

Duurzaamheid van Software

Een eerste verkenning van aandachtspunten voor inkopers

Copper8

Inhoudsopgave



Dit rapport is opgesteld in opdracht van het Ministerie van Infrastructuur en Waterstaat, voor de Buyer Group ICT.
Voor vragen over deze opdracht, of de Buyer Group ICT, kun je contact opnemen via BuyerGroupICT@rws.nl.

Auteurs
Stefan Favrin (Copper8)
Moja Reus (Copper8)

Datum
November 2022

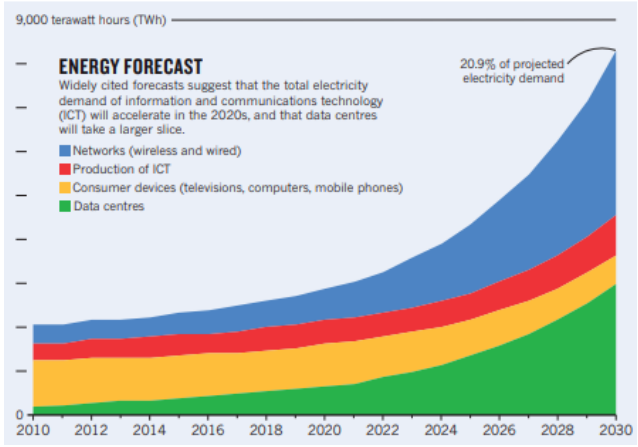
Inhoudsopgave 2
Hoofdstuk 1. Aanleiding 3
Hoofdstuk 2. De Software keten 5
Hoofdstuk 3. Duurzaamheidsthema's 9
Hoofdstuk 4. Inkooptools voor Duurzame Software 18
Bijlage A. Scope van het rapport 26

Hoofdstuk 1. Aanleiding

De afgelopen jaren is er veel aandacht besteed aan het verduurzamen van de inkoop van hardware producten. Deze hardware wordt vervolgens benut door software. Iedereen met een laptop zal bekend zijn met Microsoft Windows, Word, Excel, Powerpoint et cetera. Er zijn echter talloze applicaties die constant een rol spelen in het gebruik van hardware en de interactie met de gebruiker. Voor al deze applicaties is software nodig. Deelnemers van de Buyer Group ICT hebben de wens om software duurzaam in te kopen. Hoe dit te doen, is echter nog niet duidelijk.

Urgentie

In het Klimaatakkoord is afgesproken dat we in 2030 in Nederland 49% minder CO_2-eq uitstoten dan in 1990. Om die vermindering te behalen, wordt technologie – en daarmee ook software – naar voren geschoven als een veelbelovend middel. Zo kan automatisering helpen met efficiëntere processen en geeft het gebruik van monitoringsystemen inzicht in waar de meeste winst te behalen valt. Toch zijn er ook geluiden dat we hierin niet moeten doorslaan. Zo is op dit moment de carbon footprint van ICT al gelijk aan de totale emissies van vliegverkeer[1]. Naar verwachting zal ons elektriciteitsverbruik van ICT nog verder toenemen van 8% naar 21% van het globale elektriciteitsverbruik.

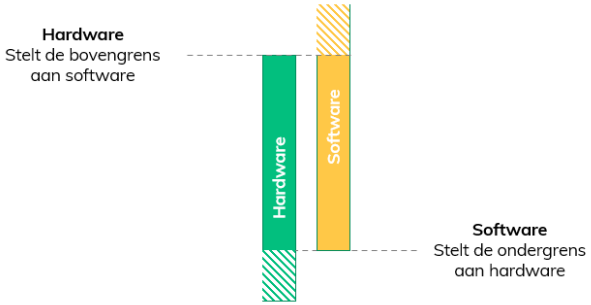


Figuur 1 | Het verwachte energieverbruik van ICT tot 2030

Het gebruik van software speelt hier een grote rol in. Om software te kunnen laten draaien, is reken- en opslagcapaciteit nodig. Het is daarom belangrijk dat de software die we gebruiken, zo duurzaam mogelijk wordt ingezet. Anders lopen we het risico dat de digitalisering, die zo geprezen wordt in de duurzaamheidstransitie, het probleem wordt.

Software i.r.t. hardware

Software en hardware hebben elkaar nodig. Hardware en software leggen beiden een limiet neer, de hardware bepaalt de bovengrens van de software en de software bepaalt de ondergrens van de benodigde hardware.



Figuur 2 | Software i.r.t. hardware

Dit is weergegeven in Figuur 2. Zoals software invloed kan hebben op hardware, kan hardware ook invloed hebben op de energie-efficiëntie van software. Dit heeft met name betrekking op de CPU (de processor): een efficiëntere CPU betekent automatisch ook minder energieverbruik van de software. Echter, er is bij LAM (Laptops, accessoires en mobile devices) geconstateerd dat

[1] Bron: Jones, N.(2018). The information factories..

de grootste milieuwinst te behalen is d.m.v. levensduurverlenging van het apparatuur.

Doel van dit rapport

Het doel van dit rapport is om een overzicht te bieden van de softwareketen en de mogelijke kansen daarin weer te geven om de inkoop van software te verduurzamen. Daarmee is de **centrale** vraag in dit rapport:

Wat zijn de effecten van software op duurzaamheidsthema's en hoe kunnen inkopende organisaties duurzaamheid in softwareketens bevorderen?

Om deze vraag te beantwoorden bespreken we achtereenvolgens:

- Wat zijn de belangrijkste duurzaamheidsrisico's en kansen in de keten van software?
- Welke stakeholders in de keten kunnen invloed uitoefenen op het verduurzamen van software, en hoe kunnen zij daar in bijdragen?
- Welke inkooptools kunnen gebruikt worden in aanbestedingen?

Dit rapport moet aangrijpingspunten geven om het gesprek in de Buyer Group verder vorm te geven en helder te krijgen waar kansen liggen.

Definitie

Software is een containerbegrip dat constant in ontwikkeling is. In dit rapport worden de volgende definities gehanteerd:

Onder **software** verstaan we computer programma's, besturingssystemen en applicaties met data die operaties op een computer systeem realiseren.

- **Standaard software** is kant-en-klare software die beschikbaar is voor de algemene markt en verschillende klanten, ook wel bekend als COTS (Commercial off-the-shelf).
- **Maatwerk software** is software die speciaal ontwikkeld is voor de opdrachtgever. Als de broncode van de software wordt aangepast gaat het vaak om maatwerk software.

De focus van dit rapport ligt op het verduurzamen van software. Voor duurzame software gebruiken we de volgende definitie:

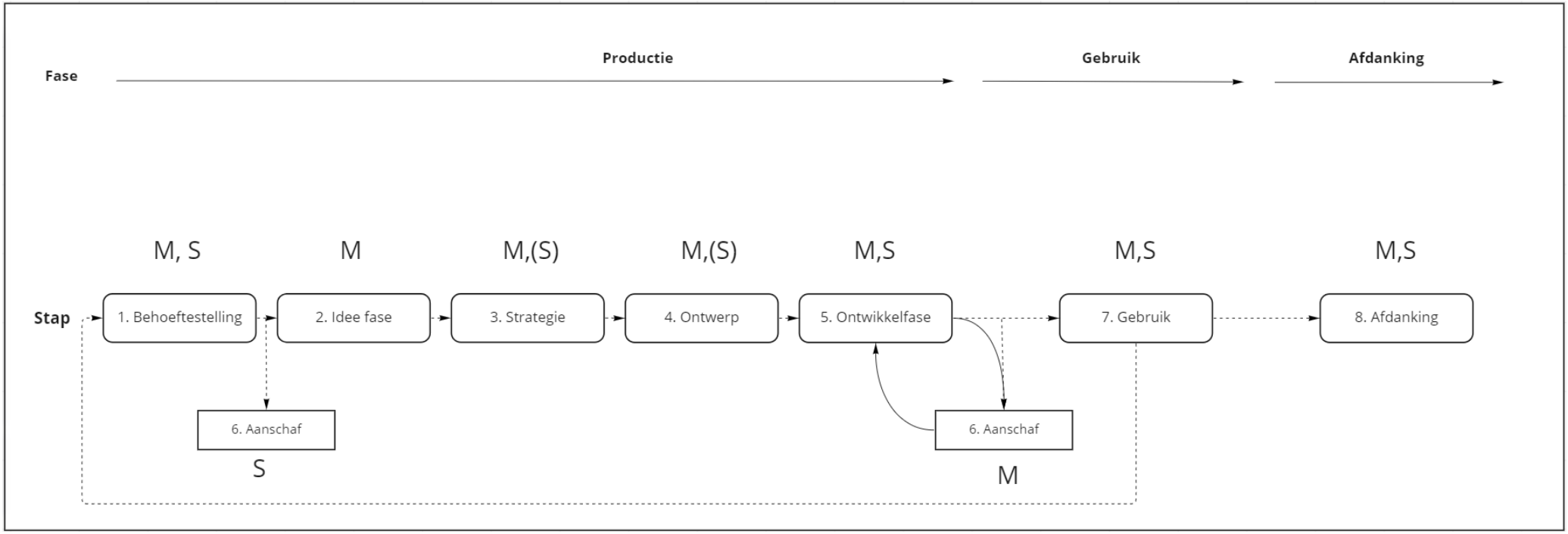
Duurzame software is efficiënt in energieverbruik. Daarmee is het efficiënt geschreven, maakt het efficiënt gebruik van hardware, wordt het efficiënt gebruikt en goed gede-installeerd. De software wordt ontwikkeld op een sociaal verantwoorde manier, zowel in de keten (het respecteren van Internationale Sociale Voorwaarden) als bij de leverancier zelf (in de vorm van Social Return en Diversiteit en Inclusie).

