DATA-DRIVEN SUSTAINABILITY

THE HEAT IS ON:

SUSTAINABLE IT IS NOT AN ILLUSION!

BY:
LAILA FETTAH
RONALD MEIJER
JAN SCHRAVESANDE
EDWIN VAN DER BURG

1ST EDITION

VIEW ON SUSTAINABLE IT:

THE JOURNEY FROM UNCONSCIOUS INCOMPETENCE TO UNCONSCIOUS COMPETENCE

FOREWORD

May 26, 1971, the moment Watson jr. indicated in a letter to his managers that IBM has a great responsibility towards the environment and that all processes and regulations regarding this must be followed in the smallest details.

With this book we would like to once again show how important and actual this topic is today. This book is about culture and taking responsibility. I dare say that sustainability is woven into IBM's DNA and that the authors of this book; Laila, Edwin, Ronald and Jan do this from a deep sense of responsibility in their domain: IT and IT Architecture. There is still so much to gain, but we see that companies do not dare or don't know how. We had to work on becoming more competent as well. However, the tide will turn and if the time comes that your company is also serious about making IT more sustainable, then this is booklet provides a broad overview.

I wrote in the foreword of the first book in the orange series “Develop a Blueprint for a data-driven Enterprise Architecture” that architects are not persé authors, but this book was received with such enthusiasm that I dare not say so anymore. I hope that this book is read with the same amount of pleasure and that every now and then a smile appears on your face at the witticisms. I hope even more that you will raise your eyebrows from time to time and that the urgency of this subject will move you to action.

A “competition” is organised in the Responsible Impact chapter. Whoever comes up with the best idea will receive a free sustainability workshop from us, resulting in an MVP (Minimum Viable Product) for your sustainable ideas! A great idea. I support this wholeheartedly and personally support this initiative.

Johan Heij
IBM Netherlands

REFLECTION OF THE AUTHORS:

We have used the Responsible Computing[1] framework as a starting point for this book. We have deepened the domains of this framework. As our journey progressed, it was clear that IT sustainability is still in its infancy for most companies. In the course of our journey, the subject has gripped us. In the first instance, sustainability is not about IT, about the CSO[2] or about ESG[3] reporting, but it is first and foremost about ourselves, about you and about me. We must always hold up the sustainability mirror to ourselves.

In an inspiring conversation with Albert Geuchies, who bears responsibility within the government for the sustainability of IT datacenters and IT infrastructure, we received two nice quotes:

“We cannot solve our problems with the same thinking we used when we created them”
Albert Einstein

This is really about a DNA change.

“Everyone knows what he does, some know how he does it, but few know why he does it.”
Simon Sinek

We use a simple model to go from unconsciously incompetent to unconsciously competent. This is paradoxical to Simon Sinek's comment, really something to think about after reading the explanation of this model.

---

[1] a membership consortium that provides a framework for setting responsible corporate policies and practices - https://responsiblecomputing.net/

[2] Corporate Sustainability Officer

[3] Environmental, social, and corporate governance

We hope that more and more people, just like the two Alberts, can turn passion into concrete steps. Laila, Ronald, Jan, Edwin

This book is arranged in a logical way, you don't have to read it from cover to cover, but you can also use it as a reference.

Last but not least, we would like to thank Vicky Bunyard, our Nordics CTO, who provided a review of our ‘Danglish’ as a native speaker. Especially withholding the urge to correct everything she encountered, but instead forced herself to just take out the really stupid stuff we put on paper.

TABLE OF CONTENTS

INTRODUCTION OF THE FRAMEWORK

1. INTRODUCTION

A question about sustainability: why do companies and organisations actually want to be sustainable? Is this because it can cut energy costs, or is it because it polishes the image of the company?
Or .... is this because we recognise that the current way of doing business is not sustainable in the long run? Our current operations are mainly focused on efficiency and (economic) growth: more, faster, better, more beautiful, bigger, and so on. Is this really responsible and sustainable in the long term?

An argument which is often used whenever we get a sustainability strategy explained by a company or organisation. "We want to cut on energy usage" which is a fine statement, but the question is, of course, 'WHY?' If it is only about cost reduction or because of regulation, well..., to be fair we will take it, it is politically correct but in the light of sustainability we believe it is not the right motivation. Real sustainability works only if it is anchored in the DNA of organisations and companies. We need a significant mutation in the current DNA (read culture) of our organisations.
It may be a little heavy for an introduction, but hopefully it provides food for thought for the rest of this book.

The good thing is that we see demand and supply growing in the area of sustainability. We see signs of sustainability all around us. In

our daily life, the call for sustainable behaviour and sustainable products is becoming ever more important, and we are seeing an awareness of the behaviour of both the consumer and the producer. This will only increase during the coming period, which will make it more and more of a lasting nature, especially in the case of businesses for products and services. You can' t afford to be anti-sustainable as a company anymore, and as a consumer you can no longer ignore the supply of sustainable products.

Sustainability is becoming more and more perceptible

An important part of running a business is, of course, technology. Technology can have a big impact on the sustainability of your organisation. Sustainable technology is a slogan that you hear all around you, and many tech company's are advertising to indicate that their technology is contributing to sustainable solutions.

In this book we deal with the question whether sustainable technology really exists, and if so, how can you make sustainability

part of an IT strategy? We have a broad view on sustainability. What we mean, besides the environmental aspects, is also the application of technology and the impact it can have on people. For example, ethics and privacy. Our view is that a business is not sustainable if it is not ethical nor does it consider people's privacy. It may be seductive, but at the long run you will run into a stone wall, by then it is too late.

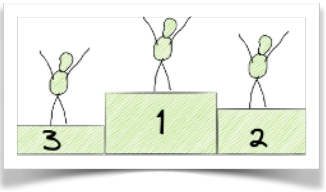
1.1. THE HEAT IS ON

What are we doing nowadays to learn something? Gamification! A game, with the purpose to learn something in a fun way. This is what we call 'serious gaming'. And sustainability is a serious topic. We developed a sustainability game with the name: "The Heat is On". Literally translated, "*The Heating Is On*", with a nod to global warming which we all want to limit to max 1.5 degrees. 'The heat is on' is a song written by Glenn Frey. The song indicates that it is about life and death: ' The pressure is high, just to stay alive ". This is in stark contrast to reality. We still don' t take sustainability serious. The purpose of the book and also the game is to create awareness about it's seriousness and that we should make it our own. It should not just take a place in our heads but sink to the heart: ' Tell me can you feel it ".

The game is aimed at improving the sustainability of your business's IT. You work with your own multidisciplinary sustainability team on a plan. Next, you

The heat is on, on the street
Inside your head, on every beat
And the beat's so loud, deep inside
The pressure's high, just to stay alive
'Cause the heat is on
Oh-wo-ho, oh-wo-ho
Caught up in the action
I've been looking out for you
Oh-wo-ho, oh-wo-ho
(Tell me can you feel it)
(Tell me can you feel it)
(Tell me can you feel it)
The heat is on (3x)
Oh it's on the street, the heat is on

work out this plan and implement it within your organisation.
In this book we describe various IT roles that can make a significant contribution to the sustainability of IT. In the starter kit for this game examples of activities are provided for various roles in order to get promoted to the next level. The promotion to the next level is done by looking at the implementation of the plans and not just the paper version. The role that has the most impact in terms of sustainability wins, but in the end, everyone who achieves an improvement is obviously a winner!



I recently spoke to someone who was confronted with a huge increase in his monthly energy bill. He decided to not look at his energy consumption any more....., because it gave him such a bad feeling. When it comes to sustainability, this probably feels familiar. Perhaps it is better not to be aware, you also don't feel the responsibility to do anything about it. How can you make energy saving a fun activity? For example, you can bring in a game element, make it a sport to use as little energy as possible. It's going to be a search to find out who the energy consumers are. When you look into this topic, there are many possibilities to save. Sometimes even with little effort.

We have come up with a game that consists of four levels: the *first* level of the game is mainly about raising awareness. You get information about opportunities, about the sustainability aspects of IT. Together with your team you will create a plan on how to become more conscious as an organisation. An important first step: the numbers tell the tale. Put the thermometer in the IT environment and see which sustainability parameters are important to record. Think of electricity consumption, but also the temperature in the computer room. In this context it is not about making sure that techies are not suffering from cold temperatures, but more about the amount of heat that systems can tolerate without becoming unstable.

In the *second* level, you are going to look at solutions for the possible improvements that you detected in the measurements of the first level. We are now thinking about opportunities for a more sustainable IT. We become aware of our shortcomings, which is the most painful level. So you want to get through to level three as fast as you can.

In level *three*, you are going to realise the improvements. You guys are going to become competent. Through this learning process, you become more and more capable in the field of sustainability. These include all aspects of IT sustainability, also the ethical and cultural aspects.

And finally, in level *four*, you are thinking about quantifying the improvements. In the end, you want to show the successes to everyone in the organisation. 'Good examples lead to good followers!' It has got to get into the DNA of your organisation.

So join us in this game. We, the writers of this book, act as your game leaders.

1.2. SUSTAINABLE TECHNOLOGY

Technology is an essential part of our existence. We are not going to explain what the last twenty years have brought us in technological innovation, but we are doing a little 'back to the future' to the last century, to the middle of the 20th century to be precise. Technology is being developed rapidly and, in particular, as a service to mankind. Domestic appliances have been coming in large numbers on the market, stereo and television are accessible to all, and in this way, the consumer market is being developed fast.
A cozy advert of the seventies with a pipe- smoking father in his lazy armchair, with the family around him, while he controls the television with a remote control...., most of us probably still remember such a picture. For the newcomers among us (generation Y&Z), this picture is less well known, but there are many other examples that will

appeal to this generation (CD player, Walkman, PC, smartphone, etc.).
This ideal picture tells a great deal about how technology makes a clear impact on our lives and how we have become accustomed to adopt 'convenience serving mankind'. The convenience often comes at a price: our social behaviour (instead of a good conversation in the train, all staring at Whatsapp or pretending to read the newspaper), demanding cheap production (involves waste and CO2 emissions), mass use of disposable products (we do not repair or reuse anymore), looting scarce resources, and so on. This consumer behaviour has a major impact on our world.

Since the introduction of computer technology in our daily existence, the 'technolofication' has increased enormously and there is actually no limit to this pattern. On the other hand, cases where we can use IT as a more sustainable accelerator, we are hesitant. For example, we look at the 150,000,000 blue envelopes sent by the Tax Service per year (in the Netherlands). Since 2014, plans have been made to reduce the number of blue envelopes by communicating with taxpayers through the 'Digital Message Box'. This has been a slow change process and it almost seems like waiting for the last digital illiterate to blow out his or her last breath. At the same time, you must sincerely continue to serve everyone, including digital illiterates. We know, one can't just pull out the paper plug, but it can probably be done smarter, with a lot less envelopes.

This is an example from the Dutch Government, the domain we are currently working in as architects/writers. That does not mean that the examples we give do not apply to other industries.

We are forced to reflect on our behaviour, both as consumers and producers, with the environmental issues all over the world and the energy crisis that we have been experiencing in 2022.
We are convinced that sustainable IT can only mean one thing: '*IT when necessary and not just because we can!'* What do we mean by this?

Take a simple example as digital newsletters, blogs, vlogs, and other e-logs. It's so simple to subscribe to these but you have to be very savvy to stop these kind of subscriptions. After twenty times clicking, you don't know what to click on anymore and with a sigh you give up: *"ah, never mind....."*
Under the hood, this is an example of *'IT just because we can'.* A simple way to bring an advert, an offer or whatever directly to a person but also to put the responsibility for subscribing (or not) directly with the recipient (true consent to recurrence instead of withholding a service or product if you do not). There is no such thing as a digital sticker on the mailbox with:
NO, unaddressed advertisements, NO, no house to house magazines. The fact that you don't read the blogs, vlogs or any other e-logs doesn't change anything about it.

And now I hear you think, "it's just a little message!" That is precisely the problem. This way of thinking is anchored in our culture. Grandpa saw a dime (that's a 10-cent coin) on the street. He picked it up, carefully put it away in his 'purse' (that's what we used to call them) and said, "that's the start of a million." Start with something small and achieve great things.

We see governmental IT organisations behave as if they were commercial companies. By using IT as a competitive playing field we see some government agencies competing instead of working together to the max. The phrase "the citizen central" is often used as an excuse to justify a certain IT project while in many cases it is just about technological innovation. In addition, it is sometimes also an excuse to introduce the latest technical gadget. Consider, for

example, a mobile app built to do a benefit request. An other example of *'IT just because we can'*..
The 'citizen central' would mean that the offer is developed in co-creation and based on of wishes/requirements from the citizen. That would be a nice thought if it were not for our laws to determine how a citizen should behave and how to conform to the law. Although it should be a hassle- free and transparent process.

A vision where Governments are fully committed to create a superior information position for the citizen, and create their (IT) services based on this information position provides a welcoming perspective (and therefore is a sustainable perspective).
This perspective includes sustainability aspects such as transparency, prevention of errors, mitigation of physical interaction with government, and improvement of the privacy of citizens.

Of course, we cannot change organisations and companies by writing a book. On the basis of a number of sustainability principles, we can try to help with simple interventions to make small steps that will help to use our IT resources in a responsible manner. For this we have used the framework of 'Responsible Computing' (RC). Through the different domains of RC we make a deep dive into the world of sustainability.

Writing about this topic in the light of IT solutions is quite tricky. We have done our best to get our thinking process in motion. There will be many more bits flowing through the cat 6 cable before each company has adopted sustainability as a guiding principle!
It is a fact we're not experts in the broad field of sustainability but we are in the field of IT!
We must get more familiar with this matter because customers demand it from us, and sometimes we even developed an opinion on this subject. We hope that this point of view will help to address possible approaches and in the light of digital transformation "sustainable by design" will be adopted as a super-principle.

1.3. AND FINALLY NUMBERS!

What do numbers say? Through this book, we use numbers regularly. When it comes to sustainability, all kinds of articles are published, and in those articles you find numbers within a certain context. The company "sustainable_to_the_bone" will let the numbers shine while the company "polluter_for_better_or_worse" will give a rather rosy perspective and will disguise their pollution using 'compensation methods'. Like for example investing in a forrest as a compensation for dumping waste water in the river. The numbers we give are mainly intended as an indication. In your organisation, in your area, it will be more clear because you work within a context which is familiar to you. You are most likely not able to take away the context you are working in, but others that are acting on a more strategic level may be able to change this context by creating a sustainable strategy.

We should all be getting a better understanding of the magnitude of the problem, but also we would like you to get a sense of the potential of possible improvements. Don't get seduced by numerical details, but pick up the essentials and get on with it. The numbers we use are sometimes factual and sometimes interpretations. Don't forget, this is a point of view where we 'sometimes' vent our own opinion.

2. INTRODUCTION TO RESPONSIBLE COMPUTING

2.1. RESPONSIBLE COMPUTING IS SUSTAINABLE!

In this edition of the orange book we concentrate on sustainable IT. Not because it's a trend, or it would be a better "sell" (even though this book is free of charge), but because it is necessary.

We have used the framework of Responsible Computing as a starting point. This framework helps us to put the right focus on the dimensions of people, processes and technology in the context of IT. Although an organisation should be able to support the United Nations Sustainable Development Goals (SDGs), we find this scope actually too big and too complex and therefore we limited ourselves to the scope of IT where we have the expertise, which also helps to confine ourselves to around 200 pages (is also sustainable).

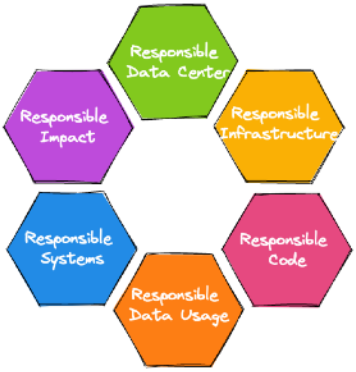
Responsible Computing is an initiative of a number of companies including IBM. This framework has been donated to the Object Management Group (OMG)[4].

Responsible Computing consist of 6 domains:

1. Responsible Datacenter
2. Responsible Infrastructure
3. Responsible Code
4. Responsible Data Usage
5. Responsible Systems
6. Responsible Impact

[4] https://responsiblecomputing.net/

All of these domains are important, because many organisations have become IT-intensive organisations. We discuss how these organisations can take responsibility in each of those domains and use resources in a responsible manner.
The first three domains are technical. In these you will see that the focus is on the optimal use of scarce resources such as electricity and water. In the other three areas, the definition of sustainability is described in a much broader perspective. It describes how you use your IT ethically and responsibly and therefore in a sustainable way. This chapter provides a brief overview of the six domains which subsequently will be discussed in more detail.



Responsible Computing Domains

2.2. RESPONSIBLE DATACENTER

When we talk about the datacenter, we talk about the physical building and the facilities you need to house a datacenter. Think of power, cooling, emergency power supply, cabling, and so on. Datacenter vendors measure their sustainability by Power Usage Effectiveness (PUE) and Water Usage Effectiveness (WUE). PUE is the ratio of the power used by the computer equipment and the entire datacenter. WUE is the amount of water divided by the amount of power used by the computer equipment. Measuring the efficiency of a datacenter is not very complicated, and most of the big datacenters do this. What usually doesn't happen is the calculation of the impact of building a datacenter. Transport of building materials and the materials themselves (mostly concrete) produce a lot of CO2.

2.3. RESPONSIBLE INFRASTRUCTURE

IT infrastructure consists of computers, storage media and network equipment. It is the applications that set the requirements for this infrastructure and ultimately determine how large, how heavy and how fast the infrastructure must be.
In 1969, we sent the Apollo 11 and crew to the moon and ensured a safe return journey. All with the help of a computer that took care of calculating the correct course and adjusting the course when necessary. This computer consisted of a processor with a clock speed of 12 microseconds and 72 Kb of memory. This system is comparable to the power of a Commodore 64.
An average smartphone today has more computer power than all of NASA could put together in 1969. Yet, we still have too little computer power!

2.4. RESPONSIBLE CODE

To run applications efficiently, efficient infrastructure is required, which in turn runs in an efficient datacenter. But the code written to make the application work has an even greater impact. With the efficient building of a datacenter we can save 10-20%, with more efficient infrastructure we can save 60-90%, but with the right code we can sometimes save up to a factor of 2. Or should we reverse it? If you program incorrectly, it can cost 2x as much computer power. Right or left, Responsible Code is an important domain that should indicate whether code is sustainable or not.

Within that domain, AI and Big Data are major power consumers. Commonly used programming languages such as Python and R are typical examples of "expensive" languages (as far as CPU usage is considered). Enough to sink your teeth in there. How do we now ensure sustainable code? Sustainable code, also called “green coding” reminded some of the writers of the old mainframe terminals:-).

Green Screen

2.5. RESPONSIBLE DATA USAGE

In a data-driven project, we make sure that the data is safe. We respect data from the customer, citizen, or other companies data. We clearly describe how we use it, and we use it only for the purpose for which permissions are granted.

We have good data governance set up for data and data usage; we know exactly where to find specific data and the meaning of this data; we also know where the data came from, how the data is composed and who has made changes to the data; the data is stored in a sustainable manner.

2.6. RESPONSIBLE SYSTEMS

A Responsible System is a system that contributes to a 'correct' application of the system. This concerns both people as well as the environment affected by the system.

Furthermore it is important when designing the system to consider non-functional requirements which are a prerequisite for a true sustainable implementation of the system.

2.7. RESPONSIBLE IMPACT

Imagine: we build systems with the best intentions for society. We make the world a little better with our systems. For example, we can build a 'house searching system' that combines a list of available houses with specific characteristics with a list of people looking for a home with such characteristics. Or we build a system that detects aggression on the street and we ensure that the police will be immediately on site to prevent escalation.

How will the system you design and build make a difference for society? Which of the United Nations SDG's (UNSDG) are addressed by your project?

2.8. RESPONSIBLE COMPUTING VALUES

In addition to these six domains, the framework has six principles. These principles are more values rather than architecture principles as we use them within the IT Profession. Since we define a number of architecture principles in this book, we will now name these six Responsible Computing values. These values can be considered as quality requirements for the system and can be interpreted as cultural values. Openness, authenticity and accountability have to be cultural values for a sustainable organisation. The following values are defined by this framework:

1. **Sustainability**. Sustainability in the framework of the UNSDG has a very broad definition. In the case of sustainability, we mainly focus on environmental aspects, with an emphasis on energy usage and CO2 emissions.
2. **Inclusiveness**. It's about making sure systems can be used by everyone and are available for everyone. Inclusivity also means consideration for the minority groups, the individual.
3. **Circularity.** Within IT, circularity is about reusability and modularity. Consider, for example, the reuse of precious metals that are in the electronics of the computer, such as gold. These materials are known as 3TG (Tantalum, Tin, Tungsten and Gold). Tantalum is a strong and flexible metal with a high melting temperature and was also used for light bulbs. Now it is used for capacitors, among other things, which are important components in the electronics industry.
4. **Openness**. Openness, authenticity and accountability go beyond the conditions that we set to our systems. These are about the behaviour of an organisation that is made up of people. Openness, transparency, shows where you stand with regards to responsible computer usage. It 's a vulnerable attitude that not only shows what's going well, but also where problems emerge and how you solved those problems. This way you can create a self learning organisation and be open to advise from others.

5. **Authentic.** Do you really want it? Are you serious? Is Responsible Computing more than just an obligation arising from reporting directives or from a (rational) realisation that we need to do something about global warming and the running out of scarce resources?
6. **Accountable.** It is about becoming the owner of the problem and accepting full responsibility for it. Later in this book we describe the different roles within IT, a central theme for these roles is accountability. Are you prepared to be liable for the domain that you are responsible for? If decisions are to be made at the expense of the sustainability principles and profit margins are more important, will you stand up and speak up?

2.9. LIABILITY (ACCOUNTABILITY) APPLIES TO EACH DOMAIN

When we map the values to the domains, we see that not all of the values are equally relevant for all domains. Below is an overview of the relevance of the values related to the domains. Regardless of the domain, "accountability" is always relevant. In the chapters about the roles such as the IT architect, the IT specialist, the product owner and the CSO, we highlight the relevance of accountability for those roles.

Domain Values	Data-center	Infra-structure	Code	Data Usage	Systems	Impact
Sustainability irt energy	V	V	V	V	V	V
Inclusiveness			V	V	V	V
Circularity	V	V	V			V
Openness			V	V	V	V
Authentic				V	V	V
Accountable	V	V	V	V	V	V

Sustainability, inclusiveness and circularity are relatively easy to make measurable and can be used to compare the Responsible Computing domains between different companies.
Openness, authenticity and accountability are less easy to measure, as these are about the 'sustainability DNA' of the company.

When we're talking about DNA, we're talking about humans. That's right, because the DNA is created by humans through principles and ethical standards! In other words, it is human work.

The principles (business and IT) used in architecture, are sets of principles that form the boundaries for Enterprise Architecture. If contradictions arise in the handling of these principles, **'super principles'** apply. These transcend all other principles. Just like **'Secure by Design'**, you should also implement the **'Sustainable by Design'** super-principle.

3. THE JOURNEY TO SUSTAINABILITY

3.1. FROM UNCONSCIOUS INCOMPETENCE TO UNCONSCIOUS COMPETENCE

We talked about the DNA (the culture) of an organisation. What we mean by this are behavioural patterns, the identity of an organisation. If you walk across the street and you eat a Protein-bar, you automatically put the wrapping back in your pocket in order to throw it into a bin on a later moment. You don't think about it. You don't wonder why you put it in your pocket because it is within your system. A nice example is that of Japanese people: when they leave a cinema or a stadium they take their rubbish with them and everything looks neat again. These patterns are 'under the skin' and don't cost energy. It's obvious. You are: unconscious competent.

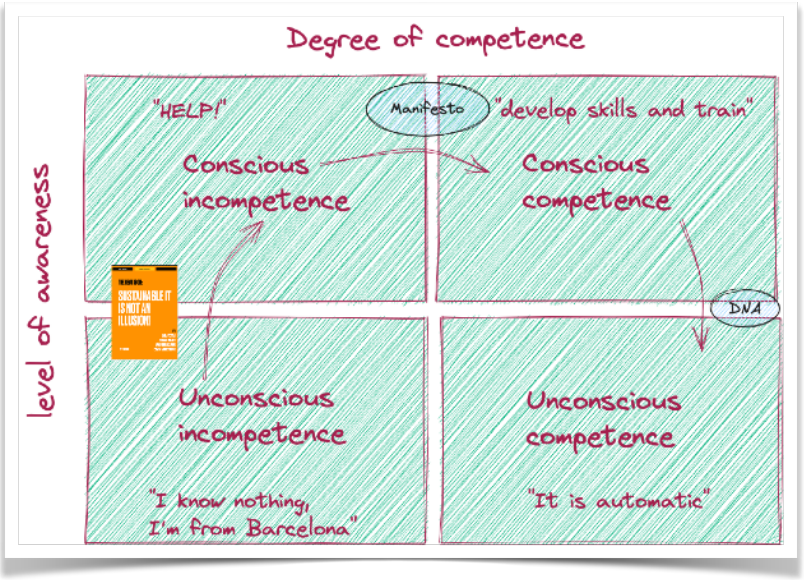
When we talk about unconscious incompetence, it's about things we don't know or don't actually realise enough. With this book we aim to make you aware of the impact of IT on sustainability and the impact of sustainability on IT. That might hurt a little bit, because sometimes it's wonderful to be oblivious. But we won't stop there because we also want to give you tools (principles) to do something, to develop and apply your IT in a sustainable way. And we hope, of course, that it becomes natural. Something you don’t think about anymore, but something that makes perfect sense, as part of your DNA.

A few examples:

Unconscious incompetence: speeding, don’t we all sometimes, but what is the impact on the environment? We are almost certain that you don't think about that at the moment you push the pedal to the metal to get to your appointment in time. Did you know that the amount of fuel you use extra is proportional to the extra amount of CO2 emissions?

The cloud also provides a temptation to use IT resources without limitation. It's cheap, it's available so... Sustainability is the problem of the cloud provider, right?

Conscious incompetence: the bill from the cloud provider landed on the doormat. How can it be so high? Should we as a company create guidelines how to deal with IT resources? The company understands that it is necessary to raise awareness to encourage better use of resources.



Degree of competence

Conscious competence: designers and developers are being driven by sustainability principles, so no excessive use of IT resources anymore. IT resources are used consciously with a principle based on scarceness of resources. Quality assurance is implemented in processes to ensure the sustainable and ethical use of IT resources.

Unconscious competence: everyone in the business unwittingly keeps to sustainability principles because it is normal, it became our standard way of working. One is intrinsically motivated to adhere to these principles. It's in the DNA of the company!

3.2. PROMOTE

As an example, we have set the conditions for promotion through the competence model.

- From unconscious incompetence to conscious competence: read this book.
- Conscious incompetence to conscious competence: implement a behavioural manifesto and principles. Examples of this are further detailed in this book.
- From conscious competence to unconscious competence: undergo a culture change. The moment you can address each other within the organisation about sustainable choices, a self-correcting ability arises. Because we do know what good choices are, but we like convenience and we find it tricky as it costs too much energy to change our behaviour.

PEOPLE

There are many roles in IT organisations, but for this book we have specifically highlighted a few. These are roles that may have a major impact on the sustainability dossier. In this chapter we therefore discuss the role of architects, the specialists, the product owner/ product manager, the strategic role of the Chief Sustainability Officer (CSO) and the employee. We have split the IT specialist into three roles: the programmer, the data scientist and the infrastructure specialist.

The 'role table' is just a model that shows where the focus lies for the different roles. It is intended to start the discussion regarding the accountability centers of gravity for these different roles. This can help in defining employee profiles and making them aware of the importance of sustainability principles.

These different roles largely determine the implementation and realisation of the sustainability aspects related to openness, authenticity and accountability. They must therefore have to incorporate it into their DNA in order to spread it like an oil slick in the organisation. (*Too bad the word 'openness' doesn't start with an 'a' then we could have made it 3A. The 3As of the DNA, nice title)*. The employees are described as a target group in this book, but are not listed in the table. They are the users of the systems. Are these happy users or is extensive automation taking away their job satisfaction? What responsibility do companies have towards their employees (users) when developing IT systems? Is job satisfaction a subject that is being looked at or is efficiency the only aspect being considered?

Domains Roles	Data-center	Infra-structure	Code	Data Usage	Systems	Impact
Architect: Business						
Architect: Application						
Architect: Infrastructure						
IT Specialist: Programmer						
IT Specialist: Data Scientist						
IT Specialist: Infrastructure						
Product Owner						
CSO						

4. THE IT ARCHITECT: DESIGNER OF SUSTAINABILITY

This book is written by IT architects. Those who find us 'suspicious' will think we 'preach for the choir', also because we discuss this role as the first role in this chapter. On the other hand by starting with this role you might also consider that we want to accept our accountability in discussing the design of sustainable IT systems.

