een Responsible System is een systeem dat transparant is over het gebruik van de rekenregels.

Voor traditionele systemen (applicaties) is dit vrij helder. Welke data is gebruikt als input om een vergelijking of een calculatie te maken? Of welke rekenregel is gehanteerd om tot deze conclusie te komen? Hierover is vrij eenvoudig verantwoording af te leggen maar waarom is dit nu een onderwerp dat specifiek aandacht krijgt binnen Responsible Computing?

De reden is dat we tegenwoordig kijken naar oplossingen die niet meer zo zwart/wit zijn en redeneren op basis van de data/informatie die ze aangereikt krijgt, en een voorspelbaar antwoord geven op de vraag. Dit is een hele nieuwe categorie van applicaties die we samenvatten als Kunstmatige Intelligentie, beter bekend als Artificial Intelligence (AI) systemen.

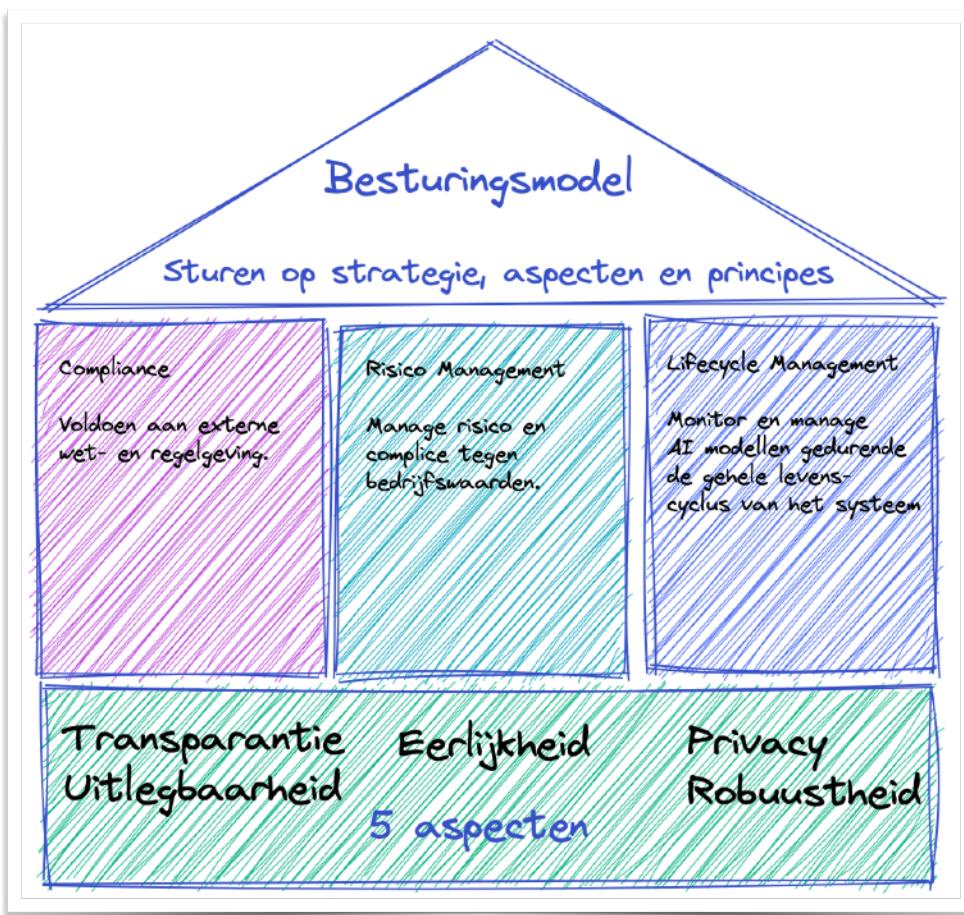
Hoe weet je nu of een systeem verantwoord omgaat met de data? Wat je ongetwijfeld wel hebt kunnen lezen in de media is dat deze applicaties zijn gebaseerd op slimme algoritmen (set instructies) die op basis van grote datasets 'zelflerend' zijn. Zelflerend wil zeggen dat deze algoritmen getraind kunnen worden door ze te voeren met data. Hierdoor leert de algoritme patronen te herkennen

waardoor er sneller en beter kan worden geanticipeerd op de uitkomst. Denk aan het ondersteunen van complexe processen waarbij veel informatie bij elkaar gehaald moet worden om bepaalde analyses te kunnen doen.

Deze nieuwe typen systemen kunnen voor een versnelling zorgen in het oplossen van technische en maatschappelijke problemen, mits goed toegepast....

## 32. RAAMWERK RESPONSIBLE SYSTEMS

**O**nderstaand figuur visualiseert het model om te komen tot een goede besturing voor het gebruik en realisatie van Responsible Systems.



Raamwerk voor Responsible Systems

Het fundamenteel van een Responsible System bestaat uit 5 aspecten.

1. **Transparantie.** Laat zien op basis van welke gegevens en rekenregels het model of het systeem werkt.
2. **Uitlegbaarheid.** Kan herleid worden hoe het resultaat tot stand is gekomen.
3. **Eerlijkheid.** Eerlijkheid kijk op een objectieve wijze naar beslissingen die door het systeem genomen worden. Deze objectiviteit wordt enerzijds door de wet bepaald (geen onderscheid in geslacht) maar is anderzijds een keuze van een organisatie. Het gaat er om dat er geen vooroordelen verwerkt in het model. Ook wel Bias genoemd.
4. **Privacy.** In acht nemen van de privacy wetgeving zoals vastgelegd in de AVG. Het gaat er daarbij vooral om dat gegevens van personen alleen met toestemming van die persoon gebruikt worden voor de doeleinden waar de betreffende persoon toestemming voor geeft.
5. **Robuustheid.** Zorg dat het model bestand is tegen gemanipuleerde data (ook dat het model zelf bestand is tegen modificaties).

Hoe hou je dat allemaal in de gaten? Deze aspecten dienen op de volgende gebieden getoetst te worden door het besturingsorgaan

- **Compliance.** Voldoen aan wet en regelgeving. Dit betekent ook dat de model op de genoemde aspecten controleerbaar moet zijn. Naast de AVG is er ook Europese regelgeving gemaakt die aangeeft wat je moet doen voor AI applicaties. In deze regelgeving onderscheid met laag risico AI applicaties en hoog risico AI applicaties. Invulling vereist dat de bovengenoemde aspecten heeft ondergebracht in het ontwikkelproces van AI systemen, echter bij hoog risico systemen is er ok de verplichting om dit regelmatig te auditen. Hierbij zijn hoog risico systemen die systemen die te maken hebben met gezondheid en/of veiligheid.<sup>17</sup>
- **Risico Management.** Denk na welke parameters je gebruikt en of dat het gewenste effect heeft en wat eventueel de risico's zijn indien de dat bijv niet compleet is.
  - In relatie tot de bedrijfs- waarde en reputatie.
  - In relatie tot de wet en regelgeving. Gegeven de Europese regelgeving betreft het hier een hoog risico of een laag risico applicatie.

---

<sup>17</sup> (<https://eur-lex.europa.eu/legal-content/NL/TXT/HTML/?uri=CELEX:52021PC0206&from=EN>)

- In relatie tot acceptatie bij de gebruikersorganisatie.
- **Lifecycle Management.** We hebben in het hoofdstuk over de Product Owner gesproken over de dynamische tijd waarin we leven. Niets is zeker, alleen dat alles onzeker is!
- Hoe draag je zorg dat gedurende de levenscyclus van het applicatie blijft valideren dat het nog steeds conform de gewenste richtlijnen het juiste resultaat wordt geleverd. Bijvoorbeeld als je model is gebaseerd op een demografische verdeling tussen leeftijdsgroepen in de jaren 70. Dan is dat model waarschijnlijk niet meer representatief voor de huidige demografie. Dit fenomeen wordt in de wereld van Artificial Intelligence model-drift genoemd.

Je moet er voor zorgen dat je organisatie zo inricht dat je kunt controleren dat de genoemde aspecten niet alleen maar verwoord zijn in een bedrijfsstrategie, maar dat iedereen zich er ook daadwerkelijk actief aan houdt. Je kunt dit doen door een (AI) Governance Board in het leven te roepen die principes voor Responsible Systems opstelt en ook in de gaten houdt dat ze daadwerkelijk worden uitgevoerd.

Nogal een opgave! In het vervolg van dit hoofdstuk worden een aantal handvatten aangereikt die kunnen helpen bij het realiseren van een besturingsmodel voor Responsible Systems:

- Methodiek en proces om te komen tot de realisatie van verantwoorde systemen
- Transformatieaanpak voor het realiseren van zo een besturingsmodel.

### **32.1. DE COMPLEXITEIT VAN ARTIFICIAL INTELLIGENCE**

Als het gaat om de systemen is er een belangrijk verschil tussen traditionele systemen en systemen die gebruik maken van kunstmatige intelligentie. Traditionele systemen maken gebruik van wiskundige regels. Als klant jonger is dan 18 kan hij geen verzekering afsluiten. Deze regels kunnen extreem complex worden. Denk maar eens aan een schaakcomputer met duizenden rekenregels....., maar allemaal met een voorspelbare uitkomst. Je kunt dus achteraf verantwoording afleggen over de beslissingen die zijn gemaakt tijdens dit schaakspel.

Maar nu systemen die gebruik maken van kunstmatige intelligentie. Deze werken anders. Stel, je gooit 100.000 plaatjes in een

systeem. Bij elk plaatje is aangegeven of er een stoplicht op staat of niet. Je genereert een model op basis van deze informatie. Je implementeert dit model in een systeem dat je “Stoplichtenherkenner” hebt genoemd. Nu pak je een ander plaatje. Op dat plaatje liggen drie ballen, een rode een groene en een paarse en je voert dit plaatje aan het systeem. De applicatie geeft aan: **het is een stoplicht!** Waar baseert de applicatie zijn keuze op? Dat mag duidelijk zijn. De kleurencombinatie natuurlijk maar het is wel een foute uitkomst. Hier wordt het dus wel een uitdaging als het gaat om een verantwoord systeem.