Hoofdstuk 2. De Software keten

Zoals elke keten bestaat de software keten uit een aantal stappen. De complexiteit van de keten is een stuk lager dan bijvoorbeeld de hardware keten, omdat er geen fysieke producten worden gemaakt. In Figuur 3 beschrijven we de software keten in acht stappen:

1. Behoeftestelling
2. Idee-fase
3. Strategie
4. Design
5. Development
6. Aanschaf en levering
7. Gebruik
8. De-installatie

Het hangt af van de expertise en behoefte van de inkopende organisatie met betrekking tot software bij welke stap in keten de opdracht verstrekt wordt. Sommige organisaties zoeken bij stap 1 een broker of ontwikkelaar. Andere organisaties, vaak met intern meer inhoudelijke kennis aan boord, betrekken een ontwikkelaar in een veel later stadium. Bij de inkoop van standaard software vindt de aanschaf (ofwel levering) vaak direct na de behoeftestelling plaats. Omdat de software dan kant-en-klaar is, hebben



Figuur 3 | Stappen in de keten voor maatwerk (M) en standaard (S) software. Voor ontwikkelaars is het proces dynamisch. Vaak is er bij gebruik weer een andere functionaliteit nodig, waardoor de ontwikkelaar weer naar stap 1 'behoeftestelling' gaat.

de stappen 2 tot en met 4 of 5 al plaatsgevonden. Bij maatwerk worden partijen vaak eerder betrokken. Welke inkooptools van toepassing zijn hangt af in welke stap de uitvraag gedaan wordt. In dit rapport gaan we uit van een uitvraag in Stap 1, en waar de aanschaf (ofwel levering) plaatsvindt in stap 6.

Stap 1: Behoeftestelling

Over het algemeen start het proces bij de opdrachtgever. De opdrachtgever heeft een bepaalde behoefte die kan worden voldaan met software. Een voorbeeld: een organisatie heeft software nodig om de boekhouding te automatiseren. De opdrachtgever heeft daarbij bepaalde eisen en wensen waar de software aan moet voldoen en heeft een bestaand ecosysteem van overige software en data waar de software aan gekoppeld kan worden. Dit uit zich in:

- Dimensionering van de software
- Eisen en criteria

Voor (publieke) aanbestedende diensten, indien er extern hulp gezocht wordt in deze stap, wordt dit opgeschreven in een aanbestedingsdocument of wordt er een marktconsultatie gehouden. Private partijen kunnen direct over hun behoefte in gesprek gaan met een bepaalde software partij. Ook worden er geschiktheidseisen geformuleerd.

Duurzame software? Refuse!

Het duurzaamst is om geen software in te kopen of te gebruiken. Vraag eerst af; is hetgeen wat we willen doen echt nodig? Mocht het nodig zijn, is er dan software voor nodig? Als er software nodig is, wordt er binnen de organisatie al software gebruikt die de taak kan uitvoeren? Vaak hebben software pakketten meer functionaliteiten dan die actief gebruikt worden.

De behoeftestelling bepaalt of de software die wordt uitgevraagd maatwerk software of standaard software moet zijn. Bestaat de software al, en voldoet dat aan de wensen? Dan kan er voor standaard software gekozen worden. Anders is het wellicht nodig om de software te ontwikkelen. In het voorgaande voorbeeld over boekhouding software is de kans groot dat er standaard software bestaat.

De volgende stap in de keten is echter afhankelijk van de keuze voor maatwerk- of standaard software, of een combinatie ervan. Een paar voorbeelden:

a) Volledig standaard software kan zonder al teveel aanpassingen ingepast worden in het huidige software ecosysteem van een organisatie. In dit geval is de volgende ketenstap stap 5 of 6.
b) Volledig maatwerk software. In dit geval gaat er nog een ontwikkelfase aan vooraf en start het proces bij stap 2.
c) Hybride software. Het kan zijn dat er een standaard software pakket wordt ingekocht, maar dat het nog niet past binnen de huidige data architectuur en ecosysteem (het kan bijvoorbeeld nog niet communiceren met ander software binnen de organisatie). In dat geval moet er een strategie opgesteld worden, en moeten er mogelijk nog wat aanpassingen worden gedaan aan de software om het binnen de organisatie passend te krijgen. In dit geval start het proces in stap 3.

Stakeholders

- Opdrachtgever
- Software partij. Dit is de partij waaraan de opdracht wordt verstrekt. Voor maatwerk software is dit vaak een partij die in Nederland gevestigd is, al hoeft dit niet te betekenen dat de ontwikkeling ook daadwerkelijk in Nederland plaatsvindt (zie stap 5). Voor standaard software kan dit ook een internationale partij zijn, zoals Oracle, Microsoft of Adobe. Dit soort grote bedrijven hebben vaak een kantoor in Nederland, ongeacht waar de ontwikkeling van de software plaatsvindt.

Stap 2: Idee fase

In stap 1 is er een partij uitgekozen die de software kan ontwikkelen. Deze softwarepartij gaat aan de slag met de behoefte van de

opdrachtgever en werkt een idee uit dat voldoet aan de wensen van de opdrachtgever. Hierbij worden criteria als haalbaarheid, designwensen en andere randvoorwaarden meegenomen.

Stakeholders
Dezelfde partijen als in stap 1.

Stap 3: Strategie

Software opereert niet in isolatie. Software moet 'samenwerken' met andere types software die bestaan binnen de organisatie of andere hardware-gebonden software. Ook moet de software ergens op draaien (hardware) en moet er bijvoorbeeld een keuze worden gemaakt wáár het gaat draaien (on-premise, off-premise, via de cloud of niet, etc.). In deze strategie fase spelen dus een aantal vragen:

- Wie is de eindgebruiker van de software? Hoe moet de software daar op inspelen?
- Hoe ziet de huidige data architectuur van de opdrachtgever eruit? Zijn daar specifieke wensen voor?
- Hoe past de software binnen het bestaande ecosysteem van de organisatie?

Na het beantwoorden van deze vragen, kan er een plan van aanpak worden geschetst waar de software aan moet voldoen en hoe de software uiteindelijk in de organisatie kan worden geïmplementeerd.

Stakeholders
Voor het strategische vraagstuk in de software-ontwikkeling kan het zijn dat er een extern 'strategy consulting' bureau wordt aangehaakt, naast de partij die de opdracht heeft aangenomen.

Stap 4: Design

In de designfase wordt de software verder ontwikkeld tot een werkend prototype. Met dit prototype kan vervolgens getest worden of dit voldoet aan de wensen van de opdrachtgever, en of er geen fouten in zitten. Deze fase draait dus om ontwikkelen en valideren van een prototype.

Stakeholders
In deze fase kan het voorkomen dat een externe partij, een zogenaamde 'digital creative agency', wordt aangehaakt.

Stap 5: Ontwikkelfase

In de ontwikkelfase wordt de definitieve versie van de software ontwikkeld. Voor maatwerk software betekent dit het verder ontwikkelen van het prototype tot volledig werkende software. Voor standaard software is deze fase bedoeld voor het up-to-date houden van de software van de leverancier, in de vorm van een update-cyclus.

Voor beide type software gebeurt het (verder) ontwikkelen van software vaak in 'development cycles' van ontwikkelen, testen en valideren.

Stakeholders
In de ontwikkelfase is de trend gaande dat steeds meer van het werk uitbesteed wordt. Dit kan gebeuren 'near-shore' in landen als Portugal of Oost-Europa, of 'off-shore' in landen als India, Pakistan en Rusland. Deze keuze wordt vaak gemaakt om kosten te drukken. Het gevaar is echter dat juist door deze kritieke stap naar het buitenland te outsourcen, de software zelf niet efficiënt wordt ontwikkeld voor de originele behoefte. Voor deze buitenlandse partijen is het namelijk vooral zaak om zo snel mogelijk een werkend stuk code te leveren.

Stap 6: Aanschaf en Levering

De oplevering van de software vindt plaats in stap 6. Het is afhankelijk van het type software en de situatie in welke stap van het proces een contract wordt afgesloten, en dus ook welke stappen in het software ontwikkelproces nog beïnvloed kunnen worden door de inkoper. Voor maatwerk gaan we in dit rapport uit van contractafspraken in stap 1 waardoor de daaropvolgende ketenstappen binnen het contract vallen. Voor standaard software vinden de contractafspraken vaak plaats in stap 6, aangezien de meeste ketenstappen zich al gezet zijn.

Stakeholders
Er zijn in deze fase drie type stakeholders te onderscheiden:

- Maatwerk softwarepartij
- Standaard softwarepartij
- Software broker

Een broker ontwikkelt zelf geen software. Het is bij wijze van spreken een groothandel voor software. Het kan wel voorkomen dat een broker ook andere diensten aanbiedt, zoals advisering rondom het implementeren van bepaalde software in hun assortiment.
Aangezien een broker geen software ontwikkeling doet, is het complexer om eisen te stellen aan producten die bij hen worden ingekocht. Zij hebben namelijk geen directe invloed op het product zelf. Dit kan betekenen dat het lastig is om te voldoen aan duurzaamheidseisen als er bij een broker wordt ingekocht. Wel kan je bij een broker wensen neerleggen. Als een broker van meerdere van zijn klanten dezelfde duurzaamheidsvraag binnen krijgt, kan hij deze wel gebundeld neerleggen bij de originele producent (OEM).

Stap 7: Gebruik

Na de inkoop van de software, volgt een gebruiksfase waarin de software wordt ingezet om de initiële behoefte te vervullen. Voor de gebruiksfase kunnen afspraken gemaakt zijn over de service gedurende de contractduur, bijvoorbeeld bij bugs of voor continue updates. Daarnaast heeft de inkopende organisatie zelf ook invloed op de eindgebruiker van de software en heeft deze daarmee invloed op het uiteindelijke energieverbruik van de software.

Stakeholders

- De gebruiker/opdrachtgever.
- De ICT service-afdeling (intern of extern)
- Serviceprovider. Dit kan dezelfde partij zijn als de ontwikkelaar, maar dat hoeft niet.