It's just how you look at it, but we find that architects have a key position when it comes to addressing and designing sustainability as a quality aspect of IT systems.

For those who are not an IT architect and for those architects who are struggling to explain their role, what is the role of an IT architect? In this role, you are responsible for ensuring that an IT system complies with the specifications of the client, taking into account "all" requirements of other stakeholders.

What do we mean by this? A client wants to build a system that must meet a number of functional specifications. Furthermore, the client (hopefully) will also have a number of non-functional requirements. Consider the availability of the system or the speed. In addition there are other stakeholders, for example the management organisation, security and a data privacy officer. An architect is a person who overlooks the complete picture and seeks the balance between the demands of the different stakeholders. Those demands can be contradictory. For example, the system should be easy to operate, but also safe. An architect writes up the possible choices and takes a decision with a motivation (called architecture decision) and has this endorsed by an architecture board.

A contradiction could also be long-term goals from a corporate strategy versus the short-term goals of a project. Think of a business strategy of wanting to be CO2 neutral by 2025 versus a project that looks for the most economical replacement of lease cars. An architect has a focus on those long-term goals and is therefore sometimes experienced as a 'pain in the ass' if something needs to be built as quickly as possible. Unfortunately, there are a lot of situations where companies would have liked to have listened better to the architect with that long-term objective.

The tool used to secure business objectives are the architecture principles. These principles are a derivative from the business strategy and ensure that IT systems that are realised will be in line with a company's long-term objectives but also with respect to technological developments.

You probably see it coming now: sustainability is part of the business strategy or should be, but we observe that at the project level there are none or hardly any demands for the deliverables in the area of sustainability.
So, architect, 'be prepared!', and realise sustainability principles will result in even more requirements that the system needs to conform to.

If we link the above tasks to the six values mentioned in the Responsible Computing framework then it demands quite some of the attitude, the mindset, and the involvement of the architect. The architect must take into account inclusiveness, circularity, openness and in addition he or she should be authentic and accept liability. With these, his or her role as a trusted advisor has become more important than ever.

In addition, designing a sustainable system requires a broad knowledge about the operation of software, hardware, networks, and datacenters to ensure that the overall solution is sustainable. Also the system must comply to laws and regulations in order to create a system that is ethical.

The aim of this chapter is to make sustainability tangible for the architect.

4.1. SUSTAINABILITY AS NON-FUNCTIONAL REQUIREMENT

There are a number of standards when it comes to mapping the quality requirements of an IT system. ISO9126 is one of the more familiar ones, but we do not see it often, probably because it is less accessible.
IEEE 830-1993 also provides a list of non-functional requirements. However, your company might be using another standard. For example, within the Dutch government we see the use of NORA (Dutch Government Reference Architecture) with its own conceptual framework and definition of quality requirements. Within these standards, various categories of implicit sustainability principles are supported.

For example, ISO9126 has usability as a quality aspect. Usability should also include inclusivity.

As for Nora, "Sustainable" is one of the goals:

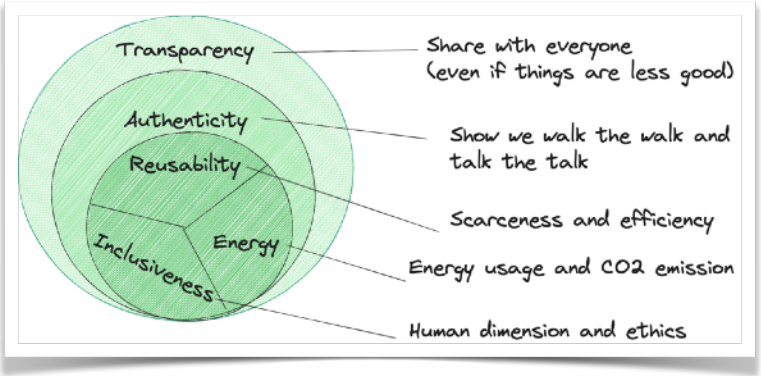
“Government services have as little impact as possible on the environment and our living environment. This leads to lower emissions, less consumption of raw materials and energy and a positive contribution to the living environment."[5]

We have chosen to identify the qualitative sustainability aspects separately from a framework. Our suggestion is to take the aspects we mention and place them within the framework that your organisation uses. We see that these quality requirements are derived from the (principles) values that in our view every company should have. Furthermore, we do not have the illusion of being complete with this list, but almost.

---

[5] https://www.noraonline.nl/wiki/Duurzaam_(Doel)

4.2. TRANSPARENCY



Five quality aspects in the context of sustainability

In the interests of transparency, we have to decide what information we publish in order to ensure that our stakeholders have a fair view of the sustainability of our delivered system. We know the expression: 'an honest man has nothing to hide'. In transparency, it's also about being transparent in giving all the evidence to stakeholders so they can assess that you actually delivered what you promised (or not). Transparency is in a way a fragile attitude. You are open and also ‘teachable’ (eager to learn). Why didn't it work out? What is the impact? What can we learn from it? What do we need to different next time?

This is not just about the result, but also about the way in which the result was established. Imagine building an energy-efficient football stadium and you achieved all the goals in this project. Have you really been transparent? What is the impact on the stadium's environment while building and constructing the place?

When we design a system, we must handle transparency as a requirement and not try to add it afterwards. A transparent system delivers information to the outside world. This can be for example monitoring information about energy usage or the datasets and

algorithms that have been applied to the system. Transparency also applies to the areas where you have not yet met your goals but that you want to achieve in the future. It also applies to the areas that you do not want or can be transparent about.

This could lead to the following set of requirements:

- *Clarity about the intent of an application* and preferably without ulterior motives. If you search for shoes on Google and you get ads about shoes, searching the Internet is not Google's (only) goal. The purpose of GDPR is to protect users' data and only use it for what it is intended for. You give "consent" to the usage. Unfortunately, we see many applications where you first have to give consent before you can access information or sites where you have to leave personal data to retrieve a document so that they can spam you later with various advertisements.

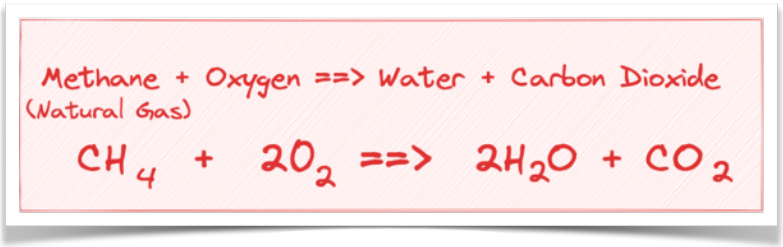
- *Measuring is knowing!* Accommodation of monitoring data related to sustainability. There is often a lot of system information to retrieve about the use of hardware and software.

- Another good example is *collecting lead times from your toolchain*. There is a relationship between the duration of your compilation and testing and the efficiency of your application.

- *Ethical use of AI*. As described in our previous orange book, you can use AI to make decisions that can have consequences for people's lives. You will have to account for the how and why of the choices. More about this in the Responsible Systems chapter.

4.3. ENERGY CONSUMPTION

The beauty of IT is that it runs on electricity and therefore does not need fossil fuels in operation... Yet electricity is largely proportional to CO2 emissions.



Most of the power plants in the Netherlands run on gas. Now you can run your datacenter on green energy, but ultimately a bit more than 10% of the generated energy is green and the rest is grey. That 10% is scarce and eventually we all have to share it. So clockwise or counterclockwise, energy is a precious thing that you have to use as if it was scarce. We can assign the following requirements to this:

- the efficiency of hardware
- production of hardware
- datacenter efficiency
- code efficiency

Finally, we would like to mention scalability here. When it comes to scalability as a non-functional requirement, we mean that with a fluctuating number of users or requests, the system can handle this by scaling horizontally or vertically. When it comes to energy consumption, we mean scaling down and shutting down systems that are used less or no longer.

4.4. REUSABILITY

The cyclical economy, perhaps better known as the Cradle 2 Cradle (C2C) concept[6] describes the way to develop a product that can be reused in another field of application. Imagine that a computer system has such a modular structure that the cabinet, power supply and cabling can be reused. That the motherboard can also handle the next generations of CPUs and that the old datacenter CPUs are very suitable for use in the automotive industry (which is currently dealing with a chip shortage). The tricky thing for us as architects is that we don't design that computer, because that's where it starts. The computer itself must be designed with reusability in mind.

Nevertheless, when selecting hardware we can at least look at its reusability and there are indeed suppliers who do not replace the entire computer, but only the processors.

- choose infrastructure that can be reused.
- purchase only from suppliers who recycle 'old' equipment responsibly.
- use modular code. Code writing and producing also costs energy. Code that is reused does not need to be built anymore. The saving, for example, could be used to optimise the code.

4.5. INCLUSIVENESS

We mentioned it at the beginning of the chapter. We have to take into account the users of the systems we design for as architects. We have to take into account people who have a physical disability, such as people with hearing impairment or visually impaired people. Too often we assume that the user is comfortable with a computer and applications.

[6] Cradle to Cradle: Remaking the Way We Make Things, Michael Braungart en William McDonough

Within the government we are dealing with a contradiction: people who are entitled and make use of various allowances can apply for these via Internet sites. But the people who need these allowances are often people who find it difficult to deal with these modern resources.

Inclusiveness also plays an important role in the implementation of automated processes in which decisions are made based on business rules or AI models. How do we deal with the exceptional situation that falls just short of the rule? Do we have enough data to train our model? The government makes laws to treat everyone equally. Those laws are often very good for the average, but nobody is average, not even the people we mentioned in the example who are entitled to benefits.

4.6. AUTHENTICITY

When we make a business case to justify a project, we generally see two things happen. First, the business case is rosier than reality. We think that the new product we put on the market will be sold 100,000 times. Usually it is less, sometimes more. We think the cost to make the product is 1 Mio. Usually it is more, never less. The second thing we see is that a business case is not or hardly used afterwards to check whether our calculation was correct. It's no use crying over spilled milk. But can we learn from it?

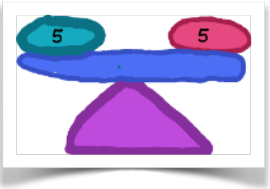
When it comes to authenticity as a non-functional requirement, it is about demonstrating that we live up to what we have promised. A definition for authenticity in this context is to demonstrate holistically by measurement. Authenticity is a concept that is used precisely in the context of sustainability because it reflects a cultural aspect. We must think in advance about the measurability of sustainability in order to demonstrate it afterwards.

For example, we have ve made use of an AI model, and we need to be able to show that we have used the right data, which algorithm is applied, and how that algorithm comes to its results.

Every time we go into production with a new release, the application uses 5% less power. That means, for example, that with Agile Development, in the final sprint, an analysis and optimisation is done on power consumption.
In the lifecycle management of our storage, the energy consumption of the new acquired storage needs to be 15% lower.

It may be, in spite of setting measurable criteria, that it is not possible to meet the criteria. With authenticity, it's about getting honest reports about these goals and not making it prettier than it really is. Authenticity and transparency have a lot in common. Where transparency indicates what information needs to be shared, authenticity addresses the correctness of the information.

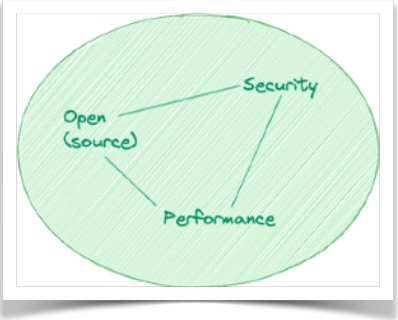
4.7. PRIORITISE IN ORDER TO BALANCE



As an architect, you have to make choices. The non-functional requirements matrix is a tool in which the requirements are prioritised. You do this together with all your stakeholders. You indicate with a factor between 0 and 10 how important a requirement is. The prioritisation helps with making choices. Interesting to see how high sustainability scores! But, lets nog get ahead of ourselves, lets get it on the list for once and for all.

It is not always the case that the non-functional requirements are opposites of each other. Precisely in the context of sustainability, we see three non-functional requirements
that reinforce each other:

- open source
- security
- performance



Open (source) systems are modular, and the result is that the modules are efficient. They do what they have to do. So the minimal code ensures maximum

performance and therefore increased sustainability. Minimum systems that only do what they have to do are therefore also less vulnerable to 'tampering'. An example of this is container technology. Docker is a well-known container technology environment that provides both application "build" as well as running the applications ("run"). By separating the "build" and "run" environments it will give you two smaller environments with one specific task that is more efficient and safer.

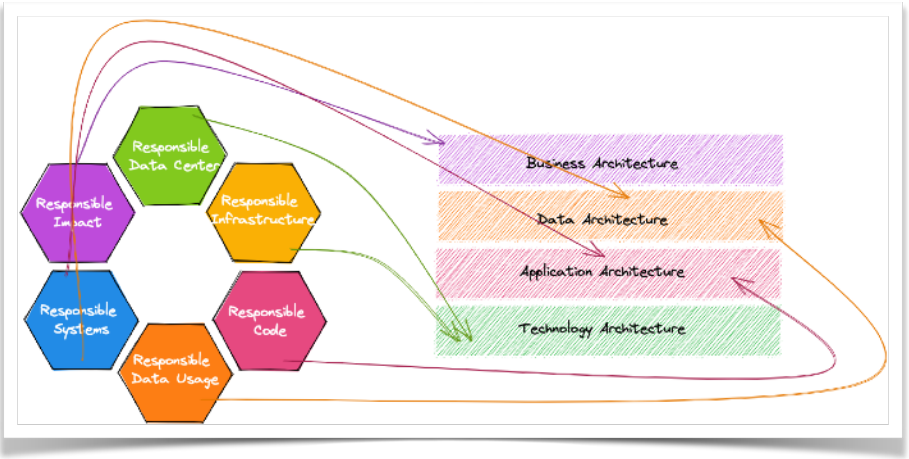
Red Hat has chosen not to use Docker but CRI-O as container technology within OpenShift. With Docker you build and run your applications. CRI-O can only run. Thus, the runtime is more efficient, but also less fragile, because build commands do not exist: no messing possible (tampering).

4.8. ARCHITECTURE FRAMEWORKS AND THE DOMAINS

Six domains have been defined in the Responsible Computing framework. In the TOGAF framework there are four domains defined:
1. Business Architecture
2. Data Architecture
3. Application Architecture
4. Technology Architecture

The six Responsible Computing domains are well-related to the TOGAF architecture domains. This, on the one hand, can help to demarcate sustainability responsibilities within your architecture role. On the other hand, it can help to better position the six Responsible Computing domains. The Enterprise Architect is responsible for the consistency between the domains.

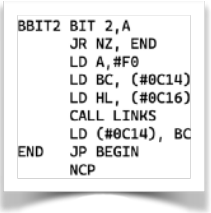
We see the mapping as follows:



TOGAF domains x Responsible computing domains

5. THE IT SPECIALIST: SUSTAINABILITY IN PROGRESS

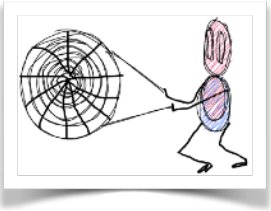
A colleague of us once said that every IT specialist should be able to write assembler in order to penetrate into the heart of the computer, really understand it and make optimal use of all resources.



Where the architect is responsible for ensuring that sustainability requirements are explicitly defined for a system, the specialist is responsible for actually creating the defined solution. It goes further: the specialist is not only responsible for realising these specifications, he or she is also responsible *for the way* the solution is realised.

It can be compared with building a house. The architect indicates which materials and building blocks are used to help the property to be optimally insulated and to make it as energy efficient as possible. They make detailed drawings of, for example, cold bridges, which state how they can be prevented or brought back to a minimum. A cold bridge, also known as thermal bridge, is created by a break in the insulation layer of the building. But when it comes to this detail, the contractor's practical experience is indispensable. Can the vision of the architect actually be realised? The contractor is the builder and possesses (powerful) tools.

Specialists, like administrators, have a lot of power in their field. In the movie Spiderman, Uncle Ben tells Peter Parker (Spiderman):



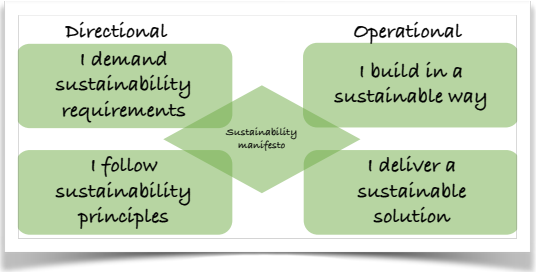
"*With great power comes great responsibility*".

There is sometimes a thin line between abuse

or proper use of that dominant position. Yet we find, when a company builds unethical systems the specialist should not act as if they were ostriches! In the event that a company is wasting scarce resources, the specialist MUST accept his or her accountability (read liability) and stand up!

5.1. MANIFESTO FOR THE SUSTAINABLE SPECIALIST

So we see that the responsibilities of the specialist are expanding. In addition to the existing responsibilities, the specialist is responsible for building a sustainable system and also for building the system sustainable. We have therefore drawn up a behavioural manifesto that the sustainable specialist should embrace:



Manifesto for the sustainable IT specialist

I demand sustainability requirements. A specialist wants to know how to build as sustainable as possible. The solution must be ethically responsible and efficient with scarce resources.
I'm following sustainability principles. The architect together with the business defines sustainability principles that support the objectives of the company in the context of the organisational culture. The specialist will follow these principles, even if pressure from the operation increases in order to cut roads and is asked not to use principles too tightly but to regard these more as guidelines. In the spirit of the law, but ...
I'm building in a sustainable way. Applies sustainability while building. Compare, for example, the construction of a datacenter. These are mostly made out of concrete and aesthetically wise not something to be very proud of. There are, however, alternatives to concrete, and why should you not pay attention to the outside of a building, for example, to prevent the pollution of the horizon? The

way in which an infrastructure or an application is being built is now being done in a sustainable way by specialists.
I deliver a sustainable system. Actively managing sustainability requirements from the business to a successful implementation of the system. Specialists also know what the ultimate intent is of the system. "I know nothing, I am from Barcelona"[7] doesn't work any more. They are accountable.

In a sustainable world, the responsibility for the specialist therefore expands. To make sustainability choices, you need more knowledge than just programming, building models or realising infrastructure. Where the specialist used to be the user of an environment, he now becomes a co-owner. Let's give a simple example to clarify this.
You have the option of leasing a car or purchasing it yourself. If we look at the difference between owner and user in a generalised way, we see the following behaviour:

Owner of the car (private)	User of the car (lease)
Driving efficiently (the eco mode and most economical route)	Always in sports mode. Drives a little faster than allowed. Time is money
Use where necessary	Use where possible
The bike is an option	Doesn't have a bike
Fuelling costs money	Refuelling takes time but does result in loyalty points
Maintenance prolongs the service life	Maintenance is mandatory by the leasing company
Mileage shortens the life span	Mileage provides an earlier prospect for a new car
Avoid any damage	Avoids more than one damage per year
The car must be efficient and drive economical	The car must be nice
Know how the car works	Know how to drive the car

[7] Quote from the English sitcom Fawlty Towers

In hot weather: AWCO (All Windows Can Open)	Airco is always on
Washes the car regularly and keeps the tyres under the right pressure	Washes the lease car when handing in the car after the lease period (at least if you don't have kids)

While this, of course, is somewhat exaggerated, there is a clear message in it; the owner feels much more involved with the car itself, the owner wants to understand what goes on under the bonnet to make the right decisions. The sustainable specialist understands the impact of the code on the processor or the impact of the choice between a router and a switch.

The Responsible Computing values are about authenticity and accountability. That is exactly what this is about. You may expect the modern specialist to be prepared to accept liability (ownership) for its code, infrastructure and models, and to take the sustainability principles to heart.

The current specialist training courses are often arranged in the same way the lease car user is using his car. Training must pay attention to the consequences of the choices made by specialists, for example with regards to the use of the hardware.

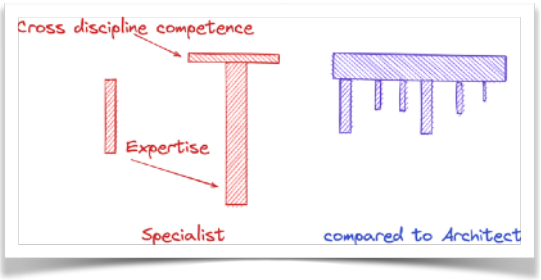
5.2. BROADER OR DEEPER?

These days, specialists who just finished school have a broad knowledge of different programming languages and computer systems. In principle, they can develop applications in any language on any system and this wide experience ensures that they can also easily switch from environment A to environment B. Over time, we see that they become more specialised in one or maybe more environments. This has the advantage that they are able to make maximum use of resources within that environment. The big downside is that every new application will be developed in that one environment. We call that the golden hammer anti-pattern. Every application is a nail for that specialist.

When it comes to sustainability, the specialist needs to broaden its scope and thus become somewhat less specialised. We are seeing a trend in this behaviour: for example SRE teams, hyper converged systems etc. Yet the value of the specialist is in his specialism. It is good to have a general practitioner, but it is also good when there are specialists in the hospital who have a lot of knowledge, about for example, the heart, or any other body part that requires specialised knowledge; a T-Shape profile.

The picture shows the profile of the sustainable specialist. They need to develop both in depth and as well in width. Compared to the architect, who has that broad knowledge (sustainability has been added to this).

We are now going into a number of specific specialist roles. We have chosen the most common roles. Of course feel free to add specialist roles to this list which are relevant to your organisation.



T-shape

5.3. IT SPECIALIST: THE PROGRAMMER

The programmer is the IT specialist who is trained for developing software. The chapter on Responsible Code provides the necessary information on sustainable coding (green coding) that is highly relevant to the programmer. Let's project the behavioural patterns for the sustainable IT specialist on the programmer.

I demand sustainability requirements.
The programmer develops the software based on requirements. These requirements are often reflected in the form of use cases, business scenarios, and non-functional requirements. With the

introduction of DevOps, the idea has become obsolete that a programmer creates a program in splendid isolation and then subsequently 'throws it over the fence' to be picked up by operations. After the adoption of DevOps, security was added to this process and it became DevSecOps. (Secure by design and not added later).

SusDevSecOps is the practice we propose to make the programmer also responsible for the development and management of software in a sustainable way. From now on, the programmer will no longer accept a request in which sustainability requirements are not clearly stated.

I follow sustainability principles
The architect has established principles for sustainability. The idea of these principles is that they support the business strategy in a sustainable way. Urgency, pressure from the business, which is often caused by pressure from the market, regularly jeopardises these principles. The specialist does not succumb to pressure, but must stick to these principles.

I build in a sustainable way
In the framework of SusDevSecOps, the programmer develops in a sustainable manner and thus thinks about its toolchain with tools and whether this is a sustainable approach. Tools include programming languages, compilers, patterns, and frameworks, but also test scripts that validate code.

I deliver a sustainable system
SusDevSecOps also requires the operation to be sustainable. The application must comply with the ethical standards, energy consumption and CO2 emissions which are defined, and must align with the UNSDGs. This might be asking a lot from your specialists, but if you know that part of the application is being built in China, by programmers working in appalling conditions, then this is something you do have to take into consideration.

Do you need to be a broad skilled programmer or a specialist? If you ask the Java programmer to build a solution then you know one thing for sure: the solution is (efficiently) developed in Java. Wouldn't it be a nice thought if the programmer would be able to select the best fit programming language and therefore develop an even more efficient solution!

In the chapter on Responsible Code, we show a number of operating models for coding. An embedded programmer is likely to benefit from an in-depth knowledge of, for example, Lua or C. The same goes for a programmer in the gaming industry or film industry using 3D tools like Blenders. If we look at our primary target group of general programming and AI, we will probably benefit most from developers who are widely employable. In that chapter we also discuss different code patterns. These patterns are based on fundamental principles such as queueing theory, sorting algorithms, caching. It is important that the programmer is aware of these patterns.

5.4. IT SPECIALIST: THE DATA SCIENTIST

In the context of this chapter, we limit the role of the data scientist to that of a model builder. In practice, the data scientists often also has a programming role. If we look at the behavioural manifesto and we apply this to the data scientist there are some observations:

I demand sustainability requirements
With respect to the sustainability requirements, the data scientist will mainly look for ethical and compliance related requirements. In the chapter on Responsible Data Usage and Responsible Systems, you can find the necessary information about this.

I follow sustainability principles
The same as for the other specialist roles applies to the data scientist. Don't yield for what is called urgent business priorities. Depending on the application, a model can have a huge impact in a decision process. That impact should be clear to the data scientist before he or she starts building the model. Whether data is allowed

to use for creating the model is a different question than whether data is available. There are plenty of situations where the data is available but should not be used.

I build in a sustainable way
The data is there, the data privacy is guaranteed and now let's build the model! The GPU cycles are spinning like a ball and if the button is pressed to generate the model, the lights in the office will dim by 25 watt, the datacenter will become 5 degrees warmer because the airco's can't keep up. There are publications[8] where the use of AI is linked to ecological disasters. All right, you are a data scientist, before you know it you're not welcome at parties anymore and people are desperately trying to ignore you!

The publication also refers to code carbon. This is a lightweight software package added to your Python code to measure CO2 emissions and also how you can make improvements. Unfortunately, you are referred to providers where you can run your code because these providers use green energy. It feels a little bit like compensating....'green IT providers'.

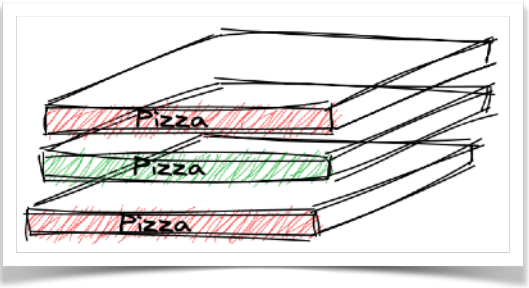
I deliver a sustainable system
Finally, you also want your model to be used for a good purpose. That can be tricky because the model is only a very small piece from a business chain.

5.5. IT SPECIALIST: THE INFRASTRUCTURE SPECIALIST

Infraspecialists in short, can have a big impact on the sustainability of IT environments. They determine to a large extent how datacenter facilities are deployed, and which infrastructure services are used in the datacenter. Often they are also involved in capacity issues and are the focal point for decisions regarding storage. In addition, the infraspecialist is the one who tests and apply new technology to the IT landscape. How these infrastructure components are configured

[8] https://datanews.knack.be/ict/nieuws/hoe-vermijden-we-dat-ai-uitdraait-in-een-ecologische-ramp/article-opinion-1865833.html?cookie_check=1670936753

to perform optimally lies in the hands of these specialists.



If there is a requirement from the business for a certain availability of a system then the infraspecialist is the one who decides how this requirement is being met. And be honest, the infraspecialist often has his or her domain perfectly well organised. In the end, it must be avoided that with every incident he or she will be called out of bed in the middle of the night in order to solve a problem. There is no better motivation to avoid problems!

Let's look at this role in the IT organisation with respect to the behavioural manifesto.

I demand sustainability requirements
As with security, the weakest link determines the strength of the chain. The infraspecialist is able to translate the policies for sustainability to his or her work environment. This can apply to technology where the specialist in many cases has a powerful position with regards to administrator rights and therefore may have access to sensitive information. Or produce incomplete documentation or misdocument technical solutions so his or her role will remain indispensable to be the hero every time in case of incidents (do not laugh this happens really). This is of course about ethics but something much more simple, that the infraspecialist gives sustainable advice on life cycle of technology and only replace/refresh when it is really necessary.

The infraspecialist needs requirements in order to make the chain for sustainability complete. In this respect the infraspecialist must play an active role.

By actively we mean: if the requirements are missing then the infraspecialist will ask for them. With the implementation of systems, the impact on sustainability through technology choice is underestimated. For data in rest (not being used actively), we can use tape media instead of online storage. Is it really necessary for the information to be super quickly available (simply as it is so convenient) or may it actually take a couple of seconds?
Do we consider consolidating applications with identical characteristics (workloads) on enterprise servers or do we continue to increase the mass footprint of 'pizza boxes' because we are familiair with this pattern?

I follow sustainability principles
The production process includes all actions required to build and test an environment.

In this, the infraspecialist has a big influence on sustainable choices. Are we implementing new applications based solely on technical specifications, or do we take sustainability requirements into account? Are we keeping the test environments online until it becomes apparent that they are no longer needed, or do we make agreements on the availability of the test environments? I build in a sustainable way

An infraspecialist understands very well for which type of application an environment should be configured. In addition to the functional aspects of the application, the infraspecialist is well informed about the non-functional requirements. The specialist secures the systems against unauthorised use and ensures that the data is protected from internal or external abuse. In order to enforce ethical behaviour in the organisation, the specialist implements technology to prevent employees from visiting unauthorised websites that the company has described as inappropriate or unethical. The specialist is able to obtain information on complaints about unethical behaviour between employees, by means of retaining e.g. log data of e-mail, text- apps, etc.