In de rest van dit hoofdstuk gaan we daarom ook verder in op kunstmatige intelligentie. Hoe leggen we verantwoording en transparantie af over de uitkomst?

# 33. AANPAK VOOR HET IMPLEMENTEREN VAN BETROUWBARE AI SYSTEMEN

**D**e vraag is nu hoe je zonder in te leveren op de snelheid van leveren (time to market) toch kan voldoen aan de gestelde randvoorwaarden zoals benoemd in het AI besturingsmodel, nl :

1. Transparantie
2. Eerlijkheid
3. Uitlegbaarheid
4. Privacy
5. Robuustheid

Voor traditionele applicatie ontwikkeling refereert men aan het DevSecOps ontwikkelproces waarbij je definieert hoe je iteratief met betrekken van de juiste stakeholders tot een applicatie komt die voldoet aan de richtlijnen. Kan dit ook gebruikt worden voor het ontwikkelen van AI systemen ? Procesmatig kan het zeker. Echter bij DevSecOps ligt de nadruk niet op de data wat voor een goed en betrouwbaar AI systeem cruciaal is: het gaat hier dus om de aangeleverde data waarmee het model wordt gegenereerd.

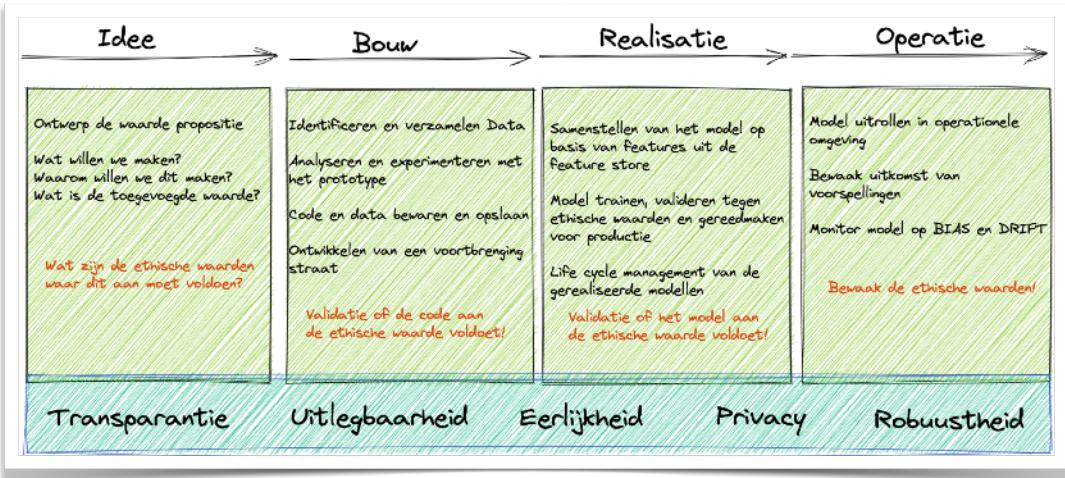
Denk nu niet dat AI iets is dat heel nieuw is. Het bestaat al jaren maar door nieuwe technologie is de AI veel nauwkeuriger geworden en is de populariteit niet meer weg te denken. In de jaren 90 is men begonnen met een manier te bedenken waarop **verantwoord** modellen ontwikkeld kunnen worden.

Dit model is CRISP-ML (*cross Industry standard process voor machine learning*) genaamd en is hieronder gevisualiseerd. Dit wordt nu gehanteerd als het Machine Learning ontwikkelproces. Het is een iteratief proces die bestaat uit 4 fasen: idee, bouw, realisatie en operatie.

Deze fasen zijn nodig om te komen tot een eindproduct. Voor elke fase zijn specifieke aandachtspunten opgenomen voor AI applicaties. Deze zijn benoemd in de 11 elementen die in de verschillende fasen in de ontwikkeling van een AI systeem worden toegepast. Hierbij zijn per fase ook aangeven welke zaken op het

gebied van ethiek de aandacht vragen voor elke fase in het voortbrengingsproces.

Complex? Ja, best wel daarom is het noodzakelijk om een duidelijk raamwerk te hanteren om te komen tot Responsible Systems.



Raamwerk voor modelontwikkeling

In de praktijk blijkt deze aanpak zeer behulpzaam te zijn. Er zijn verschillende onderzoeken gedaan, documenten geschreven en toolkits ontwikkeld op het gebied van ethiek. Deze informatie is openbaar is en we zullen er er her en der in de volgende hoofdstukken naar refereren waarbij we je een aantal hulpmiddelen geven om op bepaalde aspecten verder te kunnen verdiepen.

De volgende paragrafen gaan in op de 4 fasen en waar over men voor AI systemen over na moet denken gerelateerd aan de eerder genoemde 5 fundamentele aspecten (transparantie, uitlegbaarheid, eerlijkheid, privacy en robuustheid).

### 33.1. IDEE

Nadenken over een goed idee is over het algemeen een goede start. Bezint eer ge begint is het spreekwoord dat vooral in je opkomt als je een plank iets te kort hebt afgizaagd wat met een eenvoudige tekening voorkomen had kunnen worden.



Wat we vooral willen doen is nadenken over wat we zouden willen, wat zou kunnen en wat mag.

### **Waardepropositie**

Traditioneel gaat een waardebepaling over de kosten van een te realiseren systeem in verhouding met de opbrengsten. Dat systeem kan een probleem voor ons oplossen of een nieuwe dienst of product in de markt zetten.

Nu willen we dat altijd doen op een verantwoorde manier. De meeste bedrijven willen uiteraard eerlijk hun geld verdienen.

Specifiek voor AI systemen worden daar onder andere nog de volgende vragen aan toegevoegd:

- Hoe kunnen we tot de juiste conclusies komen op een verantwoorde manier?
- Hoe gaan we de organisatie trainen opdat ze het systeem vertrouwen en op een goede manier gaan toepassen?

De laatste vraag is een onderwerp wat wordt aangestipt in het hoofdstuk 'Cultuur en skills'.

De eerste vraag gaat met name over de 5 aspecten en **hoe** daaraan voldaan kan worden. Dat klinkt allemaal plausibel maar onderstaand voorbeeld geeft aan dat sommige zaken niet zo eenvoudig zijn.

---

#### **Voorbeeld 1. De verzekeringspremie.**

*Een verzekeraarsmaatschappij wil op basis van statische gegevens over schade haar premie differentiëren tussen verschillende klanten. Tijdens de analyse/prototyping fase komt naar voren dat geslacht (M/V) een groot verschil aangeeft in het aantal schade gevallen en men wil dit dus gaan gebruiken in de premieberekening. Dit wordt uiteindelijk voorgesteld aan de stakeholders. Helaas kan dit de toets niet weerstaan. Bias (Eerlijkheid en Privacy) zijn met name de elementen die dit classificeren als een vorm van discriminatie.*

#### **Voorbeeld 2. Diagnose ter voorkoming van Hartfalen.**

*Uit diagnostiek blijkt dat de symptomen van hartfalen nogal verschillend is tussen man en vrouw. Derhalve wil men een AI systeem maken dat in de analyse het geslacht van de patiënt meeneemt. Ook hier betreft het dus privacy gevoelige informatie.*

---

- 1.Waar is het model voor bedoeld?
- 2.Voor welk toepassingsgebied wordt het ontwikkeld?
- 3.Welke informatie bestaat er over de trainingsdata?
- 4.Welke informatie bestaat er over het model?
- 5.Wat zijn de inputs en outputs van het model?
- 6.Wat zijn de meetwaarden van het model? (5 aspecten?)
- 7.Welke informatie bestaat over de test-sets?
- 8.Kan de gebruiker uitleg krijgen hoe het model tot een conclusie komt?
- 9.Wanneer voldoet het model?
- 10.Wanneer voldoet het model niet?

#### *10 ethische vragen over het model*

*Mag het nu wel ? Mijn eigen ethische radar zou zeggen van wel, maar het is zeker iets wat getoetst moet worden aan diverse criteria*

En zo zijn er vele zaken die in deze fase helder moeten worden om te zorgen dat er uiteindelijk een systeem komt wat voldoet aan de 5 genoemde aspecten. In dit voorbeeld bovenstaande voorbeelden? wordt vastgelegd hoe je bijv Bias (eerlijkheid) dient te meten. Dit voorbeeld betreft het maken van het AI systeem. Hoe zit het eigenlijk met de klanten/patiënten van het AI systeem ?

Zou het niet mooi zijn dat je als klant/eindgebruiker/patiënt kan inzien hoe een AI systeem tot een bepaalde conclusie is gekomen? Op basis van welke gegevens is men tot een conclusie gekomen? De AVG geeft hier gelukkig wel een voorzetje. Immers jij kan als klant vragen welke informatie een organisatie over jou heeft. Maar, dit is nogal reactief en zeker niet toegespitst op een AI systeem. Welk model heeft men gehanteerd om tot deze conclusie te komen?

Naar onze mening zou het vertrouwen in AI systemen bij het publiek significant toenemen als iedere gebruiker/klant direct (proactief dus) kan zien op basis welk model / welke data men tot een bepaalde conclusie is gekomen. Dus transparantie voorop!! Misschien is dit wel wel een mooi verkoopargument, en daarmee een belangrijk aspect van jouw waardepropositie!

Je zou een eenvoudig vragenlijstje kunnen hanteren om te zorgen dat je model op een ethische manier ontwikkeld en gebruikt wordt. Er zijn diverse factsheets beschikbaar om te valideren waar het AI systeem aan moet voldoen<sup>18</sup>

## 33.2. BOUW

Genoeg nagedacht, we gaan aan de slag! Nou ja aan de slag, het gaat vooral om de voorbereidingen van de bouw fase. Deze fase gaat over het vaststellen van de juiste databronnen. Datasets die ter grondslag liggen aan het te ontwikkelen van het AI model. Ten aanzien van het verantwoord gebruik van data is eigenlijk alles al besproken in het hoofdstuk Responsible Data Usage. We geven hier nogmaals de belangrijke aandachtspunten zodat je ze echt niet vergeet.