Stap 8: Afdanking

Anders dan hardware heeft software niet per se een fysieke einde levensduur, zolang de software ondersteund blijft. Het is daarom belangrijk om goede afspraken te maken over de continuïteit van het software-pakket, aangezien zonder ondersteuning er sneller (veiligheids)problemen kunnen ontstaan. Dit resulteert niet alleen in de aankoop van nieuwe software, want soms betekent het ook dat er nieuwe hardware nodig is. Neem het voorbeeld van een smartphone: als een bepaalde Android/Apple versie niet meer wordt ondersteund, is het vaak nodig om ook een nieuw apparaat aan te schaffen dat de nieuwste software wel kan draaien.

Een andere reden voor het beëindigen van een softwarecontract is omdat het niet meer voldoet aan de behoefte die op dat moment leeft. Dit kan zijn omdat de behoeftes zijn veranderd, of omdat er andere softwarecomponenten bij zijn gekomen die de behoefte beter bedienen.

Stakeholders

- De gebruiker/opdrachtgever
- OEM

Hoofdstuk 3. Duurzaamheidsthema's

Het verduurzamen van de IT-omgeving staat al een tijd op de kaart. Toch staat het verduurzamen van software nog in de kinderschoenen. Enerzijds kan dit komen omdat hardware het gros van de duurzaamheidsimpact vertegenwoordigt. Anderzijds is duurzaamheid binnen software nog niet goed verkend. Zo leven er nog volop vragen, zoals:

- Waar ligt de impact van software?
- Welke ketenstappen hebben de meeste impact, en hoe kunnen opdrachtgevers die beïnvloeden?
- Maakt het type software (maatwerk, standaard, hybride) uit voor het uitvragen van duurzaamheid?

In dit hoofdstuk zal een deel van deze vragen behandeld worden. Een kwantitatieve duiding van de impact van software is echter lastig te geven. Er is immers nog geen standaard meetmethode. Ook kan het daadwerkelijke energieverbruik van een stuk software afhangen van andere factoren, zoals de hardware waar het op draait, de IT-omgeving waarin het wordt geïntegreerd en hoe het door de gebruiker wordt benut. Toch zijn er een aantal thema's en onderwerpen aan te duiden die zowel vanuit de literatuur als uit de markt worden aangeduid als significante onderwerpen.

Daarnaast zijn er vanuit het Maatschappelijk Verantwoord Inkopen (MVI) beleid een aantal verschillende duurzaamheidsthema's als belangrijk aangewezen voor duurzame inkoopprocessen[2], waaronder:

- Klimaat & energie
- Circulair
- Milieu (Biodiversiteit)
- Social Return
- Diversiteit & Inclusie
- Internationale Sociale Voorwaarden (ISV) of ketenverantwoordelijkheid

Uit het onderzoek blijkt dat voornamelijk het thema Klimaat (en Energie) relevant is voor het verduurzamen van software. Het gebruiken van software heeft direct impact op energieverbruik. Ook heeft het software ontwikkelproces directe impact op de arbeidsomstandigheden van ontwikkelaars (ISV).
Indirect heeft software impact op de andere thema's. De belangrijkste indirecte thema is Circulair omdat software hardware nodig heeft om te functioneren.

Dit betekent niet dat de overige thema's niet van belang kunnen zijn, of niet uitgevraagd kunnen worden. De sociale thema's 'Social Return' en Diversiteit en Inclusie zijn in dit rapport niet uitgewerkt, omdat hier al duidelijke richtlijnen voor bestaan en deze onderwerpen niet specifiek zijn voor software, maar betrekking hebben op de leverende organisatie. Een aantal voorbeelden zijn wel opgenomen in de inkooptools. Op het thema Milieu – en daarbinnen Biodiversiteit – heeft software zelf weinig impact, behalve indirect door de grondstoffen die verbruikt worden door de hardware die nodig is om software te ontwikkelen en te gebruiken.

De impact van software

De voornaamste impact van software vindt plaats in de gebruiksfase, als gevolg van het energieverbruik van het oproepen van de software componenten en de opslag van data als

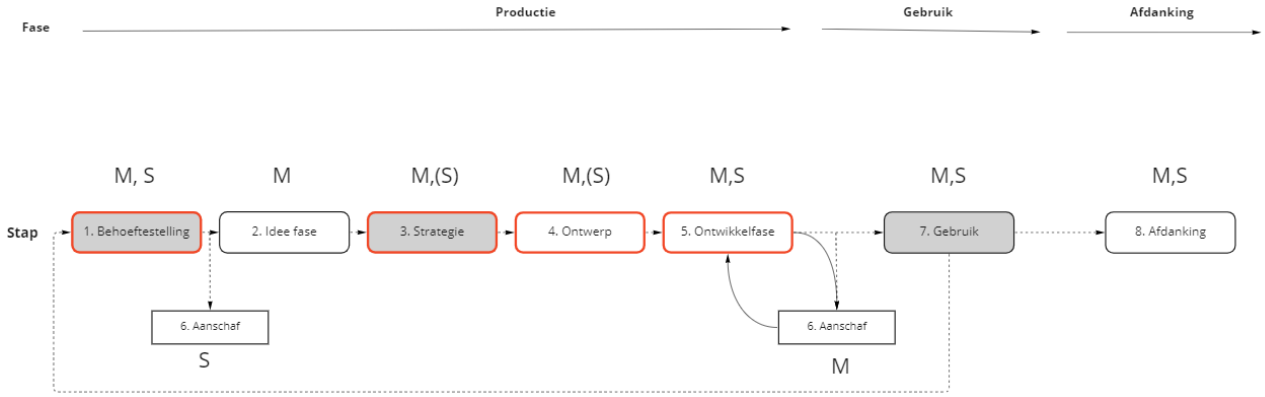
[2] Bron: Pianoo, Maatschappelijk Verantwoord Inkopen

gevolg van het gebruik van de software. Om deze impact van gebruik te verminderen zijn er verschillende maatregelen te nemen, waaronder het aanpassen van het gedrag van de gebruikers van de software. Hierbij kun je bijvoorbeeld denken aan hoe vaak en hoe lang de gebruiker de software open heeft, hoe de gebruiker omgaat met opslag zoals dubbele kopieën, et cetera. Dit ligt echter niet in de invloedsfeer van de inkoper van het softwarepakket, maar van de afdeling die verantwoordelijk is voor de implementatie, het ICT beheer en de eindgebruiker.

De inkoper speelt wel een bepalende rol in de eisen die aan de software wordt gesteld, waardoor het een hoog of laag energieverbruik heeft. Dit begint bij de *behoeftestelling* – de meest bepalende stap. In deze stap wordt namelijk bepaald waar de software aan moet voldoen. Cruciale vragen zijn:

- Is de inkoop van nieuwe software überhaupt nodig?
- Zo ja, wat zijn de hoofdzakelijke functies die het moet hebben? Zorg dan dat alleen deze functies worden vervuld, en dat functies uit kan schakelen.

Naast de behoeftestelling zijn ook de stappen ‘Strategie’, ‘Design’ en ‘Ontwikkeling’ bepalend voor de uiteindelijke duurzaamheidsprestaties. Deze stappen bepalen namelijk hoe de volgende aspecten zijn ingeregeld:



Figuur 4 | De stappen van software ontwikkeling. Grijze vakken zijn de stappen waar de meeste impact wordt gemaakt. De gebruiksfase, stap 7, heeft de grootste impact. De rood omlijnde vakken zijn de stappen waar de waar de inkoper het meeste invloed op heeft. M = Maatwerk software, S = Standaard software

- Hoe past de software binnen het eigen software/hardware ecosysteem? (Strategie)
- Hoe wordt het opgeslagen en opgeroepen? (Strategie/Design)
- Wordt de software efficiënt geschreven en ingericht? (Design/Development)

Het is vooral belangrijk dat de software niet meer functionaliteiten heeft dan nodig, efficiënt wordt ingeregeld, het oproepen ervan zo min mogelijk energie kost en dat de servers optimaal benut worden.

Duurzaamheidsaspecten per stap in de keten

Per stap in de keten zijn er een aantal onderwerpen die bepalend zijn voor de duurzaamheidsimpact van software. Deze worden hieronder per stap toegelicht. Bij elk onderwerp wordt aangegeven welk duurzaamheidsthema daarbij hoort en welke inkoopcriteria erbij passen.

Stap 1: Behoeftestelling

Thema's - Energie & Klimaat

Dimensionering

Voor iedere functie kan in principe software worden ontwikkeld. Soms kun je je echter afvragen, of die ene extra functie of het extra snelle laadvermogen wel écht nodig is. Want wat de gebruiker vaak niet ziet, is het energieverbruik dat daarmee gemoeid is. Een voorbeeld uit een ontwikkelaarsblog van Microsoft:

> "Consider what we learned when we trained an artificial intelligence (AI) model on a small, publicly available dataset of iris flowers. The AI model achieved accuracy of 96.17% in classifying the flowers' different species with only 964 joules of energy. The next 1.74%-point increase in accuracy required 2,815 joules of energy consumption. The last 0.08% incremental increase in accuracy took nearly 400% more energy than the first stage."
>
> *Source: Sustainable Software Engineering (SSE) and the role and responsibilities of a Sustainable Software Engineer*

Onnodige softwarecomponenten worden ook wel 'bloatware' genoemd. Zorg bij het uitvragen van duurzame software dat je samen op zoek gaat naar de optimale balans tussen enerzijds functionaliteit, en anderzijds energieverbruik. Wellicht kan er gevraagd worden of de gebruiker zelf functionaliteiten aan of uit kan zetten.

Stap 2: Idee fase

In de idee fase wordt de behoefte die besproken is in stap 1 verder uitgewerkt. Voor deze fase zijn er geen inkooptools.

Stap 3: Strategie

Thema's – Energie & Klimaat en Circulair

Data architectuur/Compatibiliteit met het ecosysteem

Ten eerste wordt er in de strategiefase gekeken naar de data architectuur, en daarmee de compatibiliteit met het ecosysteem van bestaande soft- en hardware. Het aantal andere systemen waar de software mee communiceert en de effectiviteit van die communicatie heeft effect op de vraag naar capaciteit van hardware en het energieverbruik. Er kan gestuurd worden op optimalisatie van de software met het ecosysteem.