I deliver a sustainable system
A system consists of more than infrastructure components. It is a chain of business functionality, applications, data and infrastructure. The infraspecialist as the last link in the chain keeps an eye on the sustainability of the complete chain.

6. THE PRODUCT OWNER: THE GUARDIAN OF SUSTAINABILITY

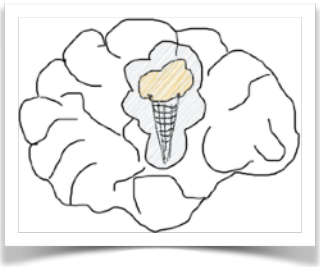
We have architects for designing sustainable solutions and we have programmers who develop sustainable solutions (and of course the data scientist and the infraspecialist). Next is the role of the Product Owner or the Project Manager who is responsible for monitoring this development process. First, we want to look at the complexity and scope of the topic of sustainability and how you can deal with it as a Product Owner. We will show that we nowadays live in a time and space that comes with great uncertainty so you cannot assume that you are going to make a plan beforehand, and implement that plan step by step. As a Product Owner, you also have to deal with cultural aspects and changing needs. We are making an initial effort to develop project plans in such a way that is as sustainable as possible. The most simple things can already add value. As we have experienced, much of the work can be done from home so driving up and down to the office on a daily base is not required. On the other hand, there is a certain need for people to meet each other face to face, not in the least for the fun and social aspect to it. The agile manifesto actually states:

"The most efficient and effective method of conveying information to and within a development team is face-to-face conversation".

Although this is a true statement, there is not much use in it because in real life the practice is different. You have to look at what is really necessary and what is not in order to exchange information.

6.1. BRAIN FREEZE: COMPLEXITY IN THE CONTEXT OF THE ECONOMIC SITUATION

This must sound familiar to you: it's a wonderful warm day, you bought a lovely cold ice cream and after the 5th greedy bite: brain freeze. You can't do anything for half a minute. Luckily after a moment this feeling is gone and you decide to eat your ice cream in a more moderate pace. It is the same with the topic of sustainability. It can sometimes be so much that your brain freezes. Being a Product Owner you have to consider all the trade offs, for a moment might be thinking, I do not know what to do anymore. Where do you want to start? If you haven't experienced this feeling yet, this paragraph has the purpose of preparing you for that feeling.



What helps is to demarcate your area into an area that is comprehensible and manageable. A friend once said when we talked about giving to charities and whether or not the money is going to be spend well: it is my responsibility to give, and the organisation to make sure that it is well spend. That is one of the goals that we want to achieve with this book: **take responsibility in the domain that you can influence.**

We are together in one of these 'perfect storms', all kinds of factors (environment, social, economic, politics, etc.) are in motion and they also have an influence on each other. These factors may reinforce each other or balance each other out, we do not know. It's unpredictable and uncertain what the next few years will look like. What can you do as a Product Owner?

In the first place make sure you're **flexible and resilient.** Dwight Eisenhower gave a good example:

'In preparing for battle I have always found that plans are useless, but planning is indispensable'.

He talks about the process of planning, the outcomes may vary by day, but if you have the process under control, you can move along with the circumstances.
As a Product Owner, you have to take care of an agile or iterative approach. In the book Agile Management[9], the writer Mike Hoogveld gives a good insight into the need for a flexible and resilient approach to achieve this.

Second, take **non-regrettable steps**. Independent of the circumstances, there are steps that are always inherently good. *Steps that do not depend on micro context but are linked to universal values.* An example of a non-regrettable step is to bring down your energy consumption. A wrong step is to transfer gas to electricity. The first is always good, the second one is dependent on a lot of context. The same applies, for example, to electric driving. Initially it was financially interesting (cost per km) relative to fuel, the 400% increase in electricity has turned this business model around within a year. A non-regrettable step would be avoiding unnecessary travel or transport (shorter distances, less frequent use, for more inspiration have a look at the chapter Responsible Impact).

What we describe here are non-regrettable steps. They are always good and independent of the context.
A drop in a bucket? A nice expression, but not very encouraging. But if everybody joins in, the bucket fills up quickly. In Dutch we say a drop on a steel plate. Well, with all of us on that hot steel plate, it will cool (literally) off. 1.5 degree is the target, which has continued to be the target during the COP27 (27th Conference of the Parties).

6.2. EFFICIENT DEPLOYMENTS OF TEAMS

We now have two important principles defined for the Product Owner. Next, we want to look at how the teams can operate as

[9] https://www.managementboek.nl/boek/9789089653185/agile-managen-mike-hoogveld

efficiently as possible. By efficient we do not mean strive to a maximal production of the employee but rather allow the employee to flourish optimally so that it will achieve optimal performance. Therefore a small piece of research that has been done with regard to the need of people.

Proxemics[10] is a study on the human use of space. A distinction is made between different spaces:

1. The intimate space
2. The personal space
3. The social space
4. The public space

These spaces are physically determined in the study of Proxemics, but you can also look at it from a social context.

Personal Space	**Distance**	**Relationships**		**Volume**
Intimate	Up to 50 cm	Partners	Core Family	2-5
Personal	0.5-1 metre	Friends	Extended Family	10
Social	1-4 meters	Acquaintances	Colleagues	150
Audience	More than 4 meters	Unknown persons	Transient	> 250

With the social context, we mean that these spaces also all have a purpose. As a Product Owner, your team has certain needs, how do you best respond to that? The intimacy of a team is crucial.

Stagnation means decline. Traditionally we see a movement towards growth to increase efficiency and influence. This means larger organisations, more specialisation and more complex

[10] https://en.wikipedia.org/wiki/Proxemics

communication structures. This is how Eckart Wintzen describes his organisational concept for the organisation BSO[11], a very agile organisation that was often geographically dispersed. Think about as the type of space you work in and the needs you have within it:

Personal Space	**Need**	**Role**	**Meetings**
Intimate	*Share personal feelings in the context of work*	*Buddy*	*Daily Meeting*
Personal	*Result/Loyalty*	*Team/mentor/ Choach*	*Meet weekly*
Social	Knowledge Sharing/ Importance/	Division/ Knowledge group	*Meet monthly*
Audience	Pride/Hearing	Company	*Meeting annually*

Wintzen believes it is important that people feel part of the department, not of the company. This could be a good model for companies larger than 150 people. Geographical grouping, that's where your involvement lies. This means that there is no longer any need to arrange large physical company meetings, which can easily be done through a virtual meeting.

So, as a Product Owner, remember that your team operates in the personal space. You have to make room for that weekly meeting. That people have their buddies within or just outside the team, that you are part of a department or knowledge group in which you stimulate monthly involvement. And finally, every team member is

[11] Bureau for System Development (Bureau Voor Systeem Ontwikkeling)

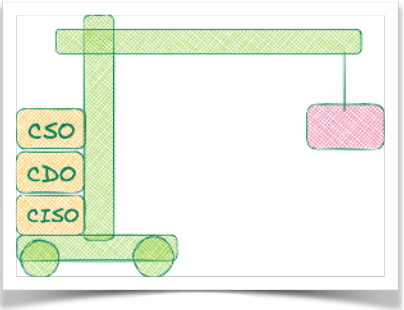
also part of the large company, which stands for sustainable standards and values.

7. THE CHIEF

The boss of sustainability is the translation of the "Chief Sustainability Officer". Well, it sounds a little different and there are probably two salary scales between them. So let's talk about the CSO. The CSO is the staff member who is engaged in the establishment and implementation of policies about the company sustainability strategy. A new role that has been created to gain more and better focus on sustainability aspects. It is likely that this first priority will be to ensure that mandatory reporting (ESG) is delivered. See also the Responsible Data Usage chapter. But also the sustainability image of the company is important.

Depending on the type of industry, the CSO will deal with IT to a greater or lesser extent. In the transport sector, this will be less, in the financial sector it will be more. Simply because financial institutions are almost IT companies and therefore IT is no longer a secondary process.

The CSO will also work together with other chiefs: the Chief Data Officer (CDO) when it comes to data privacy and ownership and the Chief Information Security Officer (CISO) when it comes to vulnerability for example.



CSO, CDO and CISO are important for the balance

There are very high and tall cranes moving construction materials. If that material hangs at the end of the crane arm, there is a huge leverage effect. You can compare the construction material to the company's quarterly objectives, the expectations of shareholders or promises of

politicians. The crane can only stand by means of counterweights. The CSO, CDO and CISO are the counterweights of the organisation. They are taking care of the perfect balance.

triage for balance

8. THE EMPLOYEE

We are talking about the responsible *use* of IT systems. Many of these systems are designed to make the life of the employee, easier?! We create systems that support users by letting chatbots handle simple questions from customers. We build case management systems so that work is assigned to the right employee in order to work more efficient. We want to optimise the process as much as possible so we are going to measure: what is the lead time and where are bottlenecks in the process? Before you know it, there is a calculation culture based on performance. Is this sustainable, where are the borders, when do you feel controlled?

A personal story. One good day I thought about the possibilities of automatic scanning of documents. I saw beautiful saving opportunities and went with a business case to my client director. Saving on ten people with 125k a year is a whopping 1.25 mio per year! An one time investment of 1 mio and a ROI within 1 year is obviously a no-brainer.
The director looked at me, said nothing, and took me to the department where the scanning of these document took place. You may recognise this type of place: in the neighbourhood of the mailroom, tucked deep in the basements of the building where men and women are busy with their work; dedicated, serious and with attention for each other. They arrive in the morning on the bike carrying their lunch boxes because food in the canteen is too expensive. They certainly don't get paid a 125k a year. This is their job, this was why they come out of their beds every day. This is their existence.

I left in my lease car, to a have lunch at a restaurant. The proposal went into the bin, having a business case or not was not interesting any more. It's about much more than just that. Thankful to this director and his life lesson. Needless to say, it is not a simple

dilemma, but there are limits to continuous improving of productivity or efficiency. A sustainable organisation has a sustainable relation with its employees.

RESPONSIBLE DATACENTER

Datacenters are used to house the physical infrastructure under the right conditions. There must be sufficient power supply and sufficient cooling to allow the infrastructure to function optimally.
In addition to the supply of power and cooling, the datacenter also provides power backup facilities when the grid goes down. In the emergence of datacenters, cooling was always a major challenge because the computer systems were 'loosely' placed on the floor, often grouped into systems of the same type. The mainframe, distributed systems such as UNIX and Windows together and the peripheral equipment (printers, tape units) were also often grouped together in a specific corner. Due to constraints in data communication, equipment was often placed in decentralised office departments in so-called Satellite Equipment Rooms (SERs). The central datacenter was known as the Main Equipement Room (MER). We no longer have the restrictions in data communications thanks to the introduction of modern communication techniques. All the information systems are now centrally located in the datacenter. Of course, that can be a datacenter within its own walls (on-premise), but of course also a cloud datacenter.

9. KEEP THE SYSTEMS COOL

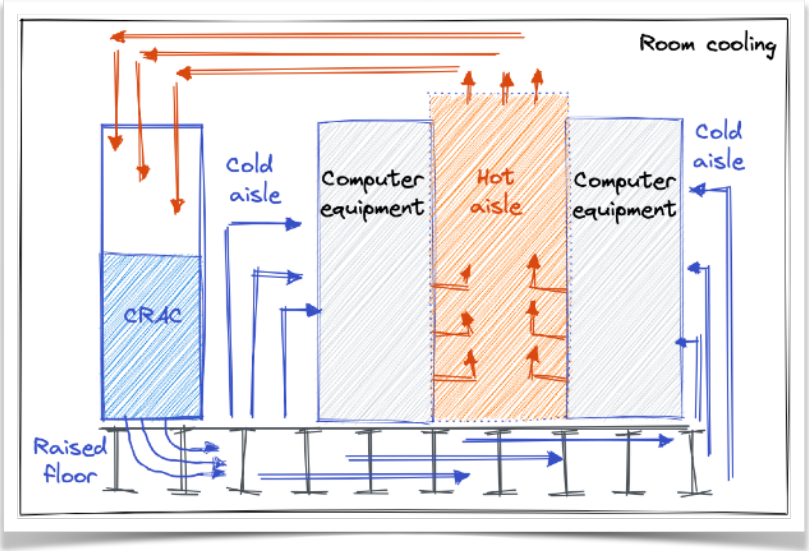
Cooling of systems has always been a challenge in a datacenter. Traditionally, it was done with large air conditioning units that were lined up in the enclosed space of the datacenter. Nowadays, the servers are arranged in locked 'corridors' which can be optimally cooled. This technique works much more efficiently than the loose units that traditionally were on the floor of the datacenter. The technique of cooling has become very advanced. The traditional 'air conditioners' that blow up chilled air are replaced with water-cooled systems that can cool down on the precise level of CPU, GPU, or memory cards.

9.1. AIR BASED COOLING

The basic principle of air cooling is quite simple. Cold air is blown into the room where the heat in the room is 'broken down' by mixing it with cold air. There are several ways in which air cooling systems are used:

- room cooling
- row cooling
- rack cooling

Room cooling: the cooling of the room is achieved by blowing cold air into the room (CRAC - Computer Room Air Conditioning). This can be done by circulating the air or bringing it to the systems via the raised computer floor. Nowadays, hot and cold aisles are used a lot so that the air circulation is more manageable and the cold air can be brought to the desired location. This makes a difference in your purse!



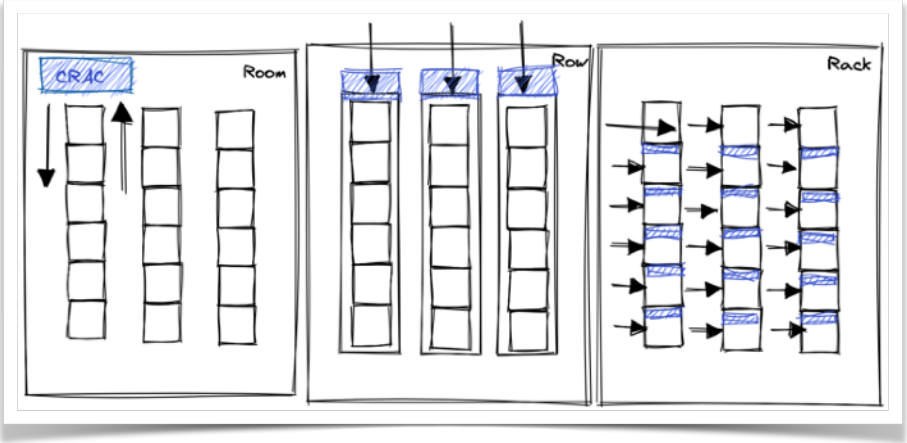
Room cooling overview

Row cooling: row-cooled solutions are more precise than room-based cooling systems. Each row has its 'own' units by which the air circulation can be brought in the row with greater precision. This technique is more efficient and therefore energy wise more economical.

Rack cooling: this form of cooling is even more efficient where the cooling is brought to rack (cabinet) level. The units are placed on top or inside the rack so cooling capacity can be adjusted to the situation of a specific rack where, for example, CPU or GPU are running intensive workloads.

Disadvantages of air-cooled solutions
There are some challenges with air-cooled solutions. Workloads are becoming more CPU and GPU intensive. This calls for a lot of cooling capacity. Air, in many cases, is no longer sufficient to supply the required cooling capacity. The cost (because of energy) is huge,

and these costs represent a large part of the budget to run a datacenter.
In addition to this financial setback, it is not really good for the health of your colleagues who regularly have to work in the computer room. The sound that the units produce can lead to hearing damage.



Room, row and rack cooling

9.2. LIQUID BASED COOLING

Liquid-based cooling is applied more and more broadly than just for mainframes and supercomputers. Liquids are 50 to 1,000 times more efficient than air! Systems are more demanding due to the 'heavy' applications. Liquid cooling promises in many cases to provide a solution to the challenge of computer concentration and heavy workloads.

Generalising there are three major liquid-based cooling techniques:
- direct-to-chip cooling/direct-to-plate cooling
- rear-door heat exchangers
- immersion cooling

Direct-to-chip cooling: brings the cooling system all the way inside the computer chassis. Cold liquid is directed to 'cold plates' that are located right next to the heat sensitive components (CPU, GPU, memory). The cold plates absorb the heat from the components. The warm liquid is then discharged back to a cool mechanism. Once the liquid has cooled, the liquid is returned to the plates.

Rear-door heat exchangers: a similar concept can be applied at rack level. The rear door is replaced by a heat exchanger and the warm air of the server fans is blown through the heat exchanger, which cools down the air again. The heat of the heat exchanger is discharged to the outside by a closed circuit.

Immersion cooling: a much newer technology by which the internal server components are immersed in a non-conductive fluid. The components and the liquid are then packed in sealed containers so that they do not leak. The heat of the components is transported to the coolant in this way. This process requires much less energy than other methods of cooling.

Drawbacks of liquid cooled solutions
Water and electronics don't go really well together. The risk of leakage is a 'thing' for many IT professionals, especially with the direct-to-chip cooling method. If this solution starts to leak, then the damage is huge!
In addition, this type of solutions require specific datacenter operators skills and specific management frameworks to configure and manage these units. This is not a sinecure and has a nice price tag.

9.3. CHOOSING THE RIGHT SOLUTION

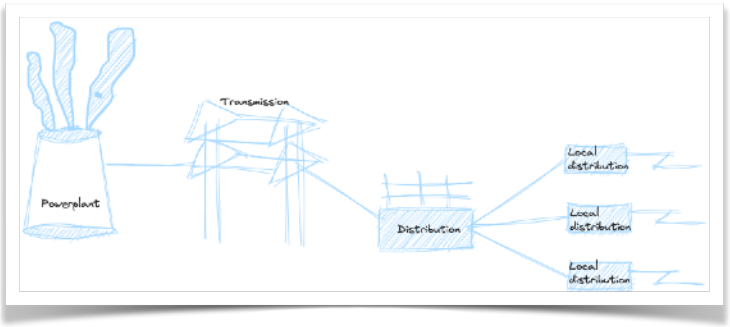
In many cases, the Total Cost of Ownership (TCO) is used as a measurement for reference. It is quite complex to make a good calculation and compare it with each other. The more efficient and more precise the cooling is, the higher the price tag will be. On the other hand, the energy consumption and water consumption of fluid-based cooling is a lot less. This is positive in a TCO comparison,

because in liquid cooling, you can concentrate more computing capacity per m2.

We hope, of course, that datacenters will have sustainability on top of the priority list, especially if you as a sustainable enterprise has it's own datacenter.
When choosing a cloud provider, make sure you have Power Usage Effectiveness (PUE) and Water Usage Effectiveness (WUE) as a selection criteria. Whether you can have influence on it or not, that question on its own makes it clear that sustainability is important to you!

10. POWER SUPPLY

We'd just like to get straight to the point. Servers that run on USB-C technology are in our opinion a sub-optimal sustainable solution. You still require power and an adapter (transformer) to convert this technology to 5, 9, 12, or 15 volts.
Adapters are everywhere these days. With your telephone, laptop, but also in kitchen equipment, hi-fi, and so on.
These power conversions cause a lot of energy loss. Every conversion step is another piece of energy loss. The energy chain is depicted on the picture below. From high voltage to voltage that we use in our household. Energy loss also occurs here. At the end of the chain there will also be a loss again, the standard 230 volts must be converted to the usual 5 or 12 volts for most equipment.



Energy chain and power conversion

Solar cells can supply 12 and 5 volts. If you can enter the generated energy (voltage) directly into the computer, you do not need a conversion (and therefore no energy loss).
A solar cell supplies about 1.1 volts. So 5 solar cells for the components that run on 5 volts and 12 solar cells that need 12 volts.

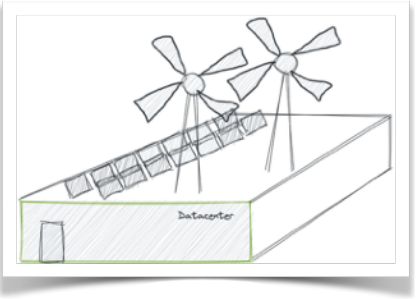
It's just a thought...

10.1. NO DATACENTER WITHOUT POWER

Of course datacenters need power, a lot of power. The critical systems are often equipped with dual power supplies and are also connected to the emergency power supply, the Uninterrupted Power Supply System (UPS) should take over the power supply from the grid when it fails. The UPS lasts for about 5 minutes so the generators (usually diesel) can take over the emergency power supply.
The power supply for datacenters is usually taken directly from the grid. Worldwide, the datacenters use 1 - 1.3% of all power, but in the Netherlands that is no less than about 2.8%! This is equivalent to 3.2 billion kilowatt hours.[12] It is logical that politics has become involved in the location strategy of large datacenters.

10.2. DATACENTER BUILDINGS

'Beauty is in the eye of the beholder', but few people will be moved by the attractive properties of a data center building.
Grey, concrete, square, all elements that characterise a data center. Naturally, there is still a lot to be gained from a sustainability point of view. Why do we keep building and why don't we reuse existing buildings? Flat roofs patiently waiting to be filled with solar panels for the data center or for local residents.



We can also become more sustainable from an aesthetic point of view. If the roofs are not filled with solar cells, then with greenery. In addition to a way to stimulate biodiversity, this is also great insulation against the heat.

[12] https://www.dutchdatacenters.nl/thema-energie/

11. DATACENTERS AND HYPERSCALERS

Perhaps you would like to implement a principle for reducing the number of square meters of the datacenter. The less 'floor space' the less electricity, cooling, UPSs and other basic facilities wil be required. The amount of servers becomes a problem for many datacenters. The choice for distributed systems (pizza boxes) was once made because of horizontal scalability. The choice of enterprise servers would be a good solution to increase the computer density.
A datacenter that can be fully or partially fed directly from renewable energy sources, such as solar and wind, is a great idea.

Large datacenters are a result of the strategy to have central control over the datacenter services and to achieve maximum consolidation of software (middleware and apps) and data. Datacenters where huge numbers of servers run (5000 or more) and therefore large in size and surface area are referred to as hyperscale datacenters. Think of the datacenters used by companies such as Amazon, Google, Microsoft, IBM and Facebook; centralisation to the max is the leading principle.

Recently there was a broad social discussion about the establishment of a new META (Facebook) datacenter in Zeewolde (place in The Netherlands). From technology to politics. Until recently, we ourselves have never heard discussions or seen parliamentary debates in the light of datacenter locations. The terms Main Equipment Room (MER) and Satellite Equipment Room (SER) have been replaced by datacenter or hyperscale datacenters. But a datacenter that uses as much energy as 460,000 households? Today, it cannot be the case that a modern datacenter demands so much energy from the network. A network that may already be up to

its limits due to the energy transition. Datacenters should be more autonomous in terms of energy supply. More on this subject in the next chapter on cloud.

Think back to the introduction to this book where the example of 'convenience serves man' illustrates how we are fed with technology and how willingly and easily we adopt and consume it. There is a growth model in our demand for technology that seems never-ending. IT out of the wall seemed like a fairy tale 20 years ago. Take a look around you and count the number of IT systems (telephone, laptop, watch, TV, radio, oven, washing machine.....) within reach, which you subconsciously don't even think about anymore that are IT systems. Will this stop at some point?
It is always dangerous to keep talking about 'the past', but 'the past' remains our reference. The image of the father in his easy lounge chair who can operate the TV with a remote control with a limited number of channels has now become a picture of choice overload. The TV has become a 'one stop shop' of streaming services, information and news channels and you can still watch everything you missed.
No more records or CDs in the room and no more cupboards full of videotapes, DVDs or Blue Rays. Sometimes new technology also has advantages :-).
It is a fact that movie and music streaming services need datacenters. As well as cloud providers who need a place to offer their services.

11.1. HOW SUSTAINABLE IS CLOUD?

To cut it short short: cloud is many times more sustainable than on-premise datacenters. The Microsoft cloud carbon study 2018[13] contains many interesting figures. For example, it is described that the Microsoft cloud is between 22% and 93% more energy efficient than traditional datacenters, but also that the CO2 efficiency scores between 72% and 98% better.

[13] https://www.microsoft.com/en-us/download/details.aspx?id=56950

Many companies are now using technology from the cloud. In America, the datacenters collectively use about 75 billion kWh. That is the same amount per year as about 6 million American households, according to a study by Berkeley National Laboratory (for European datacenters it was 100 billion kWh in 2020 according to insiders). This usage figure would be many times higher without the use of commercial datacenters (cloud).
Naturally, in addition to Microsoft, the other major cloud players are also active in the field of sustainability.
AWS claims to be 88% more efficient in terms of CO2 footprint than traditional datacenters. Among other things by using specific chip technology in the computers. The Graviton3 (similar to the chips in the phone, namely ARM) uses 60% less energy than other general purpose chip technologies.
AWS, Google, IBM, Oracle and Microsoft are all busy with sustainability.

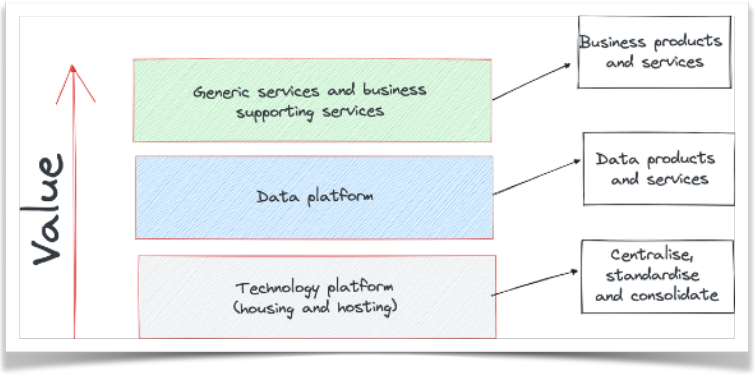
A critical note: these parties often work with purchased emission rights, but overall the cloud still seems more sustainable than on-premise datacenters.

12. ODC: A GOVERNMENT STRATEGY

Since we are architects in the government domain, we also look at what is happening here in the field of their datacenters. Our government has realised a centralisation of the datacenters through a government datacenter (ODC) strategy. The government has achieved this by centralising the approximately sixty-five datacenters into four ODCs. This has resulted in serious energy savings. Could a follow-up process of the ODC consolidation result in a truly sustainable ODC?

At the moment, the ODCs are mainly used for housing and hosting, but there are not yet many services offered from ODCs that are used government-wide. A major step forward could perhaps be made in this area. A good example that does already exists is the generic provision for workplaces! But why isn't it used by everyone? In addition, operational organisations, for example who manage social insurances have their own datacenter strategy that is not in line with the ODC strategy.
We have included a section in the next chapter that outlines a welcome perspective for the ODCs.

Perhaps a simple model can help with this discussion?



Delivery model

12.1. TECHNOLOGY PLATFORM

The ODCs currently mainly provide a technology platform. Run your applications in our datacenter and you no longer have to worry about the facilities for your platform. A nice step in itself, but wouldn't it be interesting to develop a service portfolio from the ODCs that focuses on the properties of the workloads so that the platform and application can be delivered as a generic facility.

For example, mainframe workloads could easily be accommodated into one ODC. These workloads are characterised as high transaction volume applications designed to run on a mainframe. The technical knowledge and skills (I-craftsmanship) surrounding this platform are present in that ODC, but also the knowledge and skills to modernise and develop these applications into modern applications in functional blocks instead of traditional 'stovepipe' applications.

12.2. DATA PLATFORM

Hosting applications is already a good step, but it becomes even more interesting if you can host 'chain data', data that is exchanged between different government organisations, as a service. With such a data platform, data from different sources and for different purposes is brought together and offered as a service to the various organisations. This creates one government data platform that can be used by different organisations. Based on this centrally stored data, data services can be developed that help prevent this data from having to travel throughout the entire landscape (provided organisations have the rights to do so, of course). Think of data virtualisation technology and storage virtualisation; modern technology to reduce the data footprint.
In addition to central platforms, domain-specific data services could also be hosted. These services are developed by the domains (work areas) and can be used by other organisations if authorisations permit.

12.3. SERVICES PLATFORM

Services can ultimately be developed and offered on top of the data platform. Known in the government as generic facilities. Think of facilities that are used by almost all departments, which currently have been developed in their own specific way. As a result, reuse is often not possible and that is a pity!
Examples of generic facilities are: workplace services, archiving services and development environments.

13. CODE OF CONDUCT RESPONSIBLE DATACENTERS

A code of conduct has been drawn up for datacenters by the EU. Organisations can participate voluntarily. This code of conduct was created in response to the enormous energy consumption. Participants in this initiative may even qualify for an annual EU Datacenters Code of Conduct Awards. More information can be found on the website:

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

13.1. RESPONSIBLE DATACENTER & RESPONSIBLE INFRASTRUCTURE

Naturally, these subjects are closely intertwined. In the next chapter we look at infrastructure, but occasionally you see references to datacenter facilities from infrastructure or vice versa from datacenter to infrastructure facilities.