Het gaat niet alleen maar over data. We gaan kiezen ook welke algoritmes we gaan gebruiken en welk proces we gaan volgen. Hier moet je zorg dragen dat je gaat voldoen aan de eisen die zijn gesteld tijdens de waardepropositie.

**Transparantie:** Waar hierbij ook aan gedacht moet worden is of herleidbaarheid die niet voor de hand ligt zoals bijvoorbeeld postcode gebieden die te herleiden zijn tot een percentage in etnische bevolkingsgroepen, voldoende is ganonimiseerd.

**Uitlegbaarheid:** Waar komt de data vandaan, wie is de eigenaar, wat is de classificatie, wat betekenen de velden überhaupt eigenlijk?

**Eerlijkheid:** Is de dataset van voldoende volume. Bij het trainen van modellen is het van belang dat er voldoende volume is opdat het algoritme voldoende waarden heeft om te kunnen leren? Is er voldoende spreiding in de datawaarden opdat een AI model

<sup>18</sup><https://arxiv.org/pdf/1911.08293.pdf>

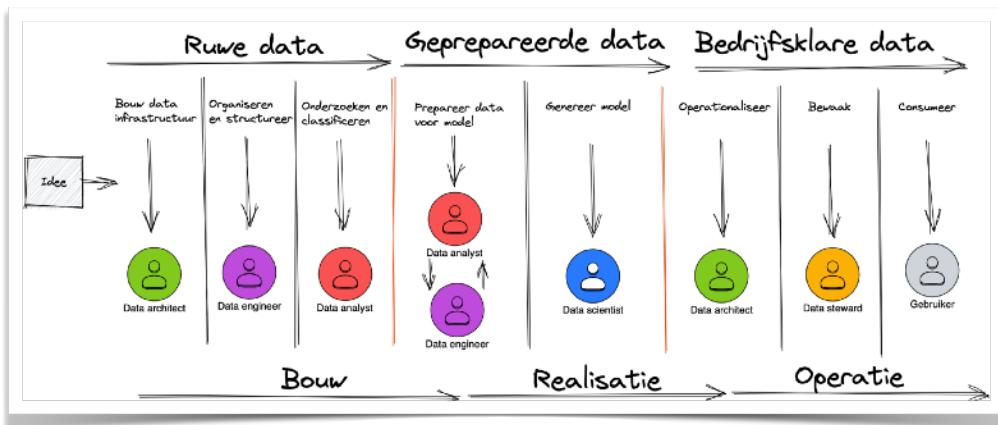
getraind kan worden om bijv aspecten als vooroordelen (Bias) te kunnen voorkomen. Je kunt wel allemaal foto's van Ferrari's laden, maar als je applicatie auto's moet herkennen herkent hij die van mij niet vrees ik.

**Privacy:** Zijn de privacy voorwaarden goed geborgd of dienen er nog extra stappen te worden genomen om gegevens te anonimiseren of te pseudonimiseren. Anonimiseren is een onomkeerbaar proces in het onherleidbaar maken van persoonsgegevens terwijl pseudonimiseren weer ongedaan kan worden dmv een sleutel.

**Robuustheid:** Met name als je datasets bestaat publieke data wil je zeker stellen dat hier niet mee is gemanipuleerd, immers als je met onjuiste data je model gaat trainen komen er mogelijk onjuiste conclusies naar voren. Je kunt in deze link meer informatie vinden waar men tijdens de dataexploratie fase naar moet kijken <sup>19</sup>.

### Selecteren van de juiste data

Dit hoofdstuk beschrijft de stappen om tot de juiste dataset selectie te komen en welke expertises in de organisatie daarbij betrokken



DataOps model

dienen te worden. Dit proces wordt ook wel het DataOps proces genoemd. Later is te zien hoe de data wordt gebruikt om het model te trainen en hoe het uiteindelijk beschikbaar wordt gesteld in de operatie ten behoeve van de gebruiker.

<sup>19</sup> <https://research.ibm.com/blog/securing-ai-workflows-with-adversarial-robustness>

In het oranje boekje '*How to Build an enterprise architecture for a data-driven resilient organisation*' vind je een beschrijving van de rollen terug. Voor iedere stap in het proces is eigenlijk een specifieke rol bedoeld. Het plaatje geeft in hoofdlijnen aan welke acties er worden genomen in het voortbrengingsproces en welke rol verantwoordelijk is voor deze stap.

### Technologische ondersteuning

Hoe hou je dit allemaal in de hand? In deze voorbereidende bouwfase zijn tools beschikbaar die helpen met:

- *Data Analysis en Exploratie*. Om dit proces te ondersteunen is er behoefte aan goede technologie die helpen bij het analyseren en exploreren van de data. Tools die hierbij helpen zijn de zogenaamde Jupyter notebooks en een hele set aan Python en/of R libraries die specifiek gemaakt zijn voor het bewerken van data.
- *Data sources en Versioning*. Het komen tot een juiste dataset kan vele exploraties en analyse iteraties bevatten. Het is derhalve cruciaal dat je goed kan vastleggen welke datasets er gebruikt zijn. Er zijn diverse tools die hierbij kunnen helpen. Een voorbeeld uit de opensource wereld is DVC.

## 33.3. REALISATIE

De datasets zijn vastgesteld en dus is het nu zaak om een verantwoord AI model/applicatie te gaan ontwikkelen die ook kan worden geoperationaliseerd. Dit lijkt nog het meest op het traditionele DevSecOps proces met ontwikkeltools en CI/CD pipelines. Maar ook hier zijn er in dit ontwikkelproces een aantal aspecten die voor een AI systeem additionele aandacht vereisen.

**Transparantie:** Zeker in het geval van "deep learning" algoritmes waar gebruik gemaakt wordt van zelf lerende systemen (neurale netwerken) is het niet zo eenvoudig om exact te herleiden hoe tot een bepaalde uitkomst is gekomen. Ten behoeve van Responsible Systems willen we dit toch helder maken!



**Uitlegbaarheid:** Vervolgens is de vraag voor wie dit inzicht bedoeld is (bijvoorbeeld een auditor of eindgebruiker) en met welk doel? Bijvoorbeeld voor de gebruiker om te valideren of het tot de juiste conclusie leidt of voor de auditor die kan vaststellen of de juiste mechanismen worden gebruikt om tot die conclusie te komen. Hier is veel onderzoek naar gedaan wat heeft geleid tot een paper en daarbij ook een toolkit die ontwikkelaars kunnen gebruiken om de uitlegbaarheid te kunnen implementeren<sup>20</sup>.

**Eerlijkheid:** Eerlijkheid wordt in het engels aangeduid als bias. Het betekent 'een vooringenomenheid in het voordeel of het nadeel tav een ding, een persoon of groepen personen die in het algemeen als oneerlijk wordt beschouwd'.

Bias is best een ingewikkeld element in het AI systeem er zijn nogal wat typen bias beschreven in online artikelen. Het gaat ons even te ver om dit zeer specialistische onderwerp in dit boekje te duiden.

Hoe eerlijk is het model eigenlijk? De eerste zorg is dat de dataset waarmee het model is getraind gebalanceerd (zie bouwfase) en niet discriminerend is op wat voor manier dan ook. Dit geldt ook voor het algoritme waarmee het model wordt gegenereerd. Er is een open source toolkit die kan helpen in het creatie proces om te valideren hoe je tegen de bias parameters bestand bent<sup>21</sup>.

**Privacy:** Hierbij is van belang dat bijv in Nederland de AVG wordt gerespecteerd. Daarbij gaat ook om de situatie dat uit het resultaat van het model ook niet valt te herleiden tot bepaalde privacy gevoelige informatie. Het vrijgeven van een postcode valt te herleiden tot een individu bijvoorbeeld. Er zijn een aantal technologieën die privacy waarborgen. Een aantal hiervan zitten nog in het onderzoeks domein en zijn nog niet beschikbaar op de markt. Een voorbeeld hiervan is '*homomorphic encryptie*': *Het model kan op basis van ge-encrypte datasets getraind worden en de data hoeft dus niet te worden ge-decrypt. Hierdoor blijft de data privacy gewaarborgd.*

**Robuustheid:** Of om het wat eenvoudiger te duiden: "is het model hack proof?".

---

<sup>20</sup> <https://arxiv.org/abs/2109.12151>

<sup>21</sup> <https://aif360.mybluemix.net/resources#overview>

Zoals al beschreven bij *data analyse en exploratie* is er onderzoek gedaan hoe te wapenen tegen manipulatie op je model. Mogelijke aanpakken hiervoor zijn beschreven in het al eerder genoemde research paper over robuustheid.

**Technologische ondersteuning:** *Code repository*: met ontwikkeltools zoals Jupyter notebooks of met een ontwikkel omgeving als Visual Studio (VSCode) of IntelliJ kan de ontwikkelaar een algoritme ontwikkelen. Vaak wordt dit in de taal Python en/of R gedaan omdat hier veel libraries vrij beschikbaar zijn voor het verwerken en bewerken van data wat uitermate geschikt is voor het ontwikkelen van AI modellen. Deze sla je dan op in een code repository. GIT is hiervoor de default standaard

*Model register en versiebeheer*: zoals hierboven wel duidelijk wordt zijn er veel tests waartegen het model gevalideerd moet worden. Het is duidelijk dat er een goed versie beheer systeem moet zijn voor het model binnen het voortbrengingsproces. Met het versie beheer systeem kan de juiste versie in de operatie worden gebracht.

### 33.4. OPERATIE

De volgende fase is dat je het model en bijbehorende applicaties die gebruik maken van het model in productie brengt. Anders dan in een standaard DevSecOps proces zal je na het deployen van het model continu moeten valideren of het model nog wel voldoet aan de gestelde criteria. Wat kan er dan nu nog misgaan ? Ik heb toch aan alles gedacht tijdens het maken van het model?

Inderdaad dat is zeker waar, maar het model is gebaseerd op een dataset van toen. Maar, wat als de dataset niet meer representatief is aan nu?