Methode van opslag

Ten tweede wordt er gekeken naar hoe de software opgeslagen wordt. Door slim om te gaan met opslag kan energie bespaard worden. Er kan gevraagd worden hoe de software met de opslag van data omgaat.
Bij het ontwikkelen van nieuwe functionaliteiten worden verschillende versies van de software in de lucht gehouden. Zorg dat na het ontwikkelen van de nieuwe functionaliteit geen extra versies in de lucht blijven.

Utilisatiefactor van servers maximaliseren

Ten derde wordt er in deze fase ook gekeken hoe de utilisatiefactor van de servers gemaximaliseerd kan worden. De utilisatie, ofwel energieproportionaliteit, is de relatie tussen de hoeveelheid geconsumeerde energie en hoeveelheid effectief werk dat verricht wordt. Door de complexe interacties tussen hardwarecomponenten is deze verhouding niet lineair. Hoe meer een computer gebruikt wordt, hoe efficiënter de computer de elektriciteit omzet in nuttige computer operaties. Een computer die aan staat maar niet gebruikt wordt heeft een relatief hoog statisch energieverbruik. Daarvoor bestaat de slaap of energie zuinige modus. Servers hebben deze modi niet. Het is verstandig om de utilisatiefactor van servers te maximaliseren zodat ze het meest efficiënt energie verbruiken voor het werk dat ze verzetten. Door minder servers optimaler te benutten zijn er wellicht minder servers in totaal nodig. Via de cloud kunnen servers optimaler benut worden, al hoeft dat niet te betekenen dat ze ook echt optimaler benut worden. Om de utilisatiefactor te maximaliseren kan je bijvoorbeeld gebruik maken van function-as-a-service of containerisatie. Je invloed op de utilisatiefactor van servers is afhankelijk van of de servers on-premise of off-premise zijn. Vraag uit hoe de utilisatiefactor van de servers gemaximaliseerd wordt.

Containerisatie zorgt ervoor dat je software efficiënt op kan schalen. Een container is een pakket met software en onderdelen die de software ondersteunen bij het draaien. Zowel bij **virtual machines** (VM) als containers wordt virtualisatie toegepast waardoor de hardware efficiënter gebruikt wordt. Bij VM's worden meerdere virtual machines op één server gerund via een hypervisor (software). Bij een container vindt de virtualisatie plaats op een andere laag van de stack en wordt het besturingssyteem gevisualiseerd in plaats van de hardware. Iedere VM heeft een eigen besturingssysteem, CPU, opslag, netwerkadapters en BIOS. Containers delen één besturingssysteem (OS), dezelfde kernel en netwerkverbinding.

Transparantie

Hoe en waar de software wordt gerund is ook van belang. Gaat het om on-premise, of om off-premise software via een derde partij? De inkoper heeft weinig invloed op hoe off-premise software gerund wordt. Probeer in dit geval transparantie bij de derde partij te bevorderen.

Efficiënt opschalen van software

Ten vierde heeft de methode van opschalen van de software effect op het energieverbruik. Via bijvoorbeeld virtualisatie en containerisatie kan je delen van de software oproepen die je nodig hebt, wanneer je ze nodig hebt en in de mate dat je ze nodig hebt. Vraag uit hoe er voor wordt gezorgd dat alleen het deel van de software benut wordt, wat op dat moment nodig is.

Stap 4 & 5: Design & Ontwikkeling

Thema's – Energie & Klimaat en ISV

Tijdens de ontwikkelfase kan er gekeken worden naar energie efficiënte software en ISV. ISV wordt behandeld onder stap 7. Onderwerpen die effect hebben op het energieverbruik in deze stap zijn: de codeertaal, de lengte van de code en de energiekosten voor het ontwikkelproces. Het is lastig uitvragen op de energiekosten van het ontwikkelproces. De milieuimpact van de deze energiekosten heeft te maken met de koolstofintensiteit van het net van de locatie waar ontwikkeld wordt, welke hardware gebruikt wordt en hoe de hardware gebruikt wordt door de ontwikkelaars. Dit laten we hier buiten beschouwing omdat we ervanuit kunnen gaan dat het energieverbruik in de ontwikkelfase vele malen kleiner is dan dat in de gebruiksfase.

Code

Des te efficiënter de code, des te efficiënter het energieverbruik. Een kortere code kost vaak minder energie. Ook de programmeertaal heeft invloed op het energieverbruik. Zo zijn R en C++ twee van de efficiëntste codeertalen, in tegenstelling tot bijvoorbeeld het populaire Python. Maar uitvragen alléén op programmeertaal lijkt niet verstandig. Elke programmeertaal heeft een bepaalde toepassing. Hierdoor kan er makkelijker of sneller een bepaalde functionaliteit ontwikkeld worden. De vraag is dan of de langere ontwikkeltijd met een energie efficiëntere programmeertaal opweegt tegen de kortere programmeertijd van een beter passende programmeertaal (zowel in energie als kosten). Bovendien kunnen niet alle programmeurs in energie efficiëntere talen programmeren. Je kunt wel de vraag stellen hoe een optimale balans wordt gevonden tussen het minimaliseren van het energieverbruik en het leveren van functionaliteit.

Modulair programmeren

Bij modulair programmeren wordt de software in modules geprogrammeerd. De functionaliteiten van de software worden hierbij in modules opgesplitst waardoor de modules zo veel mogelijk op zichzelf kunnen functioneren. Hierdoor heeft de software vaak ook een hogere herbruikbaarheid omdat de modules apart te veranderen zijn.

Stap 6: Aanschaf en Levering

Thema's - Energie & Klimaat, Social Return en ISV

Bij de aanschaf van software kan er op meer dan alleen het verduurzamen van het product zelf gestuurd worden. Dit geldt voor zowel maatwerk als voor standaard software. Zo kunnen er bijvoorbeeld eisen worden gesteld aan de

bedrijfsvoering, de keten, of aan eisen in de Service-Level Agreement (SLA).

SLA
Voor leveranciers is stabiliteit van hun software het belangrijkst omdat ze volgens de SLA een bepaalde uptime moeten realiseren. Hierdoor wordt software vaak over-gedimensioneerd. In de SLA staan ook afspraken over de responsetijd en hoe snel een bug moet worden opgelost. Bekijk de eisen die in de SLA worden gesteld goed. Een nóg snellere reactietijd van de software betekent ook meer energieverbruik. Kunnen de eisen functioneler gesteld worden?

Bedrijfsvoering
Zeker omdat duurzame software nog niet een thema is dat sterk op de kaart staat bij bedrijven, kan het helpen om in de aanbesteding aandacht te schenken aan duurzame bedrijfsvoering. Het idee hierachter is dat als bedrijven een goede visie hebben én intern aan duurzaamheid werken, dat dit uiteindelijk ook overhevelt naar het product zelf. Vraag echter niet alleen duurzame bedrijfsvoering uit, maar ook wat zij doen aan het verduurzamen van de software zelf; het is een goed begin, maar pas het halve werk.

Daarnaast geldt voor de sociale thema's wel dat het belangrijk is dat een organisatie aandacht besteed aan een (sociaal) duurzame keten. Met name in de ontwikkelfase (stap 5) wordt er vaak werk uitbesteed aan landen – zoals India en Pakistan – waar gelet moet worden op sociale omstandigheden (mensenrechten en arbeidsrechten).

Om te bepalen of een organisatie aan duurzame bedrijfsvoering doet, kunnen er verschillende aspecten worden uitgevraagd. Zo kan er gevraagd worden naar certificeringen, CO_2-prestaties van een bedrijf of hun visie op duurzaamheid of een aspect daarvan (bijvoorbeeld energieverbruik, social return of ISV). In het geval dat de wens is om kleinere organisaties (MKB) een kans te geven, is het aan te raden om minder de focus te leggen op het aanleveren van certificeringen en juist meer de aandacht te leggen op de belangrijkste aspecten van de certificering.

Energie & Klimaat
Een organisatie kan op een aantal bedrijfsonderdelen besparingen realiseren: het eigen energieverbruik, transport (eigen wagenpark), kantoorinrichting, horeca en IT-apparatuur. Dit kan enerzijds worden uitgevraagd door specifiek naar deze onderdelen te vragen, of anderzijds door certificeringen uit te vragen. Certificeringen die eventueel kunnen worden uitgevraagd zijn:

- CO_2-prestatieladder
- Ecovadis

Social Return
Social return is voor de Rijksoverheid een verplicht onderdeel bij aanbestedingen van werken en diensten met een loonsom van ten minste €250.000,- (excl. btw) en een looptijd van ten minste zes maanden. Er zijn verschillende manieren waarop een inschrijver op social return getoetst kan worden.

- *Percentage van de opdrachtsom.*

De meest voorkomende optie is het toepassen van 5% social return t.o.v. de opdrachtsom, inclusief een plan van aanpak hier invulling aan wordt gegeven.

- *Maatwerk voor Mensen.*

Deze tweede optie geeft een vrijere invulling aan de traditionele eis. De aanpak van Maatwerk voor Mensen is gericht op de samenwerking tussen HR, Inkoop en Leveranciers om een betere inzet te waarborgen dan wanneer het simpelweg geëist wordt.

- *PSO-Prestatieladder.*

Deze certificering heeft een vergelijkbare aanpak als de CO_2-prestatieladder, waarbij sociaal ondernemerschap een geïntegreerd onderdeel wordt van de bedrijfsvoering.

Internationale Sociale Voorwaarden
Met name voor de ontwikkelfase van software kunnen er risico's ontstaan op het gebied van

Internationale Sociale Voorwaarden. Normaalgesproken zijn er een tweetal aanpakken om ISV in een aanbesteding mee te nemen zoals hieronder beschreven.

- *ILO-conventies*

Er zijn de acht basisconventies van de ILO betreffende arbeid onder dwang, kinderarbeid, discriminatie en de vrijheid van vereniging en het doen van collectieve onderhandelingen (no. 29, 87, 98, 100, 105, 111, 138 en 182). Deze moeten in ieder geval gewaarborgd worden.