14. RESPONSIBLE DATACENTER PRINCIPLES

This chapter describes a number of example principles that can help to make datacenters more sustainable. We use a template from TOGAF[14] for this: an architectural approach that is used by many companies.

RD1 Principle	**Use the most efficient cooling technology**
Statement	Choose a cooling technique for the datacenter that does justice to the requirements of the infrastructure.
Rationale	By applying the right cooling technique, you avoid that the cooling system needs to run on maximum performance to cool down sufficiently. Servers where CPUs and GPUs are heavily used require a different technique than servers where only I/O takes place.
Implications	When designing the datacenter, workloads with a certain characteristic can be consolidated so that the most efficient cooling technology can be applied.

Principle RD2	**Make datacenters self-sufficient in their energy needs (net zero)**
Statement	Datacenters must obtain their energy needs as much as possible from renewable energy that is preferably generated by themselves.

[14] The Open Group Architecture Framework: https://www.opengroup.org/togaf

Rationale	By providing datacenters with solar panels, wind turbines and other energy sources to the maximum, the pressure on the local power plants is reduced and the datacenters become greener.
Implications	The green energy supply should be part of the design of datacenters.

RESPONSIBLE INFRASTRUCTURE

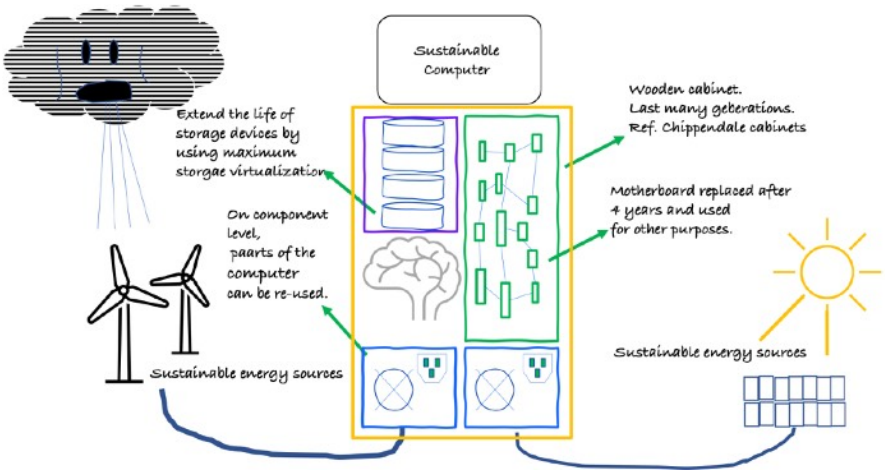
In this chapter we describe infrastructure from a sustainability perspective. Our infrastructure is modular and that is also the case with this chapter. The infrastructure is considered from different aspects. To top it all off we are going to be the boss of the ODCs for a day.

15. THE FIRST ASPECTS

15.1. MINDSHIFT

Imagine building a computer purely from a sustainable idea... .. To begin with, you naturally build on the basis of sustainable components where possible.

We'll start with the computer case. Why should it be replaced when the technical or economic lifespan comes to an end? The cabinet itself is still fine and can last for years.
How about a cabinet made of wood?
Thomas Chippendale (1718-1779) made cupboards that can still be seen today in museums or in people's homes.



'Sustainable Computer'

Naturally, the computer will then be powered from sustainable energy sources. The power supplies are modular and can be replaced in the event of defects due to age. Even stronger, if you do not have high availability requirements for such a computer, the

power supplies are not configured on an active-active basis, but on an active-standby basis.
This provides the following options:

- sun and wind, you can choose which power supply is active
- only sun, choose the right feed
- wind only, choose the left power supply
- no wind and no sun, switch to traditional energy supply

The motherboard contains the intelligence of the computer. Computers are often replaced on the basis of economic depreciation and not so much because the computer is technically worn out. The era of Moore's Law (the number of transistors in an integrated circuit doubling every two years) has ended. So the renewal due to significant improvement is not really valid anymore. So you could replace the motherboard if there is a real technical improvement or because you need different technology. For example, in addition to CPU capacity, also GPU capacity for specific analytical applications.

This also applies to the internal memory. If more capacity is required or if faster internal memory is required, memory can be replaced as a modular component.

And in the context of cradle-to-cradle thinking, you could reuse the obsolete components in applications that do not place high demands on the technology. Think of the use of processors in for example: cars, household appliances or other low-threshold technology.
It is of course not a serious proposal, but hopefully it will stimulate you as a reader to look at computer systems differently instead of the way we are used to traditionally.

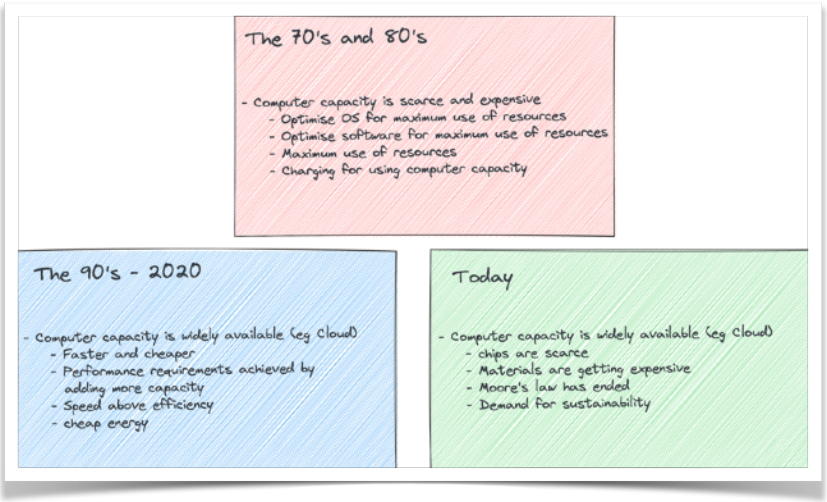
While we're at it, let's put on our sustainability glasses again. We see a number of topics that are worth examining in the context of datacenters and infrastructure:

- how sustainable is our technology?

- how sustainable are our processors?
- how sustainable is cloud?
- refresh just for the sake of refresh?

15.2. SCARCITY VERSUS ABUNDANCE

Over the years we, as IT people, have learned to deal with scarcity and abundance of IT resources. In the early years of the IT industry (1970s and 1980s) capacity and resources were scarce and expensive. An RS2-32 cable cost no less than Fl 350, so the business case at the company I worked for was quickly made. The purchase of a soldering kit was earned back in no time. This scarcity also applied to the developers; running the software as efficiently as possible in order to limit the impact on resources.



Scarcity through the years

Can you still imagine this: a user of the mainframe (in this case a Bull L66) immediately saw how much money he or she had used on the mainframe when logging on. Because the central computer had to be shared by many users, 'time sharing' was an excellent solution.

Here you had a central computer that was shared by so-called terminal users. A good solution for developers who had to edit a lot of code and then do small test runs.
By the way, this was also the period when not everyone had a telephone on their desk. It was actually a status symbol back then.
In the 1990s, computer scarcity was overcome with the introduction of personal computers. Suddenly everyone had a 'mainframe' available. There was an enormous dynamic in the development of new technology. Instead of hard wired point-to-point connections from terminals to mainframe, the local area network topology was introduced. People could share resources in smaller communities and the printer of such a community was placed at the department instead of in the central computer room (until it was discovered that a printer was not really healthy in the department either).
Client server technology resulted in considerable modernisation of programming, with the logic partly running on the PC and partly on the central computer. It went reluctantly; first the department-managers a PC, later the IT people (by no means everyone). After the massive introduction of PCs came the dilemma of email. Who is allowed to use email and who absolutely does not need it for work?

Technology was being developed at a breakneck speed and knowledge in particular was scarce. The resources were reasonably affordable due to the large supply, but also very complex to use due to a lack of standardisation in the market.
This resulted in IT entering turbulent waters. Lots of disruptions and unstable environments. And this while companies and organisations became increasingly dependent on IT facilities.
Perhaps the outsourcing wave has helped somewhat in normalising the use of IT resources. Suddenly there was a party that provided IT services based on a tight cost model (service center) instead of an internal IT organisation (cost center) that often let the costs get out of hand. Payment is made for every action and, moreover, prices are kept in line with the market and transparent, which could sometimes go off rails with an 'own' IT organisation. For example, the sandwiches for workers doing overtime in the weekend were paid

for through chargeback mechanisms to the departments for using firewalls as a common facility (for example).
Energy was not an issue at all. This only came into play at the end of the first decade in the 21st century: green datacenters. This was a very first attempt to make datacenters with their energy-guzzling systems aware of energy behaviour. These were mainly facilities-oriented solutions (air conditioning, UPS, consumption of cooling water). It was a hype for a while, but this faded into the background with the great financial crisis that started around 2008. All IT investments more or less came to a standstill. Let alone companies having an eye for an environmentally conscious IT policy.

How different it is today. Scarcity is the norm again. Lack of raw material to make computer chips, energy prices are soaring and consumers are demanding that companies embrace sustainability. That does not mean that the major sustainability transformation is a fact!

Saving energy is not the same as sustainability:

We talk a lot about energy saving. There is nothing wrong with that, but is this the right motivation for sustainability?
If you save energy from an economic aspect, there is a very good chance that you will fall back into old habits when energy prices return to normal. Then the high energy price will no longer weigh on profits and we will be back to 'business as usual'.

Fortunately, for very heavy applications, we have taken timeshare off the shelf again. With the introduction of Quantum technology, we can safely assume that this phenomenon will not get a Personal Quantum Computer (PQC) variant. Users must therefore use this technology on the basis of shared resources.

We think that sustainability should come from an intrinsic motivation. Then you don't need a Chief Sustainability Officer who desperately tries to explain to the outside world how the company is working on sustainability and tries on all sides to prevent the market from suspecting the company of 'greenwashing'. So it is largely culture! This intrinsic motivation for sustainability only arises from an

'unconsciously competent' mode.

15.3. ENERGY EFFICIENT PROCESSORS

Did you know that there is a big difference in energy consumption between the different system hardware?

AI in particular requires a lot of computing power. Intelligent use of CPU and GPU workloads can yield significant energy savings. Contrary to what you might expect from the name, the OpenPower consortium offers a technical solution that is not only open, but also uses a different processor architecture. It uses only 1/3 of the energy compared to traditional processors (Intel).

15.4. ENTERPRISE SERVERS AS A SUSTAINABLE CONCEPT

We are grateful to Intel for the ease with which we were able to modularly scale our infrastructure to the desired capacity to run our applications. Developers could easily be provided with the desired capacity by simply pressing a few buttons.

Do you also recognise this: "this server belongs to our department. It was purchased from our project budget, so we are not going to share it with another department".

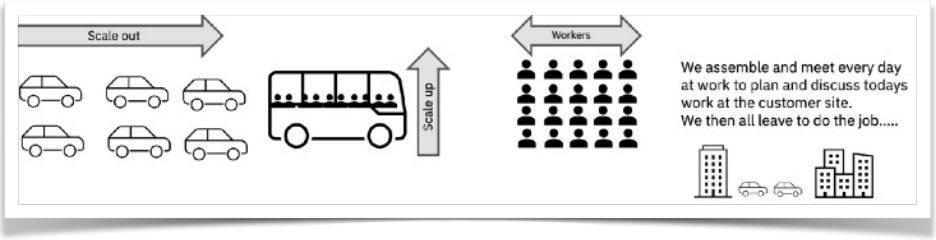
A small example of how this happened in the past (but we are almost certain that this still happens) leading to systems being used sub-optimally. Just think about the physical space that this modular solution uses: if you compare a specific workload running on X86 and an enterprise server, the X86 would use 43 m2 for this workload while an enterprise server (for example IBM LinuxOne) only needs little more than 6 m2. Especially also the energy consumption. For example, the case of Asia Pacific Insurance Company: an x86 used 890 kWh while the LinuxOne solution needed no more than 335 kWh to deliver the same performance. That is no less than 62% savings on energy and 86% on floor space. These kinds of figures are very interesting for companies and organisations that are consciously working on their CO2 footprint.

Look, you shouldn't consider such an enterprise server if you can't take full advantage of it. The great thing is that these types of systems are most useful if you can use them to the full 100%. When does the tipping point arise to consider whether such a system is attractive for your company or organisation?

Compare it to a company that provides employees with transportation to drive from the office to another work location. With 15 employees, it is quite interesting to purchase a fleet of small passenger cars and have them used by the employees to drive from location A to B. With three or four employees per car, you can easily get to a small fleet of four cars. You can still expand that to 20 employees, but it stops somewhere. Then the economic advantage of purchasing small cars is no longer profitable when you look at purchase, depreciation, maintenance, insurance, fuel, and so on. Consequently, as a company you could consider purchasing a bus and transporting the staff in it! Purchase price is higher, but if you calculate the price per seat, the purchase value is less important in this case. If you can fill the bus, you make optimal use of the bus concept.

Apparently there is always some kind of tribal war going on between distributed systems and Enterprise systems. We see it this way:

choose the right system that suits you. Not distributed because you have to, but use a number of principles on which you can base the right choice!



The bus concept

16. WORKLOAD DRIVEN DATACENTERS

We believe in workload driven datacenters. Datacenters that specialise in technology for certain workloads in order to use it as optimally (read efficiently) as possible. A “fit for purpose” and not a “one size fits all”. Incidentally, that “one size fits all” type of clothing never fits, far too long sleeves or actually too short, but that aside. We want to make an attempt to map workloads to the infrastructure that is most suitable. In this case, the most suitable also means the most efficient, the most sustainable.

16.1. SYSTEMS

Roughly speaking, five types of servers can be distinguished, with the first two having a few variants:

1. CISC based servers
   - Single Servers (also known as “pizza boxes”)
   - Blades
   - High end
2. RISC based servers
   - Single Servers
   - High end
3. GPU based servers
4. Mainframes
5. quantum computers

CISC and RISC based servers
CISC and RISC are processor architectures. CISC stands for Complex Instruction Set and RISC for Reduced Instruction Set.

RISC processors are designed with sustainability as a design principle! RISC processors are more efficient and economical and are therefore used in telephones, tablets, household appliances and we also see a lot of communication equipment in datacenters. Until recently, CISC was the most popular processor in datacenters, we see a huge increase in the use of RISC processors, especially with hyperscalers.

Because you are using services from hyperscalers, it is irrelevant as a user to know what kind of infrastructure this service runs on. For example, it can run on the the aforementioned Graviton3 processor from AWS. Windows only runs on CISC and that is the reason why many CISC processors are still used. To use a RISC processor you have to rely on a Unix variant of an Operating System.

This difference is fundamental to a sustainability discussion because RISC is much more economical, but often not the current standard. The strategic choice has been made according to many experts, but how do you deal with this in your datacenter?[15]

GPU based servers
GPU stands for Graphics Processing Unit. Once designed to do fast 3D processing. A GPU is capable of performing many calculations in parallel, ideal if you need to display the next image on your screen. Every pixel on your screen is calculated and displayed simultaneously. These servers are therefore also extremely suitable for building a neural network for AI models. We call it GPU based because all kinds of variants have been developed, for example Google's Tensor Processing Unit.

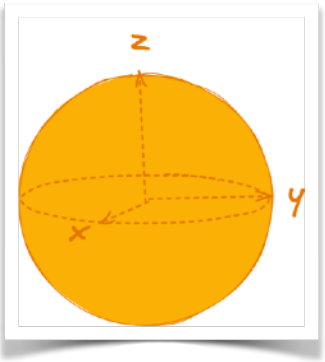
Mainframes
Mainframes are large servers that are known for their reliability, security and speed in processing transactions. This is because the components of a mainframe server are highly integrated. In the chapter on architecture we saw that a high degree of integration is

---

[15] [15] https://itnext.io/risc-vs-cisc-microprocessor-philosophy-in-2022-fa871861bc94

good for energy consumption but not so good for modularity. That is exactly the consideration you have to make here as well. In many large companies, the most important business processes run on these servers. The image may be that a mainframe is large and expensive, but mainframes come in various flavors and price ranges. A Linux mainframe is not much more expensive than a large high-end server. So depending on the size of your organisation and your sustainability strategy, a Linux mainframe could be an option.

Quantum computers
Finally, quantum computers. An extremely powerful computer which can do many calculations in parallel. Developments are still in progress. Although you will not place it in your datacenter, it is already possible to use a quantum computer via the cloud. It is still a niche product, but companies that deal with chemistry, medicine, medical science, encryption technology, etc. are fully engaged in this. We will not yet include the quantum computer in our point of view.



Quantum bit

In the next chapter we have included a table that describes the characteristics of the servers and this table shows the relationship between servers and workloads. This is not a scientific overview but gives a 'feel' for the differences.

16.2. WORKLOADS

What we also see is that more and more combinations of technologies are coming onto the market. An integration of CPU and GPU, for example. But for the moment let's put this aside..

It may well be that you design a type of workload that can make optimal use of such a combination.
Especially for large organisations, it pays off to make a distinction between the different existing workloads. What are relevant

characteristics for a server that determine which workload is suited best to run on this type of machine? Relevant for servers are actually only a few properties.

1. The processing power of the CPU. How fast can instructions be executed (in parallel)? We have also included a price/performance aspect in this.
2. The speed at which external data can be retrieved for processing by that CPU.
3. The connectivity of the CPU with external CPUs. High end servers are what we call channel attached. By linking these, very powerful systems can be built. Supercomputers are an example of this.

Workload	Description	Characteristics
Collaboration environment	Environments where workers and eco-system can collaborate, communicate, and exchange information.	Many parallel processes. Average demand for performance. Network capacity important.
Transactional systems	Transactional systems like payment, stock market, orders and insurance extensions.	Serial process with IO capacity.
Operational systems	Systems supporting operational processes. E.g. systems supporting the mobile telephony network, or a system that controls the assembly line in a factory.	Systems directly coupled to the operation. Often with real time requirements. Much communication with peripheral systems. High availability requirements. Often de-coupled from IT network for security reasons.
Reporting systems	Applications used for reporting. Often data from data warehouses.	Queries often on structured data. Much processor capacity but no real time requirements.
Information systems (Web sites / self service)	Internet sites handling many parallel user requests	Much parallelisation required. Horizontal scalability required.
Data analysis	Analytics on large date volumes.	Analysis on data is CPU/GPU intensive. Possible (near) real time requirements.
Content management	Systems for storing large amounts of unstructured data like documents, video's, chats and images.	Focus on efficient storage. Not much processing requirements. Can be used parallel and benefits from horizontal.
Development platforms	Platforms for developers of applications.	Requirements for fast deployments with fast processors and much memory.
Innovation environment	Environments for trying out new technologies or building prototypes.	Comparable to development platforms but less requirements for CPU capacity. Often de-coupled from production.
End user environments	Workplace for end users, like PC, laptop, IPAD, etc.	In some cases virtual based on e.g. VDI.

This table provides examples of common workloads with their associated characteristics. For each industry, there may be more or less focus on certain workloads.
Of course, this table only gives an indication. A workload that benefits from vertical scalability will mainly look at high-end RISC servers or a mainframe. If I/O is also important, then the mainframe is a better fit. If CPU speed is important, the RISC or CISC system is the better choice

Server	CPU Capacity	IO- Bandwith	CPU couplings	Energy
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	+ + (480 Cores)	+/-	+ +	+ +
GPU	-	-	+ +	+/-
Mainframes	+ (8 cores >5 Ghz) (Telum)	+ +	+ +	+

Server	Workload	Example
CISC - Single	Development platforms Innovation environment Shielded server in security zone	Lenovo Thinksystem SR630 7X02
CISC - Blades	Collaboration environments End user environments *VDI*	HPE B;ade system c7000
CISC High End	Operational systems Web Sites / Self Service ERP systems Content managemnt	Atos Bullion Servers
RISC - Single	Development platforms Firewall, Routers, Switches Appliances	ARM Server
RISC - High End	Operational systems Reporting Data Warehouse Content management	IBM Power E1080
GPU	Data Analytics AI model Caluculation, Graphics	Nvidia
Mainframe	Transaction systems	IBM Z16

16.3. THE WORKPLACE

In our book we hardly touch on the subject of the workplace. We are increasingly seeing a Bring Your Own (BYO) policy. Bring what you like. A virtual workplace is then offered. Simply explained, you install an application on your own device that gives you access to a workplace that runs on a server in a datacenter. The advantage is that this is safe and you do not have to support and update all kinds of devices. There is one big disadvantage: you have a computer on your desk that costs energy and you have a workplace running in a datacenter that costs energy. So this is not efficient.

The second aspect is the choice of the device itself. A new M1 (RISC) based Apple is 3x more efficient than a new Intel/AMD (CISC) based system. What do you do with that knowledge? The same goes for Chromebooks, if you do work in the cloud...
A laptop uses between 150 and 300 kWh per year. A Macbook about 1/3, that is 50 to 100 kWh, so that saves 100 - 200 kWh. If you have 20 employees, we are talking about the power consumption of one household. If you are the one reader that thinks: I don't have to pay for that as a company, then go back to page one of this book and start reading again.

Well, now that you know this, you should consider this if you have a company with a large number of employees.

17. MORE ASPECTS

17.1. HIGH AVAILABLE AS STANDARD PATTERN

The ease in which our IT resources are handled is sometimes disturbing. Often, for ease of management purpose, patterns are applied that are not necessary at all. An organisation has a twin datacenter and a beautiful IT setup that ensures that the company is operationally available 24/7.
When setting up the environment for a new application from department A, the IT department understood that this is an important application. Therefore it has been set up as high available (because IT people want the best for the company}. In IT terms, active-active. In other words, if problems arise at one datacenter, the environment will continue to run in the other datacenter without any disruption. It has been assumed that absolutely no data may be lost and that the business operation may not be disrupted.
In IT we call this RPO = 0 (data loss is 0) and the RTO = 0 (time to become operational is 0) .

And yes, this is a really important business application and this application has nothing to do with the collected recipes from the hobby cooking club of the marketing department.

In a rare session between the business owner and the team and the IT service provider (it is rare because these people hardly speak with each other), these availability assumptions turn out to be completely wrong. Even better, the application can be out for a day and the business knows exactly which data has been processed in the last 24 hours. So we can easily ask to send it again.....
As IT people, we know that this is of course not an optimal solution and that there are excellent solutions to buffer this data for later processing, but still!
It shows that we sometimes put a huge burden on IT resources for the sake of convenience (with the best of intentions). This puts the consequences for our sustainability policy at risk.

But be honest: how often do IT organisations use certain patterns because this is nice and easy and familiar from a management aspect?
There must be a lot to gain here, also for yourself.

Implement workloads on basis of SLA requirements:
Avoid using over-dimensioned patterns for workloads with low SLA requirements.

If we look at the future of IT, there is a beckoning perspective. Applications are cloud natively developed and run in containers. The infrastructure is smart enough to determine where an application should run, the availability is arranged at the application level and the capacity is automatically scaled up and down. This problem is then almost completely solved.

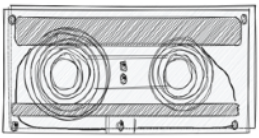
17.2. STORAGE

Data also requires a lot of storage space. Consider using tape if possible, as tape technology is a very sustainable medium. Somehow tape has disappeared from the IT landscape. Perhaps because it is considered old-fashioned or because it does not meet a non-functional requirement where the data must be available within a 'split second'. When it comes to archive data, for example, the knowledge worker can also wait 17 seconds (data retrieval time for tape) until the data of the stored file is available.
Compression technology can also be very useful to reduce the amount of storage space required.

Naturally, solutions such as data virtualisation are also important developments to reduce the storage footprint. We copy like crazy when it comes to data. Copy of copy is saved for use in test environments, for machine learning, BI, backups and so on.
With technology such as data virtualisation, but also a data fabric concept (see explanation in the book 'how to build an architecture blueprint for a data-driven organisation'), unnecessary data 'dragging back and forth' and copying is reduced. By doing much more data processing at the source and only accessing relevant

data from sources, you prevent complete data sets from having to be continuously copied for use in other processes. Also, it easier to tell truth from fiction (be sure about the quality of the data).

Consider the use of tape technology:
Is there a real need for all data to be available in a split second....or is 17 seconds fast enough....?



17.3. INNOVATION: NANO METER (NM) TECHNOLOGY

2nm is just two letters and a number. It seems nothing and it is even less. 2nm stands for two nanometers. That is small, very very tiny. During the Covid-19 pandemic, IBM's research and development center has not been idle. They have developed a chip that delivers 45% better performance while this technology consumes no less than 75% less energy. In short, quadruple the life of your smartphone! How about charging your smartphone once every four days?
The energy gain is in the size (actually 'small') of the chip. All transistors together, about 50 billion transistors on a chip the size of a nail, are very close together. For example, the processing of electrical currents can be processed super-fast. No 'long' distances between the different transistors. Distance is of course a relative concept in this context!

Quadrupling cell phone battery life, only requiring users to charge their devices every four days[i].

Slashing the carbon footprint of data centers, which account for over one (1 – 1,3%) percent of global energy use. Changing all of their servers to 2 nm-based processors could potentially reduce that number significantly.

Drastically speeding up a laptop's functions, ranging from quicker processing in applications, to assisting in language translation more easily, to faster internet access.

Contributing to faster object detection and reaction time in autonomous vehicles like self-driving cars.

We are not going to explain in detail how this technology works, but it goes without saying that these kinds of developments contribute enormously to the sustainability dossier in IT.

17.4. TECHNOLOGY PUSH

In addition to the disappearance of tape from the IT landscape, we also increasingly see that today's IT people want to get rid of batch processing. Real-time processing of information is of course much more fun, exciting and modern than processing in batches. But is this always necessary from a business perspective?
You can come up with wonderful solutions where every piece of information is picked up and automatically released through the IT landscape in an orchestrated manner. This instead of collecting and processing information when there is little business activity, usually at night.
As described in the scarcity versus abundance section, batch processing was a great solution in times of scarcity. Querying systems during the day and input of new data by users and processing at night, instead of side by side.
In this way you do not have to scale all systems for peak load, but more for continuous load, and that costs much less energy.

Who doesn't know them: the architecture principles 'always on' and access to systems based on 'anytime anywhere'? The impact of these kinds of principles on a sustainable IT policy is significant. Perhaps it is good to take a more nuanced look at these kinds of principles and to start making a distinction between systems that REALLY need to be on all the time and systems that don't need to be on at all.
Think of systems that developers work with. Development environments, test environments, acceptance environments, etc, but also systems that are used for machine learning and analytics purposes.

These systems are available 24/7 to developers and projects, but they are often not used after office hours.

Text on this picture is for safety purposes but in the context of sustainability has a complete different meaning.

They can also be turned off and restarted in the morning so they are ready for use when people go back to work during the day. It is often small things that can contribute to behaviour and awareness about sustainability being a step in the right direction in becoming 'consciously competent'.
Measuring system activity is very easy with current technology. In addition, you can activate scripts that power down these systems when idle.
A timer ensures that a script is activated in the morning to restart the systems.

17.5. DEPRECIATION OF SYSTEMS

A standard depreciation of 4 years is still often used. The technology is both technically and financially written off. Is that necessary? In

any case, you could consider using the technology for longer in non-critical environments, for example.
In this book sustainability is amongst other things linked with the environmental domain. In music, the term sustain is also used. It is the effect of making the note 'last' after the string has been 'plucked' or the key on the piano has been struck. So actually resound or last longer.
That is exactly what we have in mind when we look at the use of infrastructure. Make longer use of the technology because it still functions well. The replacement was often done in the past because after this period there were risks of disruption. The hard disk started to struggle or the system became unreliable. In fact, these are complaints and concerns that almost no longer occur.
Systems are incredibly reliable and rarely, if ever, break down.

18. BOSS OF THE ODC'S FOR A DAY

Imagine if we could be in charge of our Government's ODCs for a day.
How would we seize this opportunity to give substance to the ODCs? Naturally, we do this primarily from a sustainability perspective and a security perspective. The super principles sustainable by design and secure by design are our top priority!

18.1. WORKLOAD DRIVEN DESIGN

If we divide and organise the four locations based on the type of workloads, we get a completely different strategy. Namely, certain types of workloads are consolidated and centralised and with this new situation you could also arrange the underlying infrastructure in accordance to these types of workloads.

For example, you could make a distinction between information systems that are used by knowledge workers in the field of policy making, file processing at implementing organisations and systems to support the WOO (law for a transparant Government). Highly advanced software is used for WOO requests, which must support the WOO specialist based on AI models. To use these systems properly, you almost need an astronaut’s degree.

On the other hand you have have systems designed for workloads with large transaction volumes. Think of organisations that handle social benefits, customs declarations, taxes, state pensions, etc. These are systems where most of the transactions are processed without human intervention. Only the rejected transactions (declaration with errors, incompleteness, etc.) are 'thrown out of the process' and are further treated as cases by the office/knowledge employees. We call the transactions that are completed without human intervention 'straight through processing'. The systems that

process these types of large volumes are generally larger enterprise systems such as a mainframe.