Bij de inleiding van dit hoofdstuk hebben we daar een voorbeeld van gegeven rondom demografie van de bevolking. Uiteraard wordt in deze fase nauw gemonitord of het model nog voldoet aan de gestelde ethische eisen (de 5 aspecten). Dit doen we door de reeds ontwikkelde testen te hergebruiken en nauw te letten op

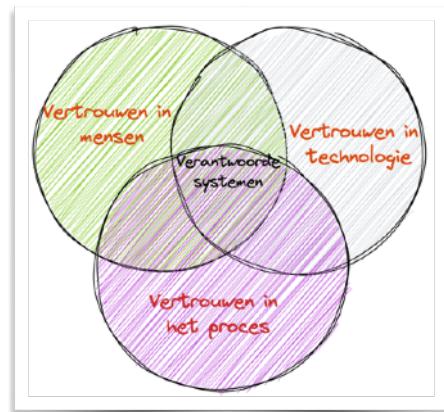


bias en model-drift. Indien het model niet meer de verwachte resultaten geeft, is dat een signaal dat het model aangepast moet worden cq opnieuw getraind moet worden.

## 34. CULTUUR EN SKILLS

**A**lsystemen worden steeds vaker ingezet in de bedrijfsvoering en leveren organisaties ondersteuning bij het nemen van (belangrijke) beslissingen.

Organisaties moeten hiervan doordringen zijn. Als de systemen op een verkeerde manier worden ingezet dan kan het de reputatie van het bedrijf of de organisatie erg schaden. We lezen hier allerlei voorbeelden van in de pers.



In voorgaande alinea's is uitgelegd hoe je procesmatig en met de juiste toolsets kan komen tot het opleveren van ethisch verantwoorde (AI) systemen. Maar het laat ook zien dat je als organisatie een aantal stappen moet zetten om dit onderdeel te maken van de bedrijfscultuur. Naast systemen en processen spelen mensen een belangrijke rol, omdat zij bepalen hoe dit soort systemen worden ontwikkeld en voor welk doeleinde deze worden ingezet. Invoering dient dus zorgvuldig te gebeuren en vraagt dus ook om een transformatie van de organisatie.

**Dit is een belangrijke stap in de evolutie van bewust onbekwaam naar bewust bekwaam.**

En betreft in hoofdlijnen de volgende drie stappen:

- **Vertrouwen in proces.** Een eerste stap die hierin gezet moet worden is het instellen van een AI Governance Board, die in ieder geval dan kan beoordelen wat de eisen zijn voor het maken en gebruiken van AI systemen. Zij is ook het adviesorgaan waar men ten rade kan gaan bij het gebruik en/of ontwikkelen van een AI systeem. Op deze wijze wordt toegezien

op inzet van AI conform de 5 ethische aspecten van AI en is dit een belangrijk onderdeel van de cultuur verandering.

- **Vertrouwen in technologie.** Daarnaast zal je de gebruikers ( de sceptici maar ook de goed gelovigen) moeten meenemen in het accepteren en het juist toepassen van de AI systemen. Gebruikers moeten kritisch zijn en begrijpen hoe de systemen worden ingezet. Vertrouwen in de technologie is de basis voor acceptatie.
- **Vertrouwen in mensen.** Zoals zo vaak is dit een evolutie die stap voor stap genomen moet worden. Begin klein en schaal op zodra hier ruimte voor is. Cultuur kun je niet in een big-bang scenario aanpakken. Zo kun je er stapsgewijs voor zorgen dat met het gebruik van Responsible Systems de organisatie uitgroeit tot een betrouwbare data-gedreven organisatie.

Over de cultuur verandering zou je een heel boek kunnen schrijven maar voor iedere organisatie vergt dit een unieke aanpak waar bovengenoemde stappen in kunnen worden uitgewerkt.<sup>22</sup>

---

<sup>22</sup> Een interessante aanpak vanuit de DevSecOps praktijk is weergegeven in het boek *Team Topologies* (Matthew Skelton & Manuel Pais, 2019). Deze legt uit hoe in een service model je autonome teams kan vormen in een zelflerende organisatie

## 35. RESPONSIBLE SYSTEMS PRINCIPLES

Principle RS1	Richt een AI Design Authority op
Statement	De AI Design Authority is de borging van de principes die gehandhaaft dienen te worden in het gebruik en het ontwikkelen van AI systemen.
Rationale	Door het oprichten van deze board worden risico's van een AI systeem op tijd gesignaliseerd en gemitigeerd waardoor de organisatie geen reputatieschade oploopt en het systeem doet wat het beoogt te doen.
Implicaties	De AI Design Authority is onderdeel van het AI besturingsmodel, maar het is niet het AI besturingsmodel. Het vraagt dus om een volledig ingericht AI besturingsmodel

Principle RS2	Automatiseer de Validatiestests voor Ethisiek in het voortbrengingsproces
Statement	Door voor elke te ontwikkelen AI systeem het validatieproces te automatiseren wordt zorgvuldigheid voor betrouwbare AI systemen afgedwongen zonder in te boeten op de time to market.
Rationale	Validatie op deze manier vereist dat er goed wordt nagedacht over de te volgen testscenario en kunnen eenduidig worden gecreëerd en de resultaten worden gevalideerd
Implicaties	Dit betekent dat er duidelijke eisen dienen te zijn voor de ethische aspecten en dat er de test gebouwd gaan worden. Sommige van deze tests zullen herbruikbaar zijn voor meerdere AI systemen. Een aantal zullen specifiek zijn voor het specifieke AI systeem.

Principle RS3	Selecteer een toolset die het proces van daselectie tot operationele inzet van AI systemen ondersteund
Statement	Datascientist maken gebruik van data die beschikbaar is. Het daarom van groot belang dat het ook helder is waar deze data vandaan komt, hoe deze tot stand is gekomen en hoe dit verder is gebruikt in het ontwikkelproces van het AI systeem
Rationale	Een geïntegreerde toolset die de gehele keten ondersteund, helpt de gehele organisatie in het borgen van de ethische eisen met daarbij minimaal verlies in de time to market
Implicaties	Er zijn diverse spelers op de markt, inclusief opensource mogelijkheden. De markt is ook in beweging hetgeen betekent dat populariteit van tools kan veranderen. Het is van belang dat ik de keuze er vooral gekeken wordt naar de openheid die de tools bieden om eventuele overgang niet uit te sluiten (vendor lock-in).

Principle RS4	Richt een organisatie in met autonome ontwikkelteams
Statement	Middels een aanpak als beschreven in “team topologie” kan je als organisatie een zelflerende schaalbare autonom ontwikkelteams inzetten zonder in te boeten op de ethische kwaliteitsnormen.
Rationale	Met autonome ontwikkelteams zorg je ervoor dat je op een schaalbare manier AI systemen kan opleveren aan de markt.
Implicaties	Dit vraagt een gedegen inrichting van je AI governance voor je ontwikkel organisatie. Hierbij deel je je organisatie op in een aantal lagen <ul style="list-style-type: none"> <li>— platform team. Deze stelt tools ter beschikking en geautomatiseerde tests ten van het de ontwikkelen AI systemen</li> <li>— AI value teams. Experts die het specifieke AI systeem ontwikkelen gebruik makend van de services die het platform team ter beschikking stelt</li> <li>— Dojo's. Een samenwerkingsmodel waarbij de verschillende disciplines van elkaar leren om als organisatie naar een hoger volwassenheidsmodel te groeien voor het ontwikkelen van AI systemen</li> </ul>

# RESPONSIBLE IMPACT



Nadenken....? Wie heeft daar nou nog tijd voor?

We worden gedreven door de waan van de dag en hebben nauwelijks tijd voor bezinning en verdieping over zaken waar we ons druk over maken.

Neem nou een onderwerp als duurzaamheid. Wat vindt jij er nou eigenlijk van of is je kijk op dit onderwerp louter gebaseerd op datgene wat je wordt verteld door de (social)media of andere kanalen?

Kijk op televisie of social media: het staat bol van de advertenties en artikelen met bedrijven die plotseling allerlei oplossingen hebben voor bezuinigingen op energie gebruik. Maar daar gaat verduurzaming in principe niet over. Een duurzame wereld is wellicht een utopie maar we moeten het wel adresseren.

One welvaart wordt gemeten aan de hand van economische groei en een vermindering van groei wordt beschouwd als een recessie. Zelfs als we nog groeien met onze welvaart maar de groeicijfers vallen enigszins tegen dan zijn we teleurgesteld en zien we terughoudendheid van investeerders en handelaren op de beurs worden minder enthousiast wat weer impact heeft op de AEX index. We horen dit soort nieuws dagelijks en we zijn het volledig gewend om in een wereld te leven waar het alleen maar over geld gaat. Zelfs onze gezondheid wordt gekoppeld aan welvaart en zijn er economische overwegingen om medische voorzieningen al dan niet voor iedereen beschikbaar te maken.

Duurzame kredieten voor bedrijven die willen verduurzamen! Alleen het geld dat een bank heeft wordt niet op een duurzame wijze verdient. Investeringen in petrochemie, wapenindustrie, autoindustrie vliegtuigindustrie en ga zo maar door. Daar worden de winsten geboekt die dan weer worden gebruikt om duurzame projecten van bedrijven te financieren. Het is een beetje krom maar zo zit onze wereld toch een beetje in elkaar.

Wij, als normale mensen, worden dagelijks verleid met nog mooiere auto's, willen we allemaal graag leven in luxe en schromen we niet om in de zomer massaal de wereld over te vliegen in het kader van vakantie.

Wat zouden we moeten opgeven om een daadwerkelijk duurzame wereld te creëren? Moeten we dan alles laten staan wat enigszins ruikt naar (overbodige) luxe en betekent verduurzamen hetzelfde als maximaal versoberen?

Wij denken dat dit niet het geval hoeft te zijn. Als we bij alles wat we doen de duurzaamheid knop zouden activeren en de afweging zouden maken wat de meest duurzame optie zou zijn bij onze keus dan zouden we al een heel eind goed op weg zijn. Het zou natuurlijk voor de hand liggen om een hele rij met duurzame afwegingen op te gaan sommen maar wij zijn er orecht van overtuigd dat het intrinsiek moet voortkomen uit de individu. Het het je niet worden opgelegd maar zodra dit soort afwegingen uit je hart en je verstand komen dan zijn we op de goed weg.