- *Gepaste zorgvuldigheid*

Van bedrijven die internationaal opereren wordt verwacht dat zij 'due diligence'[3] toepassen; een proces van gepaste zorgvuldigheid. Een tweede gangbare optie is dan ook om een verbeterplan uit te vragen in de vorm van een contractvoorwaarde bestaande uit drie onderdelen:

- Risicoanalyse
- Plan van Aanpak
- Jaarlijkse rapportage

Er kan gevarieerd worden met hoe specifiek dit verbeterplan wordt ingezet. Zo kan – in het geval van software – specifiek worden uitgevraagd hoe er wordt omgegaan met near- en off-shoring, en wat er wordt gedaan om risico's in de sociale werkomstandigheden te voorkomen.

[3] Due Diligence | OESO-richtlijnen | Nationaal Contactpunt OESO-richtlijnen (oesorichtlijnen.nl)

Het is van belang om de criteria zo te formuleren, dat ze betrekking hebben op de activiteiten in de keten waar de risico's plaatsvinden.

Stap 7: Gebruik

Thema's – Energie & Klimaat en Circulair

De inkoper kan niet direct beïnvloeden *hoe* er in de gebruiksfase met de software om wordt gegaan. Wel kan de inkoper ervoor zorgen dat de software zo goed mogelijk aansluit op de randvoorwaarden van de eigen organisatie (zoals bestaande hardware), dat de gebruiker de juiste tools ter beschikking heeft om energieverbruik te verlagen en dat de software zo lang mogelijk mee gaat. Om de impact in de gebruiksfase te verminderen, is van belang om ook de beheerders en gebruikers van de software te betrekken.

Overigens ligt er ook een duidelijke verantwoordelijkheid bij de gebruiker zelf in de keuzes hoe de software gebruikt wordt. Alles wat opgeslagen wordt vereist energie om bewaard te blijven. Door slim om te gaan met opslag kan er energie bespaard worden. Bijvoorbeeld door dubbele kopieën te verwijderen, of data te comprimeren in kleinere bestanden. Andere keuzes zijn:

- Lagere kwaliteitsinstellingen gebruiken dan het maximaal mogelijke
- Energiebesparingsmodus aanzetten
- De software niet aan laten staan als het niet in gebruik is
- De software offline gebruiken indien het niet nodig is om de online functionaliteiten te gebruiken.

Om dit te realiseren is het van groot belang dat gebruikers geïnformeerd zijn over de mogelijkheden die zij hebben om duurzaamheid te beïnvloeden.

Energie-instellingen

Om het gebruikers eenvoudiger te maken om energieverbruik te minimaliseren, kunnen inkopers een aantal eisen stellen aan de software, zoals:

- Inzicht in de benodigde hardware capaciteit
- Mogelijkheid om energie- en performance instellingen aan te passen
- Offline gebruik
- Energiemonitoring

Inzicht in benodigde hardwarespecificaties voor de software.

Let goed op of de software boven of dicht tegen het limiet van de hardwarecapaciteit van je eigen organisatie zit. Is dit het geval, is het aan te raden om goed te kijken naar de benodigde dimensionering van het product en de eigen behoefte. Het kan het er namelijk snel toe leiden

dat er nieuwe hardware moet worden ingekocht om de software te kunnen gebruiken. Nieuw hardware inkopen heeft veel meer impact dan het hele software ontwikkel- en gebruiksproces.

Mogelijkheid om energie- en kwaliteitsinstellingen aan te passen
Het kan voorkomen dat er binnen de organisatie meerdere type gebruikers zijn; medewerkers die zwaar gebruik maken van software, en medewerkers die er minder gebruik van hoeven te maken. Door het mogelijk te maken om de prestaties, snelheid of beeldkwaliteit van de software aan te passen, kan hierop worden ingespeeld. Wellicht hebben bepaalde medewerkers bijvoorbeeld niet alle functies nodig, of hoeven er niet elke dag intensief mee te werken. In dat geval kan er energie worden bespaard worden door hier op in te spelen.

Daarnaast moet het mogelijk zijn om energie-instellingen aan te passen. Een energiebesparende modus kan aantrekkelijk zijn, zeker voor gebruikers die niet de volle rekenkracht nodig hebben.

Offline gebruik
Vermijd onnodige verbindingen met een licentie server, onnodige (herhaaldelijke) downloads van bijvoorbeeld lettertypes of andere online connecties.
Energiemonitoring
Door energieverbruik te monitoren en duidelijk in beeld te brengen, wordt het eenvoudiger om hier op te sturen. Ook wordt het voor gebruikers inzichtelijk wat de consequenties zijn van bepaalde keuzes in hun energie- en kwaliteitsinstellingen.

Circulariteit
Om software zo lang mogelijk mee te laten gaan binnen de hardwarecapaciteit van de organisatie, zijn er een aantal aspecten waarop gestuurd kan worden in een aanbesteding.

Continuïteit
Het is van belang om te sturen op een periode waarin gegarandeerd ondersteuning wordt gegeven met (veiligheids)updates. Des te langer deze garantie kan worden gegeven, des te beter. Zonder veiligheidsupdates wordt het al snel nodig om nieuwe software (en daarmee soms ook hardware) in te moeten kopen.

Backward-compatibiliteit
Maak afspraken met welk referentiesysteem (bijvoorbeeld: huidig besturingssysteem) het softwarepakket compatibel moet blijven binnen de contractduur (of langer als dat nodig is).

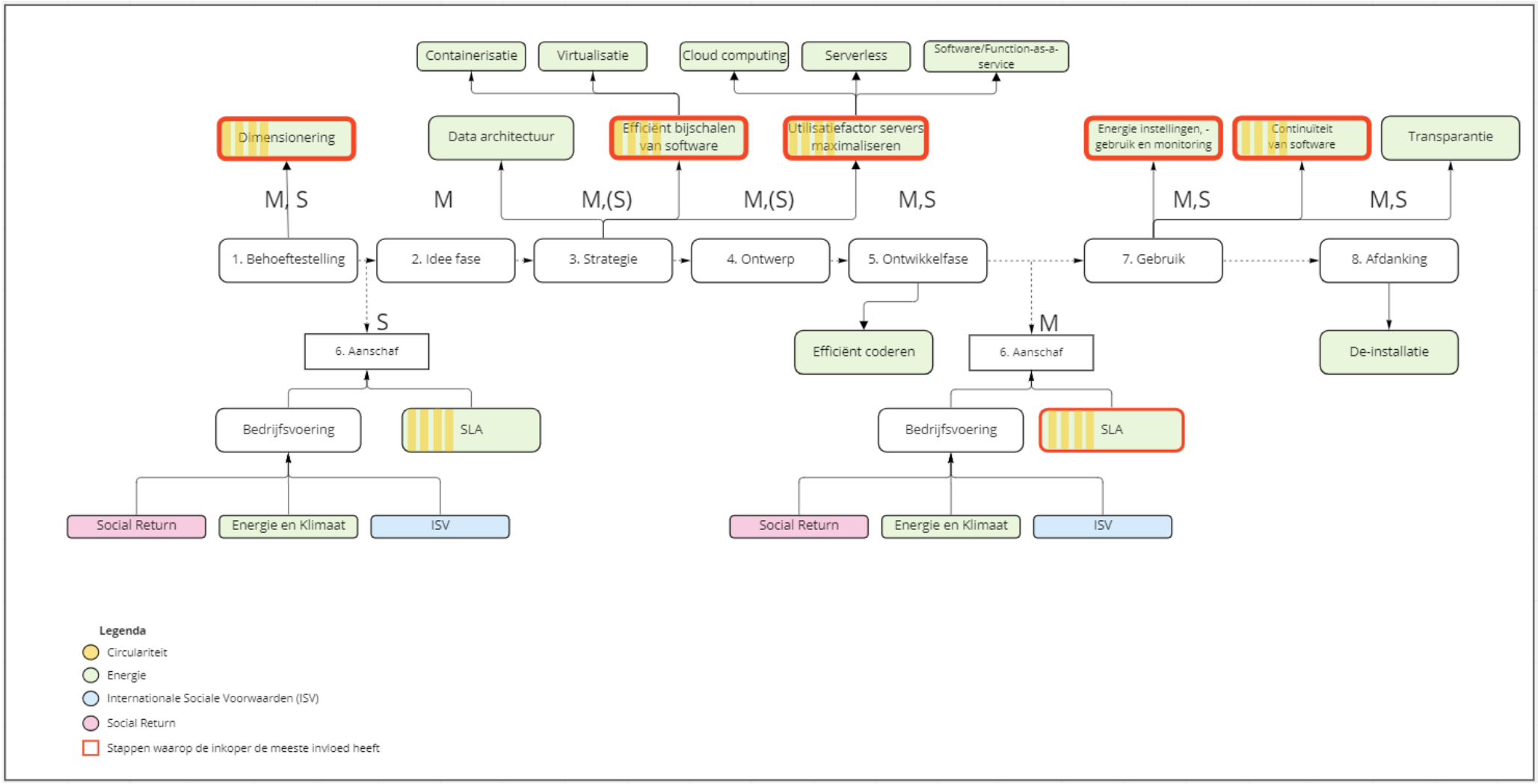
Updates / nieuwe versies
Om te blijven voldoen aan de behoefte van de gebruikers, worden constant nieuwe versies ontwikkeld of updates gedaan. Zorg ervoor dat deze updates er niet voor zorgen dat het energieverbruik sterk toeneemt, of dat de hardware het niet meer aankan.

Data format transparency
Om te waarborgen dat de data formats die worden gebruikt compatibel zijn met andere software wat in gebruik is binnen de organisatie, kan de eis worden gesteld dat de inschrijver inzichtelijk maakt; a) met welke data formats er gewerkt wordt, b) aan welke open standaarden deze voldoen en c) met welke andere data formats het kan samenwerken.