Then you can think of an ODC that is specifically designed for application development and testing. Here you have the opportunity to use these systems in the event of big calamities in the production environment and to test new versions of middleware or applications and to carry out chain tests.

Systems for analytical purposes should of course not be missing from this list. For this you could also set up an ODC where models are trained and tested on the basis of data, data warehouses are used for analytical data and reporting and where organisations such as CBS (Central Bureau for Statistics) can use data for statistical information.

18.2. PUTTING OUR SUSTAINABILITY GLASSES ON

By setting up systems based on type of workload, a high degree of standardisation is possible. After all, the system characteristics are largely determined by the non-functional requirements of the upper application layer. This standardisation also creates the possibility of massive consolidation. These types of workloads can be offered government-wide from one location in an unambiguous manner. Of course you can still distinguish between types of users where a policy officer follows a different working method (process) than an office employee in an operational organisation. The great advantage can also be found in the functional use of this type of application. Due to process standardisation, the information is also managed and (centrally) stored in an unambiguous manner. For example, by standardising metadata (description of the data) for specific user groups, it also becomes easier to comply with WOO requests because it can be retrieved at metadata level.

The result is less diverse systems (read workloads) and therefore less infrastructure footprint such as servers. In addition, lifecycle management (new versions of systems) becomes more manageable

and you can concentrate the specialist knowledge (I-craftsmanship) of these systems.

The ODC where the development environments are located will be an absolute example of sustainability. For years we have embraced principles such as on-demand, always-on and other 'convenience serves people' with a view to flexibility. But flexibility comes at a price. Imagine that we deploy technology to scan (measure) whether systems are used by developers in the datacenter.
As soon as there has been no activity for the last hour, the system is deactivated and turned off. As soon as the environment is desired again, it can be activated again.

Of course, these are just loose ideas, but if taken seriously, the ODCs could serve as a great example in the field of sustainable IT.

Workload	**ODC location**
Collaboration services	ODC Haaglanden / Cloud
Transactional systems	ODC Belastingdienst
Operational systems	ODC Amsterdam
	Own management, RWS, Politie…
Reporting systems	ODC Belastingdienst
Information services (Web sites / Self service)	ODC Noord
Data analytics	ODC Belastingdienst
Content management	ODC Haaglanden
Development platforms	ODC Noord / Cloud
Innovation space	ODC Noord / Cloud
End-user environments (Citizens and companies)	ODC Haaglanden

19. RESPONSIBLE INFRASTRUCTURE PRINCIPLES

Below are a number of example principles for infrastructure.

Principle RI1	**Consolidate workloads where possible**
Statement	Consolidate workloads with the same characteristics on enterprise servers and standardise the underlying infrastructure.
Rationale	By identifying workloads with the same characteristics, we create patterns for standardisation. By consolidating workloads on enterprise servers, we reduce the number of physical servers that are used sub-optimally.
Implications	A choice will have to be made in the investment for server technology. There is a different investment picture involved with enterprise servers.

Principle RI2	**Always off is the default**
Statement	Turn off systems that are not being used and save unnecessary energy consumption.
Rationale	Systems are always available while sometimes they are not used at all. By doing workload consolidation in a smart way, systems are deployed purpose-bound. Choices such as continuous use or use on demand can play a role here.

Implications	Systems are made available based on demand. Inactive systems are shut down and activity is measured as soon as they are available for use again. As soon as the systems are inactive again, they are switched off.

Principle RI3	**Extend the technical lifespan of the infrastructure**
Statement	Depreciation of infrastructure can be extended and 'older technology' can be used in a second life.
Rationale	Nowadays, infrastructure is much more robust and can be used longer. Components of infrastructure can be used in other appliances.
Implications	Determine the lifespan of technology based on technical health and not on standard depreciation rules.

RESPONSIBLE CODE

Sustainable programming is also known as "green coding". When we first heard this, we looked at each other with big question marks in our even bigger eyes. Green coding? What is that and how are we going to make the difference with regards to sustainability? When we talk about sustainable programming we are talking about the *process* of creating the code (the writing of the code and building of the application itself) and *running* the application in production. It is not about the application of the application, as is described in the chapter about Responsible Impact. It is also not about the sustainable usage of the application, that is described in the Responsible Systems chapter.

20. GREEN CODING

With "green coding", the emphasis is on resource usage and energy consumption. By reusing code (applications), we can save the energy needed to *develop* an application.

Developing applications also come with an ethical aspect, which is described in the Responsible Systems chapter. Programming also has to do with data privacy, which is described in the chapter about Responsible Data Usage. The focus of Responsible Code is on the usage of computers as a scarce resource and the energy we use for developing and testing applications and the efficiency of applications in production.

In practice, it is not so easy to measure the energy consumption of software. There are several initiatives and solutions, but they are often experimental. Our advice is to measure in a pragmatic way, and primarily focus on relative comparison between applications.

This is much more feasible than measuring the absolute consumption. It is easier to see if the same application uses more or less energy compared to an earlier version or to compare similar applications.

A second way to assess energy consumption is to measure the application performance. Applications that process information more quickly are generally more efficient in using resources compared to slower applications. Performance and sustainability are two quality requirements which are almost always strengthening each other.

20.1. PROGRAMMING OPERATING MODELS

Roughly we would be able to divide programming into a number of operating models:

1. *Embedded programming.* This is the programming of refrigerators, coffee machines, cars, etc. We call this the Internet of Things (IoT). Sensors at the edge of the Internet that generate information. For example, this can be a sensor that measures the water level. Sensors with small remote computers which require a battery pack and only have access to long distance, low band networks. They have to deal efficiently with their resources. This is an excellent example for other operating models when it comes to minimising resource usage.

2. Gaming, Mining, Movies, VR and 3D modelling. These models are CPU and GPU intensive. On the one hand, there could be an important educational element in this sector of applications. This is certainly true for companies and schools of interest (gamification). On the other hand, this also concerns the entertainment industry. You could compare this to a flight trip for a holiday or a marketing campaign. We can basically do without these applications, but we like it despite the effect on the environment. You cannot stop this, because there is a huge demand, it's part of our economy. How can we approach this operating model from the perspective of sustainability? The pressures on the environment of these activities should be taken into account as an "environmental tax". For

example, purchasing CO2 emissions from air travel, now voluntary, but why don't we make this mandatory? For using (online) games a usage based "environmental tax" should be paid by the consumer.

3. Applications and apps development. We are talking here about building applications that support operations. Unlike the previous example, you can build applications that support environmental objectives. Consider, for example, applying for a permit. When you can handle this digitally it saves you paper, buildings and travel.

4. Software development. Here we mean the development of software packages for a large market. For example, SAP, IBM WebSphere, SalesForce, etc. We will not look explicitly at these models but there is overlap with the third operating model. Because it's about packages in which reusability is important, we see that many of these packages focus on performance. Year over year, WebSphere's performance has been improved to stay ahead of competition. As a side effect, this competition enforces a more and more sustainable product.

5. AI development. Over the last couple of years AI is applied more often. Developing and deploying an AI model differs from application development. In our previous book[16], we briefly discuss ModelOps, the delivery process of AI models. Creating these models is a CPU and GPU intensive process which we will discuss later in this chapter.

6. Quantum development. Quantum is another branch of sport. Developments are moving rapidly, but for the time being this development is particularly interesting for certain domains. Cooling requires a lot of energy and the question that you could ask yourself is if the the power of such a computer offsets the energy usage?

We are focusing in this chapter on models 3 and 5. These are the most common models that are used in everyday life within

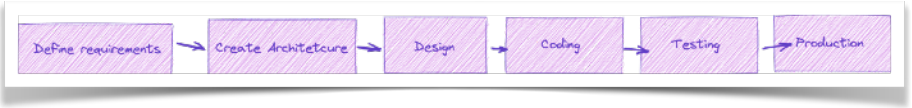
---

[16] How to: Develop a blueprint for a Data-Driven Enterprise Architecture

companies. Model 4 will have a lot of analogy with models 3 and 5. The other models are fairly specific.

	Operating model	**Impact**	**Mindset**	**Remarks**
1	Embedded programming	Very small CPU footprint Many devices	Energie efficiency is a necessary part of the design	What could we learn from this model?
2	Gaming, Mining, Movies, VR and 3D modelling	Large CPU/ GPU footprint	Focus on resolution and performance necessary	
3	Applications and apps development	Medium CPU footprint	Focus on functionality	
4	Software development	Medium footprint	Inherrit focus on performance, security and footprint	
5	AI development	Large CPU/ GPU footprint	Focus on functionality	Uses energy inefficient programming languages
6	Quantum development	Large power footprint Some devices	Focus on applicability	Large cooling systems required (15mK)

20.2. SOFTWARE DEVELOPMENT



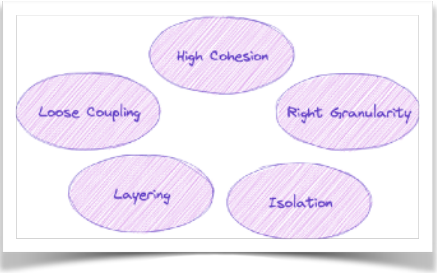
Generic software development process

Software development looks roughly like we show it in the picture above. Suppose we have got the requirements from the business, and we are going to develop the architecture and then create the design, the code, perform the tests and finally put the application into production. So everything that has to do with the SusDevSecOps pipeline. The architect and the programmer play an essential role in this. In addition to the chapter on the architect as a designer of sustainability, we have therefore added a chapter about the programmer. The programmer needs to build an application *in a sustainable way*. In the following chapters we discuss what needs to be done by the architect and the programmer throughout the various phases of software development. We will respond successively to what is to be done in the architectural phase, what is important in the design phase, what should be taken into account by coding and testing (the development street) and what is possible with regards to sustainability when putting the application in production.

21. ARCHITECTURE

When the business has established its requirements for building the software, the application architect creates a component model. A component model describes the components from which the application is built. This process consists of three phases: the first phase is to identify the components. Often this is done on the basis of reference architectures. In the second phase, the components and the connections between the components are specified in more detail. The third phase determines how components are going to be realised. Is it an application which can be bought off the shelf or is something we have to built?

In this process of component modelling, five principles are applied.



Five principles

1. Loose coupling - The components are as autonomous as possible, and communication is as loose as possible.
2. High cohesion - There is logical coherence between the functions within one component.
3. Layering - There is obviously generic components (which can be easily bought or reused) and specific components, intended for a specific business usage.
4. Granularity - The components have the right level of granularity. Not too small, not too big. Are all identified components similar in size?
5. Isolation - Technical dependencies will be realised in separate components so other components are not dependent on specific technologies and therefore better reusable.

When you evaluate these five principles, it is clear that component modelling is primarily intended to cut a complex system into smaller bits and pieces in such a way the components from which this system is built can be ***reused***. Reusability in the framework of sustainability is a good principle. At the same time, we can see that reusable components introduce a communication overhead because they provide a lightweight general usable interface technology. This means that in terms of energy consumption, additional costs are involved. For example, compare the "monolith" to a microservices architecture. The REST API calls between the individual microservices are less efficient than the internal calls in the monolith. The choice of communication between components is a relevant issue in the context of sustainability.

Although 80% of the (external) communications are nowadays handled through REST APIs and the data uses JSON (text) format, this is certainly not the most efficient way. In case of client-server communication, gRPC[17] is a better choice. For local communications, internal (LPC) calls can also be used, but always take care of a loosely coupled interface design, even when the implementation results in merging various components into one tightly integrated application.

To achieve that you must brake your applications down into the proper components and make sure that you use primary argument types to communicate between those components. Do not pass complex structures or objects. During the modelling of components, sustainability (energy consumption and reusability) must be considered as a requirement.

In the final step of "component modelling" the question is asked whether a component (or a group of components) needs to be purchased or built. This step is usually based on the Total Cost of Ownership (TCO) for that component. In most cases the TCO of self-construction is estimated far too positive. Therefor the principle "Buy

[17] https://www.wallarm.com/what/the-concept-of-grpc.

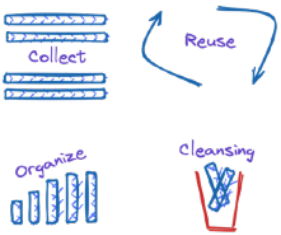
before Build" has been introduced in many organisations, but that aside. Our concern is that sustainability must be considered when making the choice between buy and build. What are the advantages when it comes to sustainability in Buy vs Build?
In both cases, the energy costs of both *building* and *running* an application must be taken into account.

Before an application is to be built, a design is made per application component. The design makes use of design patterns. In the next chapter, we discuss a number of sustainable and less sustainable patterns in case you have to build the application yourself

22. DESIGN

A good design is full of patterns. With regards to code and design patterns, it is highly recommended to take notice of the initiative of Prof. Dr. Patricia Lago (VU in Amsterdam), she published various articles on software and sustainability[18].

22.1. DESIGN PATTERNS



Software developers make use of many patterns. Patterns are a solution to a common problem in a (specific) context. It is not the intention to evaluate all patterns but rather to become aware of sustainable and less sustainable patterns!
We discuss four categories of design patterns that can increase the efficiency of your software:

1. Collect
2. Reuse
3. Organize
4. Cleanse

Design patterns help you in organising your daily life!

[18] http://patricialago.nl

By the way, these categories do not only apply to software development only, but you can also apply those perfectly in your daily life.

5. Finally, we will also discuss an anti-pattern, we call that pattern: 'passing along'.

22.2. COLLECT

If you do the dishes, you will not wash one cup or one plate. You collect them until the moment you have a reasonable amount and then you start. Unless you are a student, because then you start at the moment you run out.
Applications can use different patterns in the collect category:

1. Batch processing
2. Asynchronous communications
3. Queuing

Batch is Back
We may have thought that this is something we would like to get rid off one day, but in the context of sustainability, this is absolutely not true (and there are more reasons why batch is an extremely useful pattern).
Batch processing is hugely efficient when we compare it to a Service Oriented Architecture, or to a microservices architecture where each transaction means the call of one or more services. In batch processing, there are hundreds or thousands of transactions in one call that are processed. In addition, batch is much simpler from a performance engineering perspective because it is a series of sequential predictable steps. Also you really don' t have to wait until night before you can run your batch processes. You can run your batch process 1 times per hour if you would like.

Asynchronous communication
As human beings, we use this way of communication a lot. Think about the regular mail, electronic mail, SMS or Whatsapp messages, all asynchronous communication methods. That's nice because you

send out something at your preferred moment, and you read the reaction at your preferred moment. If you are in a hurry you grab the phone, to communicate directly, but that means that you interrupt the receiver with his regular activities. A computer system works the same way. Asynchronous communications sends a message, and a trigger returns automatically with an answer. So don't sit around waiting, but react the moment it suits you!

Queuing
When you are in the wrong line for the checkout, you may be less happy with this pattern, but from the cashiers viewpoint it is efficient. Actually, in the US the mechanism is more fair because all clients wait in one row for all the checkouts, not one row per checkout as we have in Europe. For the "gold customer" there is of course that separate checkout!

22.3. REUSE

Caching
Normally, clearing up is a very good idea. Especially if you do it efficiently. Still, sometimes, if you use something very often, for example, your phone, don't store it away, but keep it always with you. That's essentially what caching is, do not clear up, but always having it in place close by.

Share
In the context of the sharing economy, sharing is a good idea. An example of sharing is a pattern called "connection pooling", unfortunately a wrong name because it should be "connection sharing". Setting up a connection to a database for example is relatively time-consuming. Once you have set up the connection and the data has been exchanged, the connection should be closed. You can also choose to keep the connection but release it so that the next application can make use of it. You can see it as a form of

caching, so you don't clean up nicely. Being a bit sloppy has it advantages!

22.4. ORGANIZE

Sort
It hardly needs attention, but it is a very relevant pattern. You can sort out in a lot of ways. There are many different algorithms and one is more efficient than the other: Selection, Bubble, Merge, Heap, and Quick to name a few, and you can find information on the Internet about it's performance in various sites[19]. The best sort algorithm is not always QuickSort, it depends on your context and situation.

Index/Organise
If you enter a library, the books are organised in a certain way. Because of this, you can quickly find a book. Also at the supermarket, products are logically grouped together. This is efficient and in the world of IT it works just like this. By organising in the right way it costs less energy to find something back. In our publication: "How to: Develop a Blueprint for a Data-Driven Enterprise Architecture" we have included services that are focusing on storing information in such a way that it can efficiently be retrieved.

Hashing
Hashing ensures that a large amount of data is reduced into a much smaller range. If you do not need the content of the data itself, this pattern can save you energy and space.

22.5. CLEANSE

Garbage collection is a typical pattern in software development that can release memory that is not used because it will remain occupied by orphan objects that

[19] For example: https://www.toptal.com/developers/sorting-algoritms.

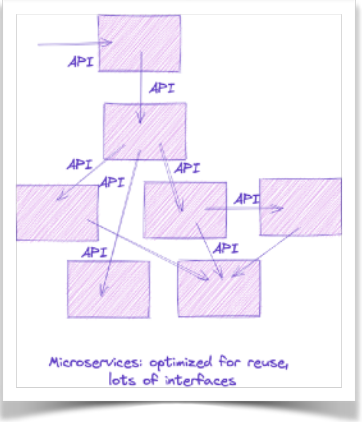
could no longer be called by the application. Also think about throwing away temporary data from the database, clean up your logging and store data no longer than necessary.
Data cleansing (lifecycle management) is a topic that we address in the domain Responsible Data Usage.

22.6. ANTI-PATTERN: PASSING ALONG

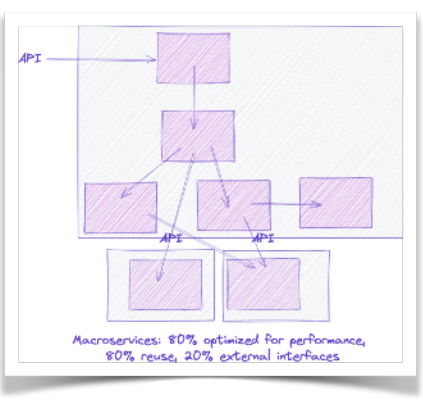
In addition to the good patterns above, there are also patterns that contain a sustainability risk. You should not implement these patterns thoughtlessly, but think carefully about the impact on energy consumption. The patterns below are about passing information from one module to another module that will return it to the next module and so on...

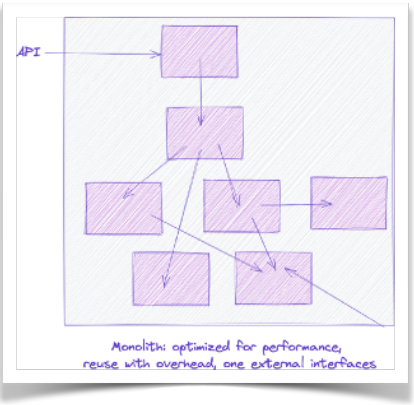
Client Server/Application Tiering
When we talk about client server or application tiering (layering) then we are talking about an application that is divided between two or more servers. In the domain Responsible Infrastructure we discussed the cost of communication infrastructure. Client server and multi-tier applications have communication overhead. The reason for this pattern is that the application is modular so that it is better positioned for reuse, is better scalable and could have different security zones. The art is to find the right balance between modularity and the most efficient communication protocol.

SOA and Microservices
Breaking up an application into components is a must.
The way in which those components communicate with each other makes a big difference. Service Oriented Architecture (SOA) services have a lower granularity compared to microservices. The higher the granularity, the higher the communication cost. The higher the granularity, the higher the reuse possibilities. Concepts like mini-services or macro services can be a good



compromise between reuse and communication overhead. Properly composing the application is essential to monitor communication costs as well as performance.





The usage of patterns tells you something about the seniority and professional competence of a programmer. Good programmers, make a lot of use of patterns. Often it is better to use a pattern that is less energy efficient in stead of writing code itself, because at the end this is less sustainable.

The second observation is that sustainable patterns positively influence the performance of a system. There is an exception to this. A pattern that ensures that code cannot run in parallel can be very sustainable but is usually not faster.

We believe that the usage of patterns should be motivated and that sustainability must be one of the motivation criteria.

23. CODING

23.1. PROGRAMMING LANGUAGES

Already some time ago, a study was performed on the efficiency of programming languages[20]. Interestingly, this study has been widely referred to recently. Unfortunately, this is one of the few studies which has been done, but the result is staggering. The difference between C and Python or Perl in efficiency, according to this study, is a factor 75!

Roughly, programming languages can be divided into two categories:
1. Compiled languages
2. Interpreted languages

In compiled languages, the code is translated into a machine language module so that the operating system performs the application directly. This machine language module can only work with the operating system for which it is compiled. Because the application and the processor are communicating directly with each other through the operating system, a compiled language is much faster and more efficient than an interpreted language. In a compiled language, the dependencies of other software libraries are known in advance. The compiler ensures that only the software libraries used are packaged with the compiled module. Therefore, the footprint of a compiled application is also smaller.

Interpreted languages use an"interpreter" that is associated with the operating system and that translates the code at run time. Every time the code is called, it needs to be retranslated. These languages are therefore less efficient. On the other hand, the same programs can be implemented in different environments for the purposes of

[20] https://greenlab.di.uminho.pt/wp-content/uploads/2017/10/sleFinal.pdf

re-use. The interpreter ensures that the operating system always understands the intent of the program. The "cost" is that there is always a need for a run-time interpreter, and it always needs to be done with an additional translation step between the application and the operating system.

23.2. LOW CODE AND NO CODE PLATFORMS

Intentional and unintentionally, low and no code platforms are being used. The idea of these platforms is that the users of these platforms do not need programming skills to develop their own applications in case of a no code platform. In the case of a low code platform very limited skills are required. In a time of scarcity in the market for good programmers, this is a message we would like to hear. The scarcity also makes it necessary to pay more for good and experienced programmers. But can we actually assume that everyone builds applications with these platforms? We are certainly not opposed to the use of low and no code platforms but a warning is in place. If it's too good to be true then it's probably not true. There are sustainability risks when using, or do we have to say abusing these platforms. The "programmers" are so far away from technology that they can't see the impact of choices on the efficiency of the generated code. How, for example, does a low code platform determine the choice for the best sorting algorithm? Therefore, we ask special attention for the importance of the knowledge of the programmer in the chapter about the people.

A second concern with low and no code platforms is the vendor lock in. Principle GC1 indicates that the code should be able to run on the most efficient platform. So it is important that the generated code should not be running within its own environment, but should be capable of running on a container platform, for example.

Now we sweep no and a low code on one belt (this is a Dutch expression translated to English), but that's not right[21]. While there is

[21] https://www.ibm.com/cloud/blog/low-code-vs-no-code

a common impact on sustainability, there is also a difference. With low code platforms, there is the possibility to custom develop some pieces. So it gives you the ability to intervene if the platform does not make sustainable choices. By no code, you will be "forced" to just take it as is. However, development capability on a low code platform are usually scripted languages , these are unfortunately not on the top of the list of most efficient languages.

Just drawbacks? Certainly not! Developing on a low or no code platform is moving faster. You save energy here in the development environment. Accelerations are promised up to 10x. But please realise that the coding phase is “just” about 30% of an application development project. The other work is in gathering the requirements, designing the application and also testing an application takes a lot of time. So even though coding would be 10x faster (than assembler?), the other activities are still required.

23.3. DEFINING VARIABLES

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

Many small ones make a big one

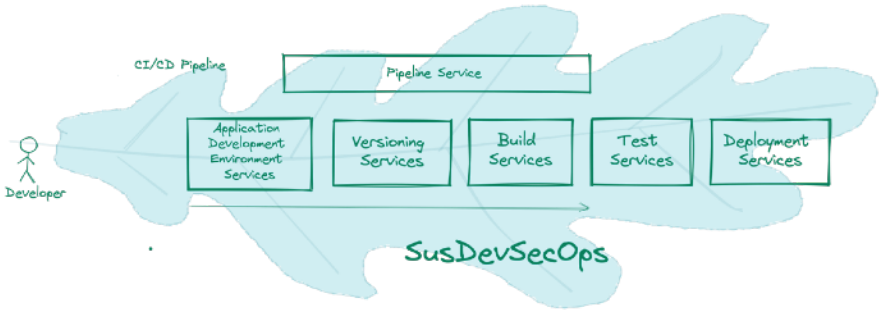
In code, you define variables such as integer, character, string floating point, etc. Depending on the definition, the computer reserves a number of bytes. Make sure you don't reserve more than you need to.

23.4. THE SOFTWARE DEVELOPMENT PIPELINE

One of the environments that is never fast enough is the software development pipeline. Developers build applications, compile these and test, adapt, compile, and test again, and this dozens of times,

day in day out. They do this preferably often and fast because time is money and waiting for a compilation to be able to test is annoying. Development platforms are energy-eating-monsters.

Let's do a little analysis. Below we have outlined a basic chain of tools:



SusDevSecOps

If we look at the sustainability aspects of the toolchain, we have several observations.

First, the pipeline services. This service is the director of the entire chain. A sustainable director is not present when it is not necessary. The director will only show up when it is needed and build the chain at that time. Serverless is a concept that can be used for implementing a sustainable director.

Next is the application development environment. This is the environment were the developer builds his application and runs unit tests. The developer prefers 32 cores rather than 16 and prefers 128 Byte of RAM than 64. If you see the specs of a modern development machine you wonder if it is required for development or if it is required for playing 3D games in the weekends? During the build of an application the laptop or desktop will turn into a heater, nice in

winter, but in the summer... These costs are often not visible because the programmer prefers to work locally on his or her computer.

If the programmer is satisfied, the code is placed in a repository. That costs little CPU, only bandwidth and some storage. Maybe it's a good thing to clean up a little bit and not keep your old code forever?

Up next, the build service. All the code of all developers are raked together and then the entire system is build. Now you can heat up an entire house with the energy that is consumed! Can you possibly do this during the night when the developers are asleep? We can then use scarce resources that we do not need at that time.

Of course, the code must be tested. A regression test, but there are also other tests possible like performance testing. In a SusDevSecOps toolchain, we test on sustainability, of course. Tests may also be performed prior to the build phase. Think of automated code inspections that check whether applications are safely and efficiently programmed.

Our application is ready, now we deploy it into production with the deployment service. Of course you automatically deploy the application in the most sustainable environment, right?!

Although we do not go into every detail and every ability to optimise the pipeline, the intent is hopefully clear by now. And especially when you look at the entire pipeline don’t forget: many little savings make a big one.

One last thing we like to mention. It is very good to measure the amount of energy used in the pipeline. The initial measurement results may not have an absolute meaning, but they can indicate a trend or be used to compare applications. What if the construction energy costs of release 3 to release 4 doubles? We must explain that.

It is also possible to establish relationships between energy use in construction and in production. Maybe make a nice Machine Learning model of it?

24. PRODUCTION

The development of applications costs energy. Most of the time, this is an activity that takes place in a period of a few months to a few years. Under application development, we also include the application maintenance. It is also the intention that the application will be used, hopefully for years and preferably by as many users as possible. Then the energy meter is really going to run.

24.1. MEASURING IS KNOWING

The best way to make good comparison is to measure the energy consumption of an application over a period of time. For applications, however, that is not so easy. The old hands among us who have worked on a mainframe still know it. On the mainframe, the usage was neatly measured by application and associated cost could be passed on to the department. The mainframe still has a built-in measurement system and reports about its usage. The resource usage of a mainframe is considered expensive but in the context of sustainability, we must regard all computers as precious.

Fortunately, more attention is being paid to measuring the sustainability of applications. Just think of your smartphone that indicates how much data an application is using or how many internal memory. Why? Exactly, if you run out of your data bundle, you have to pay additionally, therefore it's precious. How many people have not been in the process of measuring their energy use in 2022 when energy prices quadrupled? Do you have a MacBook, then you can see in the "energy tab" of your "activity monitor" how much an application used and how much it has used in the past 12 hours.

Since the use of AI libraries is relatively expensive, *hugging face* has created a tool[22] that measures CO2 emissions and they are not the

[22] https://huggingface.co/blog/carbon-emissions-on-the-hub

only ones. Another tool is *Turbonomic*[23], this tool measures the CO2 emissions during operations, reports this and gives the opportunity to move your application to a more sustainable platform. So definitely worth to look into these tools to make calculated sustainability tradeoffs. We expect that in the coming years the pressure from the CSO to measure applications for energy consumption will increase.

An organisation must ensure that a framework is created by which application usage is measured. In this way, you can see where the large consumers are and by defining objectives you can save up to 5% of CO2 emissions year over year. If you go to the cloud you have to measure it, of course. A typical example of a large user might be your Active Directory environment. It is not possible to run Active Directory on a different environment than Windows on X86. Quite a restriction considering a Linux/RISC environment could save 50%. This is just an example; the message is you need to avoid platform lock in when you choose your applications and software.