We noemen twee voorbeelden van projecten die bijdrage aan een duurzamere samenleving waarbij IT de sleutel is:

1. Een nieuwe manier van (samen)werken
2. Een milieu management systeem integreren in een Infrastructuur asset management systeem.

Vervolgens gaan we in op de Bitcoin, IT op zijn onvoordeligst, wat kunnen we daarvan leren.

En ten slotte nodigen we je uit om mee te denken en mee te dingen naar een gratis en vrijblijvende sustainability workshop!

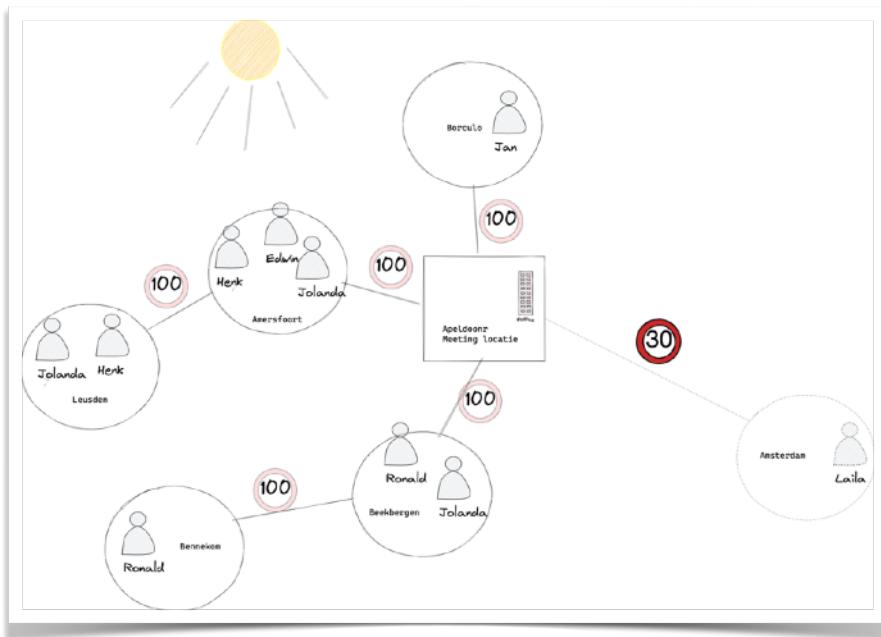
# 36. EEN VOORBEELD UIT ONS DAGELIJKS WERK

We zijn er met COVID al gedwongen mee begonnen. De 'nieuwe manier van werken'. Iedereen verplicht thuis werken en binnen no-time zijn we gewend om vanachter onze kijkbuis met elkaar te werken. Nog sterker: de economie heeft hier niet onder te lijden gehad. We zijn zelfs gegroeid met ons bedrijf. Kennelijk is het een mogelijkheid die we lang onbenut hebben laten liggen als een optie voor de moderne manier van werken. Geen lange files, mensen kunnen wonen waar ze willen want woon-werk verkeer is geen belemmering meer, en een prettige bijkomstigheid is dat de koffie thuis ook beter smaakt.

Nu de pandemie is wegebt uit ons leven dreigen we weer terug te vallen in oude gewoontes. Bedrijven (dat is het management) willen toch weer graag dat de medewerkers naar kantoor komen om te werken. En dan in ieder geval op dinsdag en donderdag waardoor het steven druk is op de weg op die dagen. Het zit in de DNA van organisaties om mensen bijeen te brengen op kantoor. Enerzijds vanuit een controle om er zeker van te zijn dat de medewerkers goed bezig zijn en anderzijds heb je sociale contacten nodig als je samenwerkt met mensen in een project of voor een klant in een klanten team. Maar op kantoor? Individueel kun je prima werken aan dossier maar voortgang en resultaten moeten bij tijd en wijlen bij elkaar worden gebracht en afgestemd om zodoende een kwalitatief geheel te kunnen leveren.

Stel je voor dat het kantoor niet meer de centrale pek is om bij elkaar te komen maar dat je de fysieke bijeenkomst met je collega's op een plek organiseert die voor iedereen een gemiddelde meest optimale afstand is. Een planningsproces dat gebaseerd is op de routeplanning theorie uit de logistieke wereld. Echter in deze planningsformule zitten ook elementen als kans op file, weersvoorspellingen en wordt gekeken of diegene die de meeste impact heeft op de reisafstand of reistijd wel degelijk een cruciaal onderdeel vormt voor deze bijeenkomst.

De tijd dat je productief bezig bent wordt zo veel hoger en daardoor heb je meer tijd om na te denken.



### *Optimaal duurzaam samenwerken*

Personalsbijeenkomsten kunnen ook op deze wijze worden georganiseerd waarbij de optie openbaar vervoer een belangrijk onderdeel kan zijn van de berekening voor de meest optimale locatie.

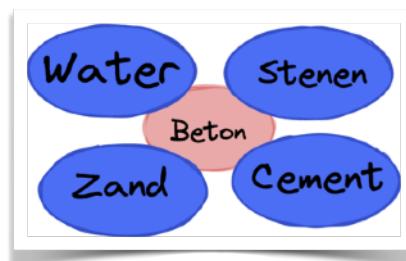
# 37. EEN MILIEU MANAGEMENT SYSTEEM (EMS): ISO 14000 NORMEN

Wanneer we het hebben over Responsible Impact dan lijken alle grenzen weg. Het gaat dan niet langer over IT, maar over de producten die een bedrijf levert en de bedrijfsvoering om dat product te leveren. Dat gaat veel verder. Denk bijvoorbeeld aan een bouwbedrijf. Hoe kun je nu als bouwbedrijf zorgen dat je duurzaamheid op een holistische manier benaderd zonder dat je, zoals we dat ook eerder zeiden, de oceaan aan het koken bent.

Om dit echt serieus aan te pakken moet je je organisatie opdelen in stukjes. Elk van die stukjes levert een positieve of negatieve bijdrage aan duurzaamheid. Dat opdelen geldt zowel voor producten als processen.

Een voorbeeld van een eenvoudig product is beton. Beton bestaat uit: water, zand, steen en cement. Voor elk van deze producten moet je weten hoe duurzaam het is om de totale duurzaamheid van beton te bepalen. Vervolgens kun je kijken naar het verduurzamen van het product. In dit voorbeeld is cement verreweg het minst duurzaam. Kun je vervangers toepassen om zo cementloos beton te maken?

Hoe zit het met het proces? Bij het maken van beton wordt zand toegevoegd. Waar komt dat zand vandaan? Is dat lokaal gewonnen of wordt het kilometers ver getransporteerd door schepen of erger nog vrachtwagens die 1 op 4 rijden en bij iedere kilometer weer de nodige CO<sub>2</sub> uitstoten?



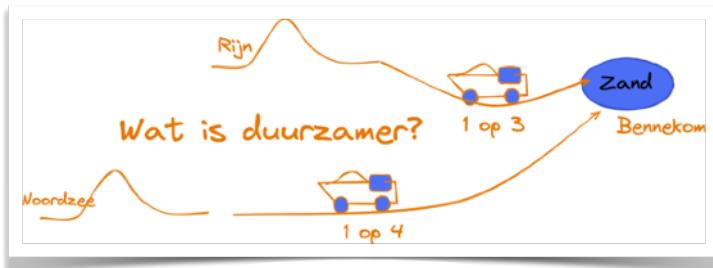
Duurzaamheid in producten

Het verduurzamen van je bedrijf vraagt dus om inzicht in producten en processen. Je moet je bedrijf ontleden in kleine stukje om zo per relevant stukje de criteria vast te leggen en vervolgens te meten! Veel bedrijven maken gebruik

van zgn. digital twins. Het is een digitale representatie van de werkelijkheid. Je legt de vrachtwagen vast in een IT systeem, hoeveel energie deze gebruikt, hoeveel CO<sub>2</sub> deze uitstoot en je kunt dan die informatie gebruiken om

beslissingen te nemen. Je laat bijvoorbeeld de vrachtwagen met minder uitstoot vaker rijden dan die met meer uitstoot. Of je besluit de minder duurzame te vervangen. Wat je zelfs nog kan doen is een IoT (Internet of Things) computertje inbouwen in de vrachtwagen die de daadwerkelijke CO<sub>2</sub> uitstoot meet. Op basis daarvan kun je ook weer bepalen of er misschien extra onderhoud nodig is.

Het beheren van je assets (Asset Management) in combinatie met IoT en een milieu management systeem wordt steeds vaker toegepast en er zijn mooie voorbeelden hoe bijvoorbeeld levensduur verlengt wordt van machines of minder vaak fysieke controles op transformatorhuisjes uitgevoerd hoeft te worden.



Duurzaamheid in processen

## 38. BITCOIN MINING

Een voorbeeld van een niet zo duurzame oplossing is het minen van bitcoins. Bitcoin is een zogenaamde cryptocurrency. Zo'n cryptocurrency is een digitaal betaalmiddel waarmee op het Internet betaald kan worden. Het is een mooi concept omdat een transactie rechtstreeks plaats vindt tussen leverancier en afnemer. Er komt geen tussenpersoon (lees bank) aan te pas. Maar er is ook een nadeel: Het controleren van zo'n transactie wordt gedaan door zgn. miners. Deze controleren een bundel van transacties en dat gaat gepaard met een enorme hoeveelheid rekenkracht. En rekenkracht daar hebben we het eerder over gehad, dat kost energie en energie resulteert in CO<sub>2</sub> uitstoot. Hoeveel zul je je afvragen? Nou schrik even niet, maar het gaat hier over 297 miljard kWh in 2024. Dat zegt je waarschijnlijk niet heel veel, maar het is ongeveer wat 118 landen in de wereld uitstoten, bijna drie keer zoveel als wat een land als Nederland per jaar uitstoot. Beng! Wat moet je daar nu mee?