Stap 8: Afdanking

Thema's - Energie & Klimaat

Als een software pakket – om wat voor reden dan ook – aan het einde van de gebruiksduur komt, is het belangrijk dat alles op een goede manier wordt gede-installeerd en er niet onnodig (data)opslag over blijft.



Figuur 5 | De software keten stappen met de relevante onderwerpen. Per onderwerp is aangegeven op welk MVI thema het betrekking heeft.

Conclusie

Inkopende organisaties kunnen duurzaamheid van software bevorderen door op verschillende duurzaamheidsonderwerpen te sturen bij de inkoop van software. De belangrijkste uitgangspunten zijn hierbij:

Koop niet in wat niet nodig is

Zet in op het verminderen van energieverbruik als de inkoop van software echt nodig is. Want de grootste impact van software zit in het energieverbruik. In de eerste plaats doe je dit door de software goed te dimensioneren; vraag niet meer functionaliteiten uit dan nodig.

Optimaliseer de software in het ecosysteem

De compatibiliteit van de software met het ecosysteem is van belang voor het energieverbruik. Vraag uit hoe de software in het ecosysteem wordt ingeregeld.

Schaal efficiënt op

Er kan uitgevraagd worden via welke methodes de software wordt opgeschaald, en hoe alleen het benodigde deel van de software benut wordt.

Ga efficiënt om met opslag

Er kan uitgevraagd worden hoe de software omgaat met de opslag van data, en hoe er om wordt gegaan met verschillende versies van de software.

Onderhoud software goed zodat hardware langer meegaat

De interactie met hardware is ook van belang. Goed onderhouden software kan leiden tot levensduurverlening van hardware.

Maximaliseer de utilisatiefactor van servers

Het maximaliseren van de utilisatiefactor van servers kan leiden tot minder servers die energetisch efficiënter runnen.

Stimuleer transparantie in de keten

Het is van belang om transparantie binnen de softwareketen te stimuleren. Er kan uitgevraagd worden waar de ontwikkeling plaatsvindt en of deze wordt uitbesteed. Near-shoring heeft minder risico's dan off-shoring, maar beiden zijn gebaat bij een goede risico-analyse en een plan van aanpak om verbeteringen in de arbeidsomstandigheden te waarborgen. Ook kan je kijken naar de bedrijfsvoering van de softwareleverancier.

Het type software en moment van inkoop heeft invloed op toepasbare inkooptools

Het inkoopproces is afhankelijk van het type software, maatwerk of standaard, en van het moment van inkoop in het ontwikkelproces. Voor standaard, kant-en-klare, software, ligt er voornamelijk de kans om transparantie in de keten te bevorderen. Doe dit door wensen in plaats van eisen te stellen. Bij maatwerk software heeft de inkopende organisatie meer controle over het ontwikkelproces.

Stimuleer transparantie bij off-premise software

Hoe en waar de software wordt gerund is ook van belang. Gaat het om on-premise, of om off-premise software via een derde partij? De inkoper heeft weinig invloed op hoe off-premise software gerund wordt. Probeer in dit geval transparantie bij de derde partij te bevorderen.

Betrek de ICT beheerder bij inkoop

Aangezien de grootste impact van software in de gebruiksfase ligt, waar de invloed van de inkoper beperkt is, kan het een goed idee zijn om de ICT beheerder van de organisatie te betrekken in het inkoopproces.

Vervolgstappen

In Hoofdstuk 4 staan inkooptools die door de inkoper gebruikt kunnen worden. Een vervolgstap kan zijn om deze uitgebreide lijst terug te brengen naar een aantal, op basis van praktijkervaringen, goed werkbare inkooptools. Ook kan een marktconsultatie gehouden worden waar de inkooptools bij de markt getoetst kunnen worden. Ten slotte vraagt de omgang met datacenters een verdiepingsslag. Dit rapport kan als basis dienen voor deze verdiepingsslag.

Hoofdstuk 4. Inkooptools voor Duurzame Software

In dit hoofdstuk staan de geformuleerde inkooptools. Als basis zijn de inkooptools uit het rapport 'Handreiking Duurzame Inkoop Informatievoorziening'[4] gebruikt. Deze zijn uitgebreid op basis van de inzichten in dit rapport. Inkopende organisaties kunnen voor hun relevante inkooptools kiezen op basis van deze uitgebreide lijst. Ook kunnen de inkooptools aangepast worden op de specifieke context van de uitvraag. Op basis van praktijkervaring kan de lijst verkort worden tot een aantal, in de praktijk, goed toepasbare inkooptools.

De inkooptools zijn als volgt gestructureerd:

- Energie & Klimaat
- Circulair
- Bedrijfsvoering
- Sociale aspecten
- Arbeidsaspecten in de keten

De omgang met datacenters valt buiten de scope van dit rapport. Een korte toelichting op datacenters, off-premise en on-premise software is te vinden in Bijlage A.

Meetbaarheid

De inkooptools zijn niet meetbaar volgens een protocol zoals of certificaat zoals dat wel bij hardware het geval is (zoals bijv. Energy Star in Laptops Accessoires en Mobiele apparatuur). Het is aan zowel de aanbestedende als de inschrijvende partij om samen te overleggen hoe meetbaarheid wordt gegarandeerd in de opdracht. Als er in meerdere opdrachten zo over meetbaarheid na wordt gedacht, ontstaat er een lerend effect. Dit kan helpen om inkopende organisaties in staat te stellen om in de toekomst een voor hen passend meetmethodiek te ontwikkelen om software te beoordelen en monitoren op verschillende aspecten van duurzaamheid.

Toekomstige ontwikkelingen

De inkooptools zijn niet statisch en zullen mee veranderen met toekomstige ontwikkelingen. Het is denkbaar dat in de toekomst gevraagd kan worden aan software partijen om aan te tonen op welke wijze zij via hun primaire proces (het ontwikkelen van software) een bijdrage leveren aan Maatschappelijk Verantwoord Ondernemen (MVO). Op termijn zou wellicht zelfs gevraagd kunnen worden om aan te tonen dat een gedeelte van hun omzet komt van software tools die een positieve bijdrage leveren aan MVO. Het is ook denkbaar dat de inkooptools die nu toepasbaar zijn op maatwerk in de toekomst ook toepasbaar zullen worden voor standaard software, om zodoende op basis van duurzame prestaties onderscheid te maken tussen leveranciers.

Legenda

GhE	Geschiktheidseis
E	Eis
SC	Selectiecriteria
GC	Gunningscriteria
CB	Contractbepaling
En	Energie & Klimaat
B	Bedrijfsvoering
C	Circulair
S	Sociale Aspecten
I	Internationale Sociale Voorwaarden
G	Gezondheid & Welzijn

[4] van Oppen, C., Favrin, S., Polet, M., & van Exter, P. (2020, April). Handreiking Duurzame Inkoop Informatievoorziening.

Energie en Klimaat

Energieverbruik van software

Een belangrijk thema binnen Energie en Klimaat is dat de Inschrijver het energieverbruik van de software kan beperken, ook bij nieuwe versies. Onderstaande inkooptools gaan in op dit thema.

Type inkooptool	Specifiek	Type	Onderwerp
Geschiktheid seisen	**En_GhE_1.** De Gegadigde heeft aantoonbare relevante competenties en ervaring in het optimaliseren van software met het doel om energieverbruik te minimaliseren.[5]	Standaard	Geschiktheidseisen
	En_GhE_2. De Gegadigde kan aantonen dat zij de relevante competenties heeft om software te optimaliseren met het doel om energieverbruik te minimaliseren. Als bewijslast wordt gevraagd om in een korte beschrijving (max. 2 A4) de aanpak en competenties te omschrijven en te onderbouwen.	Maatwerk	Geschiktheidseisen
Eisen	**En_E_1.** De te leveren software heeft de functionaliteit om de benodigde hardware capaciteit aan te passen aan de benodigdheden om de software te runnen. Hier zitten een aantal eisen aan vast: • De software kan automatisch de benodigde capaciteit minimaliseren en/of de opties geven om dit te kunnen doen bij de installatie van de software • De gebruikers hebben de optie om dit later aan te passen in de installatieopties. • En de software moet het mogelijk maken om randapparatuur of functies (tijdelijk) uit te schakelen als deze niet vereist zijn. [6]	Standaard, Maatwerk	Dimensionering
	En_E_2. Energie-instellingen van het te leveren product zijn makkelijk toegankelijk en te vinden op één plek. De energie-instellingen worden automatisch aangepast zodat energieverbruik geminimaliseerd wordt. Het is voor de beheerder mogelijk om zelf energie-instellingen aan te passen. Energiebesparingsmodus moet standaard beschikbaar zijn.[6]	Standaard, Maatwerk	Energie instellingen
	En_E_3. Energieverbruik van de software moet worden gemonitord en gebruikers moeten dit energieverbruik makkelijk kunnen inzien.[6]	Standaard, Maatwerk	Energie instellingen
	En_E_4. De software moet duidelijk aangeven als het taken in de achtergrond aan het draaien is, of als het automatisch taken start die niet per se nodig zijn voor de hoofdfunctie van het product.[6]	Standaard, Maatwerk	Energie instellingen
	En_E_5. Het product moet transparant zijn over waar data is opgeslagen. Wanneer het ge-deïnstalleerd wordt, moet het (helder) de optie geven om alle data te wissen, zonder sporen ervan na te laten.[8]	Standaard, Maatwerk	Transparantie, de-installatie
	En_E_6. Het product zal alleen verbinden met het Internet wanneer nodig. Voorbeelden van onnodige gedwongen verbindingen: verbinden met een licentie server, herhaaldelijke download van gevraagde lettertypes.[6]	Standaard, Maatwerk	Data architectuur
	En_E_7. Het is van belang om te sturen op een periode waarin gegarandeerd ondersteuning wordt gegeven met (veiligheids)updates. Zonder veiligheidsupdates wordt het al snel nodig om nieuwe software, en daarmee soms ook hardware, in te moeten kopen. De Inschrijver moet daarom de volgende informatie verschaffen:	Standaard, Maatwerk	Continuïteit

[5] Alfieri, F., Sanfelix, J., Bernad, D., Graulich, K., Moch, K. (2019), Revision of the EU Green Public Procurement (GPP) Criteria for Computers and Monitors. Beschikbaar op: https://susproc. jrc.ec.europa.eu/computers/docs/Technical_Report_GPP_Computers_v1.pdf

[6] Kern, E., Hilty, L.M., Guldner, A., Maksimov, Y.V., Filler, A., Gröger, J., Naumann, S., (2018), Sustainable software products – towards assessment criteria for resource and energy efficiency: Set of criteria for sustainable software; Future Generation Computer Systems

<table>
<tr><td></td><td>

- De periode waarin veiligheidsupdates worden verschaft. Deze periode moet minimaal X jaar zijn.
- Het aantal dagen tot een update wanneer een veiligheid kwetsbaarheid wordt ontdekt. Het maximale aantal dagen voor een update is <aantal> dagen.