[23] https://www.turbonomic.com/sustainability-calculator/

25. ARTIFICIAL INTELLIGENCE MODELS

Building AI models is a specific branch of sport. For building models a large amount of energy is needed compared to the applications build by developers we discussed in the previous chapters.

In addition, we have to deal with the ethical aspects of AI in the construction of models. These ethical aspects also apply to the application developer, but business rules are more easily explainable than a model that is trained with large sets of data. Data scientists are large users of Python and R. In the field of sustainability the most 'polluting' languages, we discussed earlier in previous chapters.

The sustainability aspects that we described in the previous chapters on developing software also apply to AI models.

In the chapter Responsible Data Usage, we describe, among other things, how we deal with privacy aspects. In Responsible Systems we describe the ethical aspects of systems. We therefore limit ourselves to a few subjects which are not covered in these chapters but can be relevant in the context of sustainability.

25.1. MODEL TRAINING SERVICES

The model training service is a service that we also describe in our first orange book. It is a batch process to create AI models that maximises resource utilisation such as GPUs. In the context of sustainability a strongly recommended service.

25.2. REAL TIME INTEGRATED ANALYTICS

Perhaps a bit strange in the context of sustainability. The idea of real time integrated analytics is that during transaction processing, an instant analysis is performed by combining a GPU and CPU on one

chip. The best known example is the M1 Chip of Apple. Because GPU, CPU and memory are on 1 chip with an operating system that is optimised to utilise those capabilities, it is up to 3x faster or more economical than a traditional Windows laptop. Windows laptops can only run on X86 processor technology. As you have been able to read in the chapters on Responsible Infrastructure, this technology is starting to lag behind in terms of sustainability and we are gradually seeing this combination disappear from cloud data centers.
Although real-time integrated analytics delivers huge profits when it comes to sustainability you have to do with a mandatory combination of CPU, GPU, Memory, Operating System, and AI Libraries. Optimal alignment between all these components is achieved, but this comes with the price of lock-in.

26. RESPONSIBLE CODE PRINCIPLES

In this chapter we give a number of sustainability principles for developing and running your application. These are just some examples and you will hopefully recognise those, because those principles are described in this chapter.

Principle GC1	**Code must run on the most efficient platform**
Statement	Code developed by your own organisation or purchased works on any platform, preferably in containers.
Rationale	Applications are more efficient on, for example, a RISC platform than on a traditional platform. The application should be platform independent in order to easily move to the most efficient platform. It also prevents vendor lockin.
Implications	It deserves additional attention to build an application independently of the underlying technology.

Principle GC2	**OTAP environments are available as code**
Statement	Whether it is a development, test, acceptance, or production environment, the environments will be deployed from code so that it can be discarded after use (immutable infrastructure).
Rationale	Environments use scarce resources such as CPU and memory. Environments that are available as code can be built, used, and discarded again. Environments are temporary and will only be built when it is needed.

Implications	This means that environments are developed as code. "Infra as code" is the slogan that is used. This requires specific knowledge of infrastructure specialists. Tools such as Ansible can be used for this purpose.

Principle GC3	**For each component, the sustainability factor is described**
Statement	During the modeling of components, sustainability (energy consumption and reusability) is considered as a requirement.
Rationale	During the design phase, consideration is given to the sustainability aspects of components. Per component, something is said about the reusability and the energy consumption.
Implications	It deserves additional effort to build an application independently of the underlying technology.

An example of the additional information from a component description could be as follows:

Component:	**Printing**	
Non-functional Requirement	***Specification***	
How often is the module called?	Generic module that is called 100,000 times a day.	
How necessary is the data?	Medium	
Development Language	Java (Compile)	
Pattern Identified (Function)	**Proposed Pattern**	**Comments**

Sorting data	Quick Sort	Large quantities of random data
Interface	API	gRPC usage
Datatransport	Asynchronous, Batch	Once every five minutes, delta's only.

Principle GC4	**We measure energy use**
Statement	During construction and the run of the application, CPU (and GPU) usage is measured.
Rationale	These do not need to be absolute numbers, but may be relative numbers to create awareness and a culture that realizes that building and running software costs energy.
Implications	Extra code must be written to measure energy consumption. One proposal may be to save 5% year over year. Results should be (centrally) collected.

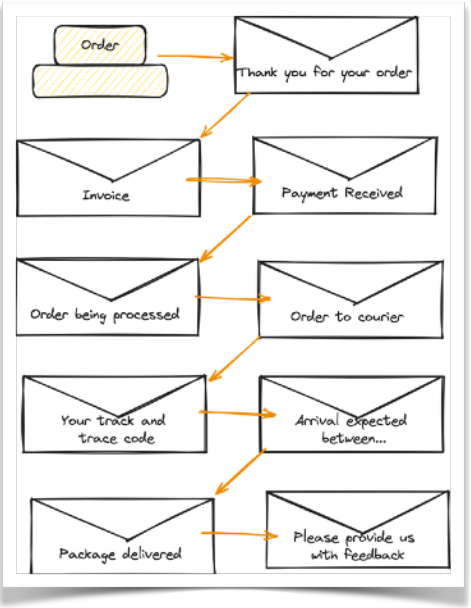
Principle GC5	**Include energy usage when deciding Build vs Buy vs Reuse**
Statement	During the selection of Build vs Buy the TCO should include energy consumption over the total lifecycle of the application.
Rationale	Part of the TCO is energy consumption. Applications that are purchased are generally more efficient. On the other hand, the generic usage of a Buy application can result in an "overkill" of unused features which can have a negative impact on runtime sustainability.
Implications	Suppliers must be asked for the energy footprint of the provided application. Suppliers have to hand over policies to develop their applications in a sustainable way.

RESPONSIBLE DATA USAGE

Responsible data usage is about the sustainable and ethical use of data. Sustainable does not have anything to do with the longevity of data, the amount of time it is kept in an organisation. It is about developing a responsible way of using and storing data and eventually disposing of it. This is only possible if we are aware of what kind of data we collect, produce and what data we (may) use.

27. INTRODUCTION

A day in the life of a seemingly mundane activity:
You are sitting at your desk, at half past 9 with your first oat flat white (it's 2023 after all).
Ready to first of all clean up your mailbox. Perhaps something you do without too much thought? Lets pause, and reflect instead.



There can easily be tens of messages in your mailbox, how many messages do you seriously keep or read? Take, for example, an order for a gift for Sinterklaas (Dutch kids festival, based on Sint Nicholas who celebrate his birthday on December 5th, comparable in many countries with Santa Claus). Within a short time you receive nine e-mails regarding your order.

Unfortunately, we all often receive multiple e-mails per order, although not always 9 pieces. But even only a few, with all the orders we place collectively, adds up considerably.

Lets pause and consider what happens in the background and while you are at it, take the time as you are reading to reflect on the data that is generated and stored. A quick view; an entire logistical process to schedule people on the work floor (data), the workflow required to receive the ordered product wrapped as a gift (data), the

communication between courier company and supplier (data), all those vans (electrical or not) that are planned (data), the processing of the package (also data). Finally, the infrastructure that those systems use (on-premises and/or in the cloud)... All these activities use IT systems that in turn consume electricity and produce data that must be processed in a responsible manner.

A simple action triggers a domino effect

All that data can subsequently be used for several purposes (provided permission has been given, at least we hope). For example, in the form of training datasets for 'cross- and up-sell' offers for mail number ten, after all, Christmas is coming!

Let it sink in, that this is only 1 order. PostNL alone delivers an average of 1.1 million parcels per day. And around the holidays (peak period) that is 70% to 100% more[24]. Just imagine if we could reduce email communication by 10%...?

In 2020, 335 million online orders were placed in the Netherlands. If you followed the transaction pattern, these transactions would result in 1.6 billion emails. It is just a simple example, but these e-mail messages are stored by the sender (the webshop) as well as the receiving party (the buyer), so 2x1.6 billion!

Sinterklaas has now left the country again, but what about the data in the systems of all these orders? Will the data also be deleted or will this data still be used in all kinds of cross- and up-sell activities?

---

[24] https://www.transport-online.nl/site/148079/postnl-ziet-geen-afname-pakketbezorging-ondanks-hoge-inflatie/

27.1. ARE WE ALL HOARDERS?

We dare to make the statement that most organisations collect more data than necessary for their operation. Also, that most organisations are not yet ready to remove and destroy data when they should.

A data hoarder; being consumed by data

You could say that by simply adopting a minimisation principle, you no longer have to collect all data. By definition you would be more sustainable and with less data, data management also becomes easier. But unfortunately, the world is not that simple. Because what if there is something magical hidden in that pile of sand? It is therefore tempting to keep as much data as possible. How do you make the choice between throwing away or keeping?
A few years ago the term “dark data” was coined. Gartner defined it as follows:
"the information assets organisations collect, process and store during regular business activities, but generally fail to use for other purposes.”

In our words: a huge mountain of data that has never seen the light and most likely will not lead to 'enlightenment', despite all the beautiful promises and attempts, and energy that is expended to keep this data available.

What if it contains some interesting information, somewhere? In the cases of road obstructions, finding causes more quickly for holes and therefore locations in the road, on the 'A-2'. By linking blowout reports from the ANWB to roadworks from Rijkswaterstaat (Ministry of Infrastructure and Water Management), the cause can be quickly found and the hole in the road can be closed.

These are hypothetical examples of data analysis. But does this mean that we should collect as much data as possible en masse?

Delete unused or old data that has no foreseeable use, unless you are legally bound to keep it!

Responsible data usage is easier said than done. With regard to this domain, we will first discuss the aspects of a data strategy and what we need to do here to apply Responsible Data Usage. Next, we look at resources we have at our disposal to implement this sustainable strategy. We call this our toolbox. The strategy and toolbox are relevant to all data. In the chapter after the toolbox, we discuss a number of aspects of data that are mainly related to AI. We conclude with the subject of operationalisation. This is an evaluation of data projects at the moment when data analysis is no longer part of a sandbox, but the business wants to make serious use of it.

28. A SUSTAINABLE DATA STRATEGY AS A STARTING POINT

In this chapter not all aspects of a good data strategy will be discussed, enough has been written about that. Particular focus will be paid to the aspects that, in our experience, require more attention in order to handle data responsibly. This is an important part of the evolution from consciously incompetent to consciously competent in the context of Responsible Computing.

Step 1: having a data strategy at all. And not a few abstract slides about the importance of data and how it can lead to insights or whether investments are made in hiring data scientists, etc. You might think is rather blunt or expect the opposite to be true but even some large organisations lack a proper data strategy. In essence a data strategy must be simply there for every organisation and should be widely supported and understood and developed 'by design'. The chief is the CDO who is at the helm and directs the organisation. Responsible Data Usage is an important spearhead in this data strategy. A maturity model can help you develop a data strategy[25]. This also helps to design and describe the principles for a sustainable data strategy.

Step 2: the data strategy must be concrete and widely supported in order to be able to use data as a resource in business operations. You can achieve a supported strategy by involving a broad group in the development of your data strategy. It is precisely by emphasising Responsible Data Usage (amongst others areas for storage, how data can be used, ethics implementation, how access is taken care of, deletion of data) that you win the trust of those involved and you prevent that the strategy development becomes a party for the data

[25] https://ictworks.org/wp-content/uploads/2019/11/Responsible_Data_Maturity_Model_10-16-19.pdf

strategists. By involving specialists in the operation, you prevent the data strategy from remaining too abstract. It also becomes clear what the impact of the use of data is on, for example, a process or even on someone's life.

Step 3: Make sure you have a good investment plan. A strategy only works if resources (technology and people) and owners are linked to it.
In the beginning, the focus was too much on tooling for solving data issues, and therefore too little on setting up data governance and data management.

Responsible Data Usage requires knowledge of the existing data in the organisation, data modelling, assigning ownership, defining quality requirements and determining origin.

Step 4: the data strategy must be widely embraced by the organisation so that people realise what it means to use data to make decisions in business processes. Where to find data that they are allowed to use and are aware and trust that the data is used according to specific ethical standards. This is another step from consciously incompetence to consciously competence.

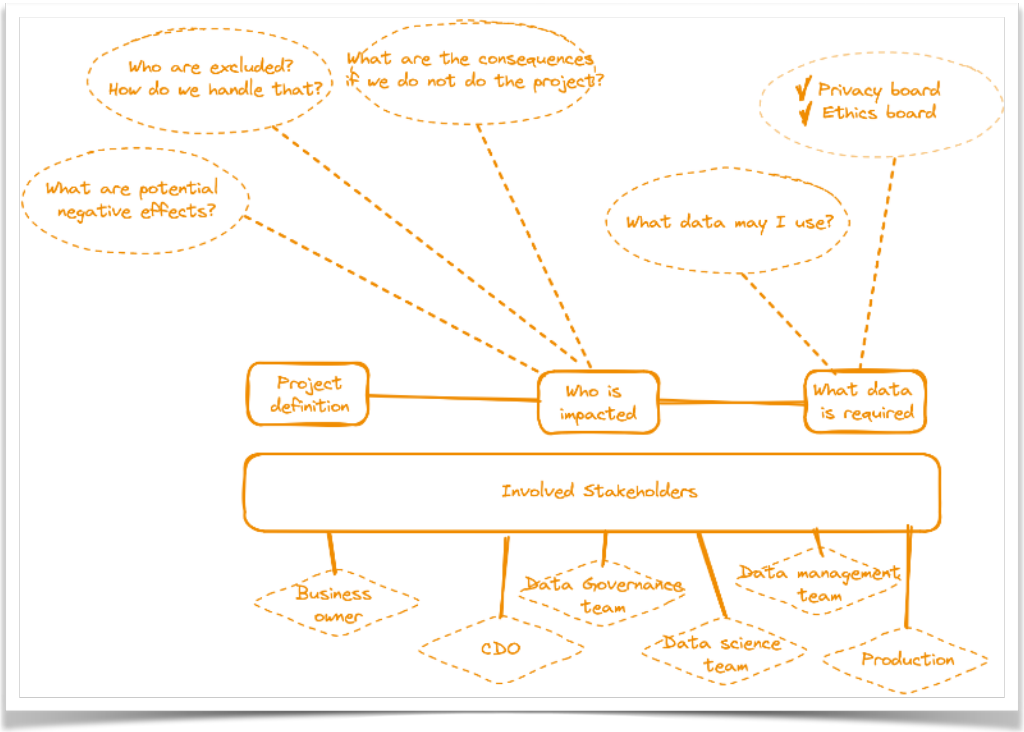
Privacy aspects
In relation to the GDPR and thus the protection of personal data, governments talk a lot about “purpose limitation”.

A definition used by NORA, which provides architecture guidelines for the Dutch government:

“data is collected and ultimately processed for a specific, explicit and legitimate purpose”

Data use within a process must be done ethically and transparently and based on the consent of the data owner. Hopefully, dashboards or portals will follow in the coming years where you, as a customer, can see at a glance what an organisation is doing with your data.

Environmental aspect
Data also has a huge impact on our environment. The right design choices within the data strategy are a foundation for the efficient storage, processing, retrieval and deletion of data.
There is still much to be gained in this area. Think of data storage on tape, reducing data transport through virtualisation, properly classifying data and metadata of data so that the search and find process is faster (less load on your CPU) and, of course, applying good lifecycle management.



A multidisciplinary team should ask many questions

Checklist

Here you will find a checklist with questions that can serve as inspiration for developing a sustainable data strategy:

29. THE TOOLBOX

When we talk about our toolbox, we refer to all the resources we have at our disposal to use data:

- technical tools such as a reporting tool, business intelligence tools, but also storage of data, for example an SSD disk or a tape.
- processes and procedures such as Data Management and Data Governance. Data governance (the guidelines and procedures) and data management (the processes and tools or the implementation of data governance).
- Cataloging of data, data cleansing or masking solutions.

On the one hand, we want to handle the data in an ethically responsible manner and, on the other hand, we also want to be efficient when it comes to energy consumption and the CO2 footprint.

29.1. RESPONSIBLE TECHNICAL TOOLS

Focus on the newest of the newest

Perhaps recognisable for some of us; children's fascination and focus on new toys in the playground or at home (and also some grown-ups among us). The latest of the latest is interesting to use and that sometimes leads to conflicts. Because even if you take Sinterklaas' bag full of new toys with you, a new release that another person is working on or talking about, wow.. I want that, that must be the answer to my challenges. As a specialist, it is a healthy characteristic to be interested in the newest and the latest. This also

applies to data specialists. Data specialists love data and even more so adore the tools that can extract value from data. Whether it concerns a reporting tool, a BI tool or an AI tool. New is interesting, but not everything new is usable for your purpose.

Do data specialists have a specific preference for a tool?
What tools do you like to use? One prefers to use the screwdriver and the other prefers to jam with a hammer and nail. In both cases a board can be fixed, but the question is for how long? P.S. There are also those who hammer in a screw, but we will leave that category aside for now.

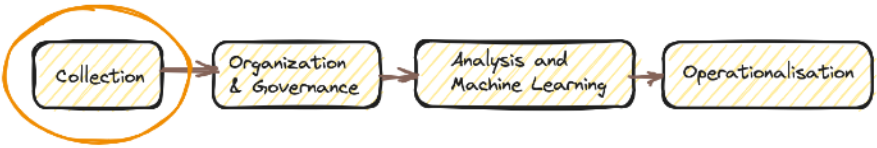
Enough tools have been developed in recent years, a number of which are described in the first orange booklet about data-driven organisations (from data to information to insights).
Competition for tools has always existed. The trick is to use the right tool for the right problem. This means that the tools should not be leading, but the results we want to achieve.

29.2. POINTS OF ATTENTION FOR TECHNICAL TOOLS

A limited number of points for attention will be discussed below to make responsible and sustainable use of all those beautiful tools (otherwise we need a bookcase full). We discuss these points, based on a simplified data analysis process.

To be able to do anything at all, data is needed (collection), which requires sound management (organisation & governance), the fun work can then start to perform analysis and develop models (analysis & machine learning), with operationalisation as a final step.

Collection



The noun “collection” is deliberately used here and not the verb to collect. This refers to the earlier section on collecting rage.

A data collection is the total collection of data from a company or organisation. This also includes: what data do I have/have access to, what data do I need and may I use, where does the data come from, for what purpose was this data produced, what do the fields mean, what about the data quality?
This can be supplemented with the data principle:
do I stick to using as less data as possible, by sincerely expanding my collection with only what I need?

Security
In addition, security for the collection plays an important role; data security can not be emphasised enough. How and what data can be made available to third parties. To another department, another organisation, or to a customer; based on their wishes and requirements, which can lead to permission for data use.
There are access management, encryption and pseudonymization techniques to achieve this. Or how about building a model, based on encrypted data from different parties, as they are not allowed to see each others data? With homomorphic encryption this is exactly the case, the *processing of the data* is encrypted. As a result, the encrypted source data does not need to be decrypted and the encrypted data can safely be ‘brought together’ and used.

For the majority of use cases though, sometimes one department may access certain data, but this may not apply to another or even an individual (role) within a department. Or maybe data may be

used, provided it cannot be traced back to whom an analysis is concerned?

Sustainable storage of data
In terms of sustainability, various technical solutions regarding data storage should not be forgotten;

- *deduplication*, using 1 truth and eliminating too many data copies to control data storage capacity
- *auto-scaling*, dynamically adjusting the environment based on the desired demand (CPU, memory, network)
- *storage tiering*, a solution in which data is stored on different types of storage devices
- *over-committing*, allocating more virtual capacity than is actually physically available
- *storage virtualisation*, bringing different storage media together virtually as a single storage medium
- *compression*, binding your 'data soup'

Sustainable availability of data
We can also use sustainable technology when making data available:

- *data virtualisation*, the virtual bringing together of data from different sources
- *remote data processing*, process data at the source and only send the results
- *optimisation of queries*, determine the most efficient way to query relational databases and other databases (e.g. NoSQL and graph databases)
- *location of data* including analysis environment, with regard to the latter, we could learn a lot from edge implementations and analyse data streaming, so to speak

Data virtualisation highlighted
Both technically and physically, using data does not have to mean copying it to your environment.

A durable and powerful tool to provide access to data is relatively new, but already widely used: data virtualisation.
This makes it possible to bring data together by unlocking, transforming and integrating it from multiple sources without physically bringing it together. You can use the capacity of the underlying technical resources that provide a view for your purposes.

Organisation and Governance

Data catalog
The core of a catalog is to provide access to data. This allows data to be under the control of the organisation; we know what kind of data is available, the meaning, and how we can use it.
Data can be trusted and complies with laws and regulations through predefined rules and policies. It provides clarity about which people or systems may or may not have access to certain data. The catalog can also mask data depending on the user requesting the data or the use case. There is much more to say about this, there are several sources that provide a great overview[26] (we obviously think this is a great reference).

This means that a catalog can describe both structured and unstructured data, as well as machine learning models. Metadata plays an important role as it makes it possible to find, profile and enrich data through an interplay of technical, process and business metadata. It also plays an important role in compliance and data quality, but also in managing the life cycle of data. Finally it also indicates which data is no longer relevant or may be used.

---

[26] https://www.ibm.com/topics/data-catalog

It remains important to realise that technology such as a data catalog is not enough. In addition to a data strategy, solid internal processes are needed and a culture (change) is needed within an organisation to use data in a responsible, ethical manner.

No matter how big your team is, even if you have an army of data engineers and stewards, meeting all data requests and ensuring that it takes place responsibly is an impossible task and also inefficient, IT was invented for a reason.

That's why self-service is important. To make this technically possible, the data catalog is essential.

Monitoring and managing the Data Collection
Monitoring data collection and use is also an important task. Dashboards that provide insight into what is viewed and used, by which organisational units and by whom. This applies not only to the data, but also to the models and the systems that use the models.

Analysis & machine learning

Everything we have described so far in the Responsible Data Usage section is about arbitrary use of data. It does not matter whether the data is used to create a report or whether the data is used to profile potential customers for marketing purposes. Ethical handling of data applies to everything and especially to PII (Personal Identifiable Information). In this chapter a few additions relevant for AI will be provided.

Shared responsibility
Suppose we assume the most favourable case: a mature organisation that has set up data governance and data

management. Data scientists have access to data that they are allowed to use for their analysis and they can trust that the data quality is in order. No matter how good the data quality is, it is always possible that fields in a database are not filled properly or are incomplete. It is important that this is clear to the data scientist so that he can take this into account when considering the influence on the analysis.

The data scientist has an important responsibility to keep a close eye on data quality. We have also written about this in the chapter on the sustainable specialist and the behavioural manifesto.

By way of illustration, if there is a field that may not be used for ethical reasons or because of purpose limitation, then the alarm must be sounded. Whether it's with the data steward, a data privacy officer or an ethics board, it's important that there is a process for reporting this. It is also necessary to have a plan in the unlikely event that things go wrong and it turns out that data has been used improperly.

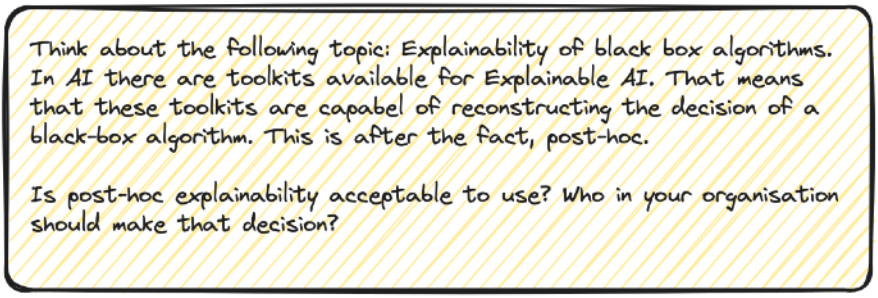
If you can derive information to which you do not have access through combinations of variables, well, we do not need to finish that sentence... A very simple example: take gender, if you can deduce that for part of the population because you have a variable with maternity leave..

Transparency, also in Data Transformations

Data transformations and manipulations to develop a model must also be recorded in the context of transparency. You can do this by using a data catalog or an AI Factsheet. An AI Factsheet[27] is, can be compared to a recipe that shows how a model was created; the data

[27] https://aifs360.mybluemix.net/examples/hmda

sources used, how the data was transformed, the training set, including the locations, the fields, the model itself, etc.



Explainability of a black-box algorithm

Sustainable Training Data

Getting to the right training data can be a chore in itself, even if data management is in order. In the context of sustainability, we add a task: challenge the organisation to arrive at a dataset as sparingly as possible.

Use a smaller set, because that saves resources, and every little bit helps. Some researchers stated that the training of a large AI model in the Natural Language Processing field can equal the carbon emission of 5 cars. In their lifetime that is[28]. A data scientist also has a responsibility in both the approach and the use of resources, with the nice advantage of performance gains, but also the explainability of a model. Especially when it comes to resource-intensive algorithms. It would have been nice if we could provide concrete tips here to arrive at an optimal dataset, but unfortunately it is not that simple.

---

[28] https://www.technologyreview.com/2019/06/06/239031/training-a-single-ai-model-can-emit-as-much-carbon-as-five-cars-in-their-lifetimes/

Explainability
Once you have created a good training set the modelling can begin. How do you ensure that you can explain the results? After all, handling data responsibly also means that results can be explained. There are plenty of algorithms that can be explained. But there are just as many that cannot: black-box[29] algorithms.
Depending on the case and the area of application, it may or may not be acceptable.

In addition to Explainability, which ensures trust, there are other indicators that are important in using data responsibly to gain insights and thus make decisions.

Fairness
Fairness is about ensuring equal treatment and preventing bias. This may or may not be consciously hidden in the data or in the algorithms that are used.
This is not only about the technical approach to preventing bias, but also affects the organisation and the cultural aspect. For example, a dedicated team specialising in diversity can assist in the development process and help ensure stakeholders are involved in the design phase.
Here too, various algorithms are available to check whether the training data and the model are fair. One of the possibilities to envision the working is like flipping a coin; the males in the dataset, are switched to females. If this produces an opposite results, depending on the case, the model might be biased.

Robustness
Another branch of sport is aimed at making algorithms 'hack proof', so that the outcome cannot be changed by outside influences. This is important for both the training data and the models themselves,

[29] The user cannot see what the underlying reasoning is of an algorithm. Only the input and output are known which makes it uninterpretable.

so that other outcomes can be prevented through manipulation.

Operationalisation

For real this time!
That's what we used to say when we played marbles. You had a fake game, then everyone got their marbles back and a game for real, where you lost your marbles or you got the other's marbles if you won. Exciting because something was at stake. The same is true with the proper use of data. We are talking about real personal data. Jan van Vliet from Vlaardingen is no longer a made-up virtual person but a human, all flesh and blood, if you squeeze him he says "ouch". That's when things get exciting and can have a profound impact!

Organisations that have been on a journey with analysis work for a long time, may have many projects that do not progress beyond the Proof of Concept phase with simulated data, will have to invest more time and energy in data management. Data management is a field that deals with the sustainable use of data. If you know what data you are talking about you can start playing for real. It is a precondition to bring the experimental phase that the analysts and data scientists are working on into a production-worthy system. This can be done practically by including data management supported by data governance for the project from start to finish.

You don't just focus on the beautiful models, but are forced to first think about the effect of the (ethical) use of the data as well as other non-functional (sustainable) aspects. This may lead to a side effect of cooperation with various layers and roles within an organisation, which probably leads to a higher success rate of the promotion of PoC to a successful production system.

30. RESPONSIBLE DATA USAGE PRINCIPLES

RDU1	**Data is accessed and not copied**
Statement	We collect data from the source and do not copy everything.
Rationale	Copying data has a number of disadvantages: – Data actuality. A copy lags behind reality. – Transport and storage. An extra copy is extra storage and transport. – Risk of inscrutable chains. What if a copy of another copy of the copy is made?
Implications	The consequence is that we have to, for instance virtualise the data and realise a connection to the data that meets requirements such as security and latency. An exception process needs to be put into place.

RDU2 Principle	**We conform to the data strategy**
Statement	There is a data strategy, it has broad support and we stick to it.
Rationale	We ensure that there is a data strategy that promotes sustainable business interests and is also feasible at the same time. Individual teams prioritize public interest over individual success.

Implications	Meeting a long-term strategy usually means making additional investments: - Setting up a Data Management team - Setting up a Data Governance board

Principle RDU3	**Data Minimisation**
Statement	We adhere to a principle of minimum storage and retention of data. If in doubt, discard and do not store.
Rationale	We trace the data we want to keep from our data strategy. So if there is no strategic reason or legal reason to keep data, we throw it away.
Implications	Maybe we throw away too much and miss an opportunity to increase sales. Too bad, sustainability and manageability come first.