Een antwoord op dit soort problemen is niet makkelijk. Belangrijk is wel dat we blijven nadenken over energieverbruik van IT. Videostreaming, gaming, advertising zijn ook grootgebruikers van computercapaciteit. Is er een weg terug? Lastig he? Laten we daarom maar beginnen met de maatregelen te nemen die in dit boekje staan. Een idee is misschien een sustainability garage? Wat is dat nu weer?

# 39. SUSTAINABILITY GARAGE

Het idee van een sustainability garage is om met elkaar een oplossing te vinden voor het halen van duurzaamheidsdoelstellingen die een bedrijf gesteld heeft. De garage aanpak hebben we kort beschreven in ons boekje over een data-gedreven organisatie. Bij de sustainability garage is de aanpak

*Maak kans op een geheel vrijblijvende sustainability garage bij IBM met mogelijk een MVP! Wacht niet te lang met je idee om je bedrijf duurzamer te maken, dien hem nu in bij één van de auteurs van dit boekje. Hun email adres staat achterin.*

identiek en we richten ons daarbij op de doelstellingen van de UN, de UNSDG. We kijken welke van de doelstellingen relevant zijn en hoe realistisch. We bouwen dan een MVP die ook daadwerkelijk moet aantonen hoe de doelstellingen gerealiseerd kunnen worden.

Aan het begin van dit boekje hebben we het al genoemd. Heb jij een goed idee? Stuur je idee op naar de schrijvers van het boekje. Mogelijk kom jij in aanmerking voor een geheel vrijblijvende Sustainability garage. Uiterlijk 1 juni 2023 kiezen we de eerste en gaan met je in overleg hoe we e.e.a. kunnen realiseren.

Voorwaarden:

- Het idee moet duurzaam zijn en de oplossing maakt gebruik van IT (sorry, onze beperking).
- Het moet gerelateerd zijn aan de organisatie waarvoor je werkt.
- Binnen je organisatie zijn stakeholders die dit ondersteunen.

We gaan vertrouwelijk met je idee om.

We zijn benieuwd naar je ideeën!

# CONCLUSIES

Met het lezen van dit boekje heb je een beeld gekregen wat we bedoelen met duurzaam computergebruik. Het gaat dus niet alleen maar over energieverbruik van datacenters en infrastructuur maar ook over “green coding” en het bouwen van ethisch verantwoorde systemen die privacy in acht nemen.

We hebben soms voorbeelden gebruikt die ver weg zitten van de mogelijkheden om aan te geven dat we een vooringenomen kijk hebben op computers, dus bias. Maar er zijn ook talloze voorbeelden van verduurzaming waar je **nu** al mee kan beginnen.

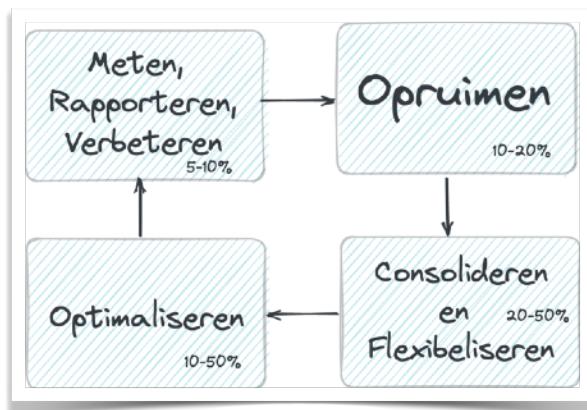
Sommige onderwerpen liggen ver buiten jouw verantwoordelijkheid. Richt je vooral op die onderwerpen die *jij* waar kunt maken!

Door duurzaamheid onderdeel te maken van je werkwijze wordt je van bewust bekwaam, onbewust bekwaam!

## 40. DE VOLGENDE STAP

**W**e hebben vier super simpele stappen voor je, kun je nu mee beginnen:

- Stap 1: Zet uit en ruim op wat je niet gebruikt! (Niet alleen servers in het datacenter maar wist je dat ook je laptop uit kunt zetten).
- Stap 2: Optimaliseer je gebruik door bijvoorbeeld te consolideren of te



De verbetercyclus

zorgen dat de omgeving kan schalen en aan en uit kan worden gezet naar gelang de behoeft.

Stap 3: Verhuis je applicaties naar duurzame infrastructuur (Koop een Laptop met een RISC processor).

Stap 4: Meet, stel (ambitieuze) doelen en wees transparant.

Heb je ooit wel eens zo'n simpel stappenplan gezien? Als je dit doet dan kun je zeker 50-70% aan energiekosten besparen. Mee eens? Waarom doe je het dan niet? Valt niet mee he, een simpele duurzame strategie implementeren. Je moet mensen meehebben, je organisatie, het moet met elkaar, het moet in je DNA zitten!

Een maturity assessment kan een heel goed middel zijn om duurzaamheid in het DNA te krijgen en mogelijk een visie te ontwikkelen en een strategie te definiëren. Een assessment geeft op de zes verschillende domeinen aan waar je staat qua volwassenheid (onbewust onbekwaam ==> onbewust bekwaam). Het geeft ook aan waar je naar toe zou willen. Op basis van deze informatie kun je een roadmap ontwikkelen die je in de komende jaren kunt implementeren. Stap voor stap, want dat is duurzaam!

# 41. SAARINEN & IBM ALS BRON VAN INSPIRATIE

We begonnen dit boekje met de brief van Watson jr. die zijn managers aangaf dat IBM een grote verantwoordelijkheid heeft ten opzichte van het milieu en dat alle processen en regels hieromtrent tot in het kleinste details moeten worden gevolgd.

We willen hierbij afsluiten met het werk van Eero Saarinen. Hij is een Fins- Amerikaanse architect bekend van zijn neo- futuristische stijl en werk zoals de Gateway Arch in St Louis, Missouri of het WA Flight Center, John F. Kennedy International Airport. Of misschien kent u hem wel van de Tulip stoel of eettafel, die hij ontwierp voor Knoll.

Bewust bekwaam of misschien wel onbewust bekwaam maakte hij keuzes die uiterst duurzaam bleken want tot op de dag van vandaag zijn zijn ontwerpen actueel, relevant en bruikbaar.

Zijn aanpak zou ons moeten inspireren. In de eerste plaats door zijn volgende uitspraak, ook in de IT, te omarmen:

"Elk tijdperk moet met de beschikbare technologie zijn architectuur creëren - één die de geest van die tijd tot uitdrukking brengt."

Eero Saarinen

De geest van deze tijd is duurzaamheid. Laten we onze beschikbare technologie inzetten om duurzame systemen te bouwen, systemen met Responsible Impact.

In 1957 kreeg Saarinen de opdracht om een nieuw wetenschappelijk onderzoekscentrum te ontwerpen voor IBM in Yorktown Heights, New York: Het Thomas J. Watson Research Center. Dit gebouw is vandaag de dag bezig met het uitvinden van

de technologie van de toekomst. Het heeft onderdak geboden aan wereldberoemde briljante wiskundige, winnaars van Nobel prijzen, aan supercomputers, aan de beroemde schaakcomputer Blue Gene dia Kasparov versloeg, aan Watson die Jeopardy won.

Tientallen uitvinding die vandaag de dag onze IT industrie bepalen, onder andere die energiezuinig RISC CPU waar we het over hadden. Maar ook de transistor die gebruikt wordt in de CPU's.

Aan ons om de briljanten uitvindingen om te zetten in Responsible Computing.

# AFKORTINGEN

3D	Drie Dimensionaal	Een technologie waarbij er in een visualisatie diepte wordt gecreëerd. Meestal door ogen afzonderlijk aan te sturen. Bijvoorbeeld door een bril.
3TG	Tin, tantalum, tungsten gold	Kostbare materialen die ontzettend belangrijk zijn om computers mee te bouwen. Toezicht op het verantwoord verkrijgen van deze materialen is een belangrijk duurzaamheidsaandachtspunt.
AI	Artificial Intelligence	Kunstmatige Intelligentie is een soort nep intelligentie die door computers gebruikt wordt. Het is niet echt intelligent maar door heel veel informatie te combineren kan de computer op basis van die informatie beslissingen nemen of inzichten verwerven. Reuze handig, maar je moet hem wel controleren anders gaat ie er mee aan de haal en krijg je ellende.
AIX	Advanced Interactive eXecutive	AIX is een Unix operating systeem dat ontwikkeld is door IBM. De systemen worden voornamelijk als enterprise servers ingezet.
API	A Programming Interface	A technisch toegankelijke manier waarop applicaties met elkaar communiceren.
AVG	Algemene Verordening Gegevensbescherming	De Nederlandse versie van de GDPR ten behoeve van het beschermen van je persoonlijke informatie.
BI	Business Intelligence	Het naar boven halen van informatie van belangrijke bedrijfsinformatie. Dit kan op basis van rapporten, dashboards, analyses, data mining.
C2C	Cradle to cradle	Van de wieg naar de wieg. Dus niet naar het graf, maar je gebruikt iets weer opnieuw in plaats van weggooien hergebruiken.
CD	Compact Disk	Een plastic plaatje waar data opgezet kan worden. De data kan muziek zijn.
CDO	Chief Data Officer	De persoon die verantwoordelijk is voor de data binnen een bedrijf. Een rol die steeds meer aandacht heeft gekregen door de privacy wetgeving.
CI/CD	Continuous Integration / Continuous Delivery	Het idee dat je continue een applicatie ontwikkeld, integreert met andere modules, test en in productie brengt.
CIO	Chief Information Officer	De baas van de IT afdeling, waarschijnlijk jou baas.
CISC	Complex Instruction Set Computer	Minimises the amount of instructions per program. The opposite of RISC.
CISO	Chief Information Security Officer	Super strenge persoon die alles op het gebied van veiligheid in de gaten houdt. Hou je aan de regels!