</td><td></td><td></td></tr>
<tr><td>Gunningscriteria</td><td>

En_GC_1. De Inschrijver geeft in een omschrijving van max. 2 pagina's aan hoe bij het geleverde product wordt omgegaan met nieuwe versies. Minimaal de volgende aspecten worden hierin meegenomen:

- Welke programmeertaal is gebruikt, en hoe deze scoort t.a.v. energie-, tijd- en geheugenefficiëntie*[7];
- Een onderbouwing van de keuze voor programmeertaal in het kader van energie-efficiëntie;
- Hoe de architectuur van de software toekomstbestendig is qua energie-efficiëntie;
- Er wordt in gegaan op de gevolgen die nieuwe versies hebben (gehad) voor hardware (capaciteit) en energieverbruik;
- Er wordt aangegeven hoe de Inschrijver voorziet in het verlagen van hardware (capaciteit) en energieverbruik.[6]

*Voor de beoordeling van dit aspect, zie bijvoorbeeld het onderstaande tabel uit het artikel van Pereira et al. (2017) en een lijst van up-to-date programmeertalen op https://blog.andwork.com/wat-spreken-developers-dit-zijn-de-8-meest-gevraagde-programmeertalen/

	Energy
(c) C	1.00
(c) Rust	1.03
(c) C++	1.34
(c) Ada	1.70
(v) Java	1.98
(c) Pascal	2.14
(c) Chapel	2.18
(v) Lisp	2.27
(c) Ocaml	2.40
(c) Fortran	2.52
(c) Swift	2.79
(c) Haskell	3.10
(v) C#	3.14
(c) Go	3.23
(i) Dart	3.83
(v) F#	4.13
(i) JavaScript	4.45
(v) Racket	7.91
(i) TypeScript	21.50
(i) Hack	24.02
(i) PHP	29.30
(v) Erlang	42.23
(i) Lua	45.98
(i) Jruby	46.54
(i) Ruby	69.91
(i) Python	75.88
(i) Perl	79.58

	Time
(c) C	1.00
(c) Rust	1.04
(c) C++	1.56
(c) Ada	1.85
(v) Java	1.89
(c) Chapel	2.14
(c) Go	2.83
(c) Pascal	3.02
(c) Ocaml	3.09
(v) C#	3.14
(v) Lisp	3.40
(c) Haskell	3.55
(c) Swift	4.20
(c) Fortran	4.20
(v) F#	6.30
(i) JavaScript	6.52
(i) Dart	6.67
(v) Racket	11.27
(i) Hack	26.99
(i) PHP	27.64
(v) Erlang	36.71
(i) Jruby	43.44
(i) TypeScript	46.20
(i) Ruby	59.34
(i) Perl	65.79
(i) Python	71.90
(i) Lua	82.91

	Mb
(c) Pascal	1.00
(c) Go	1.05
(c) C	1.17
(c) Fortran	1.24
(c) C++	1.34
(c) Ada	1.47
(c) Rust	1.54
(v) Lisp	1.92
(c) Haskell	2.45
(i) PHP	2.57
(c) Swift	2.71
(i) Python	2.80
(c) Ocaml	2.82
(v) C#	2.85
(i) Hack	3.34
(v) Racket	3.52
(i) Ruby	3.97
(c) Chapel	4.00
(v) F#	4.25
(i) JavaScript	4.59
(i) TypeScript	4.69
(v) Java	6.01
(i) Perl	6.62
(i) Lua	6.72
(v) Erlang	7.20
(i) Dart	8.64
(i) Jruby	19.84

</td><td>Maatwerk</td><td>Data architectuur, code</td></tr>
<tr><td></td><td>**En_GC_2** De Inschrijver schrijft een plan van aanpak hoe softwarefuncties en deelcomponenten van software worden opgeroepen, op welke momenten en via welke methodes. De Inschrijver beschrijft wat het effect hiervan is op het energieverbruik van de software.</td><td>Maatwerk</td><td>Efficiënt opschalen, Utilisatiefactor servers</td></tr>
<tr><td></td><td>**En_GC_3.** De Inschrijver schrijft een plan van aanpak hoe de software gedimensioneerd wordt op de behoefte van de gebruiker en licht daarmee toe hoe het energieverbruik van de software geminimaliseerd wordt.</td><td>Maatwerk</td><td>Dimensionering</td></tr>
</table>

[7] Pereira, R., Couto, M., Ribeiro, F., Rua, R., Cunha, J., Fernandes, J. P., & Saraiva, J. (2017, October). Energy efficiency across programming languages: how do energy, time, and memory relate?. In Proceedings of the 10th ACM SIGPLAN International Conference on Software Language Engineering (pp. 256-267).

	En_GC_4. De Inschrijver schrijft een plan van aanpak hoe de software wordt geïntegreerd in het huidige soft- en hardware ecosysteem binnen de organisatie, en hoe de interactie hiermee geoptimaliseerd wordt op minimale energieverbruik.[8]	Maatwerk	Data architectuur
	En_GC_5. De Inschrijver schrijft een plan van aanpak hoe de opslag van data (ook van de software zelf) data wordt geminimaliseerd.	Maatwerk	Methode van opslag
	En_GC_6. De Inschrijver beschrijft hoe de utilisatiefactor van de servers waar de software op draait wordt gemaximaliseerd.	Maatwerk	Utilisatiefactor servers
Contractbepalingen	**En_CB_1.** De Inschrijver biedt de mogelijkheid aan de gebruiker om zelf functionaliteiten van de software aan of uit te zetten om zelf regie te houden over het energieverbruik. De Inschrijver krijgt: • 0 punten als dit niet kan. • 1 punt als de opdrachtgever het kan met ondersteuning van de Inschrijver. • 2 punten als de opdrachtgever het zelf kan. • 3 punten als de opdrachtgever het zelf kan met pro-actieve ondersteuning.	Standaard, Maatwerk	Dimensionering

8 Gröger, J., & Köhn, M. (2016). Leitfaden zur umweltfreundlichen öffentlichen Beschaffung: Produkte und Dienstleistungen für Rechenzentren und Serverräume. Ratgeber. Hg. v. Umweltbundesamt (UBA). Online verfügbar unter https://www.umweltbundesamt. de/sites/default/files/medien/376/publikationen/leitfaden_zur_um weltfreundlichen_oeffentlichen_beschaffung_-_produkte_und_dienstleistungen_fuer_rechenzentren_und_serverraeume. pdf.

Circulair			
Verminderen materiaalgebruik Indirect kan bij de inkoop van software gestuurd worden op het minimaliseren van materiaalgebruik door levensduurverlenging van hardware. Verder kan in beperkte mate middels de aanbesteding gestimuleerd worden dat Inschrijvers zich gaan onderscheiden op het vlak van het verminderen van materiaalgebruik. Onderstaande inkooptools kunnen hierbij benut worden.			
Type inkooptool	**Specifiek**	**Type**	**Onderwerp**
Eisen	**C_E_1.** De software moet worden ingekocht, geïnstalleerd en geopereerd (inclusief updates) zonder het transporteren van fysieke opslag (inclusief papier) of andere materialen (inclusief verpakkingen).[6]	Standaard, Maatwerk	-
	C_E_2. De software moet compatibel zijn met eerdere versies op een referentiesysteem voor X jaar. [6]	Maatwerk	Backward compatibiliteit
	C_E_2. De Inschrijver moet de minimale hardware specificaties leveren die benodigd zijn om de software te runnen. Daarnaast moeten meerdere configuraties mogelijk zijn om te zorgen dat de software ook op apparaten kan draaien met lagere technische specificaties. [6]	Standaard, Maatwerk	Continuiteit
Contractbepalingen	**C_CB_1.** Het ideale software product neemt toe in functionaliteit bij nieuwe versies, zonder dat dit meer eisen stelt aan de hardware (capaciteit). Dit zal gemeten worden op het moment dat nieuwe versies zijn gelanceerd, aan de hand van de volgende hiërarchie: 1. "Zeer goed": nieuwe versies hebben geleid tot lagere hardware capaciteit 2. "Goed" nieuwe versies hebben niet geleid tot hogere hardware capaciteit 3. "Voldoende": nieuwe versies hebben geleid tot hogere hardware capaciteit. De toename in hardware capaciteit zijn niet overgecompenseerd door de genomen efficiëntie maatregelen als gevolg van technische factoren. 4. "Onvoldoende": als gevolg van nieuwe versies, is de hardware capaciteit meer toegenomen dan de technische efficiëntie. Gedurende de looptijd van het contract moet er een minimale score van 3 gehaald worden; wanneer bij twee achtereenvolgende meetmomenten een score van 4 wordt gehaald, heeft de Opdrachtgever het recht om het Contract te ontbinden.[6]	Maatwerk	Dimenisonering & continuiteit software

Bedrijfsvoering

Duurzame bedrijfsvoering

Aanbestedingen kunnen duurzame bedrijfsvoering van de Inschrijvers stimuleren. Voor Standaard software is dit wellicht beter inzetbaar als Selectiecriterium, voor Maatwerk software zou ook een Contractbepaling opgenomen kunnen worden.