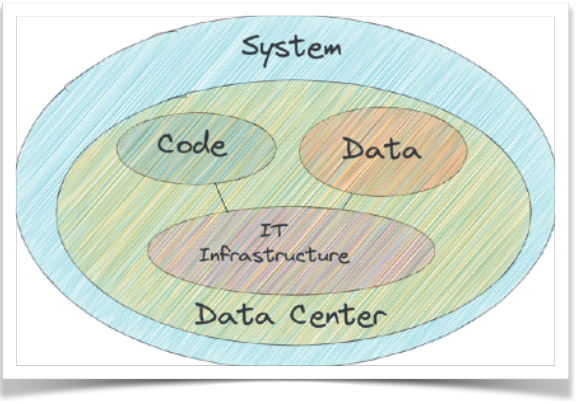
RDU4 principle	**All data elements have an owner and are in the corporate catalog**
Statement	We ensure that the data we work with is recorded in the corporate catalog with all information and ownership that goes with it.
Rationale	We have control over the data we work with. We don't let things happen to us, but we steer them.
Implications	We need administration and a tool to make sure this is done properly.

RESPONSIBLE SYSTEMS

What do we mean with Responsible Systems? By a system we mean the usage of technology to support the users of that system. Well that could be anything, right? An example could be an advanced document processing application, where the system reads metadata from a document and stores the document and the metadata in an electronic archive so that it can be found back when necessary. That support is not provided so that the users of the system have more spare time, but is provided to increase their productivity so that they can do more during the same working hours. On the other hand, we have systems that can be used by customers to give them more and better information...

The system mentioned above consists of code (the scanning software), data (metadata and document), middleware(database) and infrastructure (scanner and server) and the software most likely runs in a data center. The combination of code, data and middleware we call an application. A system can consist of multiple applications running somewhere on a piece of hardware. Traditionally we developed one large application, but we have realised that this is not convenient in relation to maintenance and life cycle management.

The “code” of the application is the mechanism that takes input data and convert that to output data based on mathematical calculations.



Responsible Systems

But who determines what the calculations are that are used? What is the input data that we use? What is the logic that is applied (e.g. business rules) and what are we going to do with the data? A lot of responsibilities for the designers of an application.

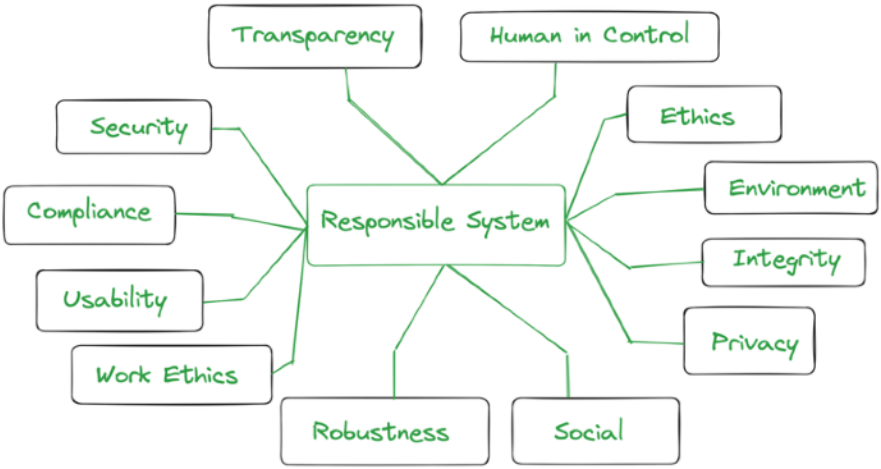
Responsible Systems bring together all the previous domains of Responsible Computing. The big question is: are we going to use our system in a responsible way? What will be the purpose of our system? Do we take into account the environment, the humans involved (both the designers, users as those eventually impacted), ethics? In other words, do we apply the usage of technology in a sustainable and ethical way?

31. RESPONSIBLE SYSTEM ASPECTS

As alluded to in the introduction above, Responsible Systems is about responsible usage of IT systems. This encompasses several aspects. In the chapter about the employee, we gave an example of how social aspects are part of a Responsible System. The rise of the Internet offers the opportunity to unlock systems to all kinds of end users. Do we take into account the diversity of end users? The impact on users by a system needs to be taken into account, those systems that you “have to” use, for example payment systems. The same applies to systems that are provided by public authorities. Think about your tax application or applying for a surcharge. There are people with disabilities, labor migrants who have a language barrier, people who struggle with learning and so on.

In addition to social aspects, we also look at the security of the system, especially now that a lot of systems are exposed to the outside world. What is the damage if the system is abused? We are also considering the impact of the IT system on the environment. What about the energy consumption and CO2 emissions of a system? In the attached figure (Responsible System Aspects) we have presented the various aspects of a Responsible System. As mentioned above, there are a lot of topics to consider:

- **Security**. The system is in compliance with security requirements. Secure by design.
- **Transparency**. It provides insight in how it come to an ‘insight’; it shows the data and computational rules that the system uses and in what context. We have worked out this aspect in more detail in this chapter.
- **Human in control**. IT systems can process mountains of work at high speed and can become very accurate. Yet we find that people should be able to intervene and overrule the decisions of IT systems. This aspect is worked out in more detail in this chapter.



Responsible System Aspects

- **Ethics**. Ethics is also called moral philosophy. What do we find acceptable and what not. This will also be worked out in more detail later on.
- **Integrity**. An integer system is a reliable system. This system ensures that cases are handled in a timely and comprehensive way, and that the person performing mutations in the system is empowered to do so.
- **Environment**. What is the impact of our IT system on the environment? You might think that this is going to be easy, but are you really? Did you actually spend the time to figure out what it takes, measure it and realise what can be done to make the system more sustainable?
- **Privacy**. Privacy legislation for Europe has been described in the GDPR. In particular, this legislation handles the usage of data from individuals. This is only allowed with the consent of the person and can only be used for the purposes for which that person has given their consent.
- **Social**. The impact of our system on society. Think about the use of digital video conferencing. This saves fuel and prevents CO2

emissions. But also, might a system be in favour of certain individuals or segments of a population?
- **Robustness**. This is about making sure that a system is resistant to manipulation and ensure that the system is robust enough to survive the evolution of IT technology, while preventing technical debt.
- **Work Ethics** A Responsible System is designed to take into account the fun of a user' s work. This is also worked out in more detail in this chapter.
- **Usability**. A Responsible System is designed to take into account the abilities and disabilities of the end user. Systems must be intuitively designed.
- **Compliance**. The system is demonstrably compliant with law and regulation, but also meets the requirements set by the company IT systems.

31.1. A FEW OF THE ASPECTS IN MORE DETAIL

Transparency
For the sake of the length of this chapter we will zoom in on AI. For traditional systems (applications), this is quite clear. What data and which rules/algorithm(s) have been used to make a calculation?

Several applications nowadays use one form or another of algorithms, Artificial Intelligence (AI) systems.
These systems are based on smart algorithms (set of instructions) that are 'trained' based on a set of data. The algorithm learns to recognise patterns that allow faster and better recognition of provided input. For example recognising what is on a picture.
These new types of systems can provide an acceleration in solving technical and societal problems, as long as they are well applied.

When it comes to systems, there is an important difference between traditional systems and these types of systems. As mentioned earlier, traditional systems make use of business rules.
An example of a business rule is; if a customer is under 18, the customer will not be able to apply for car insurance. A quite simple one. In the use of complex systems, it is very different. Just imagine

a chess computer, with thousands of computational rules; all moves and combinations of moves have a predictable outcome. So you can be held accountable afterwards about the decisions made during the chess game.

Getting back to the pictures, if you throw 100,000 pictures into a system, each picture is labeled to indicate whether a traffic-light is on it or not. You generate a model based on this information. You implement this model in a system that you have called "traffic light explorer." Now you're going to get a different picture. On that picture there are three balls, a red, a green and a purple one and you let the system classify this picture.
The system indicates a traffic light! But on what basis did the system make its choice? That may be obvious: the colour combination of the balls?

Simple business rules are therefore quite transparent, for complex systems like a chess computer it is already more difficult but the application of AI becomes a real challenge when it comes to the transparency of a Responsible System. Let alone for Large Language Models that are a hype.

Human in control
If we talk about AI, the general explanation for this acronym is Artificial Intelligence. However, companies, including IBM, prefer to talk about AI in the sense of Augmented Intelligence. This means that the system is in support of a human. It advises humans on decisions, it shows the certainty the system has in it’s answer(s) and the human takes this information and remains responsible for the decisions.

A system, whether it's AI or traditional, may be faster, more efficient and more accurate than a human being, but it lacks a few key human aspects: feeling, nuance, and what about empathy or a conscience?
Through our sustainability foundation, we are trying to develop the system in such a way that it complies with the ethical standards that

a company has, as a system otherwise works, thinks, charges and acts binary. We are not yet in a position to develop a system that has a “fingerspitgefühl” nor can we or should we want to develop a system with a conscience!

When a system is being developed for facial recognition and it is used to detect missing children in mass crowds, it is a system that is used for noble purposes. The same system can also be (mis)used to profile individuals in the same human mass who meet a 'suspicious profile' on the basis of, for example ‘unusual’ behaviour. It selects people that are limping as suspects but there is no relationship between limping people and criminal behaviour. Even if there was, does that justify the invasion of privacy?

In the end it should always be a group of human beings who determines whether the application of the system is ethical, before even embarking on the development.

Ethics
Would you be willing to cooperate on an app that secretly installs on a phone of crooks, villains or terrorists to be able to follow all their conversations and text messages? Or would you have reservations about this application as it could fall into the wrong hands?
How about this: you are asked to design a system that speeds up the production process by implementing fully automated packaging. In doing so, the majority of the workforce would become obsolete. Who should consider this, when would it be applicable or not? What measures can be taken?
There are many sustainability principles to consider in your organisation. What if there are no sustainability principles in your company? Do you, in your role as a sustainable developer, demand that there will be? Most likely there will be a much better and sustainable solution possible, if we are willing enough to consider the alternatives.

Of course there are plenty of cases, which we might take for granted that take these principles into account and benefit us. For instance

systems that ensure that logistics for transport are calculated as optimally as possible so that the impact on the environment is minimised or a system that ensures that filling in our annual tax return form will become a piece of cake (al least for most of us).

Work Ethics
A lot of focus is on efficiency, getting more done in a shorter time frame or with better quality so the satisfaction of customers improves. But efficiency in your daily work does not always have to be more fun. You may already feel a bit of the tension here, because is efficient work always more enjoyable work?

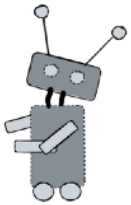
For example, if you were to handle all the easy incoming questions at a call center automatically (for example by a virtual assistant) but would route all the complex ones to live agents, you would actually burden a call center employee. By sending only the complex and escalated questions to a call center employee, the chance of having a fun interaction with the customer is smaller. What is the impact in the day of the life of these agents? How do they open their next call, what is the impact on how they feel their days goes by?

Is the work more fun? There must be a balance between increased productivity and work satisfaction; we coin it work ethics.

32. A FUTURE PERSPECTIVE

IT systems are becoming more sophisticated. 40 years ago, a school report had to be written by pen and paper. In the off chance, you had a typewriter, you were actually in an advanced position. Today, you can put together your essay by using ChatGPT[30] or have a summary of 200 words created by an AI document summary solution.

In the beginning of automation it was about the automation of simple actions, but nowadays IT is capable of simulating human intelligence through artificial intelligence. The image that has been created is that computer systems think for you, understand you and draw conclusions. In the 1980s, there was a great deal of concern about these developments. The rise of robotics reinforced this once more, and it was feared that humans would be replaced by the robot and that we would soon be ruled by robots.



Isaac Asimov, science fiction writer of the famous book 'I Robot' (film starring Will Smith in 2004), came up with three laws for robotics[31]:

1. A robot may not injure a human being or, through inaction, allow a human being to come to harm.
2. A robot must obey the orders given to it by human beings except if such orders would conflict with the First Law.
3. A robot must protect its own existence as long as such protection does not conflict with the First or Second Law.

As people have a conscience, an ethical framework within which we act. We have a feeling, most at least, certainly when it comes to

---

[30] https://openai.com/blog/chatgpt/

[31] https://en.wikipedia.org/wiki/Three_Laws_of_Robotics

decisions that affect people. And somehow, we want that robot, or that system, to add ethics and empathy. But you know what, 2d2 and c3po exist only in Star Wars. Lets keep them there and treat tech as a means and properly understand how new tech works.

33. CULTURE AND GOVERNANCE (PEOPLE AND PROCESS)

Systems are increasingly used in operations and provide support to organisations in making important decisions.
Organisations must realise the dependency on those systems and the impact these systems have on the organisation. If systems are deployed in a wrong way, it can greatly harm the reputation of the company or the organisation. We read all sorts of examples about this in the press.

As an organisation you have to take a number of steps to make sustainability part of the corporate culture. For this we developed a concept called the sustainability foundation (described in the next chapter). In addition, the human aspect plays an important role, as they define how these systems are developed and for which purposes these are deployed. The introduction of this concept should be done carefully and might need a transformation program and provides an important step in the evolution of moving from consciously incompetent to consciously competent.

These are the three main elements to consider during the transformation:

1. Trust in process. A first step to be taken is to set up a Governance Board that can assess the requirements for the creation and use of Responsible Systems. The board can monitor the appliance and deployment of systems.

2. Confidence in technology. In addition, you will have to create support, among the users (the skeptics but also the good believers), regarding the application of systems. This applies to both traditional and AI systems. Users need to be critical and understand how the

systems are deployed. Confidence in technology is the basis for acceptance.

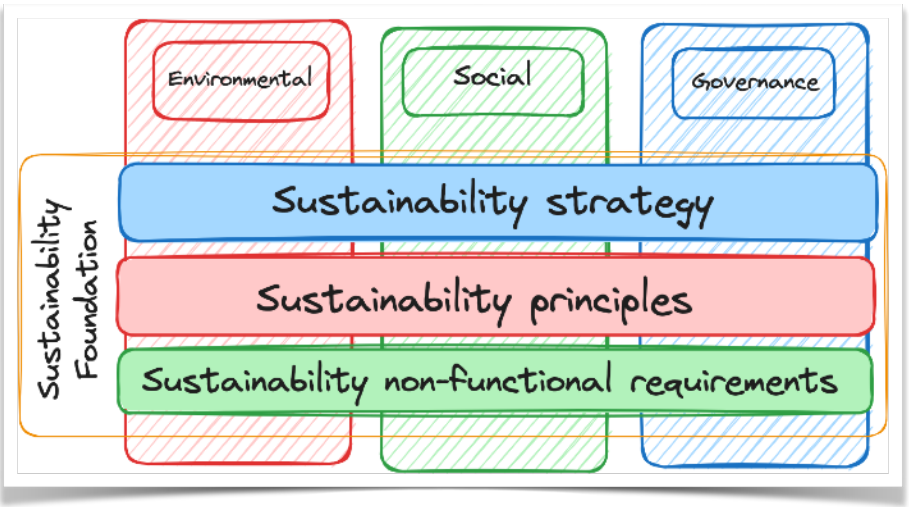
3. Trusting people. Like so many times, this is an evolution that must be taken step by step. Start small and scale as soon as there is room for it. Culture can't be tackled in a big-bang scenario. Gradually make sure that the use of Responsible Systems transforms the organisation into a reliable organisation.

In regards to cultural change, you could write a whole book, but each organisation requires a unique approach that can be worked out using the main elements described above[32].

[32] An interesting approach from the DevSecOps practice is shown in the book Team Toplogies (Matthew Skelton & Manuel Pais, 2019). This one explains how in a service model you can form autonomous teams in a self learning organisation.

34. ESG AS A COAT RACK FOR RESPONSIBLE SYSTEMS

We assume that companies already have a good operating model for business and IT strategy. If not, it would be urgent to develop one and immediately add sustainability. If you add the sustainability perspective to a business strategy, you could, for example, use the ESG framework[33] (Environmental, Social & Governance) as a coat rack[34] to describe these three important aspects. We assume that many companies know the ESG framework [28] already therefore we summarise it below.



Sustainability Foundation

[33] https://en.wikipedia.org/wiki/Environmental,_social,_and_corporate_governance

[34] This is a Dutch expression that we translated, the idea is that you can connect your idea to something, like hanging your coat on a coat rack.

The ESG framework contains three pillars:

Environmental: among other things, aspects of climate, energy usage, emission and water usage are described.

Social: this describes the ethical aspects of the framework. How do we deal with diversity, how do we treat our staff, how do we provide a safe workplace, how do we monitor privacy?

Governance: takes care of an organisation that oversees the strategic objectives and makes sure that the values and standards of the company are used in practice.

By adding a sustainability strategy to the business strategy, describing the sustainability principes and capturing the non-functional requirements you get a feeling of what the possibilities are and what is needed to change. This also sheds light on how to properly go about getting the organisation aligned and principles implemented/included in the day to day operation.

For ease of writing, we have called these three factors the sustainability foundation. Of course we are still talking about sustainability in the context of IT, but you can also use this for sustainability in a broader perspective.

35. RESPONSIBLE SYSTEMS DEVELOPMENT APPROACH

The question now is how you can still comply with the principles stated in the ESG management model without compromising on the speed of delivery (time to market):

1. The environmental factors (climate, energy, CO2)
2. The social factors (ethics, security, and Social Responsibility)
3. Monitoring and controlling of these factors

For traditional application development, one refers to the DevSecOps development process, where you define how you are iteratively engaging with the appropriate stakeholders into an application that meets the requirements. Could this also be used for developing Responsible Systems? We think that it is definitely possible by adding the sustainability foundation in the DevSecOps process. The sustainability foundation will have to be implemented to ensure that the developed system meets all the sustainability requirements. We have now called this the SusDevSecOps process.

For design, implementation and production of Responsible Systems we use the SusDevSecOps approach

Is building a responsible AI system now so much different as building a responsible system that consists of code in which business rules are coded? Yes, and no.

No. Principles and the values that apply to Responsible Systems always apply, regardless of whether AI is being used.

Yes. It's different anyway. Business rules have a logic, that may be complex, but you can decompose it in a manageable decision tree. AI models have a logic, too, but they are sometimes hidden. We put an amount of data into something we call a model. This model extracts information from the data and stores it in neurons, and then present new information, this network of neurons is used to interpret that data. That is why it is called a neural network.

In addition, business rules, are called rules, because rules are supposed to be constant, that is why they are called rules. When we played a game with marbles, a friend was constantly bending the game rules to avoid loosing his marbles. That is not the intention of rules, it should be stable, otherwise it is not fair. Data however is in motion.

Today, a litre of petrol costs 1.75, last week it was 1.65 and a month ago still 2.10. A little bit of change in data is not bad, but if gets too crazy, you will have to train your model again, otherwise it does not understand the reality anymore. Lets not forget, in the end it is a representation of a small piece of the world, one that continuously changes, thus our models and ideas should as well. Just like your old grandpa talking about dialling a phone number[35], almost literally turning it. If I turn my phone though, the screen will but there won't be anyone answering.

[35] Turning refers to old phones where you put the turntable into the right position to dial a telephone number

36. RESPONSIBLE SYSTEMS PRINCIPLES

Two examples of principles that can be applied to Responsible Systems:

Principle RS1	**Set up principles in response to the various aspects relevant to a responsible system (superprinciple)**
Statement	Translate the aspects of responsible systems to principles that fit your organisation.
Rationale	By thinking about the aspects we described in this chapter, you can determine what the important aspects for your organisation are, to ensure that a system is properly developed and used.
Implications	The developed principles should be guaranteed within the organisation, for example by an Architecture Board or Design Authority.

Principle RS2	**Make the aspects measurable, so you can validate them**
Statement	The aspects are at a conceptual level, they are quite generic so that they are relevant for a large audience. These should be defined in a specific way within your context.
Rationale	Measuring is knowing! The only way to actually implement it is by also defining KPIs. Thus there is a responsibility within the broader organisation.

Implications	For example, by making the aspects measurable, they are verifiable and, where appropriate, also implemented in test tools.

RESPONSIBLE IMPACT

Think it over....? Who still has time for that?
We are driven by the issues of the day and sometimes hardly have time to think carefully about how we can use technology in a way that clearly has an impact on and contributes to making our environment more sustainable.

Fortunately, good results have been achieved in recent years with technological developments in various areas.
For example, thanks to research and technology, we have a better picture of the state of the climate and therefore also the urgency to do something about it. The climate shows how we all, from individuals to organisations, from small to large, have an impact on the state of the world.

Another example is the way in which we have rapidly developed medicines to combat COVID. Without advanced technology, this would have taken years.

How about a simple technological application of drones? People don't have to get into dangerous shenanigans for a rooftop inspection. This dangerous work is already being carried out by drones that can easily take detailed pictures of the condition of the roof with their hands free.

If we were to activate the sustainability button in everything we do and weigh up the most sustainable option in our choice, we would be well on our way. It would of course be obvious to list a whole series of sustainable trade-offs, but we are sincerely convinced that it must come **intrinsically** from ourselves. It should not be imposed on you, but as soon as these kinds of considerations come from your heart and your mind, we are making important steps in the right direction.

This means that we have the responsibility and must take the time to think carefully about the impact we have with the use of IT.

We describe two examples of projects that contribute to a more sustainable society where IT is the key:
- A possible way of working (together).
- Integrate an environmental management system into an infrastructure asset management system.

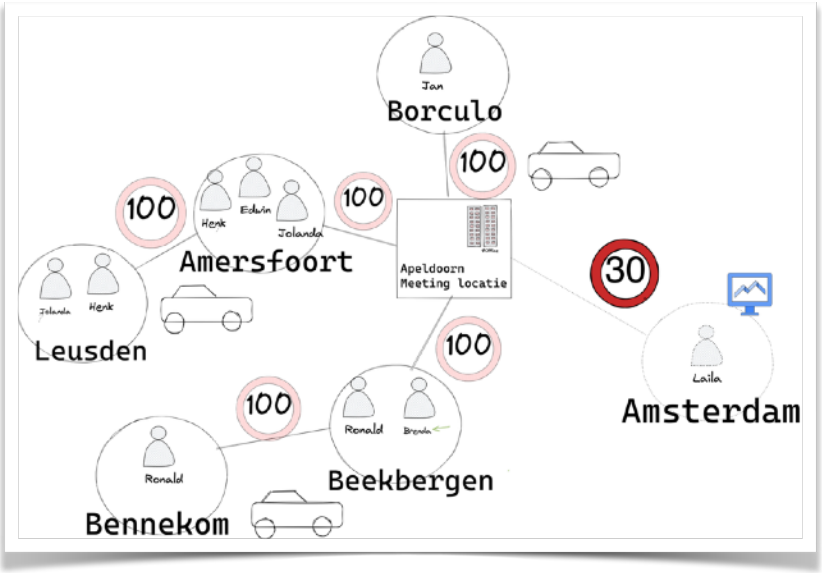
Then we go into Bitcoin, IT at its worst, what can we learn from it.

And finally, we invite you to think along and compete for a free and non-binding sustainability workshop!

37. A POSSIBLE WAY OF WORKING (TOGETHER).

We have already been forced into it with COVID. The 'new way of working'. Everyone was obliged to work at home and in no time we were used to communicating with each other from behind our ‘viewing tube’. In fact, the economy has not suffered as a result, and has actually grown during this period. Apparently it's a feature we've long neglected as an option for the modern way of working. No long traffic jams, people can live wherever they want because commuting is no longer an obstacle, less petrol is 'burned' and a nice bonus is the coffee tastes better at home.
Now that the pandemic has ebbed from our lives, we are in danger of falling back into old habits. Companies (that is management) still want their employees to come to the office to work. Collectively that is the case on Tuesdays and Thursdays, which means that it is very busy on the road on those days. It is part of the DNA of organisations to bring people together in the office. On the one hand, from a need for control to ensure that the employees are doing the right thing. On the other hand, you need social contacts when you work together with people on a project or for a customer in a customer team and, after all, we are social animals.
You can work individually on a file, but progress and results must be brought together and coordinated from time to time in order to be able to deliver a high-quality whole.

Imagine that the office is no longer the central place to get together, but that you organise the physical meeting with your colleagues in a place that is the most optimal distance for everyone. A planning process based on the route planning theory from the logistics world. However, this planning formula also includes elements such as the risk of traffic jams, weather forecasts and it is checked whether the person who has the most impact on the travel distance or travel time is indeed a crucial part of this meeting.



Sustainable Meetings

The time that you are productive becomes so much higher and therefore you have more time to 'think'.
Elaboration: the best place appears to be Apeldoorn.

Ronald leaves Bennekom and picks up Brenda in Beekbergen. Jolanda and Henk pick up Edwin in Amersfoort. Jan drives alone because he cannot take anyone on his route, but his contribution is crucial to the project. There are roadworks and traffic jams on Laila's route. She can attend this meeting via a virtual connection.
The CO2 emissions for this necessary meeting are limited in this way and the travel time is also reduced for everyone. In addition, you can already pre-meet in the car!

Staff meetings can also be organised in this way, whereby the option of public transport can be an important part of the calculation for the most optimal location.

38. AN ENVIRONMENTAL MANAGEMENT SYSTEM

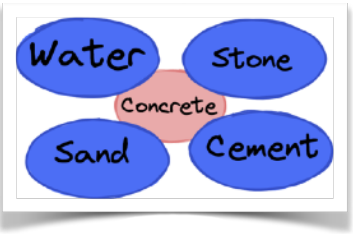
An environmental management system supports a company in achieving its sustainability objectives. This system looks at the entire business, not just IT. The standard used for this is ISO 14001.

Let's take a construction company as an example. As a construction company, how can you ensure that you approach sustainability in a holistic way without, as we said before, boiling the ocean? Or in a way that you are no longer running a healthy operation. Although we have to admit that we think there are few philanthropists around who would let it get to this point.

To really take this seriously, you have to divide your organisation into pieces. Each of those pieces makes a positive or negative contribution to sustainability. This division applies to both products and processes.

An example of a simple product is concrete. Concrete consists of water, sand, stone and cement. For each of these products, you need to know how durable it is to determine the overall durability of concrete. You can then look at making the product more sustainable. In this example, cement is by far the least durable. Can you use substitutes to make cement-free concrete?

What about the process? When making concrete, sand is added. Where does that sand come from? Is it sourced locally or is it transported for miles by ships, or even worse trucks that drive 1 in 4 and emit the quite some CO2 with every kilometer?



Product sustainability

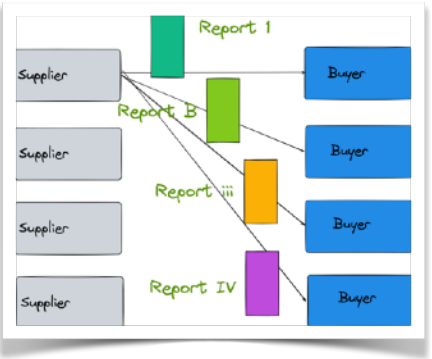
Making your company more sustainable therefore requires insight into products and processes. You have to dissect your company into small pieces in order to record the criteria for each relevant piece and then measure it. Many companies use so-called digital twins. It is a digital representation of reality. You record the truck in an IT system, how much energy it uses, how much CO2 it emits and you can then use that information to make decisions. For example, you let the truck with less emissions drive more often than the one with more emissions. Or you decide to replace the less durable ones. What you can even do is to install an IoT (Internet of Things) computer in the truck that measures the actual CO2 emissions. Based on this, you can also determine whether additional maintenance is required.

Managing your assets (Asset Management) in combination with IoT and an environmental management system is increasingly being used and there are good examples of how, for example, the lifespan of machines is extended or how physical checks on transformer stations need to be carried out less often, resulting in many car kilometers be saved.

39. THE VISMA APPROACH

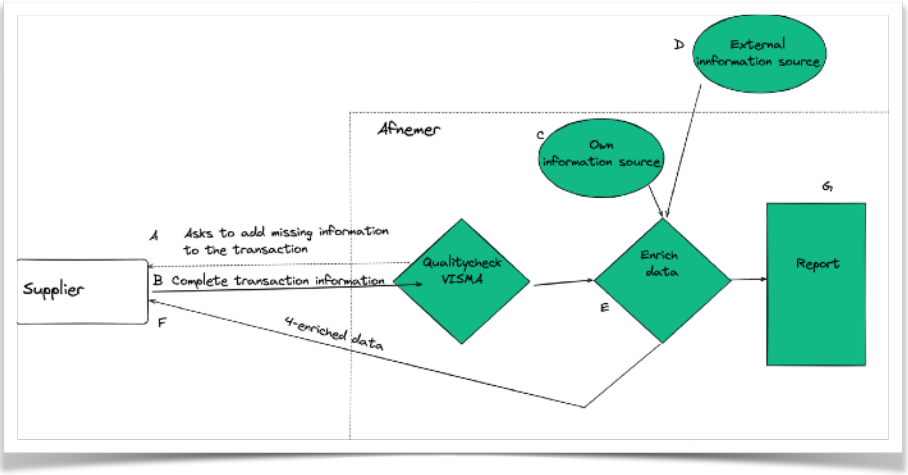
Reports. Why are sustainability reports important? For the simple reason that existing business models can have a major impact on sustainability. Whether it concerns the transport of goods or whether child labor is involved, the insights from reports will influence these existing business models. Reporting responsibility usually lies with the corporate holding companies and not with the operating companies. These operating companies are in many cases P&L (profit & loss) responsible. By reporting on the sustainability activities of the operating companies, insights are gained that normally remain 'under the table'. This leads to transparency of the sustainability activities. These insights allow 'the boss' (corporate) to give more direction to existing business models of the operating companies. Not only from 'the boss' but also from the employees and what about the shareholders, financiers and insurers. All in all, society as a whole will put 'pressure' on companies to adopt sustainable business models.