CO2	Kooldioxide	Stof die vrijkomt bij omzetting van brandstof naar energie. Beschermt de aarde tegen straling, maar we hebben er nu teveel van waardoor de aarde opwarmt. Op sommige dagen zou je dat misschien willen, maar het ijs is aan het smelten, de zeespiegel stijgt en als we zo doorgaan staat half Nederland over 50 jaar onder water.
COP27	Conference of the Parties	De 27e conferentie over het klimaat, gehouden in 2022 in Egypte.
COVID-19 COVID	COrona Virus Disease 2019	Soort van officiële naam voor het Corona virus dat in 2019 ontdekt is in China.
CPU	Central Processing Unit	De hersenen van de computer.
CRAC	Computer Room Air Conditioning	Airconditioners zijn belangrijke energieverbruikers in datacenters. Hebben veel invloed op de PUE factor. Niet verwarren met Crack.
CRISP-ML	CRoss Industry Standard Process voor Machine Learning	Aanpak om op duurzame wijze met AI om te gaan.
CSO	Chief Sustainability Officer	De baas die duurzaamheid overziet
CSRD	Corporate Sustainability Reporting Directive	De CSRD-richtlijn staat centraal in de Green Deal van de Europese Unie en moet zorgen voor meer transparantie over en betere kwaliteit van duurzaamheidsinformatie.
CTO	Chief Technology Officer	De baas van de technologie strategy, onder andere de IT strategie.
DNA	desoxyribonucleïnezuur	Bevat de erfelijke informatie van een organisme, maar in ons geval gebruiken we dit als analogie voor de cultuur organisaties.
EU	Europese Unie	Bij de meeste wel bekend.
GDPR	Global Data Protection Regulation	Europese regulerig die aangeeft hoe een bedrijf met privacy moet omgaan.
GIT	Whatever rocks your boat, but it could be referring to Torvald (the creator) himself	Open source software for het beheren van versies van software en documentatie.
GPU	Graphical Processing Unit	The second brain of the computer. Meant to address the display, but the way this processor works, it is very useful to do complex calculations for AI as well.
HR	Human Resources	Basically the workforce of an organisation, although we refer to the management of the workforce.
I/O	Input and Output	De bits and bytes (8 bits) die heen en weer gaan tussen opslag en processor.
IaaS	Infrastructure as a Service	Het aanbieden of gebruiken van IT nfrastructure als een service. IT uit de muur. Je hoeft niet langer eigen computers an te schaffen en te configureren maar dit wordt als dienst aangeleverd en je betaald naar gebruik. Het is een dienstverleningsmodel dat populair is bij cloud leveranciers, maar kan ook door de eigen IT organisatie als zodanig worden aangeleverd.

IBM	International Business Machines	Een groot en mooi (naar onze bescheiden mening) Amerikaans bedrijf. Focus op AI en cloud en heeft diverse duurzame uitvindingen gedaan zoals de elektronenmicroscoop en RISC processor.
IT	Infrastructure Technology	Met infrastructuur bedoelen we computers, netwerkapparatuur en opslag apparatuur voor data.
MER	Main Equipment Room	De centrale computerruimte waar de communicatie binnenkomt en de centrale servers staan.
ML	Machine Learning	Deze discipline binnen AI is in staat om computers zelf te laten leren op basis van aangeboden informatie zonder daarvoor expliciet geprogrammeerd worden.
MVP	Minimum Viable Product	Een initiële versie van een product dat al geschikt is om te gebruiken. De feedback van het eerste gebruik wordt weer toegevoegd aan het volgende product. Op deze manier wordt een product snel volwassen en maximaal bruikbaar.
NORA	Nederlandse Overheid Referentie Architectuur	Een referentie architectuur die binnen de Nederlandse centrale overheid gebruikt wordt: <a href="https://www.digitaleoverheid.nl/overzicht-van-alle-onderwerpen/standaardisatie-en-architectuur/nora/">https://www.digitaleoverheid.nl/overzicht-van-alle-onderwerpen/standaardisatie-en-architectuur/nora/</a> GEMMA is de variant voor de Nederlandse gemeente: GEMeentelijke Model Architectuur.
ODC	Overheids DataCenter	Typisch voor Nederland, maar misschien hebben andere landen dit ook wel. De Nederlandse overheid heeft in verband met efficientie alle IT in vier DataCenters neergezet.
PaaS	Platform as a Service	Dit is een vervolg op IaaS. Nu wordt niet alleen infrastructuur geleverd als dienst maar een heel platform. Dit kan een dataplatform zijn, maar kan ook een software ontwikkelplatform zijn. Een voorbeeld van een ontwikkelplatform is ARO (Azure Red Hat Openshift).
PC	Personal Computer	Een computer voor jou alleen. Je deelt deze niet gelijktijdig met anderen. Een computer die dat wel kan noemen we een server.
PDF	Portable Document Format	Zeker weten dat je dit formaat wel kent!
PII	Personal Identifiable Information	Informatie die gerelateerd zijn aan jou als persoon zoals je telefoon nummer of je email adres. Volgens de AVG/GDPR moeten bedrijven hier heel zorgvuldig mee omgaan.
PS	Post Script	Komt vanuit het Latijns: postscriptum. Je schrijft dat aan het einde als je iets belangrijks vergeten bent en je niet alles weer opnieuw wil omschrijven. Als je daarna nog iets vergeten bent kun je PPS gebruiken. Maar ja wat zegt het over jou als je steeds dingen vergeet?
PUE	Power Usage Effectiveness	Geeft aan hoe energieefficiënt het datacenter is. Bij een factor 1 wordt alle energie gebruikt voor de computerapparatuur. Energie zuinig DC zijn rond de 1,2.
REST	Representational State Transfer	REST is een protocol om informatie uit te wisselen tussen services op het internet.

RISC	Reduced Instruction Set Computer	Reduce the amount of cycles per instruction. The opposite of CISC.
SaaS	Software as a Service	Het gebruik van software als een dienst. Een concept dat past bij de public of private cloud aanpak.
SDG	Sustainability Development Goals	Breed gedefinieerde doelstelling van de UN op het gebied van duurzaamheid.
SER	Satellite Equipment Room	Een computerruimte op afstand die aangesloten is op de MER. Kijk bij MER voor die betekenis.
SQL	Structured Query Language	Good old SQL is a programmeertaal om data op te vragen uit relationele databases
SRE	Site Reliability Engineer	De rol van een persoon die een computer platform of keten (proactief) beheert. Traditioneel beheert een beheerder een component, een computer of een applicatie. Dit concept is bedacht door Google om te voorkomen dat bij een groeiende omgeving de beheer teams lineair meeschalen.
SusDevSec Ops	Sustainable Development Security en Operations	De nieuwe hype, bedacht door deze auteurs om aan te geven dat applicaties duurzaam ontwikkeld moeten worden en duurzaam in gebruik moeten zijn.
TCO	Total Cost of Ownership	Totale kosten van een oplossing, product of systeem waarin niet alleen aanschaf maar ook onderhoud zijn meegenomen.
TOGAF	The Open Group Architecture Framework	De-facto IT Architectuur standaard die beschrijft wat je moet doen om tot een goed werkend systeem te komen.
UNSDG	United Nations Sustainability Development Goals.	Zie SDG. Ze zijn bedacht door de Verenigde Naties, dus vandaar ook wel UNSDG genoemd.
UPS	Uninterruptible Power Supply	Een tijdelijk noodvoeding die aanslaat als onverhoopd de stroom uitvalt. Zorgt ervoor dat de systemen blijven draaien. Meestal een accu die tijdelijk stroom levert tot de noodstroomvoorziening opgestart is. Heeft dus niets te maken met de pakketbezorger.
VR	Virtual Reality	Een nep werkelijkheid, maar het lijkt soms net echt. Dit kun je bereiken door een speciale bril op te zetten waardoor je het idee hebt een andere wereld te zijn ingestapt.
WUE	Water Usage Effectiveness	Geeft aan hoeveel water er gebruikt wordt om systemen te koelen.

# BIJLAGE A: RAPPORTAGE VERPLICHTINGEN

**C**SRD staat voor Corporate Sustainability Reporting Directive. Het gaat er over dat bedrijven moeten rapporteren over hun duurzaamheidsdoelstelling en resultaten. Dat gaat niet over IT, maar over de hele bedrijfsvoering. Als je de rapporten leest dan zijn deze heel verschillend en je kunt ze niet met elkaar vergelijken. Hoe weet je nu of je samenwerkt met het meest duurzame bedrijf of niet?

Er komt dus standaardisatie aan (2024) om vergelijkingen mogelijk te maken en bedrijven ertoe aan te zetten duurzaamheid onderdeel van de bedrijfsvoering te maken.

Aan deze rapporten zitten twee IT aspecten:

1. De data waarover gerapporteerd wordt en de rapporten die gemaakt worden, worden meestal gedaan met behulp van IT systemen. Een voorbeeld van een rapportage tool is Envizi. Verschillende bedrijven kunnen helpen om dit aan de praat te krijgen.
2. De data *waarover* gerapporteerd wordt omvat ook IT. Bijvoorbeeld het stroomverbruik van datacenters. Daarover hebben we al voldoende geschreven in dit boek. Meten is weten, dat inzicht gaat helpen om steeds duurzamer te worden.

KPMG<sup>23</sup> heeft onderzoek gedaan en meer dan 2000 bedrijven in Nederland en 49.000 in de EU vallen onder de CSRD richtlijn. Alle kans dat er ook voor jou bedrijf werk aan de winkel is<sup>24</sup>.

Wij zijn niet specifiek in de CSRD richtlijnen gedoken maar we willen dit wel noemen in deze bijlage omdat dit wellicht voor jou bedrijf een opstap kan zijn naar duurzame IT. Dit kan werken als katalysator voor Responsible Computing.

---

<sup>23</sup> <https://home.kpmg/nl/nl/home/topics/environmental-social-governance/corporate-sustainability-reporting-directive.html>

<sup>24</sup> <https://www.mvonederland.nl/wat-is-de-csrd-wet-en-hoe-ga-je-ermee-aan-de-slag/>

# BIJLAGE B: DATABASE MANAGEMENT SYSTEMEN

**V**erwerken van data kost veel rekenkracht. In de loop der jaren zijn hier verschillende oplossingen voor (uit)gevonden. Er is behoefte om snel de juiste informatie terug te vinden in grote bakken met data.

We onderscheiden daarbij twee soorten data:

- gestructureerde en
- ongestructureerde.