Type inkooptool	**Specifiek**	**Type**	**Onderwerp**
Geschiktheid seisen	**B_GhE_1.** Gegadigde kan aantonen dat zij een milieumanagement systeem heeft conform ISO 14001, EMAS of vergelijkbaar. [22]	Standaard	Geschiktheids eisen
Selectiecriteria	**B_SC_1.** Gegadigde geeft aan hoe zij werkt aan duurzaamheid in haar eigen bedrijfsvoering. Beoordelingsaspecten: • Wat het huidige niveau is van de prestaties van de eigen bedrijfsvoering (waar mogelijk kwantitatief gemaakt a.d.h.v. CO_2-emissies (scope 1, scope 2 en waar mogelijk scope 3 emissies* vanuit het GHG protocol)[27]; • De reeds behaalde resultaten in de afgelopen drie jaar t.a.v. het verbeteren van prestaties ondersteund met verifieerbare prestatie informatie; • De doelstellingen die de organisatie heeft m.b.t. de verduurzaming van de eigen bedrijfsvoering; Een plan van aanpak waarin de doelstellingen worden onderbouwd, al dan niet met beleidsdocumenten. • De doelstellingen zoals hier vermeld worden, indien over wordt gegaan op definitieve gunning, opgenomen als Contractbepaling.	Maatwerk	Energie & Klimaat
Contractbepalingen	**B_CB_1.** Opdrachtnemer levert elk jaar een managementrapportage op waarbij inzicht wordt geboden in de scope 1 en 2 emissies. Ook wordt hierin aangegeven welke verbeteringen zijn doorgevoerd in dat jaar en tot welke mate van CO_2 besparing dit heeft geleid.	Maatwerk	

Sociale aspecten			
Inzet mensen met een afstand tot de arbeidsmarkt Software bedrijven kunnen ook inzetten op sociale aspecten door mensen met een afstand tot de arbeidsmarkt in te zetten voor het primaire bedrijfsproces. Een voorbeeld is de Sociale Enterprise ITVitae, die het streven heeft om een opleiding en baanbemiddeling te bieden voor personen met autisme en/of hoogbegaafden ongeacht hun vooropleiding.			
Type inkooptool	**Specifiek**	**Type**	**Onderwerp**
	S_GhE_1. Gegadigde stemt ermee in om na gunning van de raamovereenkomsten, binnen 30 dagen na een verzoek van opdrachtgever, een voorstel (plan van aanpak) in te dienen voor inrichting en uitvoering van een social return groeituin. • Een social return groeituin is erop gericht om sociale impact te realiseren door mensen met (grotere) afstand tot de arbeidsmarkt aan werk en inkomen te helpen of het perspectief van deze mensen op werk en inkomen te verbeteren. • U heeft ruimte om uw eigen ambities/ideeën voor het leveren van sociale impact te vertalen naar een concreet voorstel voor een proeftuin, die aansluit bij de mogelijkheden en wensen van de deelnemende dienst(en) en die u de mogelijkheid biedt deze ambities te realiseren. • De doelstelling voor de met deze groeituin te realiseren (additionele) sociale impact is concreet, ambitieus doch haalbaar en realistisch. • Uw voorstel voor een social return groeituin wordt na afstemming met en acceptatie door opdrachtgever en deelnemende dienst onderdeel van de raamovereenkomst. U gaat vervolgens de social return groeituin uitvoeren op basis van een inspanningsverplichting. • Er mag bij de social return proeftuin geen sprake zijn van verdringing of oneerlijke concurrentie op de arbeidsmarkt, waarbij de gerealiseerde sociale impact ten koste gaat van banen van ander personeel.	Standaard, Maatwerk	Social return
Selectiecriteria	**S_SC_1** Gegadigde levert een beschrijving aan van max. 1 A4 waarin wordt aangegeven: • Hoe Gegadigde structureel werkt aan de inzet van Social Return in de organisatie, al dan niet ondersteund met beleidsdocumenten; • Wat de prestaties zijn geweest op het vlak van Social Return in de afgelopen drie jaar; • Wat de doelstellingen zijn ten aanzien van Social Return voor de aankomende drie jaar.	Maatwerk	Social return
Contractbepalingen	**S_CB_1** Opdrachtnemer levert elk jaar een rapportage met betrekking tot inzet van mensen met afstand tot de arbeidsmarkt (MAAM), waarbij bijvoorbeeld, maar niet uitsluitend, wordt aangegeven hoeveel mensen zijn ingezet of op een andere manier een bijdrage geleverd wordt aan de ontwikkeling van mensen, op welke functies, en wat de groeipotentie is met betrekking tot dit onderwerp. Voor meer informatie aangaande dit onderwerp, zie: https://www.maatwerkvoormensen.nl/praktijkvoorbeelden/rws-groeituin	Maatwerk	Social return

Arbeidsaspecten in de keten

Internationale Sociale Voorwaarden

Wanneer er in het buitenland geprogrammeerd wordt, is het van belang om ook arbeidsaspecten in de keten mee te nemen in de aanbesteding. Zoals al eerder in dit hoofdstuk is toegelicht, liggen de werktijden van de gemiddelde programmeur in China ruim boven de norm van de Internationale Arbeidsorganisatie (IAO); dit is een voorbeeld van een arbeidsomstandigheid waarbij softwarebedrijven een goed voorbeeld kunnen zijn. Onderstaande inkooptools kunnen benut worden om een positieve impact te hebben.

Inkooptool	Specifiek	Type	Onderwerp
Geschiktheidseisen aanbieder	**I_GhE_1.** Gegadigde onderschrijft de acht basis conventies van de ILO betreffende arbeid onder dwang, kinderarbeid, discriminatie en de vrijheid van vereniging en het doen van collectieve onderhandelingen (no. 29, 87, 98, 100, 105, 111, 138 en 182).[21]	Standaard, Maatwerk	ISV
Selectiecriteria	**I_SC_1.** Gegadigde beschrijft in een plan van aanpak (max. 2 pagina's) hoe hij zich inzet voor Internationale Sociale Voorwaarden (ISV). In het plan van aanpak worden de volgende aspecten besproken: • De algemene risico's in de productieketen van softwareontwikkeling; • De wijze waarop Gegadigde zich de voorgaande drie jaar heeft ingezet voor een verbetering van ISV, waar mogelijk ondersteund met verifieerbare prestatie informatie; • De plannen die Gegadigde de aankomende drie jaar heeft voor verbetering van ISV, waar mogelijk ondersteund met beleidsdocumentatie en/of certificaten.	Standaard, Maatwerk	ISV
Contractbepalingen	**I_CB_1.** Opdrachtnemer spant zich in om de Internationale Sociale Voorwaarden na te leven conform het onderstaande: • Risicoanalyse productieketen – uiterlijk 3 maanden na definitieve gunning van de opdracht wordt een risicoanalyse door de opdrachtnemer aangeleverd ten aanzien van de keten van het productieproces. • Plan van aanpak mitigeren risico's – uiterlijk 6 maanden na definitieve gunning van de opdracht wordt een plan van aanpak door de opdrachtnemer verstrekt om de risico's die in de risicoanalyse zijn vermeld te mitigeren. • Jaarlijkse rapportage – gedurende de contractperiode rapporteert de opdrachtnemer jaarlijks ten opzichte van de ingangsdatum van de overeenkomst (of in geval van een korter lopende overeenkomst: bij de afronding daarvan) over zijn inzet ten aanzien van het naleven van de ISV. Hieruit moet blijken dat de opdrachtnemer een 'redelijke inspanning' heeft geleverd om de ISV na te leven. • De rapportage moet tevens gelijktijdig openbaar worden gemaakt, al dan niet als onderdeel van een rapportage die een breder deel van de activiteiten van de opdrachtnemer betreft, zoals een (duurzaamheids)jaarverslag. Openbaarmaking kan worden bereikt door plaatsing op de website van de opdrachtnemer. [22]	Standaard, Maatwerk	ISV

Bijlage A. Scope van het rapport

Als vervolg op dit rapport is er een verdiepingsslag nodig op de duurzaamheidsimpact van data-opslag. Hier hoort bijvoorbeeld bij hoe inkopende organisaties bij het inkopen van goederen en diensten kunnen sturen op duurzamer gebruik van datacenters / cloudoplossingen.

Datacenters

Datacenters hebben een groot energie- en materiaal gebruik. In datacenters staan servers, dit zijn computers die gebruikt worden voor rekenkracht en data opslag. Hiervoor is hardware nodig en energie om de rekenkracht te benutten en servers te koelen.

On-premise of cloud

Software kan on-premise zijn, wat betekent dat het geïnstalleerd is op hardware van de eigen organisatie zoals de laptop, smartphone of server van een bedrijf. Software kan ook gehost worden door een derde partij wat dan 'off-premise' heet. Als software op servers draait via het internet heet dit 'cloud computing'. Dan is er sprake van on-demand beschikbaarheid van opslag en computing power. Cloud computing kan over verschillende locaties gaan. Het theoretische voordeel is dat hardware gedeeld wordt, en daarmee optimaler benut kan worden.

Steeds vaker zijn datacenters off-premise en worden ze beheerd door een externe partij. Zo kan een organisatie bijvoorbeeld gebruik maken van software die via de cloud op servers van Amazon runnen. Het gebruik van software via de cloud kan via verschillende verdienmodellen zoals eigendom, as-a-service, of as-a-function.

Als inkopende organisatie is het lastig om grip te krijgen op cloud diensten. Aan de ene kant zorgen de partijen achter de cloud dat er zo efficiënt mogelijk gebruikt gemaakt wordt van de servers, aan de andere kant is er weinig transparantie op de duurzaamheidscondities van de datacenters. Een aantal inkooptools in Hoofdstuk 4 kunnen toegepast worden op off-premise gerunde software.
Hoe inkopende organisaties omgaan met het gebruik van off-premise software en datacenters vraagt een verdiepingsslag die kan voortbouwen op deze publicatie.