An environmental management system can help you to map out the impact on the environment of the activities that you manage yourself. Companies use suppliers, but the question is what about the impact on the environment of these parties? In order to implement a responsible policy and to report on this, you would like to have that insight. In an interview with Gilbert Haverkamp of Visma, he gave us an interesting perspective on this.



Take a retailer, for example, they have hundreds of suppliers. Retailers also regularly change suppliers, new ones arrive, existing

ones go. This example also applies, of course, to large corporates with their operating companies and also to governments with their implementing organisations. Should you now ask all those suppliers to report on the environmental impact of the specific transactions they do with your company? If your supplier also supplies to a hundred other companies and they have to provide their data for all their customers every year, then that is painstaking work. Customer 1 wants the data in an XML format with the name as the first field and customer 2 wants the data in JSON format with all fields in Chinese. Only the retailer benefits from this information. It's just extra work for the supplier. Visma takes a different approach.



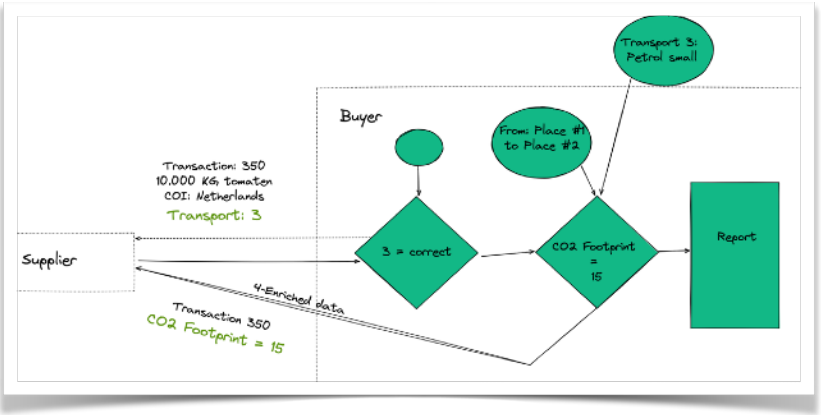
Principles for using the Visma approach:

- a lot of data is already available and needs to be supplemented and not completely rebuilt
- chains to be reported on are very dynamic and can change daily
- by unburdening the chain as much as possible instead of obliging it to provide complete reports, the quality of the data improves
- enriched information is returned to the chain
- Because the origin of the data is known, a fully auditable situation is created

Gilbert's first experience is that the retailer itself already has a lot of information. Don't ask your suppliers for that information again. Only ask the suppliers for information that you do not have (A), that you cannot request elsewhere and that you cannot calculate yourself. Add this information to each individual transaction (B). Visma then checks the quality of the added data with the same precision as checking a financial payment. Process this high-quality data (E) together with the information you already had (C + D) and give it back to the supplier (F). The supplier can then use that extra information to improve its data quality and environmental performance. This creates an environmental improvement circle, which means working together for a better environment.

Here's a hypothetical example. The means of transport is added to the transaction by the supplier. All data is now known to the customer, so that the CO2 calculation can be carried out. The transaction record is now complete and has value for both the customer and the supplier. The value for the customer is insight into the tax per transaction per product, so that it is possible to control how products can be purchased in a balanced way. If the customer makes the data available to the supplier, the supplier also controls the control mechanism.

Based on this, both the buyer and the supplier can make a list with the greatest impact on CO2 emissions and decide to have the tomatoes come from Flevoland from now on.



40. BITCOIN MINING

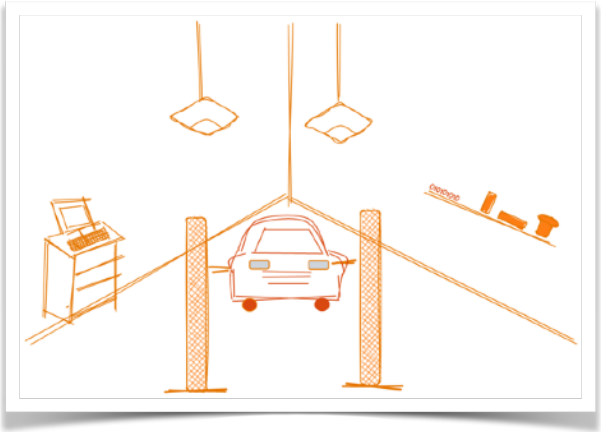
An example of a not so sustainable solution is bitcoin mining. Bitcoin is a so-called cryptocurrency. A cryptocurrency is a digital means of payment that can be used to pay on the Internet. It is a nice concept because a transaction takes place directly between supplier and customer. There is no intermediary (read: bank) involved. But there is also a disadvantage: checking such a transaction is done by so-called miners. These check a bundle of transactions and that involves an enormous amount of computing power. And computing power we talked about earlier, that costs energy and energy results in CO2 emissions. How much will you ask? Don't be alarmed, but this is about 297 billion kWh in 2024. That probably doesn't tell you very much, but it is about what 118 countries in the world emit, almost three times as much as what a country like the Netherlands emits per year. Bang! What do you do with that now? And what (but also for whom) does it actually bring to the world, especially given the image of crypto in recent months and the accompanying headlines.

An answer to these kinds of problems is not easy. It is important that we continue to think about IT energy consumption. Video streaming, gaming and advertising are also major users of computing resources. Is there a way back? Difficult huh? So let's start by taking the measures outlined in this booklet. An idea might be a Sustainability Garage?

41. SUSTAINABILITY GARAGE

The idea of a Sustainability Garage is to find a solution together to achieve the sustainability goals that a company has set. At the Sustainability Garage, the approach is identical and we focus on the goals of the UN, the UNSDG. We look at which of the objectives are relevant and how realistic. We then build a Minimum Viable Product that must actually demonstrate how the objectives can be achieved.

Make a change to win a sustainability garage workshop with IBM with a possible MVP as outcome! Do not wait too long with your idea to make your organisation sustainable, but file your idea with one of the writers of this book. Their contact information is in the back of this book.



Visit us in our garage

We mentioned it at the beginning of this booklet. Do you have a good idea? Send your idea to the authors of the booklet. You may be eligible for a completely non-binding Sustainability Garage. We will choose the first one at the end of 2023 and discuss with you how we can realise this.

Terms and conditions::
- The idea must be sustainable and make use of IT (sorry, our limitation).
- It must be related to the organisation you work for.
- You must have stakeholder support in your organisation.

We will handle your idea confidential.

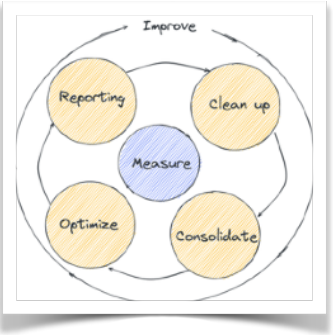
We are curious about your ideas!

CONCLUSION

Reading this booklet has hopefully given you an idea of what we mean by Responsible Computing. So it is not just about energy consumption of data centers and infrastructure, but also about “green coding” and building ethically sound systems that respect privacy.
We have sometimes used examples that are far from the possibilities to indicate that we have a biased view of computers, so bias. But there are also countless examples of sustainability that you can start with **now**.
Some subjects are far beyond your responsibility. Focus mainly on those topics that you can live up to! By making sustainability part of your working method, you become consciously competent and unconsciously competent! This can be done, for example, by including sustainability as a criterion in requests or tenders:

Requirements	Criterion	Points
Functional1.0	Integration with Office 365	15
Functional1.1	User-friendly interface	15
Non-Functional1	Security according to ISO27001 standard	20
Non-functional2	Availability 99.9%	20
Sustainability 1	**Data center is 85% self-sufficient**	10
Sustainability 2	**Decision lineage is stored in the application**	10
Price	Software and technical implementation	10

42. THE NEXT STEP



We have four super simple steps for you, you can start now:

Step 1. Clean up: turn off if you are (temporarily) not using an environment and clean up what you no longer need. This applies to servers in the data center, but did you know that you can also turn off your laptop?

Step 2. Consolidate: merge environments with limited use, even if it is temporary. For example, at night environments are used less, merge them and turn off the other environments (step 1).

Step 3. Optimise: move your applications to sustainable infrastructure. (Buy a Laptop with a RISC processor).

Step 4: Report: set (ambitious) goals and be transparent.

To know what to do and where you can improve sustainability, you will have to measure continuously. You can improve based on the measurement results.

Have you ever seen such a simple step-by-step plan? If you do this, you can certainly save 50-70% on energy costs. Agree? Then why don't you? Not easy, is it, implementing a simple sustainable strategy. You need to have people on board, your organisation, it has to be done with each other, it has to become part of your DNA.

A maturity assessment can be a very good way to get sustainability in the DNA and possibly develop a vision and define a strategy. An assessment indicates where you stand in terms of maturity in the six different domains (unconsciously incompetent ==> unconsciously

competent). It also indicates where you would like to go. Based on this information, you can develop a roadmap that you can implement in the coming years. Step by step, as that is sustainable.

43. SAARINEN & IBM AS A SOURCE OF INSPIRATION

We began this booklet with Watson Jr.'s letter telling his managers that IBM has a great responsibility to the environment and that all processes and regulations must be followed to the smallest detail.

We would like to conclude here with the work of Eero Saarinen. He is a Finnish-American architect, known for his neo-futuristic style and work such as the Gateway Arch in St Louis, Missouri or the WA Flight Center, John F. Kennedy International Airport. Or you might know him from the Tulip chair or dining table, which he designed for Knoll. Maybe you have it at home?

In 1957, Saarinen was commissioned to design a new scientific research center for IBM in Yorktown Heights, New York: The Thomas J. Watson Research Center.

Consciously competent or perhaps unconsciously competent, he made choices that turned out to be extremely sustainable, because to this day his designs are current, relevant and usable. The same goes for the center in New York; the basis was the activities of the researchers, which led to a flexible layout of the space with a focus on concentration to minimise distances. Nature was also involved in the entire design, to offer relaxation to the researchers, and to allow the building to blend in with nature[36].

Today, this building is still about inventing the technology of the future. It has hosted world-renowned brilliant mathematicians, winners of Nobel prizes, supercomputers, the famous chess computer Blue Gene who beat Kasparov, Watson who won Jeopardy. Dozens of inventions that define our IT industry today,

---

[36] Pierluigi Serraino (2010), World Architects by Taschen - De Volkskrant

including that energy-efficient RISC CPU we talked about. But also the transistor used in the CPUs.

We would like to finish our book with the following words of Saarinen:

> "Each era must create its architecture with the available technology - one that expresses the spirit of the time."
>
> Eero Saarinen

The spirit of this time is sustainability. Let's use available technology to build systems sustainable and build systems with a Responsible Impact.

ABBREVIATIONS

3D	Three Dimensional	A technology where depth is achieved in a visualization. Most commonly by controlling your eyes separately. For example through 3D glasses.
3TG	Tin, tantalum, tungsten gold	Precious materials that are incredibly important to build computers. Supervision of the responsible procurement of these materials is an important sustainability focus.
AI	Artificial Intelligence	Artificial Intelligence is a kind of fake intelligence used by computers. It's not really intelligent, but by combining a lot of information, the computer can make decisions or acquire insights based on huge amounts of information. Very handy, but you have to be careful because he can make stupid decisions as well and then you are in deep shit!
AIX	Advanced Interactive eXecutive	AIX is a Unix operating system that is developed by IBM. The systems are deployed primarily as enterprise servers.
API	A Programming Interface	A way applications can communicate to each other, a necessity so to say and allows companies to open up themselves to others.
BI	Business Intelligence	To reveal important business information. This can be based on reports, dashboards, analyses, data mining.
C2C	cradle to cradle	From the cradle to the crib, so don't go to the grave. You use something once again instead of throwing it away you are going to reuse it.
CD	Compact Disk	A plastic plate where data can be stored. The data can be music.
CDO	Chief Data Officer	The person that is responsible for everything related to data. A relative new role in organisations, became more popular with the introduction of GDPR, if you do not have one, appoint one.

CI/CD	Continuous Integration/ Continuous Delivery	Practices of continuous integration and continuous delivery or deployment.
CIO	Chief Information Officer	The boss of the IT department, probably your boss.
CISC	Complex instruction Set Computer	Minimizes the number of statements per program. That ensures that the instructions themselves are more complex. The processor is also more complex (about 3x more transistors needed) than a RISC processor.
CISO	Chief Information Security Officer	Super strict person who is watching everything in the area of security. Stick to the rules!
CO2	Carbon dioxide	Substance released during conversion of fuel to energy. Protects the earth against radiation, but we now have too much of it, causing the earth to warm up. On some days you might want to, but the ice is melting, the sea level is rising and if we continue like this half of the Netherlands will be under water in 50 years.
COP 27	Conference of the Parties	The 27th conference on climate, held in Egypt in 2022.
COVID-19 COVID	COrona VIrus Disease 2019	Type of official name for the Corona virus that is discovered in China in 2019.
CPU	Central Processing Unit	The brain of the computer.
CRAC	Computer Room Air Conditioning	Air conditioners are important energy consumers in datacenters. Have a lot of influence on the PUE factor. Don't confuse CRAC with crack.
CRISP-ML	CRoss Industry Standard Process for Machine Learning	Approach to deal in a sustainable way with AI.
CSO	Chief Sustainability Officer	The boss who overlooks sustainability. If this person does not have this book, provide it immediately!

CSRD	Corporate Sustainability Reporting Directive	The CSRD Directive is at the heart of the European Union's Green Deal and should provide greater transparency and better quality of sustainability information.
CTO	Chief Technology Officer	The boss of technology strategy, among other things the IT strategy.
DNA	desoxyribonucleic acid	Contains the hereditary information of an organism, but in our case we use this as an analogy for the culture of organisations.
EU	European Union	Most of us know.
GDPR	Global Data Protection Regulation	European regulation that indicates how a business should deal with privacy.
GIT	Whatever rocks your boat, but it could be referring to Torvald (the creator) himself	Open source software for distributed version control, in essence a stupid content tracker (so they said themselves).
GPU	Graphical Processing Unit	Many computers have a second set of brains next to the CPU, the GPU. No wonder they're getting smarter all the time. GPU were meant to run many simple calculations in parallel to direct the pixels on your screen. The characteristic of the processor lents itself excellent to AI and is now widely used for it.
LHR	Human Resources	Basically all the employees of an organisation, but usually we mean the personnel department.
I/O	Input and Output	The bits and bytes (8 bits) that go back and forth between the data storage system and the processor.
IaaS	Infrastructure as a Service	You no longer need to purchase and configure your own computers, but this is being served as a service and you pay for use. It is a service model that is popular with cloud suppliers, but can also be supplied by your own IT organisation.

IBM	International Business Machines	A great and beautiful (in our humble opinion) American company. Focus on AI and hybrid cloud and has done various sustainable inventions like the electron microscope and the RISC processor.
IT	Infrastructure Technology	By infrastructure, we mean computers, network equipment, and storage equipment for data.
MER	Main Equipment Room	The central computer room where communication enters and the central servers are located.
ML	Machine Learning	A discipline of AI that aims to teach computers how to learn and act without being explicitly programmed.
MVP	Minimum Viable Product	An initial version of a product that can be used in a business operation, and thus shared with customers. Which will iteratively be updated with the gathered feedback.
NORA	Dutch Government Reference Architecture	A reference architecture used within the Dutch central government: https://www.digitaleoverheid.nl/overzicht-van-alle-onderwerpen/standaardisatie-en-architectuur/nora/ GEMMA is the variant for the Dutch municipality: Municipal Model Architecture.
ODC	Government DataCenter	Typical for the Netherlands, but perhaps other countries have this. In connection with efficiency, the Dutch government has put a lot of IT in four datacenters.
PaaS	Platform as a Service	Providing or using an entire platform as a service. This includes infrastructure, a bundle of related software and it's management services. Examples are a development platform service like ARO (Azure Red Hat Openshift) or it could also be a data platform.
PC	Personal Computer	Just a computer for you. You don't share them with others. A computer that you share we call it a server.
PDF	Portable Document Format	Sure you know this format, but maybe you didn't know what the abbreviation stands for!

PII	Personal Information Information	Information that is related to you as a person such as your phone number or your email address. According to the GDPR, companies have to deal with this information very carefully.
PS	Post Script	It comes from Latin postscriptum, you write that at the end if you forgot something important and would like to highlight that. You can also ad PPS if you forgot another thing. But what should we say about a person that starts to forget so many things…
PUE	Power Usage Effectiveness	Indicates how energy efficient the datacenter is. For a factor of 1, all energy is used for the computer equipment. Energy efficient Datacenter do have a PUE of 1.2.
REST	Representational State Transfer	REST is a protocol to exchange information between services on the Internet.
RISC	Reduced Instruction Set Computer	Simple instruction set. Because of this, the software has to run slightly more instructions compared to the CISC processor but is far more efficient. Now it is clear that this is the future. Apple MacBooks have also moved on to this technology. A good reason to tell your boss to provide you with a MacBook!
SaaS	Software as a Service	Providing and using software as a service.
SDG	Sustainability Development Goals	Wide-defined objective of the UN in the field of sustainability. That's why they're called UNSDG. For a meaning of this abbreviation you have to look at UNSDG.
SER	Satellite Equipment Room	A remote computer space that is connected to the MER. Look at the MER for that meaning.
SQL	Structured Query Language	Good old SQL is a programming language to request data from relational databases. For example: SELECT salary FROM celebs WHERE LAST_NAME == "Clooney"; COMMIT;

SRE	Site Reliability Engineer	A role for a person that manages a computer platform rather than one system. This concept was introduced by Google to avoid linear scaling of support teams.
SusDevSecOps	Sustainable Development Security and Operations	The new hype, created by these authors to indicate that applications should be developed and run sustainable
TCO	Total Cost of Ownership	Total cost of a solution, product, or system that includes not only initial purchase but also long term maintenance.
TOGAF	The Open Group Architecture Framework	The-de facto IT Architecture standard that describes what you need to do in order to get to a good working system.
UNSDG	United Nations Sustainability Development Goals.	See SDG. They have been invented by the United Nations, so hence called UNSDG.
UPS	Uninterruptible Power Supply	A temporary emergency power supply that turns on when the public power line fails. Makes sure that the systems keep running. Usually a battery that delivers temporary power until the emergency power supply has been started. So has nothing to do with the parcel delivery company.
VR	Virtual Reality	A fake reality, but it sometimes seems like real. You can achieve this by setting up a special pair of glasses that will allow you to have the idea of being in a different world.
WUE	Water Usage Effectiveness	Specifies how much water is used to cool systems.

ANNEX A: REPORTING OBLIGATIONS

CSRD stands for Corporate Sustainability Reporting Directive. It is about companies having to report on their sustainability goals and results. This is not about IT, but about the entire business. If you read the reports, they are very different and you cannot compare them. How do you know whether you are working with the most sustainable company or not?

So standardisation is coming (2024) to enable comparisons and encourage companies to make sustainability part of their business operations.

There are two IT aspects to these reports:
The data that is reported on and the reports that are made are usually done using IT systems. An example of a reporting tool is Envizi. Several companies can help get this up and running.
The data being reported on also includes IT. For example, the power consumption of data centers. We have already written enough about this in this book. To measure is to know, that insight will help to become more and more sustainable.
KPMG[37] has conducted research and more than 2000 companies in the Netherlands and 49,000 in the EU are covered by the CSRD directive. Every chance that there is work to be done for your company as well.[38]

We have not specifically delved into the CSRD guidelines, but we would like to mention this in this appendix because this may be a stepping stone for your company to sustainable IT. This can act as a catalyst for Responsible Computing..

---

[37] https://home.kpmg/nl/nl/home/topics/environmental-social-governance/corporate-sustainability-reporting-directive.html

[38] https://www.mvonederland.nl/wat-is-de-csrd-wet-en-hoe-ga-je-ermee-aan-de-slag/

REFERENCES

We know that references are hardly consulted, but we put them in nevertheless, also for ourselves. Only some of them are in Dutch but we found out that google translate can support you with reading the site.

[1] https://responsiblecomputing.net/
[2] https://www.noraonline.nl/wiki/Duurzaam_ (Purpose)
[3] Cradle to Cradle: Remaking the Way We Make Things, Michael Braungart and William McDonough
[4] https://datanews.knack.be/ict/nieuws/hoe-vermijden-we-dat-ai-uitdraait-in-een-ecologische-ramp/article-opinion-1865833.html?cookie_check=1670936753
[5] https://www.managementboek.nl/boek/9789089653185/agile-managen-mike-hoogveld
[6] https://en.wikipedia.org/wiki/Proxemics
[7] Office for System Development
[8] https://www.dutchdatacenters.nl/thema-energie/
[9] https://www.microsoft.com/en-us/download/details.aspx?id=56950
[10] The Open Group Architecture Framework: https://www.opengroup.org/togaf
[11] https://itnext.io/risc-vs-cisc-microprocessor-philosophy-in-2022-fa871861bc94
[12] How to: Develop a Blueprint for a Data-Driven Enterprise Architecture
[13] https://www.wallarm.com/what/the-concept-of-grpc .
[14] http://patricialago.nl
[15] For example: https://www.toptal.com/developers/sorting-algorithms .
[16] https://greenlab.di.uminho.pt/wp-content/uploads/2017/10/sleFinal.pdf
[17] https://www.ibm.com/cloud/blog/low-code-vs-no-code

[18] https://huggingface.co/blog/carbon-emissions-on-the-hub
[19] https://www.turbonomic.com/sustainability-calculator/
[20] https://www.transport-online.nl/site/148079/postnl-ziet-geen-afname-pakketbezorging-ondanks-hoge-inflatie/
[21] https://ictworks.org/wp-content/uploads/2019/11/Responsible_Data_Maturity_Model_10-16-19.pdf
[22] https://www.ibm.com/topics/data-catalog
[23] https://aifs360.mybluemix.net/examples/hmda
[24] The user cannot see what the underlying reasoning is of an algorithm. Only the input and output are known which makes it uninterpretable.
[25] https://openai.com/blog/chatgpt/
[26] https://nl. wikipedia.org/wiki/Drie_wetten_van_de_robotica
[27] An interesting approach from the DevSecOps practice is shown in the book Team Toplogies (Matthew Skelton & Manuel Pais, 2019). This one explains how in a service model you can form autonomous teams in a self learning organisation.
[28] https://en.wikipedia.org/wiki/Environmental,_social,_and_corporate_governance
[29] https://www.iso.org/iso-14001-environmental-management.html
[30] Pierluigi Serraino (2010), World Architects by Taschen-De Volkskrant
[31] https://home.kpmg/nl/nl/home/topics/environmental-social-governance/corporate-sustainability-reporting-directive.html
[32] https://www.mvonederland.nl/wat-is-de-csrd-wet-en-hoe-ga-je-ermee-aan-de-slag/

The original book was written in Dutch. However there is much international demand for this book and therefore we translated it in English. With help from Watson and from Google. Don't be alarmed if you sometimes may have to read a line twice or read between the lines to understand what is says.....That's translation services for you. Ronald and Jan have spent quite some time in adjusting the outcome of the translation services. Laila made sure the boys did a good job!

The book has become longer than we had anticipated. We have tried to show cohesion between the Responsible Computing domains in a light-hearted way and to provide practical tips for becoming more sustainable.
We are from IT and have therefore limited ourselves to that. In our opinion, we are quite complete in that domain.

An important value of sustainability is openness. Open discussions and transparent reporting are a precondition for making the world more sustainable. This also applies to IT, which is why this booklet is also free to use. You can find it in PDF format on GitHub::

https://github.com/OrangeSeries/Sustainability

Laila Fettah: lailafettah@nl.ibm.com
Ronald Meijer: meijerr@nl.ibm.com
Jan Schravesande: schravesande@nl.ibm.com
Edwin van der Burg: edwinvdb@nl.ibm.com

ABOUT THE WRITERS

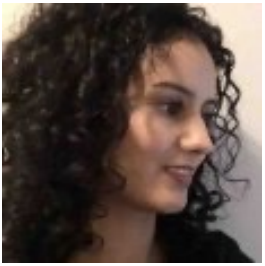
Jan Schravesande is an IBM and Open Group certified Executive Enterprise Architect and has over 40 years of experience in various industries such as insurance, supply chain, energy & utilities and government. For the last ten years he has been active as a technical consultant for organisations in the Dutch government.

"Sustainability is a subject that should concern us all. When I started this book I was still unconscious incompetent in many areas, but now I have been 'promoted' to conscious incompetent and in some areas even consciously competent. I have to say honestly that writing this booklet has changed my outlook on life somewhat”

Ronald Meijer is a certified IT Architect working at IBM. From this organisation he has worked for organisations such as Rabobank, ABN/AMRO, Shell, KLM, ING, Delta Lloyd, Aegon. He is a passionate trainer in TOGAF(R), Architecture Thinking and Microservices Modelling. His experience and knowledge extends across the breadth of IT architecture, namely business, data, application and infrastructure architecture. Graduated in electrical engineering with a specialisation in technical computer science. He regularly speaks at conferences and in various forums.

“Sustainability is something that belongs to all of us. I experience an appropriate, but also passionate “annoyance” when I see that with the current state of 2 nm chip technology, companies miss out on 75-90% energy savings. It also surprises me when people hide behind green energy. Come on, first turn off what you don't use, then save on your infrastructure and finally what you use, yes of course you do that green."

Laila Fettah is an Open Group and IBM Certified IT Specialist. She started in the world of statistics and spent the first ten years of her career finding what really matters for business processes in structured and unstructured data. She has a great passion for bringing people, processes and technology together. She is currently an Architect and works on overarching themes within the Dutch government.



"Organizations and technology have so much impact on our daily lives, but also on a macro level on the state of the world. For example, we already heard from home, "turn off the tap quickly, waste of water" and worn clothing was passed on. But somehow I hadn't found the mode enough to see the "business" possibilities. We all bear responsibility for our world and the world of our children. With all the little bits together we come to a big sum".

Edwin van der Burg is an Open Group and IBM Executive Certified Architect and has gained experience in the Travel & Transport, Retail & Consumer Goods and Government sectors over the past 30 years. For the past 4 years he has been the Technical Advisor for the Ministry of Defence. The problems are diverse, but always come down to translating business issues into a workable solution of applications with underlying infrastructure. His starting point is always that the solution must be stable, workable, modular and maintainable in order to be able to deal with changes in the future.

"Sustainability is a theme that I have personally been involved with for the past 12 years, especially in the field of energy consumption. In 2010 it was already clear that the first step towards sustainability is saving. With good insulation and more economical transport, you can already reduce energy consumption by 40%. With this booklet, we have hopefully inspired you how the IT systems can be made more sustainable and can contribute to making society more sustainable."

Dear reader,

In English we talk about Sustainability. Sustain is known in the musical world as reverberation: “making it last longer”. Just think of a piano key that you hit or a guitar string that you pluck.

Responsible computing refers to the use of IT in a responsible manner, which, so to speak, has repercussions for users of these systems or even for society. IT offers us many possibilities, it can no longer be ignored in our lives. It has even become so important that there seems to be an IT solution for every problem. Before you know it IT is a goal, instead of a means to support us. Is IT used correctly and are sustainability issues such as ethics and energy consumption an intrinsic part of the drawing board? Just realize that all together the data centers worldwide use no less than 1.5% power, let alone if you add the IT technology that is in our homes.

There are many questions that we, as the authors of this booklet, do not have answers to. In everyday life we are IT architects, not sustainability specialists (although we now dare to say that we have earned a first Boy Scout medal) and not writers! But in one area we are really good, we understand IT systems. The four of us worked on this, and 4 x good is pretty good.
When we started, we only knew a quarter of what it now contains. We are overwhelmed by data about the energy consumption of IT data centers, from AI, and how about cryptocurrency such as bitcoin.
We started writing this booklet with our minds, but you will notice as you read that we started writing more from our hearts.
We hope that you may become aware of the consequences of IT use from your subconscious. But we don't stop there. We also have ideas on how we can improve sustainability together. Your contribution is a drop in the ocean, but with 100,000 IT people in the world, together we drop that plate 1.5 degrees.

9 789083 314525