Bij ongestructureerde data denken we aan documenten, video, plaatjes, spraak, etc. Zoeken daarin is complex. AI leent zich uitstekend voor het zoeken in plaatjes. In dit hoofdstuk willen we de mogelijkheden evalueren voor het zoeken in gestructureerde data. Overigens wordt aan ongestructureerde data veelal gestructureerde data toegevoegd om dit goed vindbaar te maken. Metadata noemen we dat.

Er zijn in de loop van de jaren veel database management systemen (DBMS) ontwikkeld en we bespreken een aantal vormen. Ons pleidooi is om een DBMS te kiezen dat het beste past bij het probleem dat je wilt oplossen. Bij veel organisaties is er een standaard generieke oplossingen voor een relationeel database management systeem waardoor de kans bestaat dat de weg van de minste weerstand wordt gekozen en zelfs een simpele rijtje met waarden daarin wordt opgeslagen terwijl dat niet de meest duurzame oplossing hoeft te zijn.

Hieronder noemen we een aantal soorten database management systemen en hun karakteristieken. Je hoeft niet een expert te zijn in verschillende DBMSen maar we beschrijven de belangrijkste karakteristieken. Deze kunnen je helpen in het maken van de meest duurzame oplossing.

**Hiërarchische database management systemen.** Het eerste database management systeem was een hiërarchische database. Informatie kon je vinden door van boven naar beneden te gaan in

je structuur. Die structuur lag vast en zoeken gaat razendsnel zolang je vraag past binnen die structuur. De toepassing van hiërarchische database management systemen is door de komst van relationele database management systemen nagenoeg verdwenen. Toch wordt er nog veel gebruik gemaakt van gegevensuitwisseling m.b.v. XML dat een hiërarchische structuur heeft. Deze zou perfect passen binnen zo'n DBMS. Gaat duurzaamheid ervoor zorgen dat deze vorm van DBMS weer terug komt? In ieder geval zijn er native XML databases (NXD) op de markt die XML op een efficiënte manier opslaan.

**Relational Database Management Systeem.** Verreweg het meest bekende DBMS. Data wordt opgeslagen in tabellen en tussen tabellen worden relaties gedefinieerd. Door deze vorm ontstaat de mogelijkheid om data flexibel op te slaan en te zoeken. Tabellen bevatten sleutels en deze worden van tevoren gesorteerd in indexen waardoor razendsnel zoeken mogelijk wordt gemaakt. Een zoekopdracht wordt gedaan in een specifieke query taal: Structured Query Language (SQL). Er zijn vele leveranciers van RDMS-en waaronder Oracle, IBM bekende spelers zijn en PostgreSQL en MySQL open source varianten die steeds en steeds meer gebruikt worden.

**NoSQL.** En als SQL de standaardtaal is dan is er ook wel iemand die daartegen is. NoSQL is een oplossing die tegemoetkomt aan een eenvoudiger gebruik van data. Een voorbeeld is de opslag van een naam en een waarde die aan die naam wordt toegekend (name-value pair). Waar RDMS-en zorgvuldige planning vereisen, samenhang tussen tabellen gedefinieerd moet worden en het ontwerpen van queries de nodige expertise vereist is NoSQL flexibeler en eenvoudiger in gebruik. Het schaalt horizontaal door inzet van meerdere servers. Zeker met het gebruik van microservices technologie is NoSQL eenvoudig en lichtgewicht voor relatief eenvoudige toegang tot informatie.

**Graph database.** Waar NoSQL een eenvoudigere versie is van een DBMS, is een Graph database een complexere. Deze vorm van data opslag is vooral bedoeld om relaties tussen objecten vast te leggen en op basis daarvan analyses te doen. Waar in een RDMS de objecten in een tabel worden opgeslagen en de relatie wordt gedefinieerd op tabellniveau, geeft een Graph database de relaties aan tussen individuele objecten.

**Object-Oriented DBMS.** Wanneer je gebruik maakt van een object georiënteerde programmeertaal zoals Java, kun je objecten maken. Objecten zijn een instantie van een class. Stel je hebt een class “Auto” die erft van een class “Vervoer”. Zo’n object kan best ingewikkeld worden. In een OODBMS kan je het object “auto” opslaan en ook weer ophalen. Je ziet dat verschillende NoSQL en RDMS-en ook objecten kunnen opslaan. Wanneer je gebruik wilt maken van een OODBMS zul je de balans moeten vinden tussen het gebruik van een bestaand DBMS dat ook Objecten kan opslaan of een keuze maken tussen een efficiënt OODBMS. Uiteraard kun je er ook voor kiezen om niet het object maar alleen de data op te slaan. Je moet dan iedere keer dat je de data ophaalt er weer een object van maken. Die conversie moet je dan zelf doen, en kost rekenkracht, dus energie.

Dit zijn wellicht de belangrijkste voorbeelden van database management systemen. Belangrijk is dat een keuze gedreven wordt door de eisen die gesteld worden. Op basis daarvan moet het meest duurzame platform gekozen worden in plaats van het platform dat toevallig voor handen is. Ook hier zien we dat de kennis van de IT specialist ontzettend belangrijk is.

# OVER DE SCHRIJVERS

**Jan Schravesande** is een IBM en Open Group gecertificeerde Executive Enterprise Architect en heeft ervaring in verschillende bedrijfstakken en industrieën zoals, Verzekering, Supply Chain, Energy en Utilities en de overheid. De laatste tien jaar is hij actief geweest als technisch adviseur voor organisaties in de Nederlandse overheid.

*"Duurzaamheid is een onderwerp dat ons allemaal moet bezig houden. Toen ik begon met dit boekje was ik op veel vlakken nog onbewust onbekwaam maar ben ik inmiddels 'gepromoveerd' tot bewust onbekwaam en op sommige vlakken zelfs bewust bekwaam. Ik moet eerlijk zeggen dat het schrijven van dit boekje mijn kijk op het leven enigszins heeft veranderd".*



**Ronald Meijer** is een gecertificeerd IT Architect werkzaam bij IBM. Vanuit deze organisatie heeft hij voor organisaties gewerkt zoals Rabobank, ABN/AMRO, Shell, KLM, ING, Delta Lloyd, Aegon. Hij is een gepassioneerd trainer op het gebied van TOGAF(R), Architecture Thinking en Microservices Modellering. Zijn ervaring en kennis strekt zich uit over de breedte van IT architectuur, te weten business, data, applicatie en infrastructuur architectuur. Afgestudeerd in elektrotechniek met als specialisatie technische computerkunde. Hij spreekt regelmatig op conferenties en in verschillende forums.



*"Duurzaamheid is iets van ons allemaal. Ik ervaar een gepaste, maar ook gepassioneerde "ergernis" als ik zie dat met de huidige stand van de 2 nm chiptechnologie bedrijven 75-90% energiebesparing laten liggen. Het verbaasd me ook als men zich dan verschuilt achter groene energie. Kom op zeg, eerst uitzetten wat je niet gebruikt, dan besparen op je infra en tot slot wat je gebruikt, ja dat doe je natuurlijk groen".*

---

**Laila Fettah** is an Open Group and IBM certified IT Specialist, who started in the world of statistics and has spent the first ten years of her career in finding what matters in data-driven projects. While working on efficiency projects in industrial sector, supporting marketing projects or unraveling value from unstructured documents, bridging technology and business has been her passion.

*This has driven her to her current architect role to bring those two together more holistically, with a nod to her past: “make it simple, but significant”.*



---

**Edwin van der Burg ....**

---

Een belangrijke waarde van duurzaamheid is openheid. Open met elkaar discussiëren en transparant rapporteren is een voorwaarde om de wereld duurzamer te maken. Dat geldt ook voor IT. Daarom is dit boekje ook vrij te gebruiken en je kunt het vinden op GitHub. Een platform waar we kunnen delen.

Kopiëren en delen staat vrij zolang je het maar gebruikt om jou organisatie duurzamer te maken! Als je wilt reageren graag! Hieronder staan onze email adressen en neem gerust contact met ons op.

Het boekje is te vinden in PDF formaat via deze link: <https://github.com/SustainableIT/Downloads#readme>

Or use the QR code:

Laila Fettah: lailafettah@nl.ibm.com

Ronald Meijer: meijerr@nl.ibm.com

Jan Schravesande: schravesande@nl.ibm.com

Edwin van der Burg: evdburg@nl.ibm.com

Beste Lezer,

We leven in een tijd dat wij met elkaar een aanslag doen op onze planeet. We weten dit en we kunnen met elkaar beredeneren dat de uitstoot van CO<sub>2</sub> de aarde opwarmt. Dat schaarse bronnen met hoog tempo opgebruikt worden en dat schaars betekend dat het een keer op is. Het woord “op” komt niet in ons woordenboek voor. Misschien is er wel het gevoel dat we er wel weer wat op bedenken want we moeten vooruit!

Moeten we vooruit? Wat is vooruit?

Er zijn heel veel vragen waar wij geen antwoord op hebben als schrijvers van dit boekje. Wij zijn in het dagelijks leven IT architecten, geen schrijvers dus! Maar op één gebied zijn we goed, we hebben verstand van IT systemen. We hebben met z'n vieren hieraan gewerkt dus 4 x goed is best wel heel goed.

We zijn gestart met het schrijven van dit boekje met ons verstand, maar je zult tijdens het lezen merken dat het een hartaak is geworden. We proberen in dit boekje duidelijk te maken wat de impact van IT is op duurzaamheid. Toen we begonnen wisten wij ook maar een kwart van wat er nu allemaal instaat. We zijn overdonderd door gegevens over het energiegebruik van IT datacenters, van AI, en wat dacht je van cryptocurrency als bitcoin.

We hopen dat je vanuit je onderbewustzijn bewust mag worden van de omvang van de problemen die door de IT veroorzaakt worden. Maar daar laten we het niet bij. We hebben ook ideeën hoe we met elkaar kunnen verduurzamen. Jou bijdrage is een druppel op een gloeiende plaat, maar met 100.000-en ITers in de wereld druppelen we die plaat 1,5 graad naar beneden!

Amsterdam,  
Januari, 